A Path Decomposition Approach for Computing Blocking Probabilities in Wavelength-Routing Networks

Yuhong Zhu, George N. Rouskas, *Member, IEEE*, and Harry G. Perros, *Senior Member, IEEE*

***Abstract*—We study a class of circuit-switched wavelength-routing networks with fixed or alternate routing and with random wavelength allocation. We present an iterative path decomposition algorithm to evaluate accurately and efficiently the blocking performance of such networks with and without wavelength converters. Our iterative algorithm analyzes the original network by decomposing it into single-path subsystems. These subsystems are analyzed in isolation, and the individual results are appropriately combined to obtain a solution for the overall network. To analyze individual subsystems, we first construct an exact Markov process that captures the behavior of a path in terms of wavelength use. We also obtain an approximate Markov process which has a closed-form solution that can be computed efficiently for short paths. We then develop an iterative algorithm to analyze approximately arbitrarily long paths. The path decomposition approach naturally captures the correlation of both link loads and link blocking events. Our algorithm represents a simple and computationally efficient solution to the difficult problem of computing call-blocking probabilities in wavelength-routing networks. We also demonstrate how our analytical techniques can be applied to gain insight into the problem of converter placement in wavelength-routing networks.**

***Index Terms*—Call-blocking probability, converter placement, decomposition algorithms, wavelength-division multiplexing, wavelength-routing networks.**

I. Introduction

RECENT advances in wavelength-division multiplexing (WDM) and optical switching make it possible to contemplate the deployment of wavelength-routing networks that will provide backbone connectivity over wide-area distances and at very high data rates [7]. A wavelength-routing network consists of wavelength routers and the fiber links that interconnect them. Wavelength routers are optical switches capable of routing a light signal at a given wavelength from any input port to any output port, making it possible to establish end-to-end lightpaths, direct optical connections without any intermediate electronics. The functionality of optical switches may be enhanced by employing wavelength converters, devices that are capable of shifting an incoming wavelength to a different outgoing wavelength [14], [1]. Wavelength conversion is a desirable feature since it improves the performance of the network in terms of call-blocking probability.

While the operation of wavelength-routing networks is expected to be similar to that of conventional circuit-switched networks, several new issues arise which add significant complexity to the problems of design and performance evaluation of the former. Specifically, the existence of multiple distinct wavelengths makes it necessary to employ a wavelength allocation policy to assign one of the (possibly many) available wavelengths to an incoming call. Similarly, the wavelength conversion feature gives rise to new problems associated with evaluating the benefits of conversion and optimally placing the converters at the various nodes. Also, dynamic (or adaptive) routing is tightly coupled with wavelength allocation, since it involves a search over available wavelengths in addition to a search over the possible paths for establishing a call.

The problem of computing call-blocking probabilities under static (fixed or alternate) routing with random wavelength allocation and with or without wavelength converters has been studied in [1], [11], [2], [6], [14], [16], [15]. The model presented in [1] is based on the assumption that wavelength use on each link is characterized by a fixed probability, independently of other wavelengths and links, and thus, it cannot capture the dynamic nature of traffic. In [11] it was assumed that statistics of link loads are mutually independent, an approximation that is not accurate for sparse network topologies. The work in [2] developed a Markov chain with state-dependent arrival rates to model call blocking in arbitrary mesh topologies and fixed routing; it was extended in [6] to alternate routing. While more accurate, this approach is computationally intensive and can only be applied to networks of small size in which paths have at most three links. A more tractable model was presented in [14] to recursively compute blocking probabilities assuming that the load on link i of a path depends only on the load of link $i-1$. A study of call blocking under non-Poisson input traffic was presented in [16], under the assumption that link loads are statistically independent. Finally, in [15] a dynamic programming algorithm was developed to determine the location of converters on a single path that minimizes the average or maximum blocking probability.

Other wavelength allocation schemes, as well as dynamic routing, are harder to analyze. First-fit wavelength allocation

Manuscript received December 17, 1998; revised September 11, 1999 and June 1, 2000; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor G. Sasaki. This work was supported in part by the National Science Foundation under Grant ANI-9805016. Parts of this paper were presented at the IEEE INFOCOM'99 conference, New York, NY, March 21–25, 1999, and at the Sixteenth International Teletraffic Congress (ITC 16), Edinburgh, U.K., June 7–11, 1999.

Y. Zhu was with the Department of Computer Science, North Carolina State University, Raleigh, NC, 27695-7534 USA. He is now with the Optical Network Group at Lucent Technologies, Acton, MA 01720 USA.

G. N. Rouskas and H. G. Perros are with the Department of Computer Science, North Carolina State University, Raleigh, NC 27695-7534 USA.

Publisher Item Identifier S 1063-6692(00)10926-4.

was studied using simulation in [3], [11], and it was shown to perform better than random allocation, while an analytical overflow model for first-fit allocation was developed in [8]. A dynamic routing algorithm that selects the least loaded path–wavelength pair was also studied in [8], and in [12] an unconstrained dynamic routing scheme with a number of wavelength allocation policies was evaluated. We studied the first-fit and most-used allocation policies in [19], and we showed through analytical and simulation results that they exhibit very similar performance over a wide range of traffic loads and network topologies.

Most of the approximate analytical techniques developed for computing blocking probabilities in wavelength-routing networks [11], [2], [6], [16], [8], [12] make the assumption that link blocking events are independent and amount to the well-known *link decomposition* approach [5], while the development of some techniques is based on the additional assumption that link loads are also independent. Link decomposition has been extensively used in conventional circuit-switched networks where there is no requirement for the *same* wavelength to be used on successive links of the path taken by a call. The accuracy of these underlying approximations also depends on the traffic load, the network topology, and the routing and wavelength allocation schemes employed. While link decomposition techniques make it possible to study the qualitative behavior of wavelength-routing networks, we believe that more accurate analytical tools are needed to both evaluate the performance of these networks efficiently, as well as to tackle complex network design problems, such as selecting the optical switches where to employ wavelength converters.

In this paper, we consider a wavelength-routing network with an arbitrary topology. Each link in the network carries W wavelengths. Call requests between a source and a destination node arrive at the source according to a Poisson process with a rate that depends on the source–destination pair. Call holding times are assumed to be exponentially distributed. We consider both fixed and alternate routing [5]. In *fixed* routing, each source–destination pair is assigned a single path. If there are no wavelength converters in the path, a call is blocked if no wavelength is free on all links of the path. This is known as the *wavelength continuity* requirement, and it increases the probability of call blocking. If some nodes in the path employ wavelength converters, a call is blocked if no wavelength is free on all the links of any segment of the path consisting of links between successive nodes with converters. In *alternate* routing, a set of paths (consisting of one primary path and one or more alternate paths) is assigned to each source–destination pair. This set is searched in a fixed order to find an available path for the call.

Once a path is selected, one of the (possibly many) free wavelengths in the path must then be assigned to the call. We only consider the random wavelength assignment policy, whereby a call is allocated one of the available wavelengths at random. In a path with wavelength converters, a wavelength is randomly assigned within each segment of the path whose starting and ending nodes are equipped with converters.

We develop an iterative *path decomposition* algorithm [5] for computing call-blocking probabilities in wavelength-routing network. We analyze a given network by decomposing it into a number of path subsystems. To analyze each subsystem, we first develop an exact Markov process model for a path with and without converters. We then show how to slightly modify this process to obtain an approximate Markov process model which is time-reversible and which has a closed-form solution that resembles the product form solution in queueing networks [10]. The solution to the time-reversible Markov process provides an accurate approximation to the blocking probabilities obtained through the exact process. Because of computational requirements, both the exact and the approximate Markov process models can only be applied to relatively short paths. For longer paths, we then develop an iterative algorithm for computing the blocking probabilities by decomposing a path into a series of shorter segments connected in tandem. Once the path subsystems have been analyzed in isolation, the individual solutions are appropriately combined to form a solution for the overall network, and the process repeats until the blocking probabilities converge. Our approach accounts for the correlation of both link loads and link blocking events, giving accurate results for a wide range of loads and network topologies. Also, our algorithm can compute call-blocking probabilities in a mesh network where only a fixed but arbitrary subset of nodes are capable of wavelength conversion. Finally, the algorithm can be used to develop and evaluate converter placement strategies.

In Sections II and III we analyze a single path in wavelength-routing networks. In Section IV we present a path decomposition algorithm for analyzing mesh networks under both fixed and alternate routing. In Section V we validate our decomposition algorithm through simulation, and we also study the problem of converter placement. We conclude the paper in Section VI.

II. A Single Path in a Wavelength-Routing Network

In this section we present an exact and an approximate Markov process for a k-hop path. We first study paths without wavelength converters, and then we extend our results to paths where converters are employed at some nodes. The following notation will be used in this and the next section (refer to Fig. 1).

1) A k-hop path consists of $k+1$ nodes labeled $0, 1, \ldots, k$, and hop $i, i = 1, \ldots, k$ represents the link between nodes $i-1$ and i.
2) $\lambda_{ij}, j \geq i$, is the Poisson arrival rate of calls that use hops i through j, i.e., they originate at node $i-1$ and terminate at node j.
3) $1/\mu$ is the mean of the (exponentially distributed) holding time of all calls. Also, $\rho_{ij} = \lambda_{ij}/\mu$ is the offered load of calls using hops i through j.
4) $n_{ij}, j \geq i$, is the number of calls using hops i through j that are currently active in the network.
5) $f_{ij}, j \geq i$, is the number of wavelengths that are free on all hops i through j.

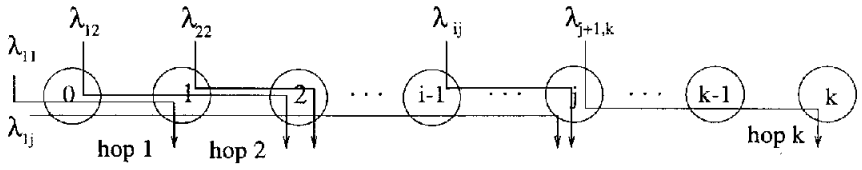


Fig. 1. A k-hop path.

A. Exact Markov Process Model for Paths with No Wavelength Conversion

Let us first consider the two-hop path (without converters) shown in Fig. 2. The evolution of this system can be characterized by the four-dimensional Markov process $(n_{11}, n_{12}, n_{22}, f_{12})$. Since, on each hop, the number of busy wavelengths plus the number of wavelengths that are free on both hops may not exceed the number W of available wavelengths, the following two constraints must be satisfied:

$$n_{11} + n_{12} + f_{12} \le W \quad \text{and} \quad n_{12} + n_{22} + f_{12} \le W. \tag{1}$$

The above result can be generalized to k-hop paths, $k > 2$. Let $\mathcal{M}_k$ denote the Markov process corresponding to a k-hop path. There are k^2 random variables in a state $\underline{n}$ of Markov process $\mathcal{M}_k$, as follows:

$$\underline{n} = (n_{11}, n_{12}, \ldots, n_{1k}, n_{22}, \ldots, n_{2k}, \ldots, n_{kk}, f_{12}, f_{13}, \ldots, f_{1k}, f_{23}, \ldots, f_{2k}, \ldots, f_{k-1,k}). \tag{2}$$

The first $(k(k+1)/2)$ random variables n_{ij}, $1 \le i \le j \le k$, in the state description (2) provide the number of active calls between all possible source–destination pairs in the path. The last $((k-1)k/2)$ random variables f_{ij}, $1 \le i < j \le k$, represent the number of wavelengths that are free on all segments of the path consisting of two or more links. The following constraints are imposed on the state space of Markov process $\mathcal{M}_k$:

$$f_{ij} \le f_{i,j-1} \le \cdots \le f_{i,i+1}, \qquad 1 \le i < j \le k \tag{3}$$

$$f_{ij} \le f_{i+1,j} \le \cdots \le f_{j-1,j}, \qquad 1 \le i < j \le k \tag{4}$$

$$\begin{cases} \displaystyle\sum_{j=1}^{k} n_{1j} + f_{12} \le W \\ \displaystyle\sum_{i=1}^{l}\sum_{j=l}^{k} n_{ij} + f_{l-1,l} + f_{l,l+1} - f_{l-1,l+1} \le W, \\ \qquad l = 2, \cdots, k-1 \\ \displaystyle\sum_{i=1}^{k} n_{ik} + f_{k-1,k} \le W \end{cases} \tag{5}$$

The set of constraints (3) [respectively, (4)] account for the fact that if a wavelength is free on all hops of an m-hop segment, then it is also free on the first (respectively, last) $m-1$ hops of the segment, while the k constraints (5) ensure that the number of wavelengths (used or free) on each hop does not exceed W.

Process $\mathcal{M}_k$ captures the correlation of wavelength use on all links of a k-hop path, and it can be used to provide an exact solution for the probability that a call request will be blocked.

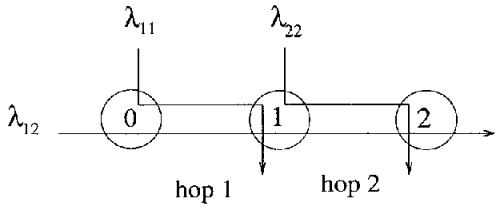


Fig. 2. Two-hop path.

Unfortunately, the large number of random variables in its state description makes it impossible to numerically solve it for anything but very short paths and small values of W. In addition, the transition rates of Markov process $\mathcal{M}_k$ are state-dependent. In Fig. 3 we show the transition diagram of Markov process $\mathcal{M}_2$ for $W = 2$ wavelengths. From this figure we can see that there exists a sequence of states, $\underline{n_1}, \ldots, \underline{n_s}$, such that

$$r(\underline{n_1}, \underline{n_2})r(\underline{n_2}, \underline{n_3}) \cdots r(\underline{n_{s-1}}, \underline{n_s})r(\underline{n_s}, \underline{n_1}) \neq r(\underline{n_1}, \underline{n_s})r(\underline{n_s}, \underline{n_{s-1}}) \cdots r(\underline{n_3}, \underline{n_2})r(\underline{n_2}, \underline{n_1}) \tag{6}$$

where $r(\underline{n}, \underline{n'})$ is the transition rate from state $\underline{n}$ to state $\underline{n'}$. Two such sequences of states are: $(1,1,1,0)$, $(1,0,1,1)$, $(1,0,0,1)$, $(1,1,0,0)$, and $(1,1,1,0)$, $(1,0,1,1)$, $(0,0,1,1)$, $(0,1,1,0)$. Therefore, Kolmogorov's criterion for reversibility [9], which states that a process is time-reversible if and only if (6) holds with equality for any and all sequences $\underline{n_1}, \ldots, \underline{n_s}$, is not satisfied. It is straightforward to show that this result is true in general, and that process $\mathcal{M}_k$, $k \ge 2$ is not time-reversible when $W > 1$.

B. Approximate Time-Reversible Markov Process Model

A closer examination of Fig. 3 reveals that the two four-state sequences mentioned above are the shortest sequences of states for which (6) holds true. We also note that these two sequences involve transitions that cause changes in the value of random variable n_{12}. Let us define $\mathcal{L}_{2,c}$ as the sub-chain of Markov process $\mathcal{M}_2$ that includes only the states for which the value of random variable n_{12} is constant, i.e., $n_{12} = c$:

$$\mathcal{L}_{2,c} = \{(n_{11}, n_{12}, n_{22}, f_{12}) \,|\, n_{12} = c\}, \qquad c = 0, \ldots, W. \tag{7}$$

Sub-chain $\mathcal{L}_{2,c}$ corresponds to a new system with $W - c$ wavelengths per hop, in which no calls using both hops ever arrive (that is, $\lambda_{12} = 0$ in this new system). Then, it can be easily verified that Kolmogorov's criterion for reversibility is satisfied by any sequence of states in sub-chain $\mathcal{L}_{2,c}$.

On the other hand, let us consider the four-state sequence $\underline{n_1} = (n_{11}, n_{12}, n_{22}, f_{12})$, $\underline{n_2} = (n_{11}, n_{12}+1, n_{22}, f_{12}-1)$, $\underline{n_3} = (n_{11}+1, n_{12}+1, n_{22}, f_{12}-1)$, and $\underline{n_4} = (n_{11}+$

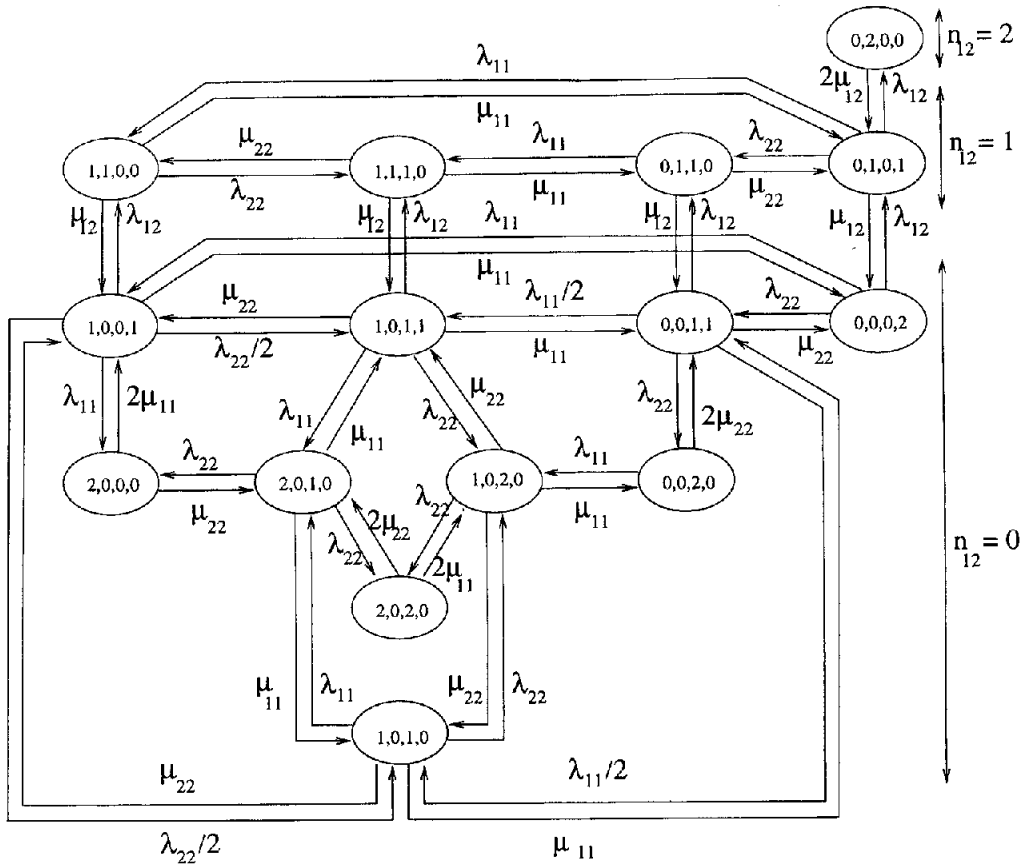


Fig. 3. State space $(n_{11}, n_{12}, n_{22}, f_{12})$ of a two-hop path with $W = 2$ wavelengths.

$1, n_{12}, n_{22}, f_{12})$, shown in Fig. 4, which includes states from two different sub-chains. We have that

$$r(\underline{n}_1, \underline{n}_2) = \lambda_{12}, \qquad r(\underline{n}_3, \underline{n}_4) = (n_{12}+1)\mu_{12},$$
$$r(\underline{n}_2, \underline{n}_1) = (n_{12}+1)\mu_{12}, \qquad r(\underline{n}_4, \underline{n}_3) = \lambda_{12} \tag{8}$$

so these transition rates balance along the two directions. However, the rates of the other transitions do not balance, since

$$\begin{aligned} r(\underline{n}_2, \underline{n}_3) &= \lambda_{11}\left(1 - \frac{f_{12}-1}{W-(n_{12}+1)-n_{11}}\right) \\ r(\underline{n}_4, \underline{n}_1) &= (n_{11}+1)\mu_{11}\left(1 - \frac{W-f_{12}-n_{12}-n_{22}}{n_{11}+1}\right) \\ r(\underline{n}_3, \underline{n}_2) &= (n_{11}+1)\mu_{11} \\ &\quad \times \left(1 - \frac{W-(n_{12}+1)-(f_{12}-1)-n_{22}}{n_{11}+1}\right) \\ r(\underline{n}_1, \underline{n}_4) &= \lambda_{11}\left(1 - \frac{f_{12}}{W-n_{11}-n_{12}}\right). \end{aligned} \tag{9}$$

Process $\mathcal{M}_2$ is not time-reversible due to transitions between states with different values of n_{12}.

Returning to Fig. 3, we note that if the transition rate from state $(1, 0, 1, 1)$ to state $(1, 1, 1, 0)$ is changed to $2\lambda_{12}$ (from λ_{12}), then the Markov process becomes time-reversible and has a closed-form solution. However, when each hop supports more than two wavelengths, a larger number of transition rates must be modified to yield a time-reversible Markov process. The rule for changing the transition rates is as follows.

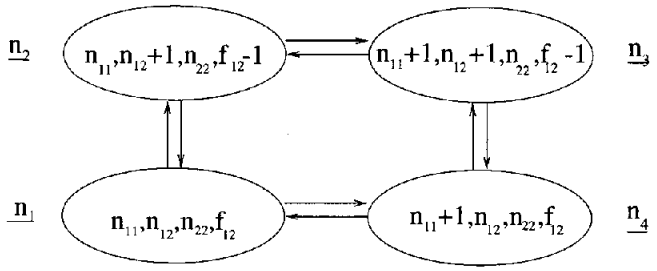


Fig. 4. Four-state sequence with states from two different sub-chains $\mathcal{L}_{2,c}$.

Consider all states $\underline{n} = (n_{11}, c, n_{22}, f_{12})$ of sub-chain $\mathcal{L}_{2,c}$ with $n_{11} > 0$ and $n_{22} > 0$, for which there is a transition from state $\underline{n}$ to a state $\underline{n}' = (n_{11}, c+1, n_{22}, f_{12}-1)$ of sub-chain $\mathcal{L}_{2,c+1}$ with rate $r(\underline{n}, \underline{n}') = \lambda_{12}$. If these transition rates are changed to

$$\begin{aligned} r'(\underline{n}, \underline{n}') &= \lambda_{12}\frac{f_{12}(W-c)}{f_{11}f_{22}} \\ &= \lambda_{12}\frac{f_{12}(W-c)}{(W-n_{11}-c)(W-n_{22}-c)} \end{aligned} \tag{10}$$

then we obtain a new Markov process which is time-reversible.

In fact, it is straightforward to show that if the transition rate $r(\underline{n}_1, \underline{n}_2)$ in (8) is modified according to this rule, then the four-state sequence in Fig. 4 will satisfy Kolmogorov's criterion for reversibility.

The above results can be generalized to a k-hop path, $k \geq 2$. Consider a sub-chain $\mathcal{L}_{k,\underline{c}}$ of $\mathcal{M}_k$ which includes all states of the process for which the number of active calls using two or more hops is constant:

$$\mathcal{L}_{k,\underline{c}} = \{\underline{n} \in \mathcal{M}_k \,|\, n_{ij} = c_{ij} = \text{constant},\ i < j\},$$
$$\underline{c} = (c_{12}, \ldots, c_{1k}, c_{23}, \ldots, c_{2k}, \ldots, c_{k-1,k}). \quad (11)$$

Sub-chain $\mathcal{L}_{k,\underline{c}}$ models a k-hop path in which there are no arrivals of calls using two or more hops (i.e., $\lambda_{ij} = 0$ for $i < j$), and in which each hop supports a fixed number of wavelengths (that can be different from the number of wavelengths supported by other hops). Then, it is straightforward to verify that any sequence of states in sub-chain $\mathcal{L}_{k,\underline{c}}$ satisfies Kolmogorov's criterion for reversibility, but there exist sequences that include states from other sub-chains that violate this criterion.

A time-reversible Markov process $\mathcal{M}'_k$ can be derived from $\mathcal{M}_k$ as follows. The new process $\mathcal{M}'_k$ has the same state space and the same transitions as $\mathcal{M}_k$. The vast majority of its transition rates are the same as the respective transition rates of $\mathcal{M}_k$. However, to ensure that the new process is time-reversible, the transition rates between some pairs of states must be appropriately modified. Consider the states $\underline{n}$ of sub-chain $\mathcal{L}_{k,\underline{c}}$ for which there exists a transition with rate λ_{lm}, $l < m$, to a state $\underline{n}'$ of another sub-chain $\mathcal{L}_{k,\underline{c}'}$, $c \neq c'$:

$$\mathcal{N}_{k,\underline{c}} = \{\underline{n} \in \mathcal{L}_{k,\underline{c}} \,|\, \exists\, i, j, l, m,\ l \leq i < j \leq m,$$
$$n_{ii} > 0,\ n_{jj} > 0,\ n_{lm} < W,\ f_{lm} > 0\}. \quad (12)$$

The transition rate $r(\underline{n}, \underline{n}') = \lambda_{lm}$ in process $\mathcal{M}_k$. In process $\mathcal{M}'_k$ the transition rate is changed to

$$r'(\underline{n}, \underline{n}') = \lambda_{lm} f_{lm} \frac{\left(\sum_{i=1}^{l} n_{il} + f_{ll}\right)}{f_{ll}} \times \frac{\left(\sum_{i=1}^{l+1} n_{i,l+1} + f_{l+1,l+1}\right)}{f_{l+1,l+1}} \cdots \times \frac{\left(\sum_{i=1}^{m-1} n_{i,m-1} + f_{m-1,m-1}\right)}{f_{mm}}. \quad (13)$$

Markov process $\mathcal{M}'_k$ has a closed-form solution for its steady-state probability that resembles the product form solution in queueing networks [10]. Let $G_k(W)$ denote the normalizing constant for a k-hop path with W wavelengths per link. Then, the solution of Markov process $\mathcal{M}'_2$ corresponding to a two-hop path is[1]

$$\pi(n_{11}, n_{12}, n_{22}, f_{12}) = \frac{1}{G_2(W)} \frac{\rho_{11}^{n_{11}} \rho_{12}^{n_{12}} \rho_{22}^{n_{22}}}{n_{11}! n_{12}! n_{22}!} \times \frac{\binom{f_{11}}{f_{12}}\binom{n_{11}}{f_{22}-f_{12}}}{\binom{n_{11}+f_{11}}{f_{22}}} \quad (14)$$

where $f_{11} = W - n_{11} - n_{12}$ and $f_{22} = W - n_{22} - n_{12}$, while the solution to Markov process $\mathcal{M}'_3$ corresponding to a three-hop path with state description $\underline{n} = (n_{11}, n_{12}, n_{13}, n_{22}, n_{23}, n_{33} f_{12}, f_{13}, f_{23})$ is given by

$$\pi(\underline{n}) = \frac{1}{G_3(W)} \frac{\rho_{11}^{n_{11}} \rho_{12}^{n_{12}} \rho_{13}^{n_{13}} \rho_{22}^{n_{22}} \rho_{23}^{n_{23}} \rho_{33}^{n_{33}}}{n_{11}! n_{12}! n_{13}! n_{22}! n_{23}! n_{33}!} \times \frac{\binom{f_{11}}{f_{12}}\binom{n_{11}}{f_{22}-f_{12}}}{\binom{n_{11}+f_{11}}{f_{22}}} \times \frac{\binom{f_{12}}{f_{13}}\binom{n_{22}+n_{12}+f_{22}-f_{12}}{f_{33}-f_{13}}}{\binom{n_{22}+n_{12}+f_{22}}{f_{33}}} \times \frac{\binom{f_{22}-f_{12}}{f_{23}-f_{13}}\binom{n_{22}+n_{12}}{f_{33}-f_{23}}}{\binom{n_{22}+n_{12}+f_{22}-f_{12}}{f_{33}-f_{13}}} \quad (15)$$

where $f_{11} = W - n_{11} - n_{12} - n_{13}$, $f_{22} = W - n_{12} - n_{22} - n_{13} - n_{23}$, and $f_{33} = W - n_{13} - n_{23} - n_{33}$.

We can write down the solution to any Markov process $\mathcal{M}'_k$, $k > 3$, by a straightforward generalization of (14) and (15). Specifically, the solution to $\mathcal{M}'_k$, for any k, is the product of k^2 terms as follows. The first $k(k+1)/2$ terms are of the form $\rho_{ij}^{n_{ij}}/n_{ij}$, and each corresponds to one of random variables n_{ij} in the state description (2). The last $k(k-1)/2$ terms are combinatorial terms, each corresponding to one of the dependent variables f_{ij} in the state description (2).

The significance of the new Markov process $\mathcal{M}'_k$ will be illustrated in Section V and in Figs. 9 and 10, where it will be shown that the blocking probabilities obtained through the closed-form solution of $\mathcal{M}'_k$ closely approximate the exact blocking probabilities obtained through the numerical solution of $\mathcal{M}_k$. In order

[1]The closed-form solution (14) is similar to the one presented in [14]. However, there are several important differences in the two approaches. First, the solution in [14] was derived by considering a three-dimensional Markov process (n_{11}, n_{12}, n_{22}), while as we have shown, the fourth parameter f_{12} is necessary to completely characterize a two-hop path. Second, we have shown that the closed-form solution (14) is the exact solution to an approximate Markov process; in contrast, in [14, Section II-C] this result was derived as a solution to the original two-hop problem, without explicitly stating that it is an approximation. Finally, whereas only a two-hop path was studied in [14], our approach is far more general, and it leads to an approximate closed-form solution for any k-hop path, $k \geq 2$ [see also (15)].

for the closed-form solution to be useful, we need to have a computationally efficient procedure for calculating the normalizing constant. Using brute-force enumeration, we can calculate the closed-form solution of $\mathcal{M}'_k$ for up to 25 wavelengths when the number of hops is $k = 3$, and for up to eight wavelengths when $k = 4$. (As a comparison, for $k = 3$, we can obtain the exact numerical solution of $\mathcal{M}_k$ for only up to $W = 4$.) The computation of the normalizing constant in time that is polynomial in W and k has turned out to be a very difficult task. In view of this, an iterative decomposition algorithm was devised for paths longer than four hops. This algorithm is described in Section III and it can be used to obtain the blocking probabilities for paths of arbitrary length in an efficient manner.

C. Paths with Wavelength Conversion

We now turn our attention to paths in which wavelength converters are employed at some nodes. Let us again refer to Fig. 2, and let us assume that wavelength converters are located at node 1 (the only interesting possibility in this case). We immediately see that the two-hop system can now be described by the three-dimensional Markov process (n_{11}, n_{12}, n_{22}). Random variable f_{12} is no longer necessary because wavelength continuity is not required, and calls continuing on both hops can now use *any* of the $(W - n_{12} - n_{22})$ available wavelengths on the second hop. In other words, the two-hop system with a converter at node 1 becomes equivalent to a two-hop circuit-switched path.

In the general case, consider a k-hop path, $k \geq 2$, with converters employed at one or more nodes. This path can be modeled by a new Markov process which is simpler than $\mathcal{M}_k$. The new Markov process has the same $(k(k+1)/2)$ random variables n_{ij} as $\mathcal{M}_k$, but some of the variables f_{ij} are no longer present in the state description. Specifically, let us consider the case when a converter is employed at node l, $0 < l < k$, of the path. Then, variables f_{ij}, $i \leq l < j$, which are required for $\mathcal{M}_k$, are not part of the state description of the new Markov process. Because of the converter at node l, a call using hops i through j can now be completed as long as there is at least one free wavelength on hops i through l, and at least one free wavelength on hops $l+1$ through j. Therefore, random variables f_{il} and $f_{l+1,j}$ (which remain part of the state description) provide all the information needed to determine whether the call can be completed, making f_{ij} redundant.

It is now straightforward to show that the Markov process for a k-hop path that employs wavelength converters is not time-reversible, except when converters are employed at *all* internal nodes of the path (a circuit-switched scenario). We can then use an approach similar to the one we followed in Section II-B to modify some of the transition rates of this process in order to obtain an approximate, time-reversible Markov process which has a closed-form solution.

III. Decomposition Algorithm for Long Paths

Let K denote the largest integer such that the closed-form solution to Markov process $\mathcal{M}'_K$ can be computed within a reasonably short amount of time. Consider a k-hop path. If $k \leq K$, the path can be analyzed approximately by solving the corresponding Markov process $\mathcal{M}'_k$. If, on the other hand, $k > K$, the approximate closed-form solution cannot be used directly. In this section, we develop an iterative decomposition algorithm to analyze paths of length greater than K.

A. Paths with No Wavelength Conversion

We analyze a k-hop path, $k = lK + m$, $l \geq 1$, $m < K$, by decomposing it into one m-hop segment and l K-hop segments in tandem. Each segment is first analyzed in isolation using the corresponding Markov process $\mathcal{M}'_m$ or $\mathcal{M}'_K$. The arrival rates of calls originating in a segment but terminating in another segment are accounted for by increasing the arrival rate of calls in the individual segments. The individual solutions are appropriately combined to obtain an initial value for the blocking probability of calls that traverse more than one segment. Using these initial estimates, the arrival rates to each segment are modified and each segment is again solved in isolation in order to obtain a new solution. These new individual solutions are again combined to update the blocking probability of calls traversing multiple segments. This is repeated until the blocking probabilities converge.

A summary of our iterative algorithm is provided in Fig. 5. Below, we describe the decomposition algorithm using the four-hop path shown in Fig. 6(a). This path is decomposed into two two-hop segments, namely segment 1 and segment 2, as shown in Fig. 6(b). Segment 1 consists of nodes 0, 1, and 2, and segment 2 consists of nodes 2, 3, and 4. Let λ_{ij}, $1 \leq i \leq j \leq 4$, be the arrival rates of calls to the original four-hop path, and let $\lambda_{11}^{(1)}$, $\lambda_{12}^{(1)}$, $\lambda_{22}^{(1)}$ and $\lambda_{11}^{(2)}$, $\lambda_{12}^{(2)}$, $\lambda_{22}^{(2)}$ denote the arrival rates of calls in the first and second segments, respectively. The interpretation of the arrival rates in the two segments is somewhat different under our decomposition algorithm. Specifically, $\lambda_{12}^{(1)}$ accounts for all the calls in the original four-hop path that originate at node 0 and terminate at nodes 2 or higher; similarly for $\lambda_{22}^{(1)}$. On the other hand, $\lambda_{12}^{(2)}$ accounts for all calls in the original path that originate at nodes 2 and lower and terminate at node 4; similarly for $\lambda_{11}^{(2)}$. The main steps of our algorithm are as follows.

Initially, we solve the first segment in isolation using

$$\lambda_{12}^{(1)} = (1 - q_{14})\lambda_{14} + (1 - q_{13})\lambda_{13} + \lambda_{12} \tag{16}$$

$$\lambda_{22}^{(1)} = (1 - q_{24})\lambda_{24} + (1 - q_{23})\lambda_{23} + \lambda_{22} \tag{17}$$

$$\lambda_{11}^{(1)} = \lambda_{11}. \tag{18}$$

Quantity $q_{ij}, 1 \leq i \leq 2 < j \leq 4$, represents the current estimate of the conditional probability that a call using hops i through j (where i lies within the first segment and j lies within the second segment) will be blocked in the second segment given that a free wavelength for the call exists within the first segment. For the first iteration, we use $q_{ij} = 0$ for all i and j; how these values are updated in subsequent iterations will be described shortly. Thus, the term $(1 - q_{14})\lambda_{14}$ in (16) represents the *effective* arrival rate of calls using all four hops, as seen by the first segment; similarly for the term $(1 - q_{13})\lambda_{13}$. Equation (17) for $\lambda_{22}^{(1)}$ includes similar terms that account for the effective arrival rate of calls which originate at node 1 and terminate at nodes 2 or higher. Equation (18) for $\lambda_{11}^{(1)}$ does not include any such terms, since this type of

Decomposition Algorithm for Paths Without Converters

A k-hop path, $k = K + m, m \leq K$, is decomposed into a K-hop segment (segment 1) and an m-hop segment (segment 2). Segment 1 consists of nodes 0 to K, and segment 2 consists of nodes K to $K = m$ of the original path. λ_{ij} refer to the call arrival rates in the original path, whereas $\lambda_{ij}^{(n)}$ refer to call arrival rates in segment $n, n = 1, 2$.

1. begin
2. $h \leftarrow 0$ //Initialization step
   // $p_{ij}^{(1)}(h)$ is the blocking probability of calls using hops i through j of segment 1
   // $F_{ij}^{(1)}$ is the average number of free wavelengths on hops i through j of segment 1
   $p_{ij}^{(1)}(h) \leftarrow 0, \quad F_{ij}^{(1)} \leftarrow W, \quad 1 \leq i \leq j \leq K$
   // $p_{ij}^{(2)}(h)$ is the blocking probability of calls using hops i through j of segment 2
   // $F_{ij}^{(2)}$ is the average number of free wavelengths on hops i through j of segment 2
   $p_{ij}^{(2)}(h) \leftarrow 0, \quad F_{ij}^{(2)} \leftarrow W, \quad 1 \leq i \leq j \leq m$
   // $q_{ij}(h)$ is the conditional probability that an inter-segment call will be blocked in
   // segment 2, given that it has found a free wavelength in segment 1
   $q_{ij}(h) \leftarrow 0, \quad 1 \leq i \leq K < j \leq K + m$
3. $h \leftarrow h + 1$ //h-th iteration
4. $\lambda_{ij}^{(1)}(h) \leftarrow \lambda_{ij}, \quad 1 \leq i \leq j < K$ //Segment 1
   $\lambda_{iK}^{(1)}(h) \leftarrow \lambda_{iK} + \sum_{j=K+1}^{K+m} \lambda_{ij}\,(1 - q_{ij}(h-1)), \quad 1 \leq i \leq K$
   // include the effective arrival rate of calls continuing to segment 2
   Solve segment 1 to obtain new values for $p_{ij}^{(1)}(h)$ and $F_{ij}^{(1)}(h)$
5. $\lambda_{ij}^{(2)}(h) \leftarrow \lambda_{K+i,K+j}, \quad 1 < i \leq j \leq m$ //Segment 2
   $\lambda_{1j}^{(2)}(h) \leftarrow \lambda_{K+1,K+j} + \sum_{i=1}^{K} \lambda_{i,K+j} \left(1 - p_{iK}^{(1)}(h-1)\right), \quad 1 \leq j \leq m$
   // include the effective arrival rate of calls continuing from segment 1
   Solve segment 2 to obtain new values for $p_{ij}^{(2)}(h)$ and $F_{ij}^{(2)}(h)$
6. // Conditional blocking probability of inter-segment calls
   $q_{ij}(h) \leftarrow p_{1,j-K}^{(2)}(h) + \left(1 - p_{1,j-K}^{(2)}(h)\right) Q_{ij}(h), 1 \leq i \leq K < j \leq K + m,$
   with Q_{ij}(h) similar to Q_{ij} in expression (23)
7. Repeat from Step 3 until the blocking probabilities converge
8. end of the algorithm

Fig. 5. Decomposition algorithm for long paths.

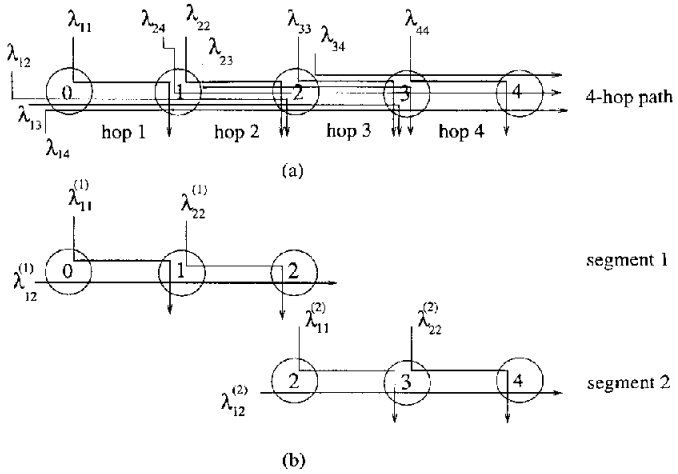


Fig. 6. (a) Four-hop path. (b) Its decomposition into two two-hop segments in tandem.

calls in the first segment do not involve calls in the original path that continue on to the second segment.

The solution to the first segment yields an initial value for the probability $p_{ij}^{(1)}, 1 \leq i \leq j \leq 2$, that a call using hops i through j of the first segment will be blocked within the segment. Therefore, the effective arrival rate of calls originating at node 0 and terminating at node 4 that is offered to the second segment can be initially estimated as $\lambda_{14}(1 - p_{12}^{(1)})$, while the effective rate of calls originating at node 1 and terminating at node 4 can be estimated as $\lambda_{24}(1 - p_{22}^{(1)})$. We can now solve the second segment using

$$\lambda_{12}^{(2)} = \lambda_{14}\left(1 - p_{12}^{(1)}\right) + \lambda_{24}\left(1 - p_{22}^{(1)}\right) + \lambda_{34} \quad (19)$$

$$\lambda_{11}^{(2)} = \lambda_{13}\left(1 - p_{12}^{(1)}\right) + \lambda_{23}\left(1 - p_{22}^{(1)}\right) + \lambda_{33} \quad (20)$$

$$\lambda_{22}^{(2)} = \lambda_{44}. \quad (21)$$

Based on the above discussion, $\lambda_{12}^{(2)}$ in (19) represents the effective arrival rate of calls using the last two hops of the four-hop path, as seen by the second segment. Equation (20) for $\lambda_{11}^{(2)}$ can be explained using similar arguments. Equation (21) for $\lambda_{22}^{(2)}$ contains only one term since, as seen in Fig. 6, it does not involve calls that originate in segment 1. The solution to the second segment provides an estimate of the blocking probabilities $p_{ij}^{(2)}$, $1 \leq j \leq 2$, of calls traversing hops 1 and 2 of the second segment (i.e., hops 3 and 4 of the original path).

We can now obtain new values for the conditional blocking probabilities q_{ij}, $1 \leq i \leq 2 < j \leq 4$, used in (16) and (17), as follows. Consider a call using hops i through j, where i lies in the first segment and j lies in the second segment. Given that at least one free wavelength exists on hops i through 2 (i.e., the call successfully makes it through the first segment), the call will be blocked if

1) there is no free wavelength in the links it uses in the second segment, or
2) there do exist free wavelengths in the second segment, but they are not the same as the free wavelengths in the first two hops.

The probability of the first event occurring is equal to $p_{1,j-2}^{(2)}$, which is obtained through the solution of the second segment. The probability of the second event is equal to $(1-p_{1,j-2}^{(2)})Q_{ij}$, where parameter Q_{ij} represents blocking due to the wavelength continuity requirement for calls using hops i through j, where i lies in the first segment and j lies in the second segment. Probability Q_{ij} cannot be computed exactly since each segment is solved independently of the other, and thus, it is not possible to determine whether a wavelength which is free in one segment will also be free in the other. An approximate value for this probability can be obtained as follows. Let $P_{i,2}^{(1)}[W=n]$ (respectively, $P_{1,j-2}^{(2)}[W=m]$) denote the probability that there are n (respectively, m) free wavelengths on the hops of the first (respectively, second) segment used by the call. Let $R(n, m)$ denote the conditional probability that there are no common wavelengths for the call to use in the two segments, given that there are n (respectively, m) free wavelengths on the hops it uses in the first (respectively, second) segment. Because of the random wavelength assignment policy, we have that

$$R(n,m) = \begin{cases} 0, & n=0 \text{ or } m=0 \\ 1, & n+m \geq W \\ \dfrac{\binom{W-n}{m}}{\binom{W}{m}}, & \text{otherwise.} \end{cases} \tag{22}$$

Then, we approximate the probability Q_{ij} of blocking due to the wavelength continuity requirement as

$$Q_{ij} = \sum_{n=1}^{W}\sum_{m=1}^{W}\left[P_{i2}^{(1)}[W=n]\times P_{1,j-2}^{(2)}[W=m]\times R(n,m)\right] \times \left[\frac{1}{2}\left(\frac{\lambda_{ij}}{\lambda_{i2}^{(1)}}+\frac{\lambda_{ij}}{\lambda_{1,j-2}^{(2)}}\right)\right]. \tag{23}$$

The double summation in the right-hand side of (23) is the probability that there are no common wavelengths for the call to use in the two segments, assuming that the two segments are independent. We have found experimentally that adjusting this probability by the last factor in (23) reduces the effect of the independence assumption and accurately approximates the probability of blocking due to the wavelength continuity requirement for a wide range of arrival rates. Note that the term $\lambda_{ij}/\lambda_{i2}^{(1)}$ in (23) represents the fraction of traffic requiring free wavelengths on hops i through 2 of the first segment that is due to calls on hops i through $j > 2$ in the original path [refer also to (16) and (17)]. Similarly, the term $\lambda_{ij}/\lambda_{1,j-2}^{(2)}$ in (23) represents the fraction of traffic requiring free wavelengths on hops 1 through $j-2$ of the second segment that is due to the calls under consideration. Hence, the last factor in (23) adjusts the blocking probability obtained through the independence assumption to capture the contribution of the calls using hops i through j of the original path.

The new estimates for q_{ij} are then used in (16)–(18) to update the arrival rates for the first segment. The first segment is then solved again, the estimates for $p_{ij}^{(1)}$ are updated and used in (19)–(21), and so on. We iterate in this fashion until the blocking probabilities for all calls in the original path converge within a certain tolerance. In all cases studied, we have found that the algorithm converges in only a few (less than ten) iterations even for long paths, and that the blocking probabilities obtained closely match simulation results (more on this in Section V).

The decomposition algorithm described above is similar in spirit to the decomposition algorithms developed for tandem queueing networks with finite-capacity queues (see [13]). This algorithm can be easily extended to handle paths decomposed into more than two segments. We note that when employing the decomposition algorithm, the selection of the segment size will depend on the following factors: length of the original path, how efficiently we can calculate the closed-form solution of the Markov process $\mathcal{M}'_k$ associated with each segment, and how accurate the approximate solution of $\mathcal{M}'_k$ is. It is well known in decomposition algorithms that the larger the individual subsystems that have to be analyzed in isolation, the better the accuracy of the decomposition algorithm. Thus, as we mentioned at the beginning of this subsection, we decompose a path in segments of the largest size K for which we can efficiently analyze the Markov process $\mathcal{M}'_K$, plus, possibly, a segment of smaller size, if the path length is not a multiple of K.

B. Paths with Wavelength Conversion

The iterative algorithm described above can also be used for paths with converters. We note, however, that the addition of $l < k$ converters leads to a natural decomposition of a k-hop path into $l+1$ segments, with each segment consisting of the links between successive nodes where converters are employed. Given such a decomposition, the blocking probability of calls spanning several segments now depends only on the number of calls within each segment (similar to the circuit-switching case), and *not* on the actual wavelengths used by those calls. Hence, the probability that a call spanning multiple segments will be successfully established becomes equal to the product of the probabilities of finding a free wavelength (not necessarily the same one) within each segment. Therefore, we decompose a path into

segments, each consisting of all the links between two successive nodes with converters. Each segment is then analyzed in isolation as described above. Specifically, if it is feasible, we analyze the segment's underlying approximate time-reversible Markov process $\mathcal{M}'_k$. Otherwise, we analyze it using the decomposition algorithm presented in Section III-A.

As an example, let us consider a k-hop path with a single converter located at node $K < k$. This path can be analyzed using the decomposition algorithm in Fig. 5 after making the following single modification: in Step 6, the expression for the conditional blocking probabilities is changed to $q_{ij} = p^{(2)}_{1,j-K}$. (The second term in this expression represents blocking due to the wavelength continuity requirement, and since $Q_{ij} = 0$ in this case, it drops out.) The decomposition algorithm can be extended in a straightforward way to handle paths with more than one converters.

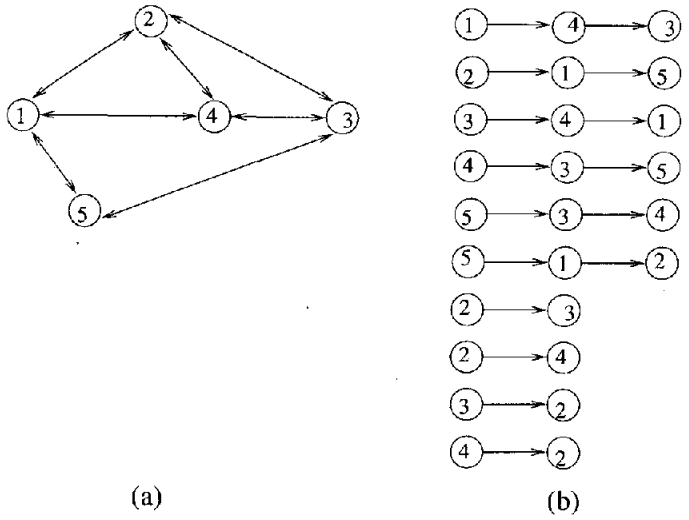


Fig. 7. (a) Original network. (b) Set $\mathcal{R}'$ of paths into which the network is decomposed.

IV. Path Decomposition Algorithm for Mesh Networks

A. Fixed Routing

We analyze a mesh network by decomposing it into a number of subsystems where each subsystem is a single path. Each subsystem is analyzed in isolation using the algorithms developed in Sections II and III. Specifically, subsystems consisting of three links or less are analyzed by solving the corresponding approximate time-reversible Markov process of Section II-B. Subsystems longer than three hops are analyzed using the iterative decomposition algorithm of Section III to obtain the call-blocking probabilities. The individual solutions obtained from the subsystems to which the mesh topology is decomposed are appropriately combined by modifying the call arrival rates to each subsystem to reflect the newly computed blocking probabilities. The process is repeated until all blocking probabilities converge within a prescribed tolerance.

Let $\mathcal{R}$ denote the set of paths assigned to the source–destination pairs, with $|\mathcal{R}| = N(N-1)$. The first step in analyzing a given network is to decompose it into a set $\mathcal{R}' \subseteq \mathcal{R}$ of paths such that: a) no path $r \in \mathcal{R}'$ is contained within a path $q \in \mathcal{R}$, $q \neq r$, and b) any path $q \in \mathcal{R}$ either belongs to $\mathcal{R}'$ or is completely contained within a path $r \in \mathcal{R}'$. These requirements ensure that a minimal set of subsystems that includes all possible paths is used. We can construct such a set $\mathcal{R}'$ by using the following steps. First, the paths in $\mathcal{R}$ are sorted in a list in order of decreasing length. The first path r in the list is removed and inserted in $\mathcal{R}'$. Then, any sub-paths of r that are also in the list are removed from it. The process continues with the next path in the list and is repeated until the list becomes empty. It is straightforward to show that this algorithm will construct a set $\mathcal{R}'$ which satisfies the above two properties. Fig. 7(b) shows the set of subsystems $\mathcal{R}'$ obtained by applying this algorithm to the network of Fig. 7(a). As we can see, while there are 20 source–destination pairs and corresponding paths in the network, only ten-path subsystems are used. For instance, the blocking probability on the path from, say, node 1 to node 4, will be obtained as a byproduct of the solution to the subsystem corresponding to the path from node 1 to node 3.

Once the set $\mathcal{R}'$ of subsystems has been selected, for each path $r \in \mathcal{R}'$ we need to determine the set of paths $\mathcal{S}(r) \subseteq \mathcal{R}$ that intersect (i.e., have at least one link in common) with path r. As an example, path $(1,4,3)$ in Fig. 7 intersects with path $(4,3,5)$. The significance of set $\mathcal{S}(r)$ lies in the fact that the blocking probability experienced by calls using the links of path r may be affected by the calls using the links of a path $q \in \mathcal{S}(r)$, and vice versa. Thus, when we compute the solution to path r, we must appropriately modify the call arrival rates along this path to account for the effect of calls along paths that intersect with r.

We are now ready to present the decomposition algorithm used for the analysis of a wavelength-routing network with an arbitrary topology. A detailed description of our decomposition algorithm is provided in Fig. 8. We will illustrate the operation of the algorithm using the network shown in Fig. 7. We will show how to update the arrival rates along each path subsystem after each iteration of the algorithm by considering only paths $(1,4,3)$ and $(4,3,5)$. The other path subsystems are handled in a similar way. Without loss of generality, we assume that shortest paths are used for fixed routing in this network. Let λ_{sd} be the arrival rate for source–destination pair (s, d). We also let $\hat{\lambda}_{sd}$ denote the arrival rates used to solve the various path subsystems. As explained next, the rate $\hat{\lambda}_{sd}$ accounts for all calls of the original network that use the links between nodes s and d within a path r, and is updated at the beginning of each iteration of the algorithm.

Initially, we solve path $(1,4,3)$ in Fig. 7 in isolation using these arrival rates:

$$\hat{\lambda}_{14} = \lambda_{14} \tag{24}$$

$$\hat{\lambda}_{13} = \lambda_{13} \tag{25}$$

$$\hat{\lambda}_{43} = \lambda_{43} + (1 - P_{45})\lambda_{45}. \tag{26}$$

We note that only calls from node 1 to node 4 use link $(1,4)$ of path $(1,4,3)$, thus, the arrival rate of calls using this link as seen by the path subsystem $(1,4,3)$ is given in (24). Similarly, (25) can be explained by the fact that only calls from node 1 to node 3 use both links of subsystem $(1,4,3)$. On the other hand, (26) for $\hat{\lambda}_{43}$ is slightly different because, in addition to calls from node 4 to node 3, calls from node 4 to node 5 also use

Decomposition Algorithm for Mesh Networks with Fixed Routing
Input: Network topology, set $\mathcal{R}$ of paths for all source-destination pairs, and arrival rates λ_{sd}
Output: Call blocking probabilities P_{sd} for all source-destination pairs in the network

1. begin
2. From $\mathcal{R}$ construct the set of path sub-systems $\mathcal{R}'$ into which the network will be decomposed, as described in Section IV-A
3. For each $r \in \mathcal{R}'$ construct the set $\mathcal{S}(r) = \{q \in \mathcal{R}' \mid q \text{ intersects with } r\}$
4. $h \leftarrow 0$ // Initialization step
   $P_{sd}(h) \leftarrow 0 \ \forall\, s, d$ // All blocking probabilities initialized to zero
5. $h \leftarrow h + 1$ // h-th iteration
   For each path $r = (r_1, r_2, \cdots, r_k) \in \mathcal{R}'$ do // compute the arrival rates for this iteration
   For each path $q = (q_1, \cdots, r_i, \cdots, r_j, \cdots, q_m) \in \mathcal{S}(r)$ that intersects with r from node r_i to r_j do
   // Calls using path q affect the blocking probability of calls using path r; the call arrival rate
   // seen by path r must be increased appropriately to account for the effect of these calls
   $\hat{\lambda}_{r_i,r_j}(h) \leftarrow \hat{\lambda}_{r_i,r_j}(h) + (1 - P_{q_1,q_m}(h-1))\lambda_{q_1,q_m}$
   Solve each path sub-system $r \in \mathcal{R}'$ using the algorithms in the previous section to obtain new values for the blocking probabilities $P_{sd}(h)$
7. Repeat from Step 5 until the blocking probabilities converge
8. end of the algorithm

Fig. 8. Path decomposition algorithm.

the second link of path $(1, 4, 3)$ since paths $(1, 4, 3)$ and $(4, 3, 5)$ intersect. Quantity P_{45} in (26) represents the current estimate of the probability that a call from node 4 to node 5 will be blocked on subsystem $(4, 3, 5)$. For the first iteration, we use $P_{45} = 0$; how this value is updated in subsequent iterations will be discussed shortly. Therefore, the term $(1 - P_{45})\lambda_{45}$ represents the *effective* arrival rate of calls from node 4 to node 5 as seen by subsystem $(1, 4, 3)$, since a fraction $P_{45}\lambda_{45}$ of these calls will be blocked in subsystem $(4, 3, 5)$. Consequently, the right-hand side of (26) is the effective arrival rate of calls that use the link $(4, 3)$ of path $(1, 4, 3)$ when the latter is considered in isolation.

We also solve path $(4, 3, 5)$ in isolation by using the following arrival rates:

$$\hat{\lambda}_{45} = \lambda_{45} \tag{27}$$

$$\hat{\lambda}_{35} = \lambda_{35} \tag{28}$$

$$\hat{\lambda}_{43} = \lambda_{43} + (1 - P_{13})\lambda_{13}. \tag{29}$$

Equations (27)–(29) can be explained using arguments similar to the ones used for (24)–(26). In particular, the second term in the right-hand side of (29) represents the effective arrival rate of calls originating in subsystem $(1, 4, 3)$ and using the link $(4, 3)$ of subsystem $(4, 3, 5)$.

The solution to the path subsystems $(1, 4, 3)$ and $(4, 3, 5)$ will yield an initial value for the probabilities P_{45} and P_{13} that a call using links $(3, 5)$ and $(1, 4)$, respectively, will be blocked. The new estimates for P_{45} and P_{13} are then used in (26) and (29), respectively, to update the arrival rates for the two path subsystems, the subsystems are solved again and new estimates for the blocking probabilities are obtained, and so on. We repeat the process until the blocking probabilities for all calls in the original network converge within a certain tolerance. In all the cases we have studied, we have found that the algorithm converges in only a few iterations, and that the blocking probabilities obtained closely match simulation results.

B. Alternate Routing

In order to improve the call-blocking performance, a source–destination pair (s, d) may be assigned m paths (one primary and $m - 1$ alternates) which are searched in a fixed order. In common implementations, the m shortest paths from s to d in the physical topology are used. If a call is blocked on the primary path, the first alternate path is examined. If available wavelengths exist on this path, the call is established. Otherwise, the next alternate path is examined, and so on. In other words, the traffic offered to alternate path i, $i = 2, \cdots, m$, is the *overflow* traffic from path $i - 1$. The call is blocked if no free wavelength can be found on any of the m paths, i.e., if it overflows from the last alternate path.

Although the traffic offered to the primary path for source–destination pair (s, d) is Poisson with rate λ_{sd}, it is clear that the overflow traffic offered to the alternate paths is not Poisson. The overflow traffic model has been studied extensively in the literature, and moment matching techniques have been used to characterize the overflow traffic model in circuit-switched networks with alternate routing [5]. Overflow models have also been used in the study of blocking probabilities in wavelength-routing networks in [8], [12]. In this paper we make the assumption that overflow traffic is also Poisson with an appropriate rate. This assumption permits us to use the algorithm developed in the previous subsection to analyze networks with alternate routing. Despite this assumption, we have found that the iterative path decomposition approach is

quite accurate for both regular and irregular topologies, and for a wide range of traffic loads.

We will now describe our approach to computing call-blocking probabilities in networks with alternate routing by assuming that there is one primary and one alternate path per source–destination pair. This approach can be easily extended to handle a larger number of alternate paths, as well as situations where the various source–destination pairs are assigned a different number of alternate paths.

Let $\mathcal{R}$ denote the set of primary and alternate paths for all node pairs, with $|\mathcal{R}| = 2N(N-1)$. From $\mathcal{R}$ we construct the set of path subsystems $\mathcal{R}'$ as described in the previous subsection. In other words, we construct a decomposition of the original network based on both the primary and alternate paths. We solve decomposition $\mathcal{R}'$ using the algorithm of Fig. 8 to obtain an initial estimate of the call-blocking probabilities $P_{sd}^{(1)}$ and $P_{sd}^{(2)}$ along the primary and alternate paths, respectively. Because of our approximation, the arrival rate for the overflow traffic offered to alternate paths is simply given by the product of the arrival rate of the traffic to the primary path times the blocking probability along this path. Also, if a primary path r intersects with an alternate path q, the arrival rate on the alternate path q (primary path r) is taken into account when solving path r (path q). This approach captures the effect that calls established over alternate (primary) paths have on calls established over primary (alternate) paths.

Once estimates for blocking probabilities $P_{sd}^{(1)}$ and $P_{sd}^{(2)}$ have been obtained, an estimate of the blocking probability of calls for the source–destination pair (s, d) can be computed as $P_{sd} = P_{sd}^{(1)} \times P_{sd}^{(2)}$. These estimates are used to update the arrival rates of calls to the network, and the decomposition is solved again. This process is repeated until the blocking probabilities P_{sd} converge for all s, d.

V. Numerical Results

In this section we demonstrate the accuracy of our analytical techniques by comparing approximate results to either exact numerical results or simulation results. Simulation results are plotted along with 95% confidence intervals estimated by the method of replications. The number of replications is 30, with each simulation run lasting until each type of call has at least 100 000 arrivals. For the approximate results, the iterative decomposition algorithm terminates when all blocking probability values have converged within 10^{-7}.

A. Validation of the Time-Reversible Markov Process

In Fig. 9, we plot the blocking probability of calls for each source–destination pair in a two-hop path without converters, against the number W of wavelengths per hop. For each type of call we show two curves. The first curve is obtained through a numerical solution of the exact Markov process $\mathcal{M}_2$, and is referred to in the figure as "exact solution." The second curve is obtained from the closed-form solution of the approximate Markov process $\mathcal{M}_2'$, and is referred to as "approximate solution." As we can see, the overall behavior of the two curves is similar for all types of calls, and, more importantly, the approximate blocking probability is always very close to the exact value.

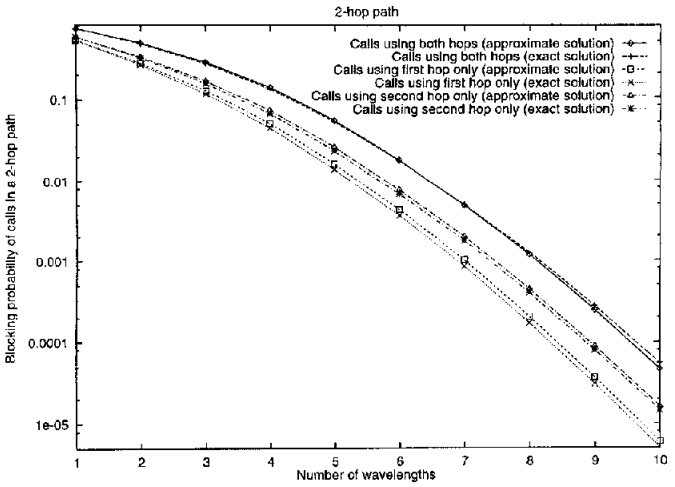


Fig. 9. Exact and approximate blocking probabilities of the various calls in a two-hop path.

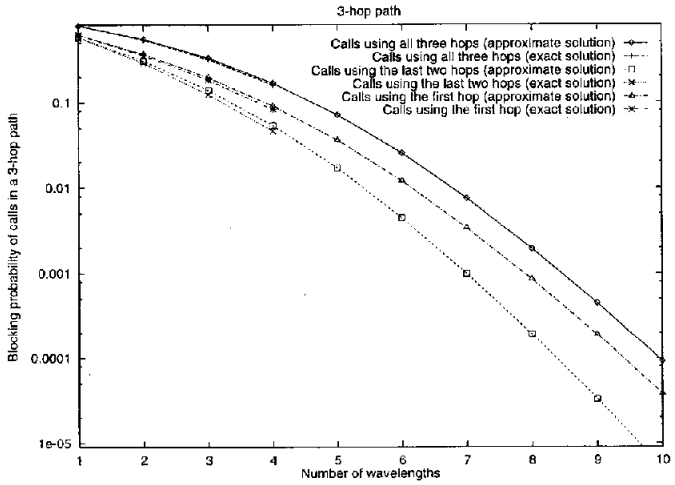


Fig. 10. Exact and approximate blocking probabilities of various calls in a three-hop path.

Fig. 10 is similar to Fig. 9, but it presents results for a three-hop path. We only plot the blocking probability of calls for three of the six source–destination pairs, namely, calls that traverse all three hops, calls that use only the last two hops, and calls that use only the first hop. The blocking probability curves for the other three types of calls are very similar to the ones shown in Fig. 10 . Again, we observe that the values of the blocking probabilities obtained through the closed-form solution of the time-reversible process $\mathcal{M}_3'$ are very close to the exact numerical values obtained from the process $\mathcal{M}_3$. However, the figure does not include values for the exact blocking probability when $W > 4$ because of the state space explosion of the exact process. In general, for the same value W, the closed-form solution of process $\mathcal{M}_3'$ can be obtained significantly faster, up to two orders of magnitude, than the numerical solution of the exact process.

Overall, the results shown in Figs. 9 and 10 indicate that the approximate time-reversible process is quite accurate for short paths.

B. Validation of the Decomposition Algorithm for Long Paths

In this subsection, we give approximate and simulation results for ten-hop paths with $W = 10$ wavelengths; results for paths of different length can be found in [17]. We include results for paths with and without converters. Because of the very large number of parameters that can potentially be varied, for the results presented here we used the following values: $\mu = 1$, $\lambda_{ij} = 0.1$, $i < j$, and $\lambda_{ii} = \lambda$. In other words, we let the mean holding time to be equal to one for all calls, we fix the arrival rate of all calls traversing two or more hops to the value 0.1, and we also set the arrival rate of calls traversing exactly one hop to λ. Figs. 11 and 12 plot the call-blocking probability as a function of λ.

In a ten-hop path there are 55 different source–destination pairs, making it impossible to present results for all of them. Thus, we show results for only two source–destination pairs. The blocking probability of calls traversing all ten hops in the path is plotted in Fig. 11, and the blocking probability of calls traversing hops 2 through 6 of the path is shown in Fig. 12. In both figures, the value of λ is varied from 0.05 to 0.21, while the arrival rate of all other calls is fixed to 0.1, as mentioned above. Each figure contains two sets of plots, one for the ten-hop path without converters, and one for the same path employing three converters. Each set consists of two plots, one for the results from our decomposition algorithm, and the other for the simulation results. For the no-converter case, approximate results are obtained through a $1 \times 3 \times 3 \times 3$ decomposition, that is, we decompose the path to a one-hop segment followed by three three-hop segments. For the converter case, the three converters are assumed to be at nodes 1, 4, and 7, a configuration that also results in a $1 \times 3 \times 3 \times 3$ decomposition.

From the figures we observe that as the value of the load λ increases, the blocking probability of both types of calls increases. We also note that, when there are converters, the blocking probability for both types of calls is significantly lower than when there is no converter. Both these results are expected. The most important observation from these figures, however, is the fact that the values of the blocking probability obtained through our iterative decomposition algorithm are close to the values obtained through simulation, with the results in the lower two graphs of Fig. 12 representing a worst-case scenario regarding the performance of the algorithm. Overall, based on a very large number of validation tests performed over a variety of different parameter values and reported in [20], we have found that the decomposition algorithm gives accurate results for long paths and for a wide range of traffic loads. We have also observed that the algorithm always converges in a few iterations, taking one minute for a ten-hop path, while the simulation takes several hours.

C. Validation of the Decomposition Algorithm for Mesh Topologies

In this section, we validate the decomposition algorithm of Section IV for mesh networks by comparing the approximate blocking probabilities to simulation results for the NSFNET irregular topology. Because of space considerations, we only present results for fixed routing. Results for other topologies and

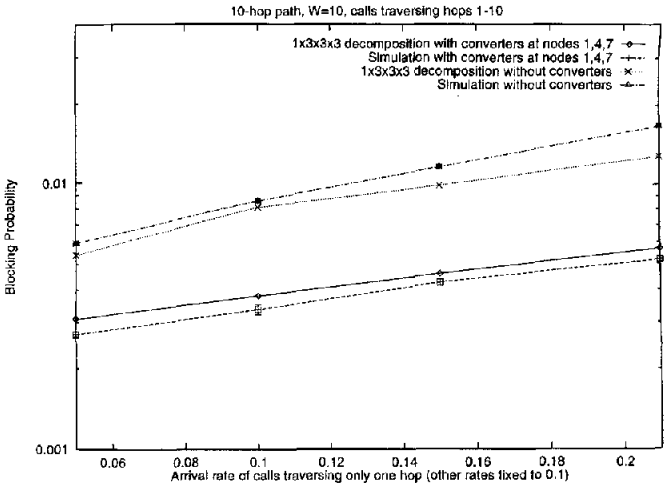


Fig. 11. Blocking probability of calls traversing all links of a ten-hop path with $W = 10$.

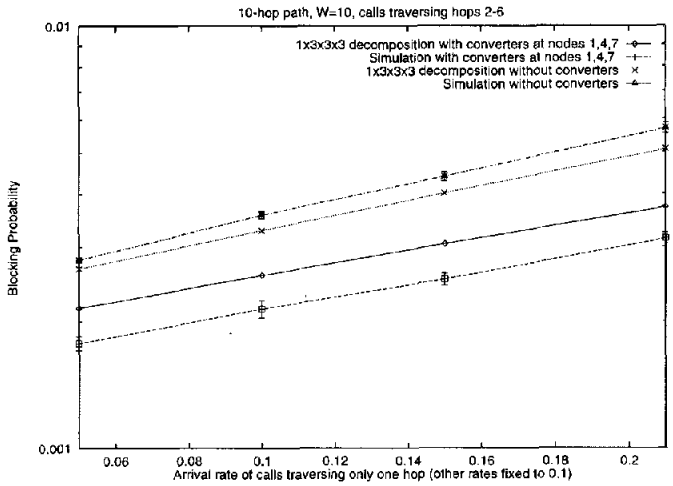


Fig. 12. Blocking probability of calls traversing links 2 through 6 of a ten-hop path with $W = 10$.

for alternate routing with one, two, or three alternate paths per source–destination pair can be found in [20], and are very similar to the ones presented here.

Since we will be using traffic data reported in [4], following that study, we have also augmented the 14-node NSFNET topology by adding two fictitious nodes, nodes 1 and 16 in Fig. 13, to capture the effect of NSFNET's connections to Canada's communication network, CA*net. The resulting topology consists of 16 nodes and a total of 240 source–destination pairs, as shown in Fig. 13. We assume that each link carries $W = 10$ wavelengths. We present detailed results for the call-blocking probabilities of only a small number of pairs, and summarize the results for the whole network. Specifically, we present detailed results for the blocking probabilities of calls involving nodes along the path $(3, 5, 6, 7, 9, 12, 15, 16)$. The 28 source–destination pairs in this path, along with the corresponding shortest path lengths and the labels used in Figs. 14 and 15 are shown in Table I.

We have used two traffic patterns with the NSFNET topology. The first traffic pattern is designed to capture the locality of traffic that has been observed in many networks. Specifically,

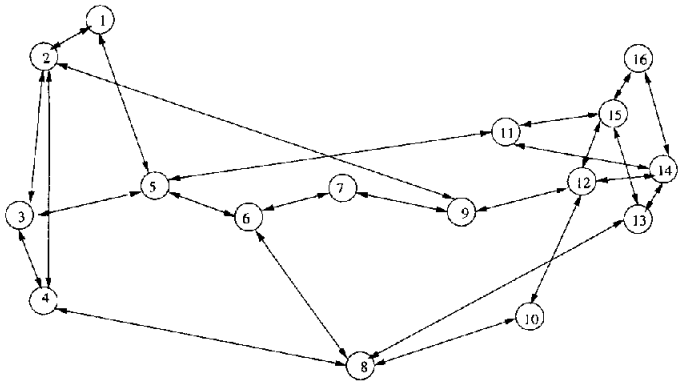


Fig. 13. NSFNET topology

the arrival rate for a source–destination pair (s, d) with a shortest path of length l_{sd} is given by $\lambda_{sd} = 0.6 - l_{sd}$ (note that no shortest path is longer than four hops). The second traffic pattern was designed to reflect actual traffic statistics collected on the NSFNET backbone network, as reported in the traffic matrix in [4, Fig. 6]. The data in this traffic matrix represent the measured number of bytes transferred from a node s to a node d in the NSFNET backbone within a certain 15-minute interval. Clearly, this data, collected over a packet-switched network, cannot be directly applied to a circuit-switched wavelength-routing network, such as the one considered in this work. However, our intention is simply to capture the relative traffic demands among the different source–destination pairs. To this end, we first divide the entries of the matrix in [4, Fig. 6] by the link capacity (T3 links) to obtain the "offered load" ρ_{sd} per source–destination pair. Since the resulting values are too small, we multiply them by a constant to obtain reasonable values for the offered load. Then, assuming that all calls have a mean holding time $1/\mu = 1$, the offered load values become the arrival rates λ_{sd} used in the experiments. As a result, the relative values of these arrival rates reflect the relative traffic requirements among the different source–destination pairs according to the traffic pattern reported in [4].

Figs. 14 and 15 present the call-blocking probabilities for the selected pairs of Table I and for the first and second traffic patterns, respectively. The fixed route assigned to each source–destination pair is the shortest path between the source and destination nodes. For the results of Fig. 14 the link utilizations is in the range [1.846, 5.668] with an average of 3.494, while for Fig. 15 the utilization is in the range [0.015, 8.059] with an average of 3.976.

From the two figures we see that calls established over longer paths tend to experience higher blocking probability than calls using short paths. However, because of the irregular topology, the blocking probability can be significantly affected by the actual load along the path taken by a call. For instance, we observe in Fig. 14 that the blocking probabilities of calls established over, say, one-hop paths vary widely depending on the number of other calls using the same path. Regarding the accuracy of the decomposition algorithm, we note that, despite the wide range of blocking probability values involved, the curve obtained analytically closely follows the simulation curve for the 28 source–destination pairs shown in Figs. 14 and 15. The

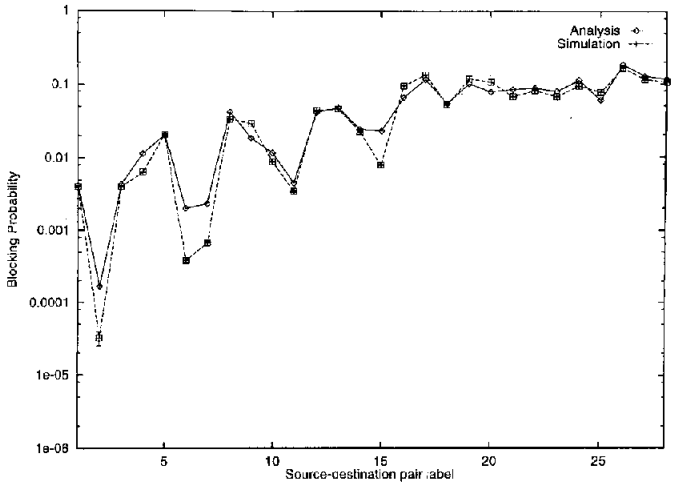


Fig. 14. Blocking probability for selected source–destination pairs in the NSFNET with $W = 10$ and fixed routing (first traffic pattern).

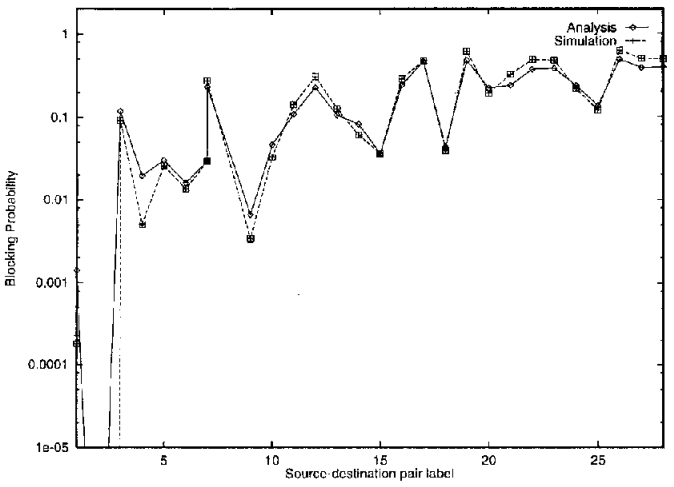


Fig. 15. Blocking probability for selected source–destination pairs in the NSFNET with $W = 10$ and fixed routing (second traffic pattern).

analytical results are accurate despite the fact that in Fig. 15 blocking probability values as high as 0.5 are involved. For these high values, however, we can see that our analysis starts to underestimate the simulation, while it overestimates it at lower blocking probability values. Despite this behavior, the analytical and simulation results are always very close even at high loads.

Finally, Tables II and III present a summary comparison of analytical and simulation results for all 240 source–destination pairs of the NSFNET topology for fixed routing and for both traffic patterns used. The high maximum relative difference values can be explained by the fact that, in an irregular topology such as the NSFNET in Fig. 13, some paths are underutilized and the corresponding blocking probabilities are very low. Although our analysis correctly predicts low probabilities in these cases, the corresponding simulation results give zero (or very close to zero) values. For instance, the blocking probability for the second source–destination pair in Fig. 15 [i.e., pair (15, 16) in Fig. 13] obtained by the simulation was zero. Although the analytically computed probability was less than 10^{-6} (not plotted in the figures), the relative error was 100%.

TABLE I
SELECTED SOURCE-DESTINATION PAIRS FOR THE NSFNET TOPOLOGY

Pair	(5,6)	(15,16)	(6,7)	(12,15)	(9,12)	(7,9)	(3,5)	(5,15)	(5,7)	(6,9)
Label	1	2	3	4	5	6	7	8	9	10
Shortest Path Length	1	1	1	1	1	1	1	2	2	2
Pair	(12,16)	(9,15)	(7,12)	(3,6)	(3,9)	(5,16)	(5,12)	(5,9)	(6,15)	(6,12)
Label	11	12	13	14	15	16	17	18	19	20
Shortest Path Length	2	2	2	2	2	3	3	3	3	3
Pair	(9,16)	(7,15)	(3,15)	(3,12)	(3,7)	(6,16)	(7,16)	(3,16)		
Label	21	22	23	24	25	26	27	28		
Shortest Path Length	3	3	3	3	3	4	4	4		

TABLE II
SUMMARY OF RESULTS FOR THE NSFNET TOPOLOGY WITH FIXED ROUTING(FIRST TRAFFIC PATTERN)

Length of shortest path	Absolute Difference			Relative Difference		
	minimum	average	maximum	minimum	average	maximum
1	0.0000e+00	1.6249e-03	1.0968e-02	0.00%	50.21%	91.68%
2	0.0000e+00	6.2733e-03	2.5302e-02	0.00%	26.93%	72.96%
3	0.0000e+00	1.5380e-02	8.0166e-02	0.00%	15.54%	53.04%
4	0.0000e+00	2.1954e-02	9.0578e-02	0.00%	13.37%	48.41%

TABLE III
SUMMARY OF RESULTS FOR THE NSFNET TOPOLOGY WITH FIXED ROUTING (SECOND TRAFFIC PATTERN)

Length of shortest path	Absolute Difference			Relative Difference		
	minimum	average	maximum	minimum	average	maximum
1	0.0000e+00	6.7571e-03	5.8573e-02	0.00%	46.26%	100.00%
2	3.1409e-06	2.2312e-02	1.4344e-01	0.06%	26.47%	100.00%
3	0.0000e+00	4.2773e-02	1.3520e-01	0.00%	16.79%	71.86%
4	9.4132e-03	7.2882e-02	1.3322e-01	3.92%	16.18%	31.31%

Overall, however, we can see that the average absolute and relative difference between analytical and simulation values is very small, indicating that discrepancies between simulation and analysis are limited to blocking probability values that are very low.

D. Converter Placement

We now consider the problem of determining the best placement of l converters on a k-hop path, $k > l$, that minimizes the blocking probability of calls that travel over all k hops. To find the best converter placement we first enumerate all possible ways of placing l converters on a k-hop path; then, we calculate the blocking probability of interest for each alternative using the decomposition algorithm. The best placement is the one with the minimum blocking probability.

We consider a ten-hop path with $W = 10$, and three different traffic load patterns. Fig. 16 plots the load of each hop in the path for each pattern. In the "uniform" pattern, all hops are equally loaded, while the "bowl" (respectively, "inverted bowl") pattern is such that the load decreases (respectively, increases) from hop 1 to hop 5, and then it increases (respectively, decreases) from hop 6 to hop 10. The load values were chosen so that the total network load is the same for all patterns. Note that the same link loads are achievable with many traffic patterns, each of which may lead to different blocking probabilities. Since for

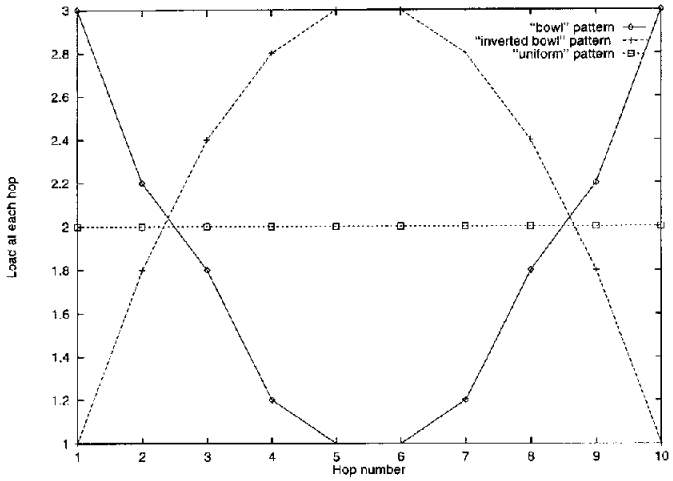


Fig. 16. Bowl, inverted bowl, and uniform load patterns.

a ten-hop path there are 55 node pairs, due to space constraints, we decided not to include the 55 load values for each pattern. Furthermore, it may be difficult for the reader to tell the difference between two sets of 55 load values, while Fig. 16 clearly illustrates the difference in the load patterns.

In Fig. 17 we plot the blocking probability of calls using all ten hops of the path for the optimal placement of l converters,

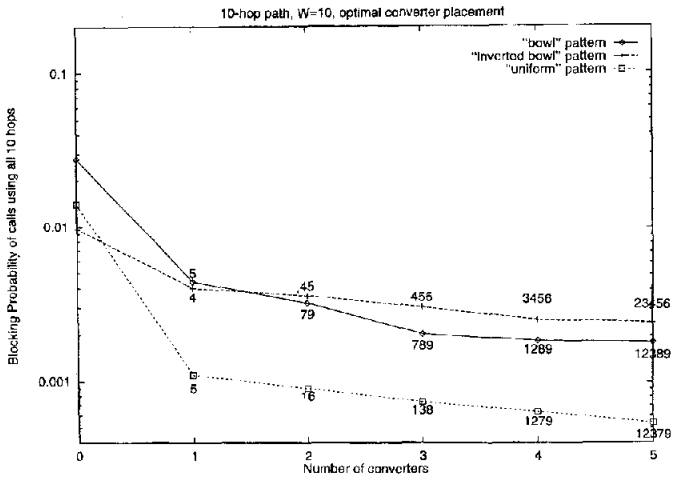


Fig. 17. Blocking probability and optimal converter placement for the load patterns of Fig. 16.

$1 \leq l \leq 5$. For comparison purposes, we also plot the blocking probability of these calls on a path without converters (the values for zero converters in these figures). The optimal location of the converters for each load pattern is also given next to each point of the curves.

As expected, the blocking probability drops as the number of converters increases. However, after an initial steep drop, the curves in general flatten as the number of converters increases. This behavior is consistent with the results of earlier work [14], [15]. We also observe that the effect of converters on the blocking probability is strongly dependent on the actual traffic pattern. Regarding the optimal node location of converters for the different traffic patterns, we first note that the results are intuitively obvious. The figure indicates, for example, that converters be placed at the middle of the path for the "inverted bowl" pattern. However, we observe that the optimal placement also depends strongly on the load pattern. This result suggests that in a dynamic environment where traffic patterns vary over time, there is no single assignment of converters to nodes that will work well for all possible loads. Consequently, simple optimization approaches, such as the one considered here, that seek to minimize the blocking probability under a specific traffic pattern may lead to poor performance if the pattern changes. Instead, more comprehensive approaches to the converter placement problem are needed, such as providing bounds for the blocking probability over a wide range of load patterns. Similar results have been obtained for mesh topologies [18].

VI. Conclusion

We have presented a path decomposition algorithm to evaluate accurately and efficiently the call-blocking performance of wavelength-routing networks with an arbitrary topology. Our algorithm is applicable to networks with either fixed or alternate routing and random wavelength allocation. Our iterative algorithm analyzes the original network by decomposing it into single-path subsystems. These subsystems are analyzed in isolation either by applying an approximate Markov process model, or, in the case of longer paths, by decomposing them into shorter segments. Results from individual subsystems are appropriately combined to obtain a solution for the overall network. Our algorithm can also be applied to the problem of converter placement in wavelength-routing networks.

Acknowledgment

The authors would like to thank the reviewers for their constructive comments.

References

[1] R. A. Barry and P. A. Humblet, "Models of blocking probability in all-optical networks with and without wavelength changers," *IEEE J. Select. Areas Commun.*, vol. 14, pp. 858–867, June 1996.

[2] A. Birman, "Computing approximate blocking probabilities for a class of all-optical networks," *IEEE J. Select. Areas Commun.*, vol. 14, pp. 852–857, June 1996.

[3] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath communications: An approach to high bandwidth optical WANS," *IEEE Trans. Commun.*, vol. 40, pp. 1171–1182, July 1992.

[4] B. Mukherjee *et al.*, "Some principles for designing a wide-area WDM optical network," *IEEE/ACM Trans. Networking*, vol. 4, pp. 684–696, Oct. 1996.

[5] A. Girard, *Routing and Dimensioning in Circuit-Switched Networks*. Reading, MA: Addison Wesley, 1990.

[6] H. Harai, M. Murata, and H. Miyahara, "Performance of alternate routing methods in all-optical switching networks," in *Proc. INFOCOM'97*, Apr., pp. 517–525.

[7] *IEEE J. Select. Areas Commun., Special Issue on High-Capacity Optical Transport Networks*, vol. 16, Sept. 1998.

[8] E. Karasan and E. Ayanoglu, "Effects of wavelength routing and selection algorithms on wavelength conversion gain in WDM optical networks," *IEEE/ACM Trans. Networking*, vol. 6, pp. 186–196, Apr. 1998.

[9] F. P. Kelly, *Reversibility and Stochastic Networks*. New York: Wiley, 1979.

[10] L. Kleinrock, *Queueing Systems, Volume 1: Theory*. New York: Wiley, 1975.

[11] M. Kovacevic and A. Acampora, "Benefits of wavelength translation in all-optical clear-channel networks," *IEEE J. Select. Areas Commun.*, vol. 14, pp. 868–880, June 1996.

[12] A. Mokhtar and M. Azizoglu, "Adaptive wavelength routing in all-optical netowrks," *IEEE/ACM Trans. Networking*, vol. 6, pp. 197–206, Apr. 1998.

[13] H. Perros, *Queueing Networks with Blocking: Exact and Approximate Solutions*. London, U.K.: Oxford Univ. Press, 1994.

[14] S. Subramaniam, M. Azizoglu, and A. Somani, "All-optical networks with sparse wavelength conversion," *IEEE/ACM Trans. Networking*, vol. 4, pp. 544–557, Aug. 1996.

[15] S. Subramaniam, M. Azizoglu, and A. K. Somani, "On the optimal placement of wavelength converters in wavelength-routed networks," in *Proc. IEEE INFOCOM*, Apr. 1998, pp. 902–909.

[16] S. Subramanian, A. K. Somani, M. Azizoglu, and R. A. Barry, "A performance model for wavelength conversion with nonpoisson traffic," in *Proc. IEEE INFOCOM*, Apr. 1997, pp. 500–507.

[17] Y. Zhu, G. N. Rouskas, and H. G. Perros, "Blocking in wavelength-routing networks—Part I: The single-path case," in *Proc. IEEE INFOCOM*, Mar. 1999, pp. 321–328.

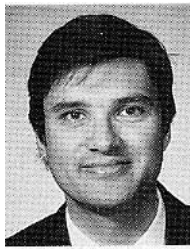
[18] ——, "Blocking in wavelength-routing networks—Part II: Mesh topologies," in *Proc. 16th Int. Teletraffic Congress (ITC 16)*, June 1999, pp. 1321–1330.

[19] ——, "Bounds on the blocking performance of allocation policies in wavelength-routing networks and a study of the effects of converters," North Carolina State Univ., Raleigh, NC, Tech. Rep. TR-99-01, Jan. 1999.

[20] Y. Zhu, "Computation of call-blocking probabilities in wavelength-routing networks," Ph.D. dissertation, North Carolina State Univ., Raleigh, NC, May 1999.

Yuhong Zhu received the B.S. degree in electrical engineering from Beijing University in 1990, and the Ph.D. degree in computer science from the North Carolina State University, Raleigh, in 1999.

He is currently a Member of Technical Staff in the Optical Network Group at Lucent Technologies, Acton, MA. His interests include WDM networks, IP Multicast, MPLS, and network performance evaluation.



George N. Rouskas (S'92–M'95) received the Diploma in electrical engineering from the National Technical University of Athens (NTUA), Athens, Greece, in 1989, and the M.S. and Ph.D. degrees in computer science from the College of Computing, Georgia Institute of Technology (Georgia Tech), Atlanta, GA, in 1991 and 1994, respectively.

He joined the Department of Computer Science, North Carolina State University, Raleigh, in August 1994, and he has been an Associate Professor since July 1999. His research interests include network architectures and protocols, optical networks, multicast communication, and performance evaluation.

Dr. Rouskas is a recipient of a 1997 NSF Faculty Early Career Development (CAREER) Award, and a co-author of a paper that received the Best Paper Award at the 1998 SPIE Conference on All-Optical Networking. He also received the 1995 Outstanding New Teacher Award from the Department of Computer Science, North Carolina State University, and the 1994 Graduate Research Assistant Award from the College of Computing, Georgia Tech. He is a co-guest editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, Special Issue on Protocols and Architectures for Next-Generation Optical WDM Networks, and is on the editorial boards of the IEEE/ACM TRANSACTIONS ON NETWORKING and the Optical Networks Magazine. He is a member of the Association for Computing Machinery and of the Technical Chamber of Greece.

Harry G. Perros (M'87–SM'97) received the B.Sc. degree in mathematics in 1970 from Athens University, Athens, Greece, the M.Sc. degree in operational research with computing from Leeds University, U.K., in 1971, and the Ph.D. degree in operations research from Trinity College, Dublin, Ireland, in 1975.

From 1976 to 1982, he was an Assistant Professor in the Department of Quantitative Methods, University of Illinois, Chicago. In 1979, he spent a sabbatical term at INRIA, Rocquencourt, France. In 1982, he joined the Department of Computer Science, North Carolina State University, Raleigh, as an Associate Professor, and since 1988, he has been a Professor. During the academic year 1988–1989, he was on a sabbatical leave of absence first at BNR, Research Triangle Park, NC, and subsequently at the University of Paris 6, France. During the academic year 1995–1996, he was on a sabbatical leave of absence at Nortel, Research Triangle Park. He has published extensively in the area of performance modeling of computer and communication systems, and has organized several national and international conferences. He also published a monograph entitled *Queueing Networks with Blocking: Exact and Approximate Solutions* (London, U.K.: Oxford Press, 1994). He is the Chairman of the IFIP W.G. 6.3 on the Performance of Communication Systems. His current research interests are in the areas of optical networks and their performance, and software performance evaluation.