Blocking in Wavelength Routing Networks, Part II: Mesh Topologies

Yuhong Zhu George N. Rouskas Harry G. Perros

TR-98-10

October 2, 1998

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 accurately and efficiently evaluate the blocking performance of such networks with and without wavelength converters. Our iterative algorithm analyzes the original network by decomposing it into single path sub-systems. These sub-systems are analyzed in isolation by using our previous algorithms for a single path of a wavelength routing network, and the individual results are appropriately combined to obtain a solution for the overall network. The path decomposition approach can naturally capture the correlation of both link loads and link blocking events, giving accurate results for a wide range of loads and network topologies. Our model also allows non-uniform traffic, i.e., call request arrival rates that can vary with the source-destination pair, and it can be used when the location of converters is fixed but arbitrary. Our algorithm represents a simple and computationally efficient solution to the difficult problem of computing call blocking probabilities in wavelength routing networks. We validate our algorithm by applying it to both regular and irregular topologies of realistic size and comparing to simulation results. We also demonstrate how our analytical techniques can be applied to gain insight into the problem of converter placement in wavelength routing networks.

Keywords: Wavelength division multiplexing, wavelength routing networks, call blocking probability, decomposition algorithms, converter placement

Department of Computer Science
North Carolina State University
Raleigh, NC 27695-7534

1 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 [9, 4]. A wavelength routing network consists of wavelength routers and the fiber links that interconnect them [7, 6, 8]. 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 [15, 1]. Wavelength conversion is a desirable feature since it improves the performance of the network in terms of call blocking probability. However, any gains in performance must be weighted against the cost of wavelength converters.

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 network 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, 13, 2, 11, 15, 17]. 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 [13] 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 [11] 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 [15] to recursively compute blocking probabilities assuming that the load on link i of a path depends only on the load of link $i-1$. Finally, a study of call blocking under non-Poisson input traffic was presented in [17], under the assumption that link loads are statistically independent.

Other wavelength allocation schemes, as well as dynamic routing are harder to analyze. First-fit wavelength allocation was studied using simulation in [3, 13], and it was shown to perform better than random allocation, while an analytical overflow model for first-fit allocation was developed in [12]. A dynamic routing algorithm that selects the least loaded path-wavelength pair was also studied in [12], and in [14] an unconstrained dynamic routing scheme with a number of wavelength allocation policies was evaluated. Except in [15, 16], all other studies assume that either all or none of the wavelength routers have wavelength conversion capabilities. The work in [15] takes a probabilistic approach in modeling wavelength conversion by introducing the converter density, which represents the probability that a node is capable of conversion independently of other nodes in the network. While this approach works well when the objective is the estimation of the expected call blocking performance, it cannot be used to calculate the actual blocking probability on individual paths when the placement of converters is known, nor can it be used to compare various converter placement strategies. Finally, in [16] a dynamic programming algorithm to determine the location of converters on a single path that minimizes average or maximum blocking probability was developed under the assumption of independent link loads.

Most of the approximate analytical techniques developed for computing blocking probabilities in wavelength routing networks [13, 2, 11, 17, 12, 14, 16] make the assumption that link blocking events are independent and amount to the well-known *link decomposition* approach [10], 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 efficiently evaluate the performance of these networks, 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 the problem of computing call blocking probabilities in mesh wavelength routing networks with fixed and alternate routing and random wavelength allocation. Unlike previous studies, we develop an iterative *path decomposition* algorithm [10] for analyzing arbitrary network topologies. Specifically, we analyze a given network by decomposing it into a number of path sub-systems. The latter are analyzed in isolation using our previous results on blocking in a single path in a wavelength routing network [18]. 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. It also allows non-uniform traffic, in the sense that call request arrival rates vary for each source-destination pair. Finally, 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. Therefore, our algorithm can be an important tool in the development and evaluation of converter placement strategies.

The rest of the paper is organized as follows. In Section 2 we review the results of our work in computing blocking probabilities in a single path of a wavelength routing network. In Section 3 we present a new path decomposition algorithm for analyzing a mesh wavelength routing network with an arbitrary topology under both fixed and alternate routing. In Section 4 we validate our algorithm through simulation using both regular and irregular topologies and various traffic loads. We also demonstrate how our algorithm can be applied to the problem of converter placement in a wavelength routing network. We conclude the paper in Section 5.

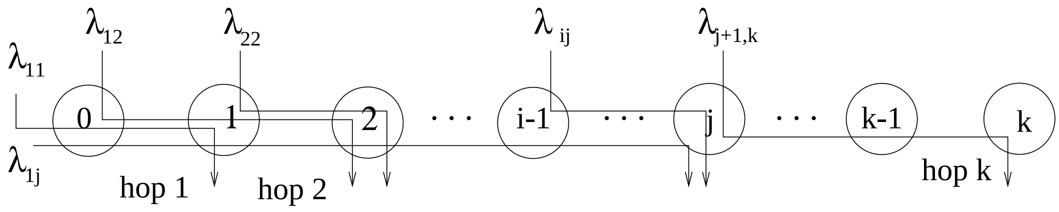

Figure 1: A k-hop path

2 Review of Single Path Results

We considered the problem of computing blocking probabilities in a k-hop path (see Figure 1), $k \geq 2$, of a wavelength routing network with and without converters in [18]. Since our work on mesh topologies builds upon the approximate analytical techniques developed for the single-path case, in this section we review the pertinent results.

Figure 1 shows a k-hop path consisting of $k+1$ nodes labeled $0, 1, \cdots, k$. We use $\lambda_{ij}, j \geq i$ to denote the Poisson arrival rate of calls that use hops i through j, i.e., calls that originate at node $i-1$ and terminate at node j; the mean holding time of all calls is denoted by $1/\mu$. Each link in the path supports exactly W wavelengths, and each node is capable of transmitting and receiving on any of these W wavelengths. One or more nodes in the path may employ wavelength converters. These nodes can switch an incoming wavelength to an arbitrary outgoing wavelength. When there are converters at all nodes, the situation is identical to that in classical circuit switched networks, a special case of the more general scenario discussed here.

If there are no wavelength converters in the path, a call can only be established if the *same* wavelength is free on all links used by the call. This is known as the *wavelength continuity* requirement, and it increases the probability of call blocking. If a call cannot be established due to lack of available wavelengths, the call is blocked. On the other hand, if a call can be accommodated, it is randomly assigned one of the wavelengths that are available on the links used by the call (in a path with wavelength converters, a wavelength is randomly assigned within each segment of the path consisting of links between successive nodes with converters).

We let n_{ij}, $j \geq i$, be a random variable representing the number of currently active calls using hops i through j of the path. We also let $f_{ij}, j \geq i$, be a random variable representing the number of wavelengths that are free on all links i through j of the path. Then, our approach to obtaining the call blocking probabilities in a k-hop path consists of three steps, as follows.

We first showed in [18] that the evolution of k-hop system can be characterized by a k^2-dimensional Markov process. The first $\frac{k(k+1)}{2}$ random variables $n_{ij}, 1 \leq i \leq j \leq k$, in the state description of the Markov process provide the number of active calls between all possible source-destination pairs in the path. The last $\frac{(k-1)k}{2}$ random variables $f_{ij}, 1 \leq i < j \leq k$, represent the number of wavelengths that are free on all segments of the path consisting of two or more links. For example, the Markov process for a 3-hop path has the state description $\underline{n} = (n_{11}, n_{12}, n_{13}, n_{22}, n_{23}, n_{33}, f_{12}, f_{13}, f_{23})$. We also showed that the Markov process for a k-hop path, $k \geq 2$, is not time-reversible when the number of wavelengths $W > 1$. However, because of the huge state space, it takes a very long amount of time to obtain the exact call blocking probabilities, by solving the Markov process directly, even for 3-hop paths with $W = 4$ wavelengths.

We then introduced a new time-reversible Markov process by modifying the original exact Markov process as follows. The new Markov process has the same state space and the same transitions as the original one. The vast majority of the transition rates of the new process are the same as the respective rates of the exact process. However, to ensure that the new Markov process is time-reversible, the transition rates between some pairs of states were appropriately modified; the reader is referred to [18] for the specific rules regarding which transition rates to be modified and by what factor. This new time-reversible Markov process has a product form solution for its steady-state probability that can be easily written down. For example, the solution to the Markov process corresponding to a 3-hop path with the state description $\underline{n}$ above is given by:

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

where $f_{11} = W - n_{11} - n_{12} - n_{13}$, $f_{22} = W - n_{12} - n_{22} - n_{13} - n_{23}$, and $f_{33} = W - n_{13} - n_{23} - n_{33}$ and $G_3(W)$ is the normalizing constant for a 3-hop path with W wavelengths.

We showed in [18] that the blocking probabilities obtained through the product form solution of the time-reversible Markov process closely approximate the exact blocking probabilities obtained through the numerical solution of the exact Markov process. Although we have not been able to devise a recursive procedure for efficiently computing the normalizing constant, using brute force enumeration we can compute the solution to the approximate Markov process in a reasonably short amount of time for 2- and 3-hop paths with up to 30 wavelengths, and even 4-hop paths with up to 8 wavelengths. For longer paths, we have devised an iterative decomposition algorithm to efficiently obtain the blocking probabilities.

We analyze a k-hop path, $k \geq 4$, by decomposing it into 3-, 2-, or 1-hop segments in tandem. Each segment is first analyzed in isolation using the corresponding time-reversible process. The arrival rates of calls originating (terminating) in a segment but terminating (originating) in another segment are accounted for by modifying the arrival rate of calls of the individual segments. The solutions to individual segments are then appropriately combined to obtain an initial value for the blocking probability of calls that traverse more than one segment. The effect of the wavelength continuity requirement is captured by an approximate continuity factor that increases the blocking probability of calls continuing to the next segment to account for the possible lack of common free wavelengths in the two segments. Using these initial estimates, the arrival rates to each segment are modified and each segment is again solved in isolation to obtain a new solution. The process is repeated until the blocking probabilities converge.

The decomposition algorithm has been shown to converge in a handful of iterations even for long paths (up to 10 hops). The blocking probabilities obtained by using this algorithm closely approximate those obtained through simulation for a wide range of offered loads [18]. Finally, the decomposition algorithm can be easily extended to paths in which some nodes contain wavelength converters, making it possible to compute call blocking probabilities when the placement of converters is arbitrary but fixed.

3 Path Decomposition Algorithm for Mesh Networks

3.1 Network Model

We consider a circuit-switched wavelength routing network with an arbitrary topology. There are N nodes and L (unidirectional) links in the network, with each link supporting W wavelengths. Call requests between a source node s and a destination node d arrive at the source node according to a Poisson process with a rate of λ_{sd}. The call holding times are exponentially distributed with mean $1/\mu$. We also let $\rho_{sd} = \lambda_{sd}/\mu$ be the offered load of these calls.

In wavelength routing networks, there are two parts in the routing problem. When a call request arrives, a path over which the connection will be established must first be determined. In this work we consider both fixed and alternate routing [10]. 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. 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. As in [18], we only consider the random wavelength assignment policy in this work, whereby a call is allocated one of the available wavelengths in the selected path at random.

We let $\mathcal{R}$ denote the set of paths assigned to the various source-destination pairs. For fixed routing, $\mid \mathcal{R} \mid = N(N-1)$. If alternate routing with m paths (one primary and $m-1$ alternates) for each source-destination pair is used, then $\mid \mathcal{R} \mid = mN(N-1)$. We also let $P_{sd}^{(n)}, 1 \leq n \leq m$, denote the probability that a call originating at node s and terminating at node d will be blocked on the n-th path assigned to this source-destination pair.

3.2 Fixed Routing

We analyze a mesh network by decomposing it into a number of sub-systems where each sub-system is a single path. Each sub-system is analyzed in isolation using the analytical techniques developed in [18] and summarized in Section 2. Specifically, sub-systems consisting of three links or less are analyzed by solving the corresponding approximate time-reversible Markov process. Sub-systems longer than three hops are analyzed using the iterative decomposition algorithm to obtain the call blocking probabilities. The individual solutions are appropriately combined (as explained shortly) by modifying the call arrival rates to each sub-system to reflect the newly computed blocking probabilities. The process is repeated until all blocking probabilities converge within a prescribed tolerance.

Before we proceed we emphasize that the number of sub-systems into which the network is decomposed is significantly smaller than the total number $\mid \mathcal{R} \mid$ of paths. This is because many of the shorter paths are completely contained within other longer paths [1]. Therefore, these shorter paths do not need to be considered as separate sub-systems. Instead, the blocking probability along these paths can be obtained as a by-product of computing the solution of a long sub-system. Since a k-hop path, such as the one shown in Figure 1, may contain up to $(k+2)(k-1)/2$ shorter paths as sub-paths, by selecting long paths as sub-systems we can drastically reduce the number of sub-systems into which the original network is decomposed.

The first step in analyzing a given network is to decompose it into a set $\mathcal{R}' \subseteq \mathcal{R}$ of paths such that:

- no path $r \in \mathcal{R}'$ is contained within a path $q \in \mathcal{R},\ q \neq r$, and
- any path $q \in \mathcal{R}$ either belongs to $\mathcal{R}'$ or is completely contained within a path $r \in \mathcal{R}'$.

These two requirements ensure that a minimal set of sub-systems 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. Figure 2(b) shows the set of sub-systems $\mathcal{R}'$ obtained by applying this algorithm to the network of Figure 2(a). As we can see, while there are 20 source-destination pairs and corresponding paths in the network, only 10 path sub-systems are used. For instance, the blocking probability on the path from, say, node 1 to node 4, will be obtained through the solution to the sub-system corresponding to the path from node 1 to node 3.

Once the set $\mathcal{R}'$ of sub-systems 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 Figure 2 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. Note also that $q \in \mathcal{S}(r)$ implies that $r \in \mathcal{S}(q)$.

We are now ready to present the decomposition algorithm used to analyze a wavelength routing network with an arbitrary topology. We will illustrate the operation of the algorithm using the network shown in Figure 2. We will show how to update the arrival rates along each path sub-system after each iteration of the algorithm by considering only paths (1,4,3) and (4,3,5). The other path sub-systems are handled in a similar way. Without loss of generality, we assume that shortest paths are used for fixed routing in this network. Recall that $\lambda_{sd}, 1 \leq s,d \leq N$, are the arrival rates to the original network. We also let $\hat{\lambda}_{sd}$ denote the arrival rates used to solve the various path sub-systems; these rates are updated at the beginning of each iteration

[1] We say that a path q is completely contained within another path r if q is a sub-path of r.

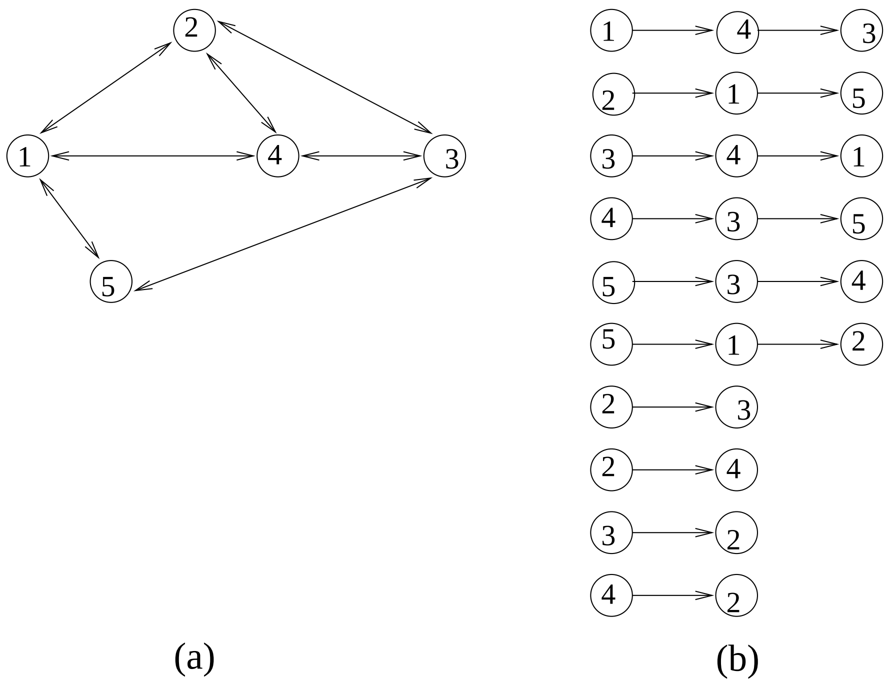

Figure 2: (a) Original network, (b) set $\mathcal{R}'$ of paths into which the network is decomposed

of the algorithm. As will be 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.

Initially, we solve path (1,4,3) in Figure 2 in isolation using these arrival rates:

$$\hat{\lambda}_{14} = \lambda_{14} \tag{2}$$

$$\hat{\lambda}_{13} = \lambda_{13} \tag{3}$$

$$\hat{\lambda}_{43} = \lambda_{43} + (1 - P_{35})\lambda_{45} \tag{4}$$

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 sub-system (1,4,3) is given in expression (2). Similarly, expression (3) can be explained by the fact that only calls from node 1 to node 3 use both links of sub-system (1,4,3). On the other hand, expression (4) 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 the second link of path (1,4,3) since paths (1,4,3) and (4,3,5) intersect. Quantity P_{35} in (4) represents the current estimate of the probability that a call will be blocked on the link (3,5) of path (4,3,5). For the first iteration, we use $P_{35} = 0$; how this value is updated in subsequent iterations will be discussed shortly. Therefore, the term $(1 - P_{35})\lambda_{45}$ represents the *effective* arrival rate of calls from node 4 to node 5 as seen by sub-system (1,4,3), since a fraction $P_{35}\lambda_{45}$ of these calls will be blocked in sub-system (4,3,5). Consequently, the right hand side of expression (4) 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 (the other sub-systems in Figure 2(b) are solved in a similar manner):

$$\hat{\lambda}_{45} = \lambda_{45} \tag{5}$$

$$\hat{\lambda}_{35} = \lambda_{35} \tag{6}$$

$$\hat{\lambda}_{43} = \lambda_{43} + (1 - P_{14})\lambda_{13} \tag{7}$$

Expressions (5) through (7) can be explained using arguments similar to the ones used for expressions (2) to (4). In particular, the second term in the right hand side of (7) represents the effective arrival rate of calls originating in sub-system (1,4,3) and using the link (4,3) of sub-system (4,3,5)

The solution to the path sub-systems (1,4,3) and (4,3,5) will yield an initial value for the probabilities P_{35} and P_{14} that a call using links (3,5) and (1,4), respectively, will be blocked. The new estimates for P_{35} and P_{14} are then used in expressions (4) and (7), respectively, to update the arrival rates for the two path sub-systems, the sub-systems 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 (less than ten) iterations, and that the blocking probabilities obtained closely match simulation results (the performance of the decomposition algorithm will be discussed in detail in Section 4).

A detailed description of our decomposition algorithm is provided in Figure 3. We note that this path decomposition algorithm explicitly accounts not only for the correlation of link loads among the various links of the network, but also for the fact that link blocking events are not independent. This is in sharp contrast to link decomposition algorithms for wavelength routing networks that have appeared in the literature (e.g., see [13, 2, 11, 17, 12, 14, 16]) which compute the blocking probability along a path by assuming that blocking events on each link of the path are independent. Therefore, we expect, and numerical results to be presented in Section 4 will confirm, that our algorithm will provide a good approximation to the blocking probabilities.

3.3 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 model is a well-known model that has been studied extensively in the literature, and moment matching techniques have been used to analyze blocking probabilities in circuit-switched networks with alternate routing [10].

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 3.2
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 $r \in \mathcal{R}'$ 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

Figure 3: Path decomposition algorithm

Overflow models have also been used in the study of blocking probabilities in wavelength routing networks in [12, 14].

We nevertheless make the assumption that overflow traffic is also Poisson with an appropriate rate. This assumption makes it possible to use the path decomposition algorithm developed in the previous subsection to analyze networks with alternate routing. 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 $\mathcal{R}_1$ (respectively, $\mathcal{R}_2$) denote the set of primary (resp., alternate) paths for all node pairs. From $\mathcal{R}_1$ and $\mathcal{R}_2$ we construct the sets of path sub-systems $\mathcal{R}'_1$ and $\mathcal{R}'_2$, respectively, as described in the previous subsection. In other words, we construct two decompositions of the original network, one based on the set of primary paths and one based on the set of alternate paths. We solve the first decomposition $\mathcal{R}'_1$ using the algorithm of Figure 3 to obtain an initial estimate of the call blocking probabilities $P_{sd}^{(1)}$ along the primary paths. From these blocking probabilities we can obtain an initial estimate of the overflow traffic offered to the alternate paths. Because of our approximation, the arrival rate for this overflow traffic is simply given by the product of the arrival rate of the traffic to the primary path times the blocking probability along this path.

We then solve the second decomposition $\mathcal{R}'_2$, with the overflow rates as the arrival rates for these paths, using the algorithm in Figure 3. However, if an alternate path $r \in \mathcal{R}'_2$ intersects with a primary path $q \in \mathcal{R}'_1$, the arrival rate on the primary path q is taken into account when solving path r. This approach captures the effect that calls established over primary paths have on calls established over alternate paths. The solution to this decomposition gives an estimate of the blocking probabilities $P_{sd}^{(2)}$ along alternate paths.

After this first iteration, we again solve the first decomposition $\mathcal{R}_1$. This time, however, we take into account the effect of calls established over alternate paths which intersect with primary paths in $\mathcal{R}_1$. A new estimate of the call blocking probabilities $P_{sd}^{(1)}$ along primary paths is obtained, and it is used to solve the second decomposition $\mathcal{R}_2$. This two-step process is repeated until the blocking probabilities $P_{sd}^{(1)}$ and $P_{sd}^{(2)}$ converge for all s, d. The final blocking probability of calls for the source-destination pair (s, d) can then be computed as [2]:

$$P_{sd} \quad = \quad P_{sd}^{(1)} \times P_{sd}^{(2)} \tag{8}$$

Numerical results to be presented in Section 4 will demonstrate that, despite the assumption that overflow traffic is Poisson, this iterative path decomposition approach is quite accurate for both regular and irregular

[2] In general, when there is one primary and $m-1$ alternate paths assigned to each source-destination pair, each iteration of our algorithm consists of m steps. Specifically, during step i, $i = 1, \cdots, m$, we compute the solution to a decomposition of the original network into a set of i-th path (primary or alternate) sub-systems. Within each step, the effect of calls established over other (primary or alternate) paths intersecting with the sub-systems solved is considered. Finally, the expression for the overall blocking probability P_{sd} consists of a product of m factors in the right hand side of (8)

topologies, and for a wide range of traffic loads. We believe that the high accuracy of our algorithm is due to the fact that the Poisson assumption for overflow traffic is a good approximation for

- low to moderate loads (and call blocking probabilities), since then the call arrivals seen by the alternate paths will approach Poisson arrivals, and
- the second and higher alternate paths.

Although we do expect our decomposition algorithm to yield less accurate results under high or very high offered loads (due to the peakedness of overflow traffic in this case), it is unlikely that wavelength routing networks will be operated at loads which will result in high call blocking probabilities. As we will show in the next section, our approximate approach works well for blocking probability values as high as 0.5, which we feel is significanlty higher than the blocking probabilities that can be tolerated in such an environment.

4 Numerical Results

In this section, we first validate our decomposition algorithm by comparing the analytically derived blocking probabilities to simulation results for two network topologies: a regular 5×5 torus network, and the NSFNET irregular topology. We then demonstrate how our analytical techniques can be applied to the problem of determining a near-optimal placement of converters within the NSFNET network. In all the figures, we plot 95% confidence intervals along with simulation results. We have also assumed that the mean holding time $1/\mu = 1$ for all source-destination pairs.

4.1 The 5×5 Torus Network

Let us first consider the 5×5 torus network shown in Figure 4 with $W = 10$ wavelengths per link. There are 600 source-destination pairs in this network, making it impossible to present numerical results for all of them. We have therefore decided to present the call blocking probabilities (obtained through simulation and analysis) for only 24 different source-destination pairs, and to summarize the results for the remaining pairs. The source-destination pairs for which detailed results are provided are those with node 1 as the source, and are listed in Table 1 along with the corresponding labels used in Figures 5 and 6. The table also gives the length of the shortest path from node 1 to node d of the corresponding source-destination pair $(1, d)$. Note that in the 5×5 torus there are 100 pairs with a shortest path length of 1, 200 pairs with a shortest path length of 2, 200 pairs with shortest path length of 3, and 100 pairs with a shortest path length of 4. Because of the regular topology, the selected pairs in Table 1 are a representative sample of the various source-destination pairs.

Figure 5 plots the blocking probability for the 24 source-destination pairs of Table 1 for the 5×5 torus network under fixed routing, where the path assigned to each pair (s, d) is the shortest path from s to d.

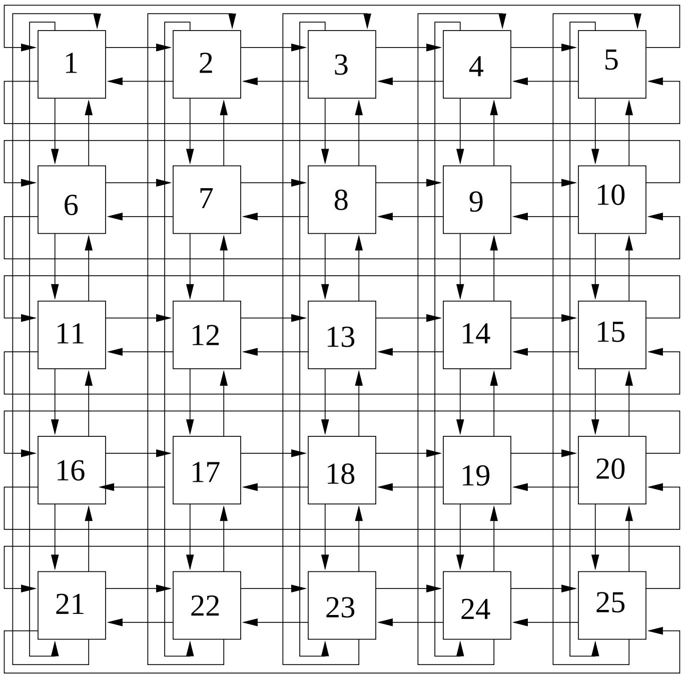

Figure 4: A 5 × 5 bidirectional torus network

Table 1: Selected source-destination pairs for the torus network

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

The call arrival rates were selected such that

$$\lambda_{sd} = \begin{cases} 0.4, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 1 \\ 0.3, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 2 \\ 0.2, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 3 \\ 0.1, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 4 \end{cases} \tag{9}$$

This selection of arrival rates was intended to capture the locality of traffic that has been observed in many networks. The utilization of each link in the network for these arrival rates is in the range [3.140, 3.144]. The tight range of link utilizations can be explained by the fact that both the topology and the traffic load are symmetric.

As we can see in Figure 5, the blocking probability increases significantly with the length of the path that a call uses. In fact, there is a difference of almost two orders of magnitude in the blocking probability experienced by calls using 1-hop paths (i.e., those with labels 1 through 4) and calls using 4-hop paths (the ones labeled 20 to 24). This increase in blocking probability with path length is expected, and has been observed in previous studies. We also note that the blocking probability of calls using the same number of hops is very similar. This behavior is again due to the symmetry of both the topology and of the traffic load. Regarding the accuracy of our decomposition algorithm, we observe that the analytical results closely match the simulation results for all source-destination pairs shown (very similar results were obtained for the other pairs as well, and are discussed below). Although the relative error of our analysis is greater for calls using 1-hop paths, this is only because of the very low blocking probability values involved. In fact, the absolute error is always very small, regardless of the length of the path used by a call.

In Figure 6, we plot the blocking probability for the same 24 source-destination pairs under alternate routing. Specifically, two paths are assigned to each pair of nodes, the shortest and second shortest ones, as the primary and alternate path, respectively. The utilization per link in this case is in the range [3.195, 3.198]. The higher utilization is due to the fact that more calls are accepted since the blocking probabilities are lower when alternate routing is used. Indeed, when alternate routing are used, the blocking probability drops by about one order of magnitude for all calls, but the general behavior of the curves in Figure 6 is very similar to that in Figure 5. We also observe that the blocking probability values obtained analytically under the assumption that the overflow traffic is Poisson, are very accurate. In fact, the accuracy of our decomposition algorithm is similar to that in the case of fixed routing, when there is no overflow traffic offered to the network.

Since call arrivals to alternate paths in the simulated system are burstier that Poisson traffic, these call requests experience higher blocking probability. Consequently, because of our Poisson approximation for overflow traffic, one would expect that our analysis would underestimate the simulation under alternate routing. However, as we can see from Figures 5 – 6, the analytically derived blocking probability values tend to overestimate the simulation values for both fixed and alternate routing. This result demonstrates that the Poisson approximation is accurate within this range of loads and for this regular topology. Similar results have been obtained for irregular topologies and higher load values, and will be discussed in the next subsection.

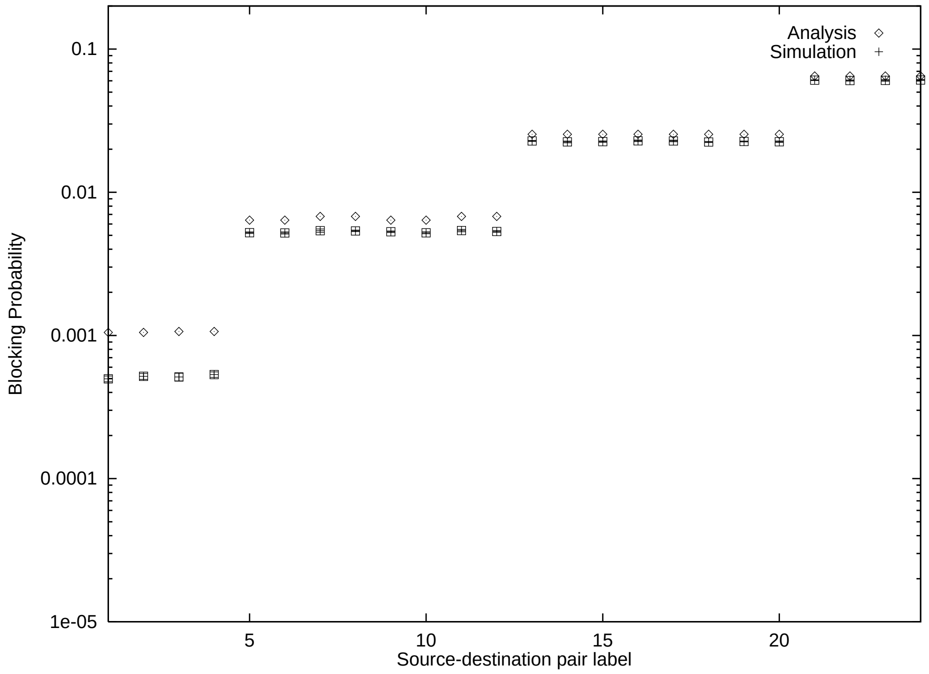

Figure 5: Blocking probability for selected source-destination pairs in the 5×5 torus network with $W = 10$ and fixed routing

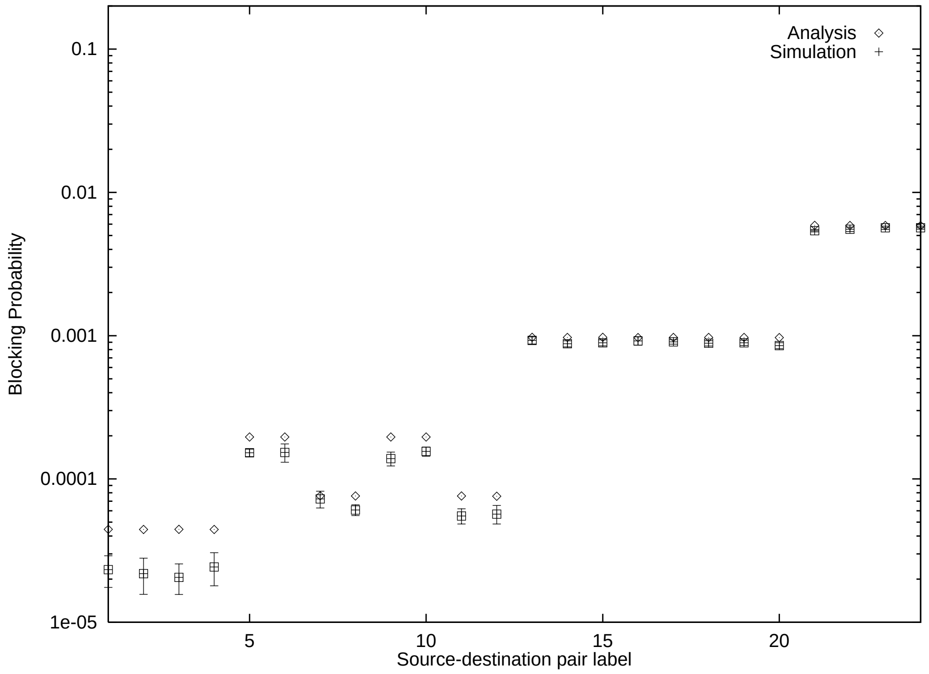

Figure 6: Blocking probability for selected source-destination pairs in the 5×5 torus network with $W = 10$ and alternate routing (one alternate path

Table 2: Summary of results for the 5×5 torus network with fixed routing

Length of shortest path	Absolute Difference			Relative Difference		
	minimum	average	maximum	minimum	average	maximum
1	4.8448e-04	5.1969e-04	5.5454e-04	46.16%	49.13%	52.02%
2	8.9264e-04	1.1658e-03	1.4208e-03	13.99%	17.67%	20.96%
3	2.2500e-03	2.5791e-03	2.8660e-03	8.85%	10.14%	11.27%
4	2.9144e-03	3.7916e-03	4.3548e-03	4.50%	5.85%	6.72%

Table 3: Summary of results for the 5×5 torus network with alternate routing (one alternate path)

Length of shortest path	Absolute Difference			Relative Difference		
	minimum	average	maximum	minimum	average	maximum
1	9.4551e-06	1.6180e-05	2.3559e-05	21.30%	36.53%	53.27%
2	0.0000e+00	1.7607e-05	5.0914e-05	0.00%	12.05%	26.45%
3	0.0000e+00	4.9275e-05	1.2382e-04	0.00%	5.08%	12.76%
4	0.0000e+00	1.7276e-04	3.7408e-04	0.00%	2.94%	6.37%

In Table 2 we present a summary of the results for all 600 source-destination pairs in the network under fixed routing. Since we have observed very similar blocking probability values for all calls with a certain path length, we have chosen to present summary results based on path length. The second, third, and fourth columns of Table 2 give the minimum, average, and maximum absolute differences, respectively, between the analytical and simulation blocking probability values over all calls with a certain path length (e.g., over all 100 calls using 1-hop paths). The last three columns are similar, except that they present the minimum, average, and maximum percentage difference (relative error), respectively, between the analytical and simulation values values among the corresponding calls. Table 3 presents similar results for the same network with one alternate path per source-destination pair.

The results shown in the two tables further support the observations made earlier regarding the accuracy of the decomposition algorithm. Specifically, we can see that the absolute error between analytical and simulation values is very small for all calls, regardless of the number of hops in the path used. Consequently, the percentage difference (i.e., the relative error) diminishes as the blocking probability values increase. Furthermore, the analytical results for alternate routing are at least as good as those for fixed routing, indicating that the assumption that the overflow traffic is Poisson does not affect the performance of our path decomposition approach.

4.2 The NSFNET Topology

We have applied our iterative decomposition algorithm to a realistic example of a backbone network, namely, the NSFNET irregular topology shown in Figure 7. Since we will be using traffic data reported in [5], following that study, we have also augmented the 14-node NSFNET topology by adding two fictitious nodes, nodes 1 and 16 in Figure 7, 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 in the previous subsection, we have decided to present detailed results for the call blocking probabilities of only a small number of pairs, and to 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). (We note, however, that the shortest path used by some of these calls is not a sub-path of (3,5,6,7,9,12,15,16); for instance, the shortest path for calls between nodes 3 and 15 is (3,5,11,15).) The 28 source-destination pairs in this path, along with the corresponding shortest path lengths and the labels used in Figures 8 through 11 are shown in Table 4.

We have used two different traffic patterns with the NSFNET topology. The first traffic pattern is similar to that used with the torus network. Specifically, the arrival rate λ_{sd} for a source-destination pair (s, d) is given by (note that no shortest path is longer than 4 hops):

$$\lambda_{sd} = \begin{cases} 0.5, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 1 \\ 0.4, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 2 \\ 0.3, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 3 \\ 0.2, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 4 \end{cases} \tag{10}$$

The second traffic pattern was designed to reflect actual traffic statistics collected on the NSFNET backbone network, as reported in the traffic matrix in [5, Figure 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 [5, Figure 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 specific traffic pattern reported in [5].

Figures 8 and 9 present the call blocking probabilities for the selected pairs of Table 4 and the first traffic pattern. Figure 8 shows results for fixed routing, while Figure 9 shows results for alternate routing with one alternate path per call. Because of the irregular topology, the alternate paths for some of the calls are 6 hops long. The link utilizations are in the range [1.846, 5.668] with an average of 3.494 under fixed routing, while for alternate routing they are slightly higher, in the range [1.964, 5.722] with an average of 3.646.

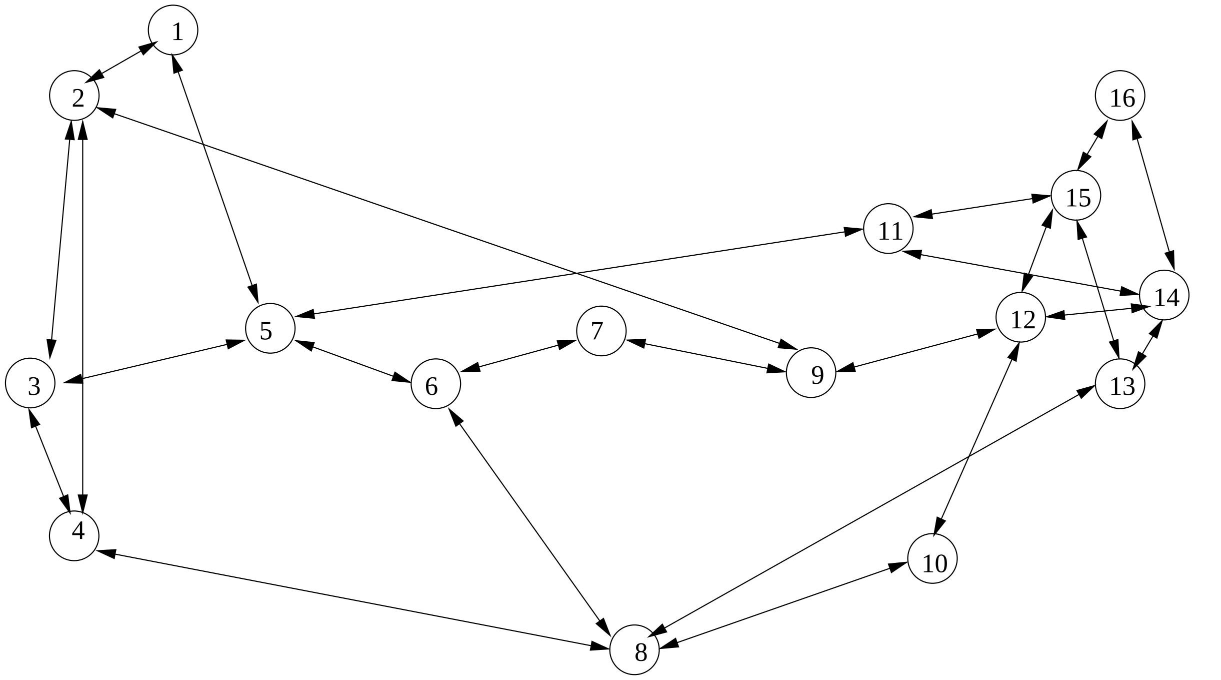

Figure 7: The NSFNET topology

Table 4: Selected source-destination pairs for the NSFNET topology

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

Comparing Figure 8 to Figure 5 we observe that, in the NSFNET topology, as in the regular 5×5 torus network, 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 Figure 8 that the blocking probabilities of calls established over, say, 1-hop paths vary widely depending on the number of other calls using the same path. Similar observations can be made by comparing Figures 9 and 6. 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 Figures 8 and 9. As with the torus network, we also note that the assumption that overflow traffic is Poisson does not appear to affect the performance of our algorithm when alternate routing is used.

Figures 10 and 11 are similar to Figures 8 and 9, respectively, except that they present results for the second traffic pattern derived from the traffic statistics presented in [5]. The utilization under this traffic pattern is in the range [0.000, 0.000] with an average of 0.000 for fixed routing, and in the range of [0.000, 0.000] with an average of 0.000 for alternate routing with one alternate path per call. As we can see, the relative behavior of the two curves (obtained through the analytical techniques and simulation, respectively) in Figures 10 and 11 is very similar to that in Figures 8 and 9, and all our previous conclusions regarding the accuracy of our decomposition algorithm are still valid, despite the fact that 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 5 to 8 present a summary comparison of analytical and simulation results for all 240 source-destination pairs of the NSFNET topology for both fixed and alternate routing and for both traffic patterns used. The high maximum relative difference values can be explained by the fact that, in a irregular topology such as the NSFNET in Figure 7, 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 Figures 10 and 11 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%. 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.

4.3 Converter Placement

We now consider the problem of determining the best placement of K converters in a N-node network, $N > K$, that optimizes a given objective function. We emphasize that our intention is simply to demonstrate how the decomposition algorithm presented earlier can be used to gain insight into the problem of converter placement in a wavelength routing network. A comprehensive study of this potentially difficult problem is outside the scope of this paper, and will be undertaken in future work.

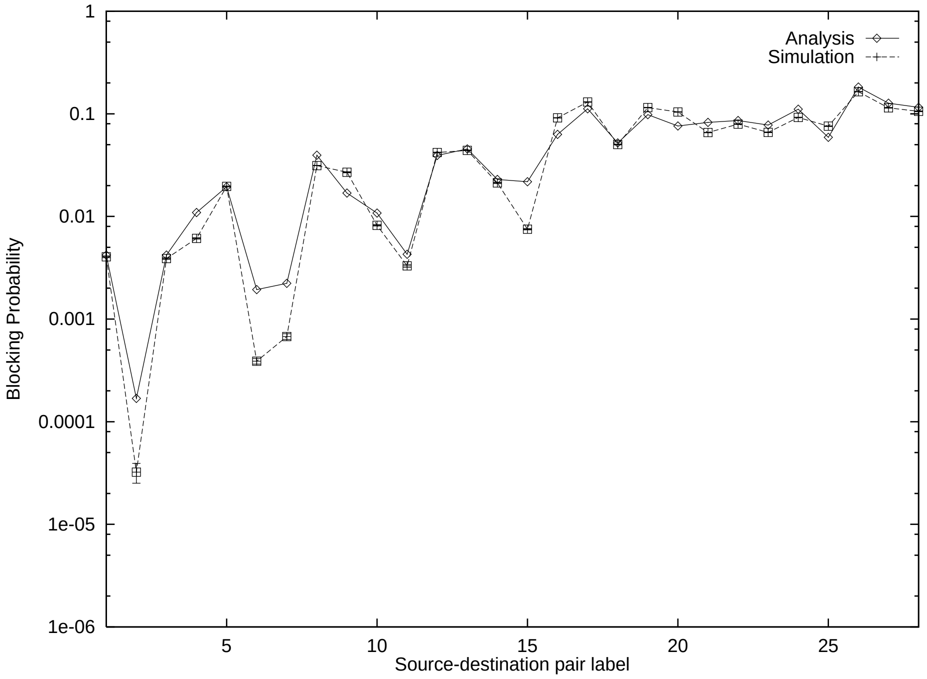

Figure 8: Blocking probability for selected source-destination pairs in the NSFNET with $W = 10$ and fixed routing (first traffic pattern)

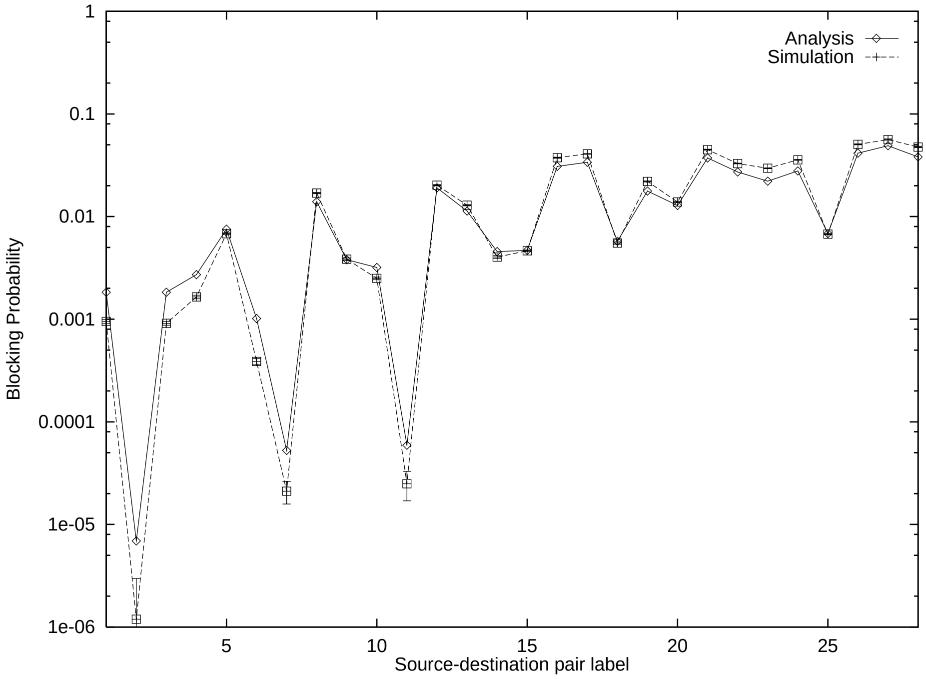

Figure 9: Blocking probability for selected source-destination pairs in the NSFNET with $W = 10$ and two alternate paths per pair (first traffic pattern)

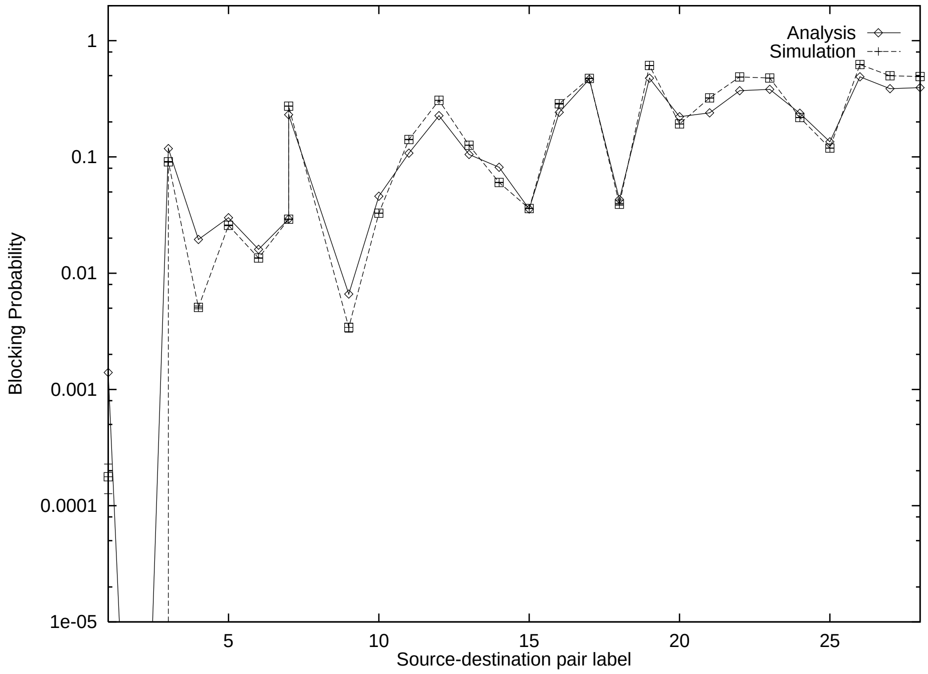

Figure 10: Blocking probability for selected source-destination pairs in the NSFNET with $W = 10$ and fixed routing (second traffic pattern)

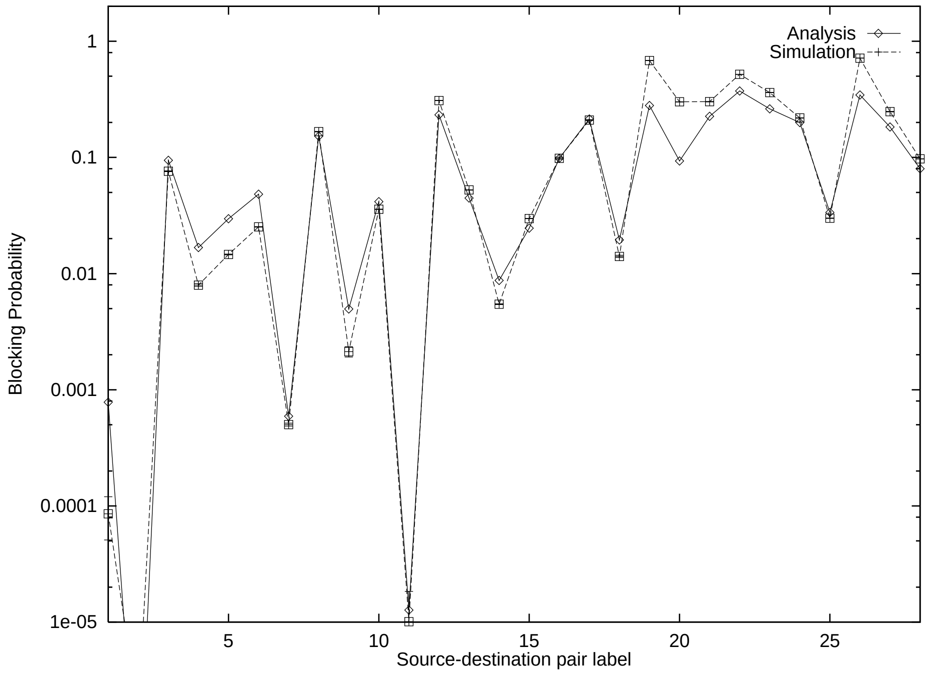

Figure 11: Blocking probability for selected source-destination pairs in the NSFNET with $W = 10$ and two alternate paths per pair (second traffic pattern)

Table 5: 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 6: Summary of results for the NSFNET topology with one alternate path (first traffic pattern)

Length of shortest path	Absolute Difference			Relative Difference		
	minimum	average	maximum	minimum	average	maximum
1	0.0000e+00	7.3909e-04	1.1693e-02	0.00%	47.81%	100.00%
2	0.0000e+00	5.3932e-03	5.0883e-02	0.00%	54.96%	99.30%
3	2.3918e-04	1.8906e-02	1.0082e-01	0.52%	62.34%	97.24%
4	1.5798e-03	4.2221e-02	1.4450e-01	3.20%	50.86%	84.31%

Table 7: 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%

Table 8: Summary of results for the NSFNET topology with one alternate path (second traffic pattern)

Length of shortest path	Absolute Difference			Relative Difference		
	minimum	average	maximum	minimum	average	maximum
1	0.0000e+00	6.5743e-03	8.5910e-02	0.00%	55.16%	100.00%
2	0.0000e+00	3.1628e-02	3.8645e-01	0.00%	36.75%	95.55%
3	0.0000e+00	7.9681e-02	4.2839e-01	0.00%	28.68%	95.74%
4	1.3680e-04	1.1138e-01	3.7035e-01	0.10%	26.33%	59.55%

The issue of converter placement in a single path of a network has been studied in [18, 16]. In [18], we used the single-path decomposition algorithm summarized in Section 2 to obtain the location of l converters in a k-hop path so as to minimize the end-to-end blocking probability (note that the blocking probability of end-to-end calls is always the highest among all calls in a single path). In [16], a dynamic programming algorithm for converter placement in a single path was presented. The objective was to minimize either the average or the end-to-end blocking probability, assuming negligible correlation of link loads.

In a mesh network, average blocking probability is not an appropriate measure of performance since it may not be representative of the actual blocking probability experienced along individual paths in the network. Therefore, our objective in this work is to show how our techniques can be used to obtain the best placement of K converters so as to minimize the maximum blocking probability over all source-destination pairs without reducing the aggregate utilization of the network. One approach to the converter placement problem would be to enumerate all possible ways of placing K converters in the network topology and to evaluate each by computing the maximum blocking probability. Since there are $\binom{N}{K}$ ways of placing K converters at the N nodes of the network, this approach will be prohibitively time consuming for anything but very small networks. Therefore, we apply a heuristic consisting of the following straightforward steps [3] to obtain a near-optimal placement of K converters.

1. Initialize the number of converters k to zero.
2. Solve the network with k converters using the appropriate decomposition algorithm (for fixed or alternate routing) described in Section 3.
3. Find the source-destination pair and corresponding path r with the highest blocking probability such that not all nodes of r have converters.
4. Place a converter on the node of path r with the highest amount of transit traffic that does not already have a converter.
5. Repeat from Step 2 with $k = k + 1$ until all K converters have been placed.

In Figure 12 we plot the maximum blocking probability in the NSFNET topology with the first traffic pattern under the near-optimal placement of $k,\ k = 1, \cdots, 16$, converters determined by the above heuristic. Results for both fixed and alternate (with one alternate path per source-destination pair) routing are presented. For comparison purposes, we also plot the maximum blocking probability for a network without converters (the values for zero converters in the figure). Figure 13 shows similar results for the second traffic pattern. The node at which a converter is placed at each step of the heuristic is also shown next to the various curves in Figures 12 and 13.

[3] We emphasize that other heuristics may be possible for this problem, and may even have better performance than the one considered here. However, as we mentioned earlier, our goal is not to study the converter placement problem in depth, but rather to illustrate how our decomposition algorithm can be an important tool in such a study.

As expected, the blocking probability drops as the number of converters increases. However, after an initial steep drop, the curves become essentially flat as the number of converters increases (note that the blocking probability values are plotted in logarithmic scale, therefore, the initial decreases in blocking probability are much more significant, in absolute value, than later ones). This behavior is consistent with the results of earlier work [15, 16]. An important observation from Figures 12 and 13 is that the effect of converters on the blocking probability is strongly dependent on the actual traffic pattern. For instance, consider the first traffic pattern with fixed routing. When four converters are employed, the maximum call blocking probability drops from to a value of 0.121 from a value of 0.187 when there are no converters, a difference of 35%. In contrast, increasing the number of converters has a more dramatic effect for the second traffic pattern. For the latter, the maximum call blocking probability drops from 0.219 when there are no converters to 0.096 with four converters, a decrease of 56%.

We also note that the effect of wavelength conversion is more pronounced when alternate routing is used. Let us again consider the improvement in maximum blocking probability when four converters are used over a network with no converters. For the first traffic pattern, the maximum blocking probability decreases from 0.061 to 0.025, a drop of 59%, higher than the 35% decrease for fixed routing. Similarly, under the second traffic pattern, the maximum call blocking probability drops to 0.015 from 0.039, a drop of 62% compared with the 56% drop under fixed routing. This increase in the performance improvement can be explained by noting that, under fixed routing, a converter at node f will only benefit those source-destination pairs whose path passes through node f. With alternate routing, on the other hand, a converter at node f will benefit a larger number of source-destination pairs, namely, those with at least one path passing through node f. Consequently, the performance benefits of wavelength conversion are higher when multiple paths per source-destination pair are used.

Let us now consider the location of converters for the two traffic patterns, as determined by our heuristic. We observe that the placement depends both on the load pattern and on whether fixed or alternate routing is used. The fact that the near-optimal location depends on the load suggests that in a dynamic network environment where the traffic pattern varies 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 maximum call 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 call blocking probability over a wide range of load patterns.

5 Concluding Remarks

We have presented a new path decomposition algorithm to accurately and efficiently evaluate the call blocking performance of wavelength routing network 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 sub-systems. These sub-systems are analyzed

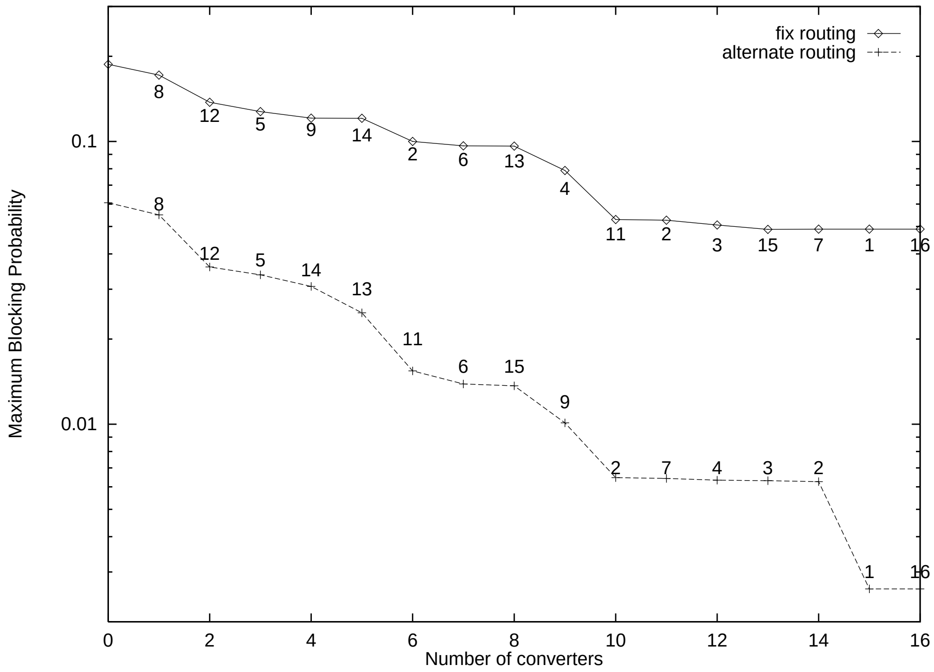

Figure 12: Maximum blocking probability in the NSFNET topology with converters (first traffic pattern)

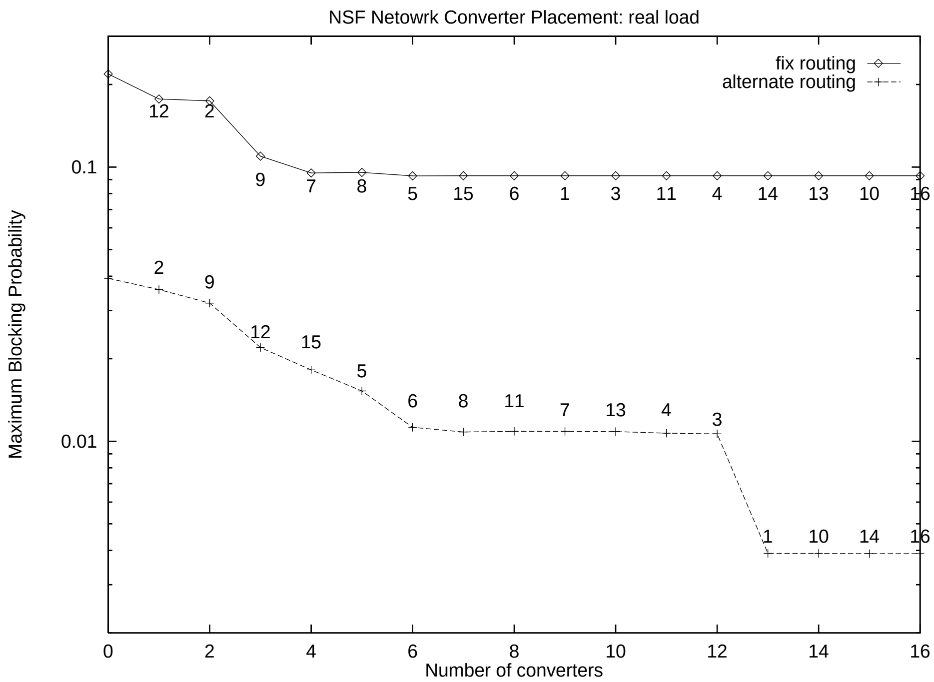

Figure 13: Maximum blocking probability in the NSFNET topology with converters (second traffic pattern)

in isolation by using our previous algorithms for a single path of wavelength routing networks, and the individual results 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.

We are currently working on extending our algorithm to work with other wavelength allocation policies, such as first-fit, which are more likely to be implemented in practice.

References

[1] R. A. Barry and P. A. Humblet. Models of blocking probability in all-optical networks with and without wavelength changers. *IEEE Journal Selected Areas in Communications*, 14(5):858–867, June 1996.

[2] A. Birman. Computing approximate blocking probabilities for a class of all-optical networks. *IEEE Journal Selected Areas in Communications*, 14(5):852–857, June 1996.

[3] I. Chlamtac, A. Ganz, and G. Karmi. Lightpath communications: An approach to high bandwidth optical WANS. *IEEE Transactions on Communications*, 40(7):1171–1182, July 1992.

[4] R. Cruz, G. Hill, A. Kellner, R. Ramaswami, G. Sasaki, and Y. Yamabayashi (Eds.). Special issue on optical networks. *IEEE Journal Selected Areas in Communications*, 14(5), June 1996.

[5] B. Mukherjee *et al.* Some principles for designing a wide-area WDM optical network. *IEEE/ACM Transactions on Networking*, 4(5):684–696, October 1996.

[6] I. P. Kaminow *et al.* A wideband all-optical WDM network. *IEEE Journal Selected Areas in Communications*, 14(5):780–799, June 1996.

[7] R. E. Wagner *et al.* MONET: Multiwavelength optical networking. *Journal of Lightwave Technology*, 14(6):1349–1355, June 1996.

[8] S. P. Monacos *et al.* All-optical WDM packet networks. *Journal of Lightwave Technology*, 14(6):1356–1370, June 1996.

[9] M. Fujiwara, M. Goodman, M. O'Mahony, O. Tonguz, and A. Willner (Eds.). Special issue on multiple wavelength technologies and networks. *Journal of Lightwave Technology*, 14(6), June 1996.

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

[11] H. Harai, M. Murata, and H. Miyahara. Performance of alternate routing methods in all-optical switching networks. In *Proceedings of INFOCOM '97*, pages 517–525. IEEE, April 1997.

[12] E. Karasan and E. Ayanoglu. Effects of wavelength routing and selection algorithms on wavelength conversion gain in wdm optical networks. *IEEE/ACM Transactions on Networking*, 6(2):186–196, April 1998.

[13] M. Kovacevic and A. Acampora. Benefits of wavelength translation in all-optical clear-channel networks. *IEEE Journal Selected Areas in Communications*, 14(5):868–880, June 1996.

[14] A. Mokhtar and M. Azizoglu. Adaptive wavelength routing in all-optical netowrks. *IEEE/ACM Transactions on Networking*, 6(2):197–206, April 1998.

[15] S. Subramaniam, M. Azizoglu, and A. Somani. All-optical networks with sparse wavelength conversion. *IEEE/ACM Transactions on Networking*, 4(4):544–557, August 1996.

[16] S. Subramaniam, M. Azizoglu, and A. K. Somani. On the optimal placement of wavelength converters in wavelength-routed networks. In *Proceedings of INFOCOM '98*, pages 902–909. IEEE, April 1998.

[17] S. Subramanian, A. K. Somani, M. Azizoglu, and R. A. Barry. A performance model for wavelength conversion with non-poisson traffic. In *Proceedings of INFOCOM '97*, pages 500–507. IEEE, April 1997.

[18] Y. Zhu, G. N. Rouskas, and H. G. Perros. Blocking in wavelength routing networks, Part I: The single path case. Technical Report TR-98-02, North Carolina State University, Raleigh, NC, February 1998.