## ABSTRACT

YUHONG ZHU. Computing Call Blocking Probabilities in Wavelength Routing Networks. (Under the direction of Dr. Harry G. Perros and Dr. George N. Rouskas.)

We study a class of circuit switched wavelength routing networks with fixed or alternate routing, with or without converts, and with various wavelength allocation policies. We first construct an exact Markov process and an approximate Markov process which has a closed-form solution for a single path. We also develop an iterative decomposition algorithm to analyze long paths with or without wavelength converters effectively. Based on this algorithm, we then present an iterative path decomposition algorithm to evaluate the blocking performance of mesh topology networks with fixed and alternate routing accurately and efficiently. The 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. Finally we show through numerical and simulation results that the blocking probabilities for the random wavelength allocation and the circuit-switched case provide upper and lower bounds on the blocking probabilities for two wavelength allocation policies that are most likely to be use in practice, namely most-used and first-fit allocation. Furthermore, we demonstrate that using these two policies has an effect on call blocking probabilities that is equivalent to employing converters at a number of nodes in the network.

# COMPUTING CALL BLOCKING PROBABILITIES IN WAVELENGTH ROUTING NETWORKS

by

**Yuhong Zhu**

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

**COMPUTER SCIENCE**

Raleigh

1999

**APPROVED BY:**

| | |
|---|---|
| Co-Chair of Advisory Committee | Co-Chair of Advisory Committee |

# BIOGRAPHY

Yuhong Zhu was born August 18, 1968 in Beijing, People's Republic of China. He received his B.S. in Electrical Engineering in 1990 from Beijing University, his M.S. in Chemical Physics in 1994 from The Ohio State University, and his M.S. in Computer Science in 1995 from North Carolina State University.

## ACKNOWLEDGEMENTS

First of all I would like to thank my wife, Yuying Zhang, without which this work would have been impossible. I also would like to thank the members of my PhD committee for their comments and guidance, particularly Dr. George Rouskas and Dr. Harry Perros.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

We study a class of circuit-switched wavelength routing networks with and without wavelength converters, with fixed routing and alternate routing, and with various wavelength allocation policies. We present a new analytical framework to evaluate the blocking performance of such networks accurately and efficiently. Our model is fairly general and it allows non-uniform traffic, i.e., call request arrival rates can vary with the source-destination pair. It also accounts for the correlation among the loads at all links in a path, and it can be used when the location of converters is fixed but arbitrary.

We first present the solution to a single path of a wavelength routing network with and without wavelength converters. Then we extend our algorithm to a mesh topology network with fixed routing and alternate routing. Finally we study several wavelength allocation policies.

The following sections briefly summarize our work on each of the problems.

## 1.1 Blocking in a Single Path of a Network

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 efficiently computed for short paths. We then develop an iterative algorithm to analyze approximately arbitrarily long paths. The algorithm decomposes a path into shorter segments which are then studied in isolation using the corresponding approximate Markov process. The individual solutions are appropriately combined to obtain a solution for the original path. Finally, we demonstrate how our

analytical techniques can be used to gain insight into the problem of converter placement along a path.

## 1.2 Blocking in Mesh Topologies

We present an iterative path decomposition algorithm to evaluate the blocking performance of such networks with and without wavelength converters, and with fixed routing or alternate routing. 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. We also demonstrate how our analytical techniques can be applied to gain insight into the problem of converter placement in mesh topologies.

## 1.3 Comparison of Wavelength Allocation Policies

We show through numerical and simulation results that the blocking probabilities obtained through our previous analytical expressions for the random wavelength allocation and the circuit-switched case provide upper and lower bounds on the blocking probabilities for two wavelength allocation policies that are most likely to be used in practice, namely, most-used and first-fit allocation. Furthermore, we demonstrate that using either of these two policies has an effect on call blocking probabilities that is equivalent to employing converters at a number of nodes in the network. These results have been obtained for both single-path and general mesh topology networks, and for a wide range of loads. The main conclusion of our work is that the gains obtained by employing specialized and expensive hardware (namely, wavelength converters) can be realized cost-effectively by making more intelligent choices in software (namely, the wavelength allocation policy).

## 1.4 Thesis Organization

In Chapter 2 we present the main features of optical networks and introduce the subject of all-optical WDM networks. In Chapter 3 we review the previous work done in the area of call blocking in wavelength routing networks. Chapter 4 contains a new analytical

framework for a single path. In Chapter 5 we extend the single path method to mesh topologies. In Chapter 6 we compare different wavelength allocation policies. Finally in Chapter 7 we conclude our work and discuss future research.

# Chapter 2

# Wavelength Routing Networks

## 2.1 Why Optical Networking?

The growing demand for more bandwidth has been pushed by the increasing number of users in the Internet. Multi-media traffic which combines the voice, data and video consumes large amount of bandwidth. There is not enough capacity of our network today to support the exponential growth in users' traffic. The traditional copper wire, even coaxial cable, is limited by electronic speeds to a few Gbps. Radio has a total bandwidth of 25 GHz which is still insufficient. In contrast, fiber can offer huge bandwidth nearly 50 terabits per second (Tbps). Optical networks are emerging as a replacement of traditional copper wire networks.

In addition to its huge bandwidth, fiber offers low signal attenuation ( as low as 0.2dB/km) and low signal distortion. The low error rate of fiber can simplify the conventional reliability improvement methods used on coppers. Optical networks are also easy to maintain because of their lower power requirement, low material usage, and small space requirement. From many years now, fiber has been installed at the rate of 4000 strand miles per day. Also, the sales of optical networking equipment based on wavelength division multiplexing (WDM) technology has risen to more than one billion dollars. Meanwhile, all-optical networks (AON), using optical switches instead of electronic devices along the signal path, are emerging from laboratories into field trials [21, 25, 11, 26].

## 2.2 Evolution of Optical Networks

The first-generation optical networks simply use optical fiber as a transmission medium to replace copper cables. Electronic devices handle the switching and processing of the bits. Such kinds of networks as SONET (synchronous optical network) and SDH (Synchronous digital hierarchy) are deployed in wide area networks (WAN), while FDDI (fiber distributed data interface) is used in local area network (LAN). The transfer rate can reach up to10Gb/s.

To increase the capacity on existing fiber, wavelength division multiplexing (WDM) is employed. The traditional time devision multiplexing (TDM) demands that each port handles not only its own bits but also those belonging to other ports. The TDM bit rate may be much higher than the electronic processing speed. WDM, on the other hand, which handles only its own bits at each port, does not have such requirement. While the technology of optical TDM and CDM (code division multiplexing) are still studied in research laboratories, WDM technology is mature and widely used in the industry. WDM transmits data simultaneously on multiple carrier wavelengths over the same fiber, i.e., each wavelength supports a single communication channel operating at whatever rate one desires. Furthermore, WDM is more economical than the traditional way for higher baud rates of transmission. Today, WDM involves mainly point-to-point communication systems. However, fiber interconnection devices, such as passive star and optical switches, provide the possibility of a purely optical network, or an all-optical network (AON).

### 2.2.1 Point-to-Point WDM Systems

WDM point-to-point communication systems are being deployed by several telecommunication companies. Figure 2.1 shows a typical four-channel point-to-point WDM transmission system. The maximum number of channels is about 40 so far, but this number is likely to increase in the near future. However, these systems are not optical networks by themselves but a convenient multiplexing scheme.

### 2.2.2 Wavelength Add/Drop Multiplexers

A Wavelength Add/Drop Multiplexer (WADM) enables a small number of contiguous wavelengths to be added and dropped without demultiplexing the entire wavelength bundle. Figure 2.2 shows a WADM consisting of a demux, a set of 2 x 2 switches and a

mux. With the exception of $\lambda_i$, $\lambda_1$ and $\lambda_w$ flow through the 2 x 2 switches in the "bar" state. However, the 2 x 2 switches for $\lambda_i$ is configured in the "cross" state. The signal on the wavelength $\lambda_i$ is "dropped" locally, and a new signal can be "added" on to the same wavelength at this WADM location.

### 2.2.3 Broadcast-and-Select Networks

The broadcast and select star network shown in Figure 2.3 is most useful for LANs and MANs. Network nodes are connected to a passive star via two-way fibers. Different nodes transmit on different wavelengths. Their signals are optically combined by the star and broadcasted on their receive fibers to all of the nodes. Each node uses an optical filter in order to receive the desired wavelength. The passive star network can also support multicast services because any number of receiving nodes can tune to the same particular wavelength. Since the end nodes perform the switching functions in terms of wavelength tunability, the broadcast and select networks can still function if one node is down. There are two types of broadcast and select star networks: single-hop and multihop. The single-hop network assumes that a tunable device can access all the available wavelengths, which means that any node can reach any other node. In a multihop network, a tunable device can only tune to some of the available wavelengths but not to all of them, which means that not every node is capable of communicating with any other node within a single hop.

### 2.2.4 Wavelength Routed Networks

A wavelength-routed optical network is shown in Figure 2.4. Such a network is highly scalable and suitable for wide area networks (WANs). A wavelength routing network consists of wavelength routers and the fiber links that interconnect them. A wavelength router is an optical switches capable of routing the light signal of a given wavelength from any input port to any output port. In this way, it is possible to establish end-to-end lightpaths, i.e., direct optical connections without any intermediate electronics. The same wavelength may be spatially reused to carry multiple connections through these devices. A lightpath uses the same wavelength on every link in its path. This requirement is referred as the wavelength continuity constraint. For example in Figure 2.4, a signal travels from Host 2 to Host B using wavelength $W_2$. If a signal originates at Host 3 on wavelength $W_2$ and reaches the middle wavelength router $R_2$, it cannot continue toward Router $R_3$ because the

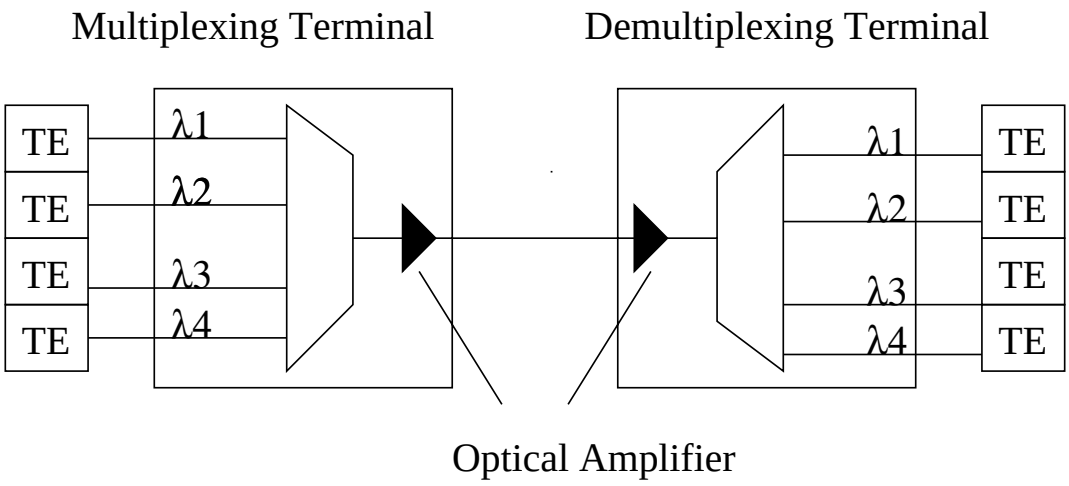


Figure 2.1: A four-channel point-to-point WDM transmission system

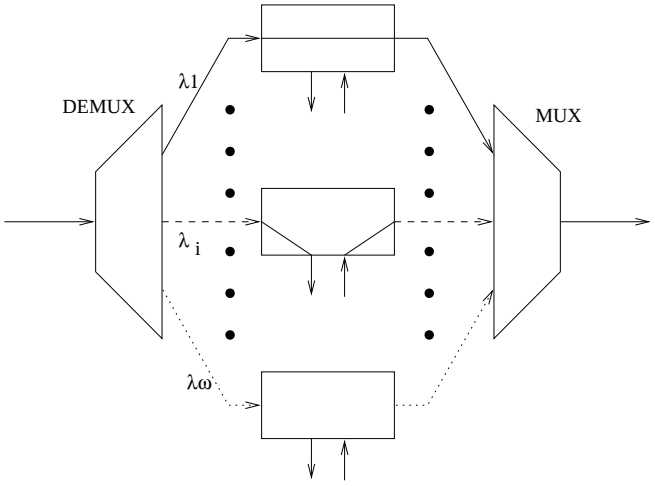


Figure 2.2: A Wavelength Add/Drop Multiplexer (WADM)

wavelength $W_2$ is already taken by the signal from Host 2. The signal cannot be switched to wavelength $W_1$ even though $W_1$ is free.

## 2.3 Routing and Wavelength Assignment

In wavelength-routed optical networks, a lightpath can be set up and taken down upon demand. A source-destination connection may be established over several different paths. Routing algorithms must be used to determine which path to choose. Once the path is found, a wavelength is assigned to it.

Routing algorithms can be classified into static and adaptive. A static route process does not vary with time, whereas an adaptive route may vary with time. Two widely used static routing techniques are: fixed routing and fixed alternate routing. In fixed routing, each source-destination pair is assigned to a single path. If there is no free wavelength available on the path, calls are blocked. In fixed alternate routing, a set of paths (consisting of one primary path and one or more alternate paths) is assigned to each source-destination pair. A call use the alternate paths if it is blocked on the primary path. The set of alternate routing is searched in a fixed order in order to find an available path. Adaptive routing in circuit-switched networks such as Dynamic Alternate Routing and Dynamic Nonhierarchical Routing provide flexibility, efficiency and robustness to the systems. Dynamic routing tries to find the "best" path through the network based on the state information of the entire network.

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 an incoming call to one of the (possibly many) available wavelengths. In conventional circuit switched networks, all circuits are similar.

The following wavelength allocation policies have been proposed: random, most-used, least used and first-fit. Random method means that a wavelength from the set of available wavelengths is randomly selected. Most-used method selects an available wavelength that is most utilized on other links. Least-used method selects an available wavelength that is least utilized on other links. In the first-fit method, all wavelengths are numbered from 1 to $N$. The available wavelength with the smallest index is chosen. Random and

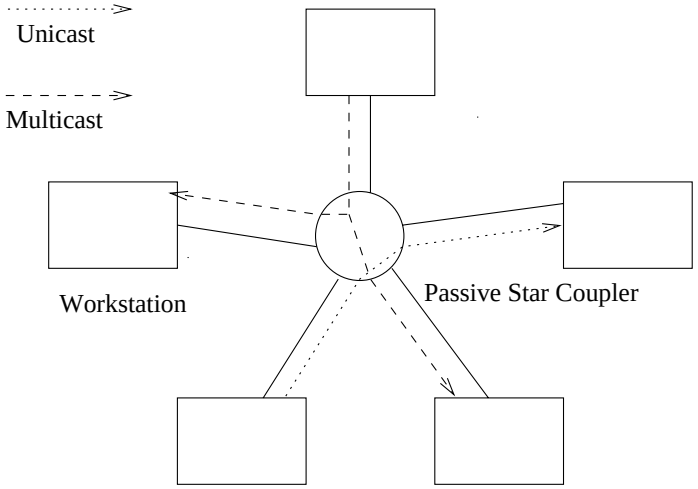


Figure 2.3: A broadcast-and-select (local) optical WDM networks

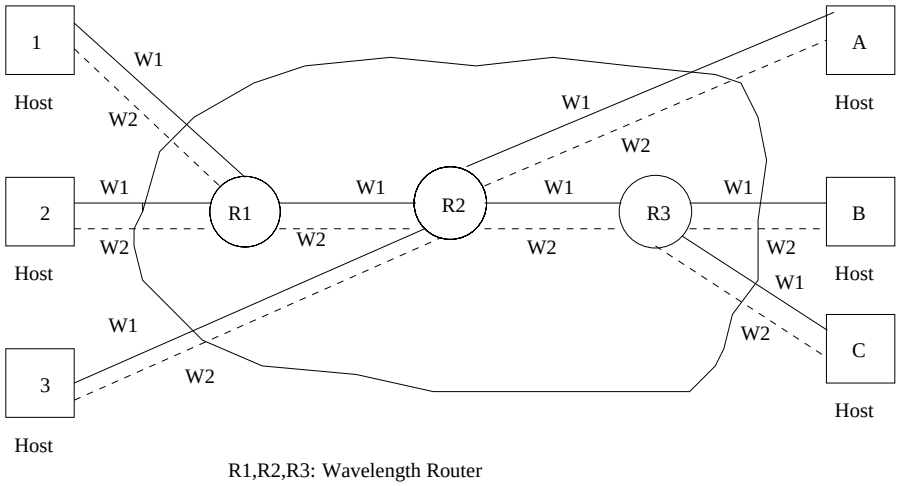


Figure 2.4: A wavelength-routed (wide-area) optical WDM network

first-fit policies are easy to implement in practice. On the other hand, most-used and least-used policies have to select a wavelength based on the current state of the whole set of wavelengths.

## 2.4 Wavelength Conversion

Wavelength conversion can improve the efficiency of WDM networks. A wavelength converter is capable of shifting an incoming wavelength to a different outgoing wavelength. It eliminates the wavelength continuity constraint. Back to the example in Figure 2.4, if we put a wavelength converter at the middle wavelength router $R_2$, the signal from Host 3 can be switched from wavelength $W_2$ to $W_1$, and it can continue to travel towards router $R_3$.

A WDM network with a wavelength converter at every node behaves in the same way as the traditional circuit-switched network, and it has a lower blocking probability. However, any gains in performance must be weighted against the cost of wavelength converters. Thus, limited wavelength conversion raise a new problem: optimal placement of converter. Given a network topology and a fixed number of converters, an optimal placement algorithm can decide where the converters should be placed to achieve the best performance for different types of traffic. In practice, wavelength converters can only convert wavelengths in a certain range. Limited-range converters may still be economical and achieve the same performance as the full-range converters.

# Chapter 3

# Previous Work

## 3.1 Wavelength Routing Algorithms

The call blocking performance of optical wavelength routing networks with static (fixed or alternate) routing, random wavelength allocation policy has been studied in [4, 2, 18, 3, 12, 28, 15, 20, 31]. In [4], a six-node network is used with fixed and dynamic alternate routing (using a small set of alternate paths for each origin-destination pair) under the random and first-fit wavelength selection algorithms. Blocking probabilities with and without wavelength converters are obtained by simulation. The routing strategy used is not adaptive to network load, and the simulations were performed only for a particular value of the offered load. The model presented in [2] 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 does not capture the dynamic nature of traffic. In [18] it was assumed that statistics of link loads are mutually independent, an approximation that is not accurate for sparse network topologies. In [3] a Markov chain with state-dependent arrival rates was developed to model call blocking in arbitrary mesh topologies with fixed routing; it was extended in [12] 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 [28] to compute recursively blocking probabilities assuming that the load on link $i$ of a path depends only on the load of link $i-1$. A dynamic routing algorithm that selects the least loaded path-wavelength pair was studied in [15], and in [20] an unconstrained dynamic routing scheme with a number of wavelength allocation policies was evaluated. Finally, a study of call

blocking under non-Poisson input traffic was presented in [31], under the assumption that link loads are statistically independent.

Most of the approximate analytical techniques developed for computing blocking probabilities in wavelength routing networks [18, 3, 12, 31, 15, 20, 29] make the assumption that link blocking events are independent, 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, more accurate analytical tools are needed both to evaluate efficiently the performance of these networks, as well as to tackle complex network design problems, such as selecting optical switches on which to employ wavelength converters.

## 3.2 Wavelength Conversion

The work in [24] gives lower bound on the blocking probabilities in networks with and without wavelength converters using an integer linear programming formulation. The study considers only fixed shortest path routing and the first-fit wavelength allocation algorithm. A wavelength conversion gain corresponding to $10\% - 40\%$ increase in wavelength reuse (utilization) is shown. However, these gains are obtained at a blocking probability of $10^{-2}$. In [33], an integer linear programming formulation of the same problem was presented for multihop networks. Blocking probabilities for a 24-node mesh network are reported, where wavelength assignment and path selection are performed by using separate heuristic algorithms.

In [14], it was shown that intermediately connected networks, such as a mesh network, have the largest gain with wavelength conversions. In [19], unconstrained routing is used in conjunction with an exhaustive search over the wavelength set in order to evaluate the effects of wavelength converters, assuming that a limited number of wavelength converters and that each converter has no restrictions on the wavelengths of the channels it can connect.

In [2], an analytic algorithm for a single path with converters under simple traffic models is presented. The work in [28] takes a probabilistic approach in modeling wavelength conversion by introducing the converter density, which represents the probability that a node is capable of wavelength 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 [29] 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.

Limited-range wavelength conversion, i.e., those wavelength converters can only switch one wavelength into a certain set of wavelengths, also has been studied in the literature. In [23], it was shown that there are ring and star networks with minimal wavelength conversion capabilities that can perform off-line channel assignment as well as networks with full wavelength conversion. In [32] , a network with limited wavelength conversion is used to study the performance due to limited wavelength shifting capability of devices based on four wave mixing. There are some recent results on the online channel assignment problem for ring and tree networks [7, 9], where lightpath requests arrive and leave the network dynamically. The problem of recovering from link and node faults in ring networks using limited wavelength conversion is addressed in [8]. In [27], the results demonstrate that limited wavelength translation of fairly small degree is sufficient to obtain benefits comparable to those obtained by full wavelength translation, using an analytical model. [13] develops an approximate analytic method to allow flexible wavelength assignment policies and limited-range wavelength conversion by using virtual of layered-graph approach.

The performance of arbitrary network topologies with any number of wavelength converters has not been studied in the literature yet. Although an optimal converter placement for a single path was proposed in [29], the converter placement problem for mesh topologies has not been addressed yet.

## 3.3 Wavelength Allocation Policies

In [1], an integer programming formulation of the optimal routing and wavelength assignment problem combined with randomized rounding was presented. The objective of

the solution is to minimize the number of wavelengths needed, given a set of lightpath requests. The work in [30] evaluates the performance improvement due to the deployment of multiple fibers between node pairs. The most-used wavelength selection algorithm selects the available wavelength which is currently utilized on the largest number of fibers. First-fit wavelength allocation was studied using simulation in [5, 18], and it was shown to perform better than random allocation, while an analytical overflow model for first-fit allocation was developed in [20], and was extended to non-Poisson model in [15]. However, current research work does not relate the performance of wavelength allocation policies (software) and wavelength converters (hardware).

# Chapter 4

# Blocking in Single Path of a Network

## 4.1 Wavelength Routing Network Model

We consider a single path in a circuit-switched wavelength routing network. Each link in the path supports exactly $W$ wavelengths, and each node is capable of transmitting and receiving on any of these $W$ 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. If the request can be satisfied, an optical circuit is established between the source and destination for the duration of the call. Call holding times are exponentially distributed with a mean that also depends on the source-destination pair.

In our model, we allow some of the nodes in the path to 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-switching networks, a special case of the more general scenario discussed here.) If no wavelength converters are employed in the path, a call can only be established if the *same* wavelength is free on all the 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 [1]. Thus, we only consider the random wavelength assignment

[1] In a path with wavelength converters, a wavelength is randomly assigned within each segment of the

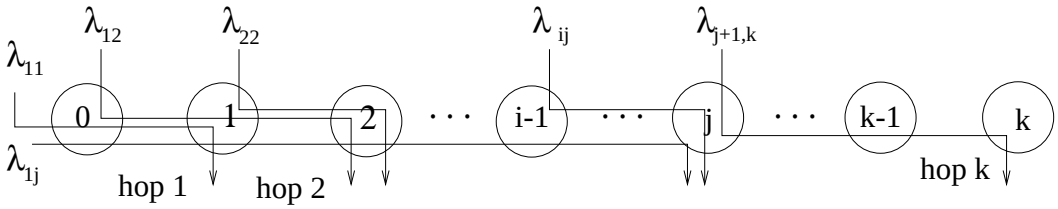


Figure 4.1: A $k$-hop path

policy in this chapter.

We define a path "segment" as a sub-path consisting of one or more consecutive links of the original path. (Unless noted otherwise, the terms "hop" and "link" will be used interchangeably.) The following notation will be used throughout the paper (refer to Figure 4.1).

- A $k$-hop path consists of $k+1$ nodes labeled $0, 1, \cdots, k$. Hence, hop $i, i = 1, \cdots, k$, represents the link between nodes $i-1$ and $i$.
- $\lambda_{ij}, j \geq i$, is the Poisson arrival rate of calls that use hops $i$ through $j$, inclusive. For instance, $\lambda_{22}$ is the arrival rate of calls that only use hop 2 (that is, those arriving at node 1 and leaving at node 2), while $\lambda_{12}$ is the arrival rate of calls using hops 1 and 2 (i.e., those that arrive at node 0 and leave at node 2).
- $1/\mu_{ij}, j \geq i$, is the mean of the (exponentially distributed) holding time of calls using hops $i$ through $j$, inclusive. Also, $\rho_{ij} = \lambda_{ij}/\mu_{ij}$ is the offered load of these calls.
- $n_{ij}$, $j \geq i$, is the number of calls using hops $i$ through $j$, inclusive, that are currently active in the network.
- $f_{ij}, j \geq i$, is the number of wavelengths that are free on all hops $i$ through $j$, inclusive.

## 4.2 Exact and Approximate Markov Process Models

In this section we present exact and approximate Markov process models that capture the dynamic behavior of a $k$-hop path in terms of wavelength use. The approximate Markov process has a closed-form solution for its steady-state probability, and it provides

path consisting of links between successive nodes with converters.

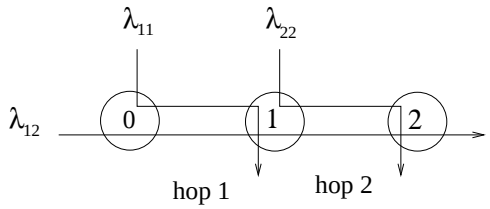


Figure 4.2: A two-hop path

an accurate approximation to the blocking probabilities obtained through the exact Markov process. We first study paths without wavelength converters (in Section 4.2.1), and then we extend our results to paths with converters employed at some nodes (in Section 4.2.2).

### 4.2.1 Paths With No Wavelength Conversion

#### Exact Markov Process Model

Let us first consider the 2-hop path (without converters) shown in Figure 4.2. It is straightforward to verify that the evolution of this system can be characterized by the four-dimensional Markov process $(n_{11}, n_{12}, n_{22}, f_{12})$. In addition, 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. This condition leads to the following two constraints that the parameters of the Markov process must satisfy:

$$n_{11} + n_{12} + f_{12} \quad \leq \quad W \quad \text{and} \quad n_{12} + n_{22} + f_{12} \quad \leq \quad W \tag{4.1}$$

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 $k^2$ parameters in a state $\underline{n}$ of Markov process $\mathcal{M}_k$, as follows:

$$\underline{n} = (n_{11}, n_{12}, \cdots, n_{1k}, n_{22}, \cdots, n_{2k}, \cdots, n_{kk}, f_{12}, f_{13}, \cdots, f_{1k}, f_{23}, \cdots, f_{2k}, \cdots, f_{k-1,k}) \tag{4.2}$$

The first $\frac{k(k+1)}{2}$ parameters $n_{ij}, 1 \leq i \leq j \leq k$, in the state description (4.2) provide the number of active calls between all possible source-destination pairs in the path. The last $\frac{(k-1)k}{2}$ parameters $f_{ij}, 1 \leq i < j \leq k$, represent the number of wavelengths that are free on all segments of the path consisting of two or more links. The following constraints are imposed on the state space of Markov process $\mathcal{M}_k$:

$$f_{ij} \quad \leq \quad f_{i,j-1} \quad \leq \quad \cdots \quad \leq \quad f_{i,i+1}, \quad 1 \leq i < j \leq k \tag{4.3}$$

$$f_{ij} \leq f_{i+1,j} \leq \cdots \leq f_{j-1,j}, \quad 1 \leq i < j \leq k \tag{4.4}$$

$$\begin{cases} \sum_{j=1}^{k} n_{1j} + f_{12} \leq W, & l = 1 \\ \sum_{i=1}^{l} \sum_{j=l}^{k} n_{ij} + f_{l-1,l} + f_{l,l+1} - f_{l-1,l+1} \leq W, & l = 2, \cdots, k-1 \\ \sum_{i=1}^{k} n_{ik} + f_{k-1,k} \leq W, & l = k \end{cases} \tag{4.5}$$

The set of constraints (4.3) (respectively, (4.4)) account for the fact that, if a wavelength is free on all hops of a $m$-hop segment, then it is also free on the first (respectively, last) $m-1$ hops of the segment. The $k$ constraints (4.5) ensure that the number of wavelengths (used or free) on each hop of the path does not exceed $W$. For instance, let us refer to the first constraint in (4.5); the others can be explained using similar arguments. The sum in the left hand side of the constraint corresponds to the total number of active calls that use hop 1, while $f_{12}$ is the total number of wavelengths that are free on both hops 1 and 2. By definition (see Section 4.1 and constraint (4.3)), parameter $f_{12}$ includes the free wavelengths on all path segments of two or mode hops starting with hop 1. Therefore, the terms in left hand side of this constraint must sum up to at most $W$ (if they sum up to *less than* $W$, the difference represents the number of wavelengths that are free on hop 1 but are not free on hop 2, and hence, can only be assigned to calls using only hop 1).

We have the following lemma.

**Lemma 4.2.1** *The stochastic process defined in (4.2) is a Markov process.*

**Proof.** See Appendix A.1. □

Markov 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. Unfortunately, the large number of parameters in its state description makes it impossible to numerically solve the MC 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 Figure 4.3 we show the state space $(n_{11}, n_{12}, n_{22}, f_{12})$ and the transition rates of Markov process $\mathcal{M}_2$, for $W = 2$ wavelengths. From the figure we can see that there exists a sequence of states, $\underline{n_1}, \cdots, \underline{n_s}$, such that

$$r(\underline{n_1}, \underline{n_2}) r(\underline{n_2}, \underline{n_3}) \cdots r(\underline{n_{s-1}}, \underline{n_s}) r(\underline{n_s}, \underline{n_1}) \neq r(\underline{n_1}, \underline{n_s}) r(\underline{n_s}, \underline{n_{s-1}}) \cdots r(\underline{n_3}, \underline{n_2}) r(\underline{n_2}, \underline{n_1}) \tag{4.6}$$

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 [16, Theorem 1.8] is not satisfied, and the

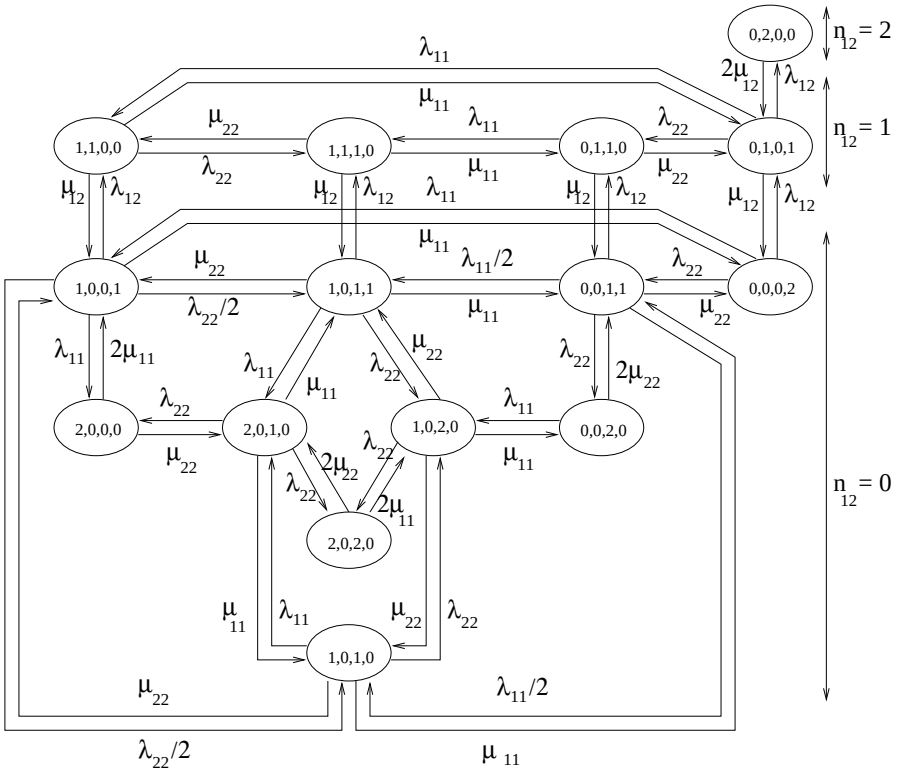


Figure 4.3: State space $(n_{11}, n_{12}, n_{22}, f_{12})$ of a 2-hop path with $W = 2$ wavelengths

Markov process is not time-reversible. It is straightforward to show that this result is true in general, and that Markov process $\mathcal{M}_k, k \geq 2$ is not time-reversible when $W > 1$.

In the following subsection we show how to modify $\mathcal{M}_k$ to obtain an approximate, time-reversible Markov process that has a closed-form solution.

### Approximate Time-Reversible Markov Process Model

A closer examination of Figure 4.3 reveals that the two four-state sequences mentioned above are the shortest sequences of states for which (4.6) holds true; any other sequence that satisfies (4.6) contains one of these two sequences. We also note that both four-state sequences involve transitions that cause changes in the value of parameter $n_{12}$. 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 parameter $n_{12} = c$:

$$\mathcal{L}_{2,c} \;=\; \{(n_{11}, n_{12}, n_{22}, f_{12}) \;:\; n_{12} \;=\; c\} \;, \qquad c = 0, \cdots, W \tag{4.7}$$

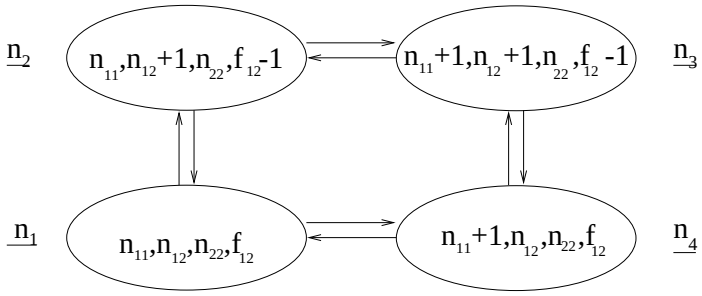


Figure 4.4: Four-state sequence with states from two different sub-chains $\mathcal{L}_{2,c}$

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 that includes only states from a certain 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 $\underline{n_4} = (n_{11}+1, n_{12}, n_{22}, f_{12})$, shown in Figure 4.4, which includes states from two different sub-chains [2]. We have that:

$$\begin{aligned} r(\underline{n_1}, \underline{n_2}) &= \lambda_{12}, & r(\underline{n_3}, \underline{n_4}) &= (n_{12}+1)\mu_{12} \\ r(\underline{n_2}, \underline{n_1}) &= (n_{12}+1)\mu_{12}, & r(\underline{n_4}, \underline{n_3}) &= \lambda_{12} \end{aligned} \tag{4.8}$$

so these transition rates balance along the two directions. However, the rates of the other transitions do not balance, since [3]:

$$\begin{aligned} r(\underline{n_2}, \underline{n_3}) &= \lambda_{11}\left(1 - \tfrac{f_{12}-1}{W-(n_{12}+1)-n_{11}}\right), & r(\underline{n_4}, \underline{n_1}) &= (n_{11}+1)\mu_{11}\left(1 - \tfrac{W-n_{12}-n_{22}}{n_{11}+1}\right) \\ r(\underline{n_3}, \underline{n_2}) &= (n_{11}+1)\mu_{11}\left(1 - \tfrac{W-(n_{12}+1)-n_{22}}{n_{11}+1}\right), & r(\underline{n_1}, \underline{n_4}) &= \lambda_{11}\left(1 - \tfrac{f_{12}}{W-n_{11}-n_{12}}\right) \end{aligned} \tag{4.9}$$

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

Returning to Figure 4.3, we note that if the transition rate from state (1,0,1,1) to state (1,1,1,0) is changed to $2\lambda_{11}$ (from $\lambda_{11}$), then the Markov process becomes time-reversible. This is an important result because, as we will see in a moment, we can obtain

[2] Similar arguments apply when $\underline{n_3}$ and $\underline{n_4}$ are obtained from $\underline{n_2}$ and $\underline{n_1}$, respectively, by incrementing the value of parameter $n_{22}$ instead of that of parameter $n_{11}$.

[3] To obtain $r(\underline{n_2}, \underline{n_3})$ we multiply the arrival rate $\lambda_{11}$ by the probability that the wavelength assigned to the new call is one of the wavelengths used on the second hop, or equivalently, the probability that the new state will have the same value for parameter $f_{12}$. To obtain $r(\underline{n_4}, \underline{n_1})$, we multiply the rate $(n_{11}+1)\mu$ by the probability that the terminating call will free a wavelength that is used on the second hop, and thus the new state has the same value for $f_{12}$. Similarly for the other rates.

a closed-form solution for the time-reversible process. However, when each hop supports more than 2 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.

> 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 with rate $r(\underline{n}, \underline{n'}) = \lambda_{12}$ to a state $\underline{n'} = (n_{11}, c+1, n_{22}, f_{12}-1)$ of sub-chain $\mathcal{L}_{2,c+1}$. If these transition rates are changed to:
>
> $$r'(\underline{n}, \underline{n'}) \;=\; \lambda_{12}\,\frac{f_{12}(W-n_{12})}{f_{11}f_{22}} \;=\; \lambda_{12}\,\frac{f_{12}(W-n_{12})}{(W-n_{11}-n_{12})(W-n_{22}-n_{12})} \tag{4.10}$$
>
> 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 (4.8) is modified according to this rule, then the four-state sequence in Figure 4.4 will satisfy Kolmogorov's criterion for reversibility.

The above observations 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 Markov process for which the number of active calls using two or more hops is constant:

$$\mathcal{L}_{k,\underline{c}} \;=\; \{\underline{n} \in \mathcal{M}_k \;:\; n_{ij} = c_{ij} = \text{constant},\; i<j\}\;,\;\; \underline{c} = (c_{12}, \cdots, c_{1k}, c_{23}, \cdots, c_{2k}, \cdots, c_{k-1,k}) \tag{4.11}$$

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 that includes only states from sub-chain $\mathcal{L}_{k,\underline{c}}$ satisfies Kolmogorov's criterion for reversibility, but there exist sequences that include states from different sub-chains that violate this criterion.

We now introduce a new time-reversible Markov process $\mathcal{M}'_k$ which is derived from $\mathcal{M}_k$ as follows. The new Markov process $\mathcal{M}'_k$ has the same state space and the same transitions as $\mathcal{M}_k$. The vast majority of the transition rates of the new Markov process are the same as the respective rates of $\mathcal{M}_k$. However, to ensure that the new Markov 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 $\lambda_{lm}, l < m$, to a state $\underline{n'}$ of another sub-chain $\mathcal{L}_{k,\underline{c'}}, c \neq c'$:

$$\mathcal{N}_{k,\underline{c}} = \{\underline{n} \in \mathcal{L}_{k,\underline{c}} : \exists\ i,j,l,m,\ \ l \leq i < j \leq m,\ \ n_{ii} > 0, n_{jj} > 0, n_{lm} < W, f_{lm} > 0\} \tag{4.12}$$

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

$$r'(\underline{n}, \underline{n'}) = \lambda_{lm} \frac{f_{lm} \left(\sum_{i=1}^{l} n_{il} + f_{ll}\right) \left(\sum_{i=1}^{l+1} n_{i,l+1} + f_{l+1,l+1}\right) \cdots \left(\sum_{i=1}^{m-1} n_{i,m-1} + f_{m-1,m-1}\right)}{f_{ll}\ f_{l+1,l+1}\ \cdots\ f_{m,m}} \tag{4.13}$$

Markov process $\mathcal{M}'_k$ has a closed-form solution for its steady-state probability that resembles the product form solution in queuing networks [17]. 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 the 2-hop path in Figure 4.2 is [4]:

$$\pi(n_{11}, n_{12}, n_{22}, f_{12}) = \frac{1}{G_2(W)} \frac{\rho_{11}^{n_{11}} \rho_{12}^{n_{12}} \rho_{22}^{n_{22}}}{n_{11}! n_{12}! n_{22}!} \frac{\binom{W - n_{11} - n_{12}}{f_{12}} \binom{n_{11}}{W - n_{12} - n_{22} - f_{12}}}{\binom{W - n_{12}}{W - n_{12} - n_{22}}} \tag{4.14}$$

while the solution to Markov process $\mathcal{M}'_3$ corresponding to a 3-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:

$$\begin{aligned} \pi(\underline{n}) = & \frac{1}{G_3(W)} \frac{\rho_{11}^{n_{11}} \rho_{12}^{n_{12}} \rho_{13}^{n_{13}} \rho_{22}^{n_{22}} \rho_{23}^{n_{23}} \rho_{33}^{n_{33}}}{n_{11}! n_{12}! n_{13}! n_{22}! n_{23}! n_{33}!} \frac{\binom{W - n_{11} - n_{12} - n_{13}}{f_{12}} \binom{n_{11}}{W - n_{12} - n_{22} - n_{13} - f_{12}}}{\binom{W - n_{12}}{W - n_{12} - n_{22}}} \\ & \times \frac{\binom{W - n_{11} - n_{12}}{f_{12}} \binom{n_{11}}{W - n_{12} - n_{22} - f_{12}}}{\binom{W - n_{12}}{W - n_{12} - n_{22}}} \frac{\binom{W - n_{11} - n_{12}}{f_{12}} \binom{n_{11}}{W - n_{12} - n_{22} - f_{12}}}{\binom{W - n_{12}}{W - n_{12} - n_{22}}} \end{aligned} \tag{4.15}$$

We can write down the solution to any Markov process $\mathcal{M}'_k,\ k > 3$, by a straightforward generalization of expressions (4.14) and (4.15). Specifically, the solution to $\mathcal{M}'_k$,

[4] The closed-form solution (4.14) is similar to the one presented in [28]. However, there are several important differences in the two approaches. First, the solution in [28] 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 2-hop path. Second, we have shown that the closed-form solution (4.14) is the exact solution to an approximate Markov process; in contrast, in [28, Section II.C] this result was derived as a solution to the original 2-hop problem, without explicitly stating that it is an approximation. Finally, whereas only a 2-hop path was studied in [28], 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 (4.15)).

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 parameters $n_{ij}$ in the state description (4.2). The last $k(k-1)/2$ terms are combinatorial terms, each corresponding to one of the dependent variables $f_{ij}$ in the state description (4.2).

The significance of the new Markov process $\mathcal{M}'_k$ will be illustrated in Section 4.4 and Figures 4.7 and 4.8, 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 $\mathcal{M}_k$. Hence, our main contribution in this area is the development of a time reversible Markov process, with a closed-form solution for its steady-state probability that can be easily written down, which accurately captures the correlation of wavelength use among the links of a path.

Of course, the main concern in any product-form solution is the computation of the normalizing constant. Using brute force enumeration, we can obtain the product form solution of $\mathcal{M}'_k$ for up to 25 wavelengths, when the number of hops is $k = 3$, and for up to 8 wavelengths when $k = 4$. (As a comparison, for $k = 3$ we can obtain the numerical solution of $\mathcal{M}_k$ only up to $W = 4$.) Despite our efforts, we have not been able to devise a recursive procedure that would allow us to compute the normalizing constant in time that is polynomial in $W$ and $k$. Therefore, a different approximate method is needed for paths longer than four hops. In Section 4.3 we describe an iterative decomposition algorithm that can be used to efficiently obtain the blocking probabilities for paths of arbitrary length.

### 4.2.2 Paths With Wavelength Conversion

Let us now turn our attention to paths in which wavelength converters are employed at some nodes. We show how the results of Section 4.2.1 can be extended to this case, leading to a method which can be used to obtain the call blocking probabilities in a $k$-hop path when the placement of converters is known.

Let us again refer to Figure 4.2, and let us assume that wavelength converters are located at node 1 (the only interesting possibility in this case). We immediately see that the 2-hop system can now be described by the three-dimensional Markov process $(n_{11}, n_{12}, n_{22})$. Parameter $f_{12}$ 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 2-hop system with a converter at node 1 becomes

equivalent to a 2-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 $\mathcal{M}_k$, as follows. The new Markov process has the same $\frac{k(k+1)}{2}$ parameters $n_{ij}$ as $\mathcal{M}_k$, but some of the parameters $f_{ij}$ are no longer present in the state description. More specifically, let us consider the case when a converter is employed at node $l, 0 < l < k$, of the path. Then, parameters $f_{ij}, i \leq l < j$, 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 $j$, inclusive, 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, parameters $f_{il}$ 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 for $k = 2$. We can then use an approach similar to the one we followed in Section 4.2.1 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. Thus, we have developed exact and approximate Markov process models for computing the call blocking probabilities in paths with wavelength conversion capabilities.

## 4.3 Decomposition Algorithm for Long Paths

Let $K$ denote the largest integer such that the normalizing constant of the closed-form solution to Markov process $\mathcal{M}'_k$ can be computed within a reasonably short amount of time (currently, $K = 3$ or 4, depending on the number $W$ of wavelengths per link, but this value may increase if we can improve our method for computing the normalizing constant). 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. As before, we first consider paths without converters, and we then extend the algorithm to handle wavelength conversion.

### 4.3.1 Paths With No Wavelength Conversion

We analyze a $k$-hop path, $k = lK + m, l \geq 1, m < K$, by decomposing it into $l$ $K$-hop segments and one $m$-hop segment in tandem. Each segment is first analyzed in isolation using the corresponding Markov process $\mathcal{M}'_k$. The arrival rates of calls originating (terminating) in a segment but terminating (originating) in another segment are accounted for by increasing the arrival rate of calls of the individual segments. The individual solutions can then be 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 to obtain a new solution. These new individual solutions are again combined to update the blocking probability of calls traversing multiple segments. The process is repeated until the blocking probabilities converge.

A detailed description of our iterative algorithm is provided in Figure 4.5. We now illustrate the operation of the algorithm by referring to Figure 4.6 which shows a 4-hop path and its decomposition into two 2-hop segments. We let $\lambda_{11}, \lambda_{12}, \cdots, \lambda_{44}$, be the arrival rates of calls to the original 4-hop path, and we use $\lambda_{11}^{(1)}, \lambda_{12}^{(1)}, \lambda_{22}^{(1)}$ and $\lambda_{11}^{(2)}, \lambda_{12}^{(2)}, \lambda_{22}^{(2)}$ to denote the arrival rates of calls to the first and second segments, respectively. However, the interpretation of the arrival rates for the 2-hop segments is somewhat different under our decomposition approach. Specifically, $\lambda_{12}^{(1)}$ accounts for all the calls of the original 4-hop path that originate at node 0 and terminate at node 2 or higher; similarly for $\lambda_{22}^{(1)}$. On the other hand, $\lambda_{12}^{(2)}$ accounts for all calls of the original path that originate at nodes 2 and lower and terminate at node 4; similarly for $\lambda_{11}^{(2)}$. Below, we describe briefly the main steps of our algorithm.

Initially, we solve the first segment in isolation by using

$$\lambda_{12}^{(1)} = (1 - q_{14})\lambda_{14} + (1 - q_{13})\lambda_{13} + \lambda_{12} \tag{4.16}$$

$$\lambda_{22}^{(1)} = (1 - q_{24})\lambda_{24} + (1 - q_{23})\lambda_{23} + \lambda_{22} \tag{4.17}$$

$$\lambda_{11}^{(1)} = \lambda_{11} \tag{4.18}$$

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 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

**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. $\lambda_{ij}$ 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
   $p_{ij}^{(1)}(h) \leftarrow 0, \quad F_{ij}^{(1)} \leftarrow W, \qquad 1 \leq i \leq j \leq K$
   // $p_{ij}^{(2)}(h)$ 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
   $p_{ij}^{(2)}(h) \leftarrow 0, \quad F_{ij}^{(2)} \leftarrow W, \qquad 1 \leq i \leq j \leq m$
   // $q_{ij}(h)$ 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
   $q_{ij}(h) \leftarrow 0, \qquad 1 \leq i \leq K < j \leq K + m$
3. $h \leftarrow h + 1$ //$h$-th iteration
4. $\lambda_{ij}^{(1)}(h) \leftarrow \lambda_{ij}, \qquad 1 \leq i \leq j < K$ //Segment 1
   $\lambda_{iK}^{(1)}(h) \leftarrow \lambda_{iK} + \sum_{j=K+1}^{K+m} \lambda_{ij} \left(1 - q_{ij}(h-1)\right), \qquad 1 \leq i \leq K$
   // 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. $\lambda_{ij}^{(2)}(h) \leftarrow \lambda_{K+i,K+j}, \qquad 1 < i \leq j \leq m$ //Segment 2
   $\lambda_{1j}^{(2)}(h) \leftarrow \lambda_{K+1,K+j} + \sum_{i=1}^{K} \lambda_{i,K+j} \left(1 - p_{iK}^{(1)}(h-1)\right), \qquad 1 \leq j \leq m$
   // 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
   $q_{ij}(h) \leftarrow p_{1,j-K}^{(2)}(h) \; + \; \left(1 - p_{1,j-K}^{(2)}(h)\right) \; Q_{ij}, 1 \leq i \leq K < j \leq K + m$, with $Q_{ij}$
7. Repeat from Step 3 until the blocking probabilities converge
8. end of the algorithm

Figure 4.5: Decomposition algorithm for long paths

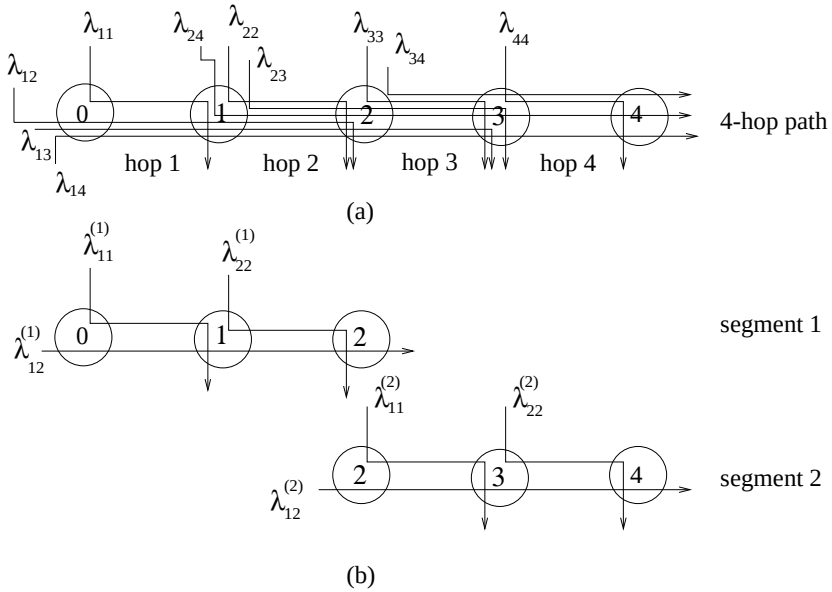


Figure 4.6: (a) A 4-hop path and (b) its decomposition into two 2-hop segments in tandem

these values are updated in subsequent iterations will be described shortly. Thus, the term $(1-q_{14})\lambda_{14}$ in (4.16) represents the *effective* rate of calls using all four hops, as seen by the first segment; similarly for the term $(1-q_{13})\lambda_{13}$. Expression (4.17) for $\lambda_{22}^{(1)}$ includes similar terms that account for the effective rate of calls which originate at node 1 and terminate at nodes 2 or higher. On the other hand, expression (4.18) for $\lambda_{11}^{(1)}$ does not include any such terms, since this type of calls in the first segment do not involve calls of the original path continuing to the second segment.

The solution to the first segment will yield 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 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 by using

$$\lambda_{12}^{(2)} = \lambda_{14}(1-p_{12}^{(1)}) + \lambda_{24}(1-p_{22}^{1)}) + \lambda_{34} \tag{4.19}$$

$$\lambda_{11}^{(2)} = \lambda_{13}(1-p_{12}^{(1)}) + \lambda_{23}(1-p_{22}^{1)}) + \lambda_{33} \tag{4.20}$$

$$\lambda_{22}^{(2)} = \lambda_{44} \tag{4.21}$$

Based on the above discussion, $\lambda_{12}^{(2)}$ in (4.19) represents the effective rate of calls using the

last two hops of the 4-hop path, as seen by the second segment. Expression (4.20) for $\lambda_{11}^{(2)}$ can be explained using similar arguments, while expression (4.21) for $\lambda_{22}^{(2)}$ contains only one term since, as seen in Figure 4.6, it does not involve calls continuing from segment 1. The solution to the second segment will provide 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 (4.16) and (4.17), 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 $n$ (resp., $m$) free wavelengths on the hops of the first (resp., 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 $n$ (resp., $m$) free wavelengths on the hops it uses in the first (resp., second) segment. Because of the random wavelength assignment policy, we have that:

$$R(n,m) = \begin{cases} 0, & n = 0 \text{ or } m = 0 \\ 1, & n + m \geq W \\ \dfrac{\binom{W-n}{m}}{\binom{W}{m}}, & \text{otherwise} \end{cases} \tag{4.22}$$

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

$$Q_{ij} = \sum_{n=1}^{W} \sum_{m=1}^{W} \left[ P_{i2}^{(1)}[W = n] \times P_{1,j-2}^{(2)}[W = m] \times R(n,m) \right] \times \left[ \frac{1}{2} \left( \frac{\lambda_{ij}}{\lambda_{i2}^{(1)}} + \frac{\lambda_{ij}}{\lambda_{1,j-2}^{(2)}} \right) \right] \tag{4.23}$$

The double summation in the right hand side of (4.23) 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 (4.23) 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 (4.23) 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 expressions (4.16) and (4.17)). Similarly, the term $\lambda_{ij}/\lambda_{1,j-2}^{(2)}$ in (4.23) 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 (4.23) 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 expressions $(4.16) - (4.18)$ to update the arrival rates for the first segment, the first segment is solved again, the estimates for $p_{ij}^{(1)}$ are updated and used in expressions $(4.19) - (4.21)$, and so on. We repeat the process until the blocking probabilities for all calls of the original path converge within a certain tolerance. In all cases studied, we have found that the algorithm converges in only a few (under ten) iterations even for long paths, and that the blocking probabilities obtained closely match simulation results (more on this in Section 4.4).

The decomposition algorithm just described is similar in spirit to the decomposition algorithms developed for tandem queuing networks with finite capacity queues (see [22]). This algorithm can be easily extended to handle paths decomposed into more than two segments. We note that when employing the decomposition algorithm, the eventual selection of the segment size will depend on the following factors: length of the original path, how efficiently we can calculate the normalizing constant for 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 sub-systems that

have to be analyzed in isolation, the better the accuracy of the decomposition algorithm. Thus, as we mentioned in the beginning of this subsection, we have decided to decompose a path in segments of the largest size $K$ for which we can efficiently compute the normalizing constant, plus, possibly, a segment of smaller size, if the path length is not a multiple of $K$.

### 4.3.2 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), *not* the actual wavelengths used by those calls. Hence, the probability that a call spanning multiple segments will be successfully completed becomes equal to the product of the probabilities of finding a free wavelength (not necessarily the same one) within each segment. Therefore, we choose to decompose a path into segments between successive nodes with converters. These segments do not contain converters, and they can be analyzed in isolation as described previously. If one of these segments is too long, it can be analyzed using the iterative decomposition algorithm in Figure 4.5.

Based on these observations, an overall solution can be obtained by using a modified version of the iterative algorithm developed for the no-converter case. In particular, a $k$-hop path with a single converter located at node $K < k$ can be analyzed using the algorithm in Figure 4.5 after making a single modification as follows: in Step 6, the expression for the conditional blocking probabilities must be changed to $q_{ij} = p^{(2)}_{1,j-K}$. The second term in the expression of Step 6 in Figure 4.5 (as well as in expression (4.23)) represents blocking due to the wavelength continuity requirement, and since $Q = 0$ in this case, it drops out. The algorithm can be extended in a straightforward way to handle paths with more than one converters.

## 4.4 Numerical Results

We now demonstrate the accuracy of the blocking probabilities obtained with our analytical techniques by comparing to either numerical or simulation results. We then apply

our decomposition algorithm in order to determine the optimal placement of converters in a path.

### 4.4.1 Accuracy of the Approximate Markov Process

In Figure 4.7 we plot the blocking probability of calls for each source-destination node pair in a 2-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 as "exact solution" in the figure. 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.

Figure 4.8 is similar to Figure 4.7, but presents results for a 3-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 behavior of the blocking probability of the other three types of calls is very similar to that shown in Figure 4.8. Again, we observe that the values of blocking probability obtained through the closed-form solution to Markov process $\mathcal{M}'_3$ are an accurate approximation of the exact values obtained from process $\mathcal{M}_3$. However, the figure does not include any values for the exact blocking probability in a 3-hop path when $W > 4$ because the size of the exact Markov process is so large in this case that we could not solve it numerically. (For the same reason we do not show any comparison results for 4-hop paths.)

Overall, the results shown in Figures 4.7 and 4.8 indicate that it is appropriate to use the approximate Markov process when analyzing short paths. We emphasize that exact Markov process cannot be solved numerically for a large number of wavelengths. Also, for the same values of $k$ and $W$, the closed-form solution of the approximate Markov process can be obtained significantly faster (up to an order of magnitude) than the numerical solution to the exact process.

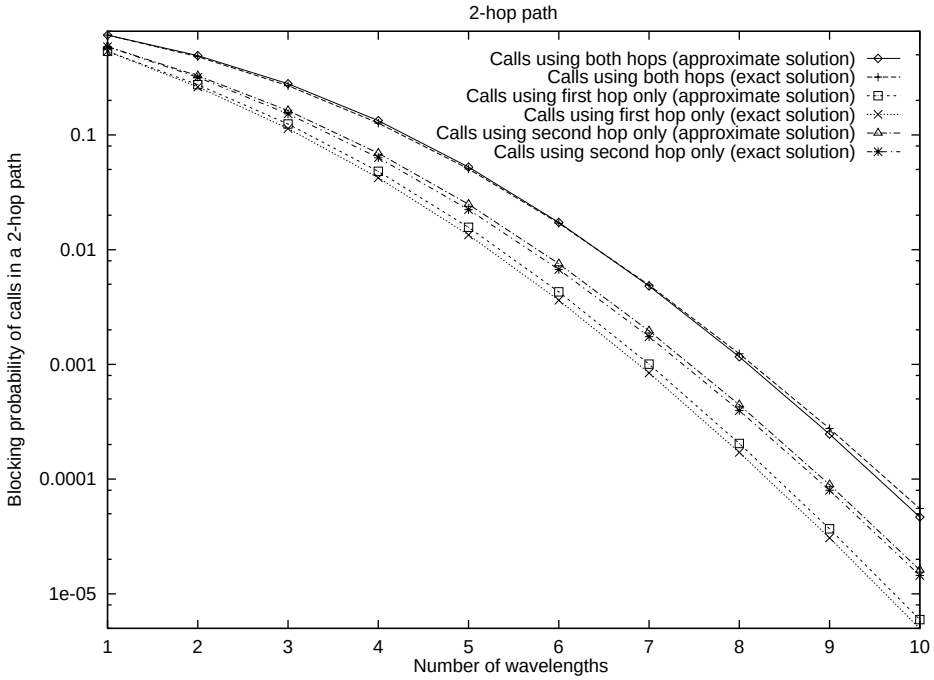


Figure 4.7: Exact and approximate blocking probability of the various calls in a 2-hop path

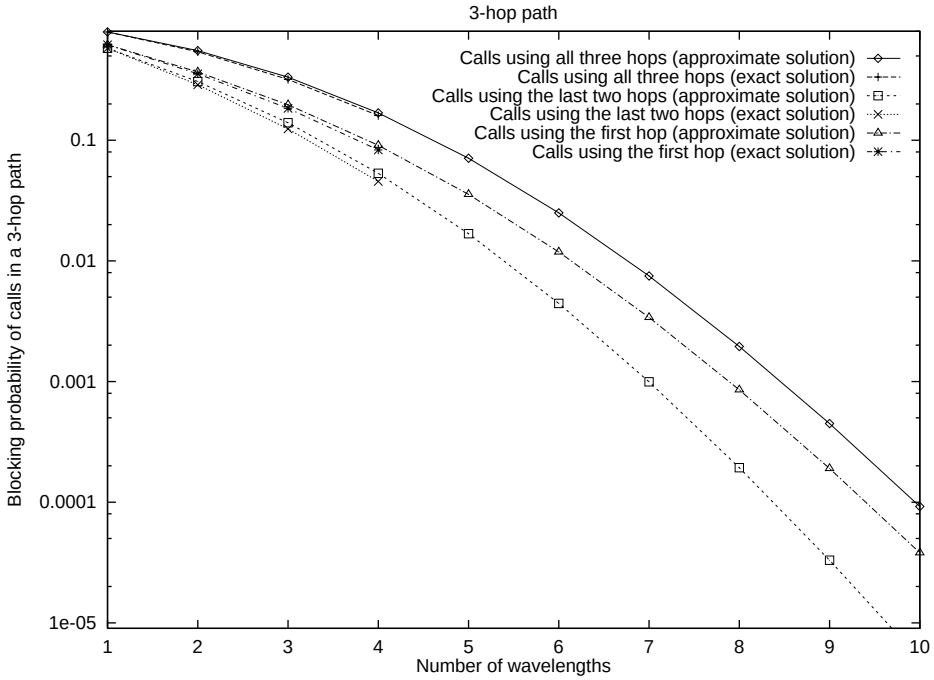


Figure 4.8: Exact and approximate blocking probability of various calls in a 3-hop path

### 4.4.2 Validation of the Decomposition Algorithm

The figures in this subsection show the results of applying our iterative decomposition algorithm to 6-hop and 10-hop paths with $W = 10$ wavelengths, and compare them to results obtained through simulation. We include results for paths with and without converters. Because of the very large number of parameters (i.e, the arrival rates $\lambda_{ij}$ and mean call holding times $1/\mu_{ij}$) that can potentially be varied, for the results presented here we have decided to use the following values:

$$\mu_{ij} = 1, \ \forall\, i,j, \quad \text{and} \quad \lambda_{ij} = \begin{cases} 0.3, & i < j \\ \lambda, & i = j \end{cases} \tag{4.24}$$

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 0.3, and we also set the arrival rate of calls traversing exactly one hop to $\lambda$. Figures 4.9 – 4.12 plot the call blocking probability as we vary the value of $\lambda$.

In Figures 4.9 and 4.10 we present results for a 6-hop path with and without converters, and for $W = 10$. We note that there are 21 different source-destination pairs in a 6-hop path, making it impossible to present results for all of them here. Thus, we show results for only two source-destination pairs. The blocking probability of calls traversing all six hops in the path is plotted in Figure 4.9, while the blocking probability of calls traversing hops 1 through 4 of the path is shown in Figure 4.10. In both figures, the value of $\lambda$ is varied from 0.1 to 0.5, while the arrival rate of all other calls is fixed to 0.3, as described above. Each figure contains two sets of plots, one for a path without converters, and one for a path with a single converter. Each set consists of two plots, one corresponding to results from our decomposition algorithm, and one corresponding to simulation results. Recall that, for $W = 10$ wavelengths, we can efficiently compute the closed-form solution of the approximate Markov process for a 3-hop path, but not for a 4-hop path. Hence, we let $K = 3$ in the decomposition algorithm of Figure 4.5, and we analyze a 6-hop path without converters by decomposing it into two 3-hop segments. We refer to this decomposition as a "$3 \times 3$ decomposition". For the case when there is a single converter in the path, we decided to place the converter at node 3, since this arrangement will also result in a $3 \times 3$ decomposition.

From Figures 4.9 and 4.10 we observe that as the value of the load $\lambda$ increases, the blocking probability of both types of calls increases. We also note that, when there is

a converter at node 3, 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 very close to the values obtained through simulation. Results similar to the ones shown in these figures were obtained for calls for all other source-destination pairs, and for a wide range of traffic loads.

The results in Figures 4.11 and 4.12 are very similar to those in Figures 4.9 and 4.10, but they correspond to a 10-hop path with $W = 10$ wavelengths. Again, we only plot the blocking probability of calls for two source-destination pairs against the arrival rate $\lambda$, as the latter increases from 0.05 to 0.25. Figure 4.11 shows the blocking probability of calls traversing all ten hops of the path, while Figure 4.12 presents the blocking probability of calls using hops 2 through 6. As before, there are two sets of plots, one for a path without converters, and one for a path employing three converters. For the no-converter case, in addition to simulation results, we present the blocking probability values obtained through $1 \times 3 \times 3 \times 3$ decomposition approaches for a 10-hop path. The path is decomposed into a 1-hop segment and three 3-hop segments For the converter case, the three converters are assumed to be at nodes 3, 6, and 9, a configuration that also results in a $3 \times 3 \times 3 \times 1$ decomposition. The general behavior of the curves in Figures 4.11 and 4.12 as $\lambda$ increases are very similar to that of the curves in Figures 4.9 and 4.10. Regarding the accuracy of our analytical techniques, we note that the results of our decomposition algorithm closely match the simulation results for the path with converters.

Compare decomposition method to simulation. In particular, mention that the decomposition algorithm converges in only a handful of iterations and takes a few minutes, while the simulation takes several hours.

### 4.4.3 Converter Placement

We now consider the problem of determining the best placement of $l$ converters on a $k$-hop path, $k > l$, that optimizes a given objective function. It is important to emphasize that our goal is simply to demonstrate how the analytical techniques developed 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. All the results presented in this section

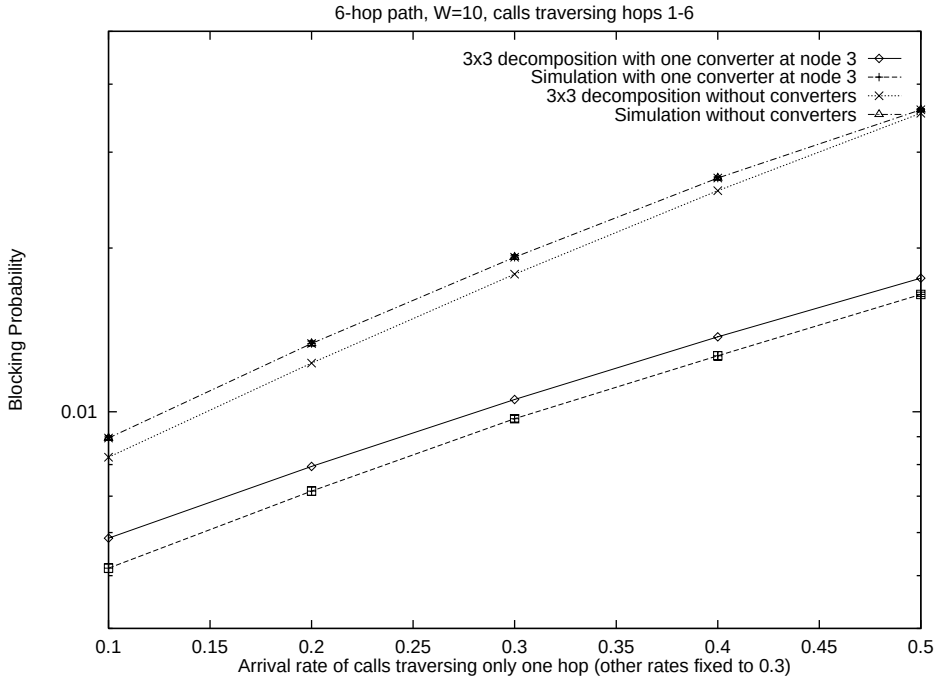


Figure 4.9: Blocking probability of calls traversing all links of a 6-hop path with $W = 10$

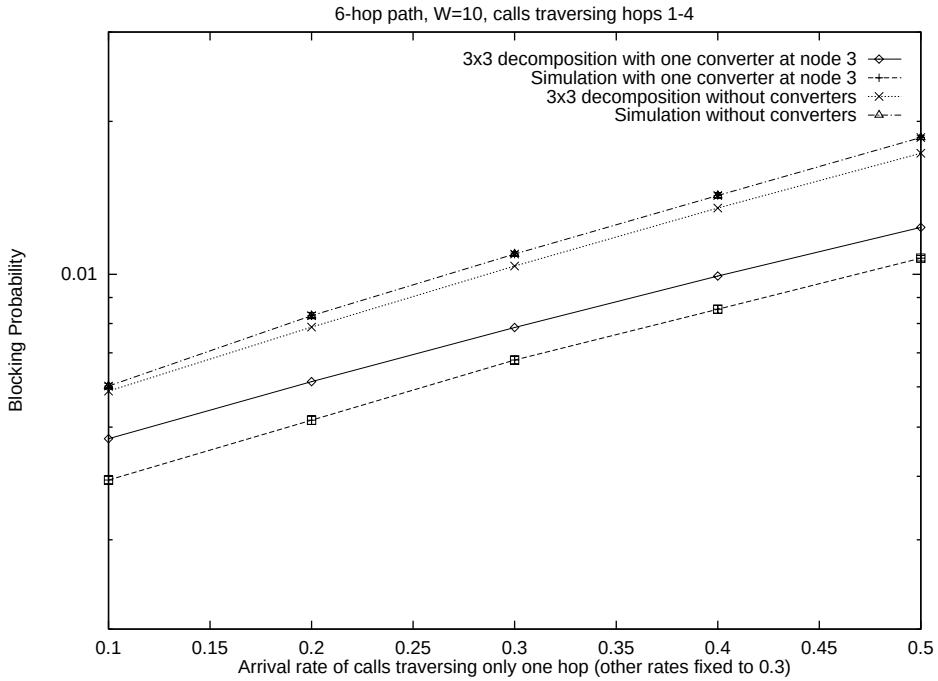


Figure 4.10: Blocking probability of calls traversing links 1 through 4 of a 6-hop path with $W = 10$

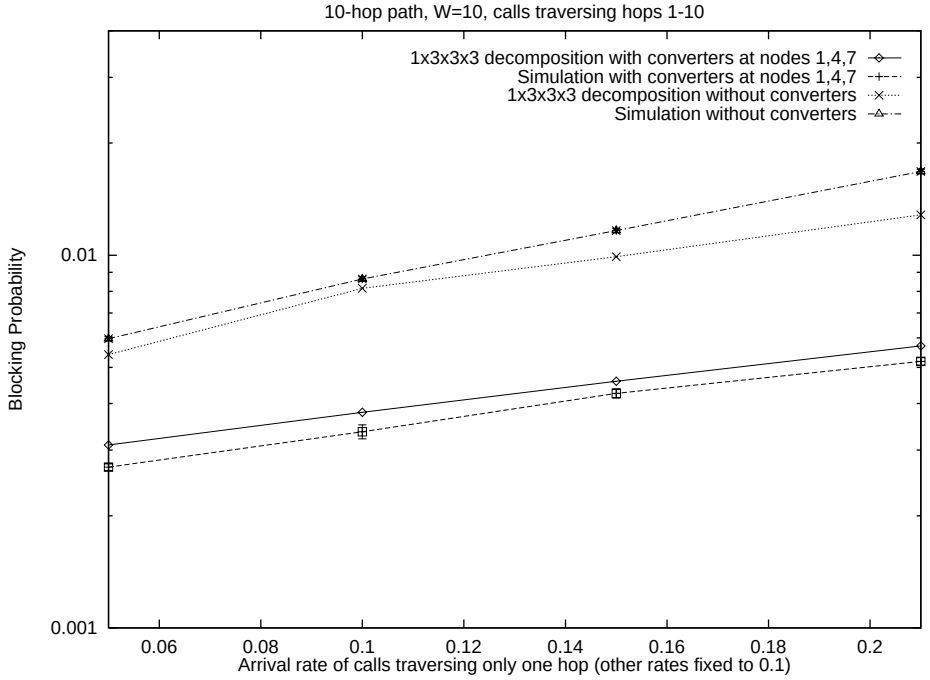


Figure 4.11: Blocking probability of calls traversing all links of a 10-hop path with $W = 10$

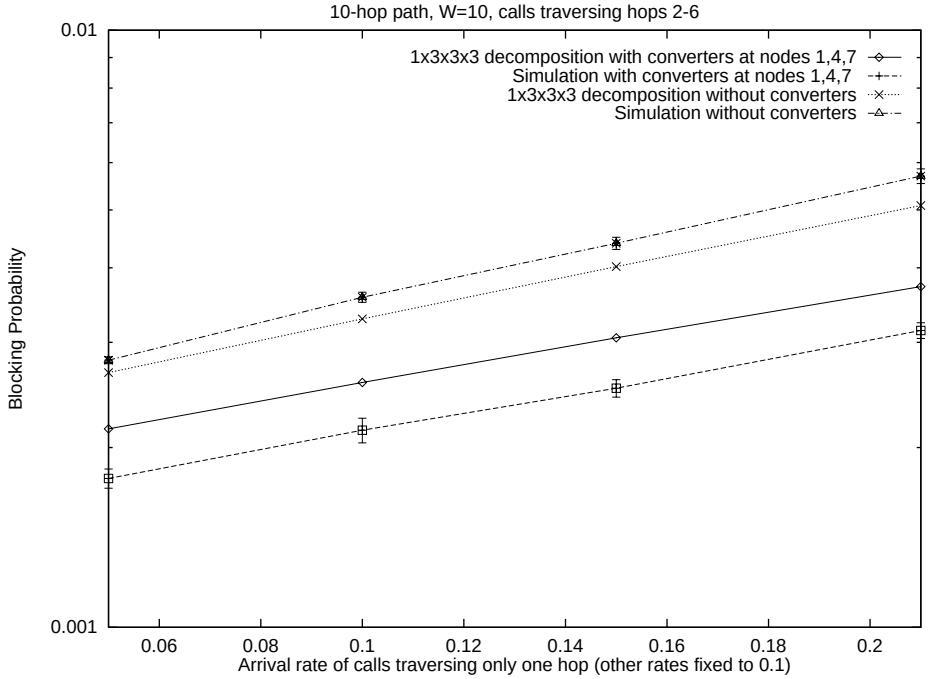


Figure 4.12: Blocking probability of calls traversing links 2 through 6 of a 10-hop path with $W = 10$

are obtained by using the following approach:

- The performance parameter of interest in a $k$-hop path is the blocking probability of the calls that travel over all $k$ hops (i.e., the ones with arrival rate $\lambda_{1k}$, refer also to Figure 4.1). Thus, our objective is to determine the location of $l$ converters that minimizes the blocking probability of these calls.

- To find the best converter placement we apply a straightforward method as follows. We enumerate all possible ways of placing $l$ converters on a $k$-hop path, and we evaluate the blocking probability of interest for each using the decomposition algorithm presented in previous sections. The optimum placement is the one with the minimum such probability.

We note that calls traveling over all $k$ hops of a $k$-hop path will experience the highest blocking probability. Hence, minimizing this parameter effectively minimizes the maximum blocking probability of any call. On the other hand, there are $\left( \begin{array}{c} k-1 \\ l \end{array} \right)$ ways of placing $l$ converters at the $k-1$ internal nodes of a $k$-hop path. Therefore, we expect our straightforward enumeration approach to work well for reasonably long paths (e.g., $k \leq 20$). For longer paths, when this method becomes prohibitively time consuming, heuristic approaches aimed at determining a near-optimal placement of converters are needed. The specification and evaluation of such heuristics will be the subject of future research.

For our study, we consider a 10-hop path with $W = 10$ wavelengths, and six different traffic load patterns. Figures 4.13 and 4.14 plot the load $\rho$ of each hop in the path for each load pattern [5]. The six patterns are representative of the wide range of loading situations one expects to encounter in practice. In the "uniform" pattern, all hops are equally loaded. The "bowl" (respectively, "inverted bowl") pattern is such that the load decreases (resp., increases) from hop 1 to hop 5, and then it increases (resp., decreases) from hop 6 to fop 10. These patterns are shown in Figure 4.13. The "ascending" and "descending" patterns are such that the load increases or decreases, respectively, from hop 1 to hop 10. Finally, in the oscillating pattern the load at each hop alternates between a low and a high value. The last three load patterns are shown in Figure 4.14. To ensure that the results are comparable across the different patterns, the load values were chosen so that the total network load (or, equivalently, the average load per hop) is the same for all patterns.

[5] The load $\rho$ for hop $l$ is the sum of the offered loads $\rho_{ij}, i \leq l \leq j$, for all calls that use hop $l$.

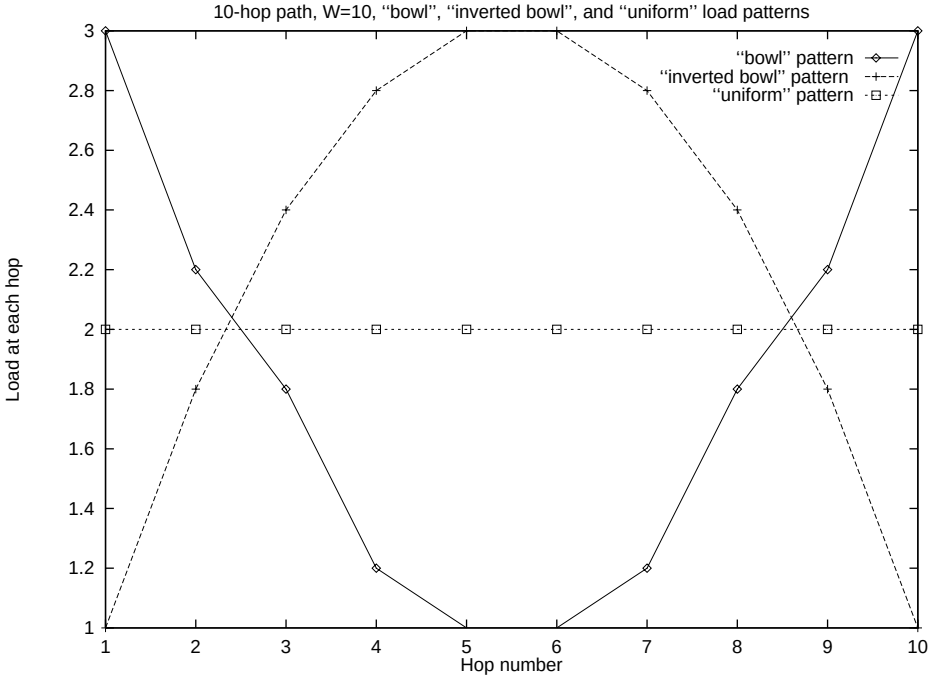


Figure 4.13: The “bowl”, “inverted bowl”, and “uniform” load patterns

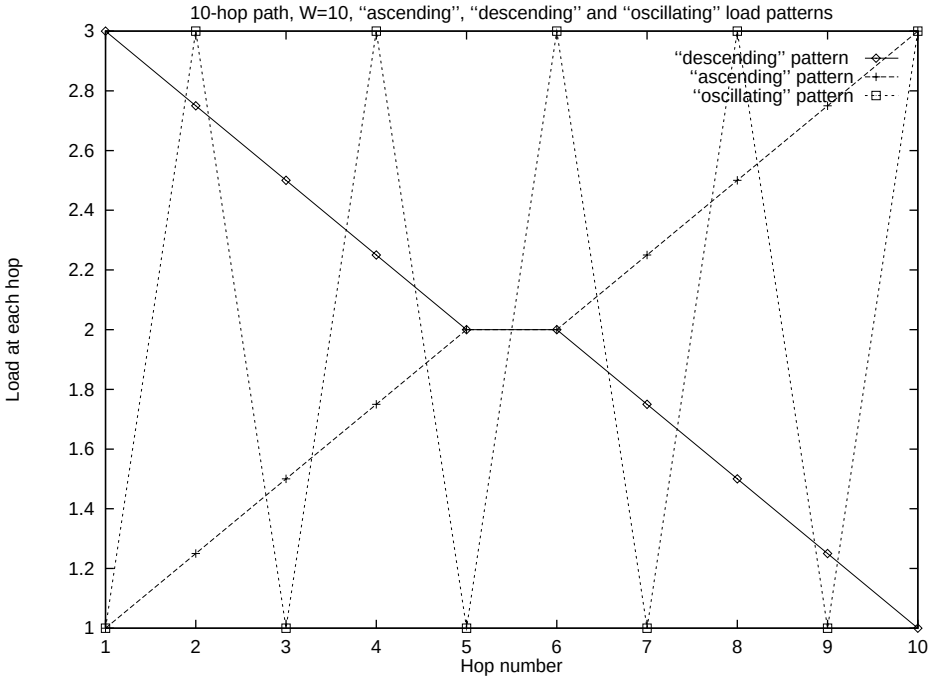


Figure 4.14: The “ascending”, “descending” and “oscillating” load patterns

In Figures 4.15 and 4.16 we plot the blocking probability of calls using all 10 hops of the path for the optimal placement of $l$ converters, $1 \leq l \leq 5$. 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 blocking probabilities shown in Figures 4.15 and 4.16 correspond to the load patterns of Figures 4.13 and 4.14, respectively. The optimal location of converters for the six different load patterns is shown in Table 4.1.

From Figures 4.15 and 4.16 it is immediately obvious that the blocking probability drops with the number of converters, as expected. However, after the initial steep drop, the curves in general flatten as the number of converters increases. This behavior is consistent with the results of an earlier work [28] which studied the effect of converter density on the blocking probability. But the results presented here shed new light into the problems of determining the importance of converters and their location, which have previously been studied only under uniform traffic loading.

A striking observation from Figures 4.15 and 4.16 is that the effect of converters on the blocking probability is strongly dependent on the actual traffic pattern. Let us compare, for example the curves for the "inverted bowl" (in Figure 4.15) and "ascending" (in Figure 4.16) load pattern. The former curve is essentially flat: the blocking probability decreases from a value of 0.0041 for one converter to a value of 0.0028 for five converters. In contrast, increasing the number of converters has a dramatic effect for the "ascending" load pattern, with the blocking probability dropping from a value of 0.0083 for one converter to 0.0018 for five converters, a 78% decrease.

Let us now refer to Table 4.1 which shows the optimal node location of converters for the different traffic patterns. We first note that the results are in agreement with intuition. The table indicates, for example, that converters be placed at the middle of the path for the "inverted bowl" pattern, and towards the end and beginning of the path for the "ascending" and "descending" patterns, respectively. However, we also observe that the optimal placement also depends strongly on the load pattern: compare, for instance, the converter locations for the "ascending" and "descending" patterns. The fact that 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 blocking probability under a specific traffic pattern may lead to poor performance if the pattern changes. Instead, more comprehensive

Table 4.1: Optimal node location of converters for the various load patterns

| Load Pattern | Number of Converters | | | | |
|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 |
| bowl | 8 | 18 | 789 | 1589 | 12589 |
| inverted bowl | 4 | 45 | 457 | 3456 | 34567 |
| uniform | 7 | 28 | 479 | 1279 | 12379 |
| ascending | 8 | 79 | 689 | 5789 | 56789 |
| descending | 2 | 12 | 123 | 1235 | 12345 |
| oscillating | 7 | 58 | 479 | 4789 | 24789 |

approaches to the converter placement problem are needed, such as providing bounds for the blocking probability over a wide range of load patterns.

## 4.5 Concluding Remarks

We presented a new analytical framework to accurately and efficiently evaluate the call blocking performance in a single path of a wavelength routing network. We derived exact and approximate Markov process models, and we developed an iterative algorithm to analyze arbitrarily long paths by decomposing them into shorter segments connected in tandem which are studied in isolation. We also demonstrate how our analytical techniques can be used to gain insight into the problem of converter placement in such a network.

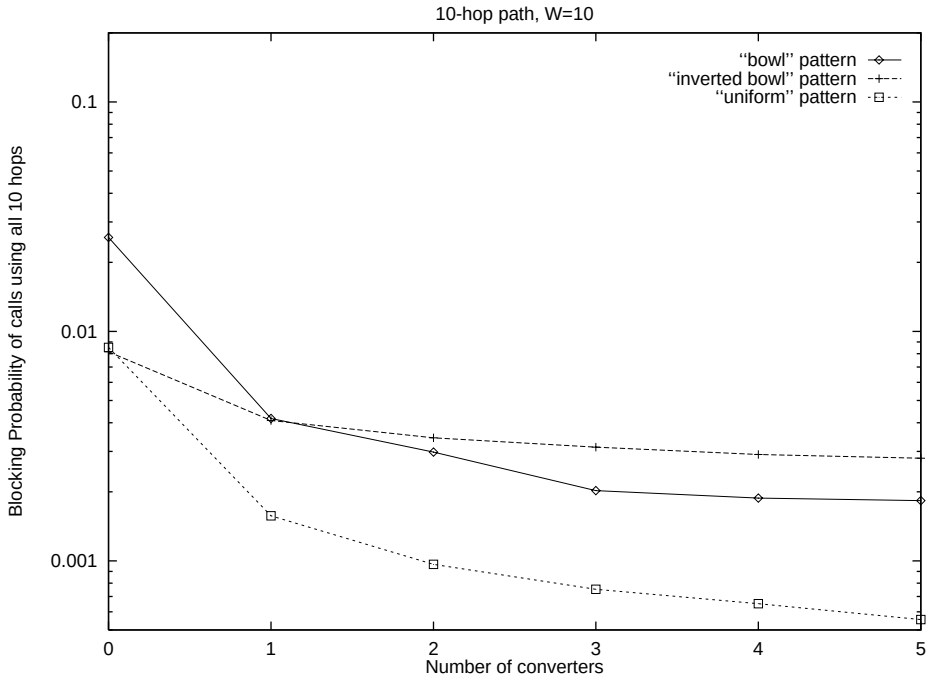


Figure 4.15: Blocking probability under optimal converter placement for the "bowl", "inverted bowl", and "uniform" load patterns

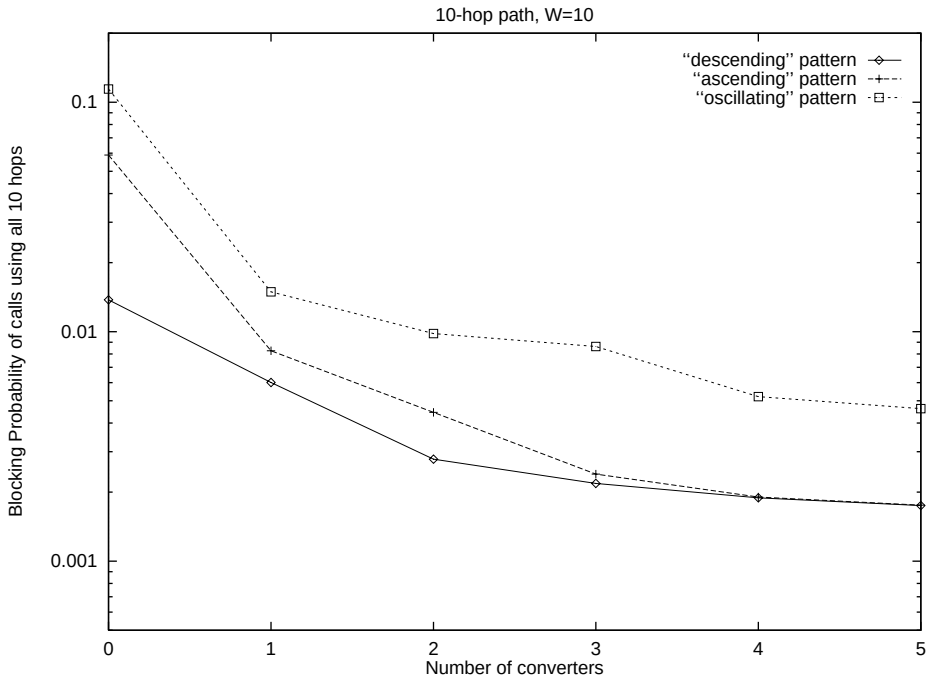


Figure 4.16: Blocking probability under optimal converter placement for the "ascending", "descending" and "oscillating" load patterns

# Chapter 5

# Blocking in Mesh Topologies

In the previous chapter, we presented an exact Markov process and an approximate Markov process for a single path. We also developed a decomposition algorithm to effectively compute long paths. Based on the solution to the single path, we now extend our decomposition algorithm for mesh networks with arbitrary topology.

## 5.1 Path Decomposition Algorithm for Mesh Networks

### 5.1.1 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 $\lambda_{sd}$. 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 Chapter 4, 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.

### 5.1.2 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 Chapter 4. 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 4.1, 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.

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

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 5.1(b) shows the set of sub-systems $\mathcal{R}'$ obtained by applying this algorithm to the network of Figure 5.1(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 5.1 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 5.1. 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

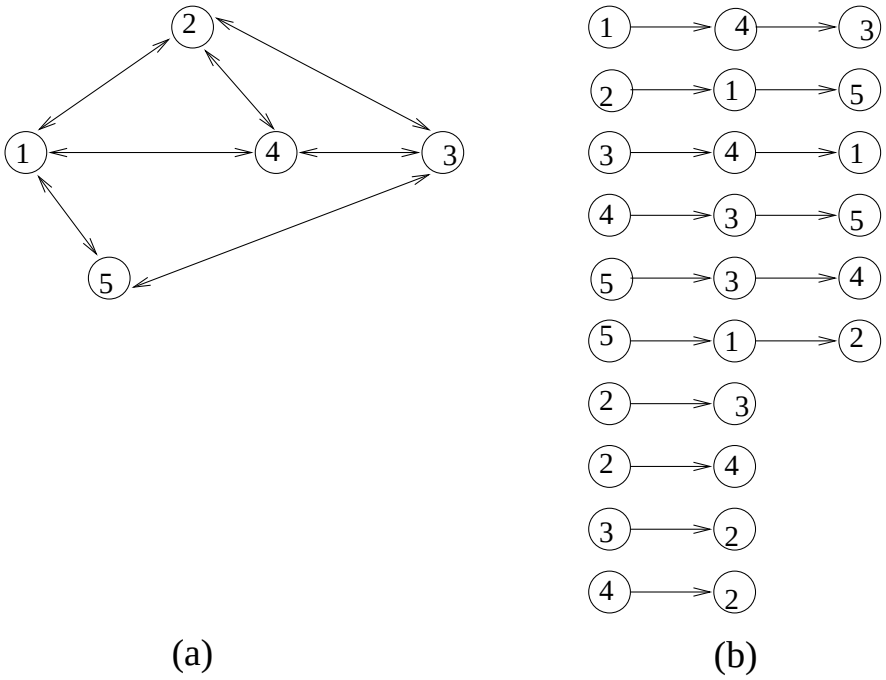


Figure 5.1: (a) Original network, (b) set $\mathcal{R}'$ of paths into which the network is decomposed

at the beginning of each iteration 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 5.1 in isolation using these arrival rates:

$$\hat{\lambda}_{14} = \lambda_{14} \tag{5.1}$$

$$\hat{\lambda}_{13} = \lambda_{13} \tag{5.2}$$

$$\hat{\lambda}_{43} = \lambda_{43} + (1 - P_{45})\lambda_{45} \tag{5.3}$$

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 (5.1). Similarly, expression (5.2) 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 (5.3) 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 $P_{45}$ in (5.3) represents the current estimate of the probability that a call from node 4 to node 5 will be blocked on sub-system (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 sub-system (1,4,3), since a fraction $P_{45}\lambda_{45}$ of these calls will be blocked in sub-system (4,3,5). Consequently, the right hand side of expression (5.3) 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 5.1(b) are solved in a similar manner):

$$\hat{\lambda}_{45} = \lambda_{45} \tag{5.4}$$

$$\hat{\lambda}_{35} = \lambda_{35} \tag{5.5}$$

$$\hat{\lambda}_{43} = \lambda_{43} + (1 - P_{13})\lambda_{13} \tag{5.6}$$

Expressions (5.4) through (5.6) can be explained using arguments similar to the ones used for expressions (5.1) to (5.3). In particular, the second term in the right hand side of (5.6) 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 $P_{45}$ 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 expressions (5.3) and (5.6), 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 5.2).

A detailed description of our decomposition algorithm is provided in Figure 5.2. 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 [18, 3, 12, 31, 15, 20, 29]) 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 5.2 will confirm, that our algorithm will provide a good approximation to the blocking probabilities.

### 5.1.3 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 $\lambda_{sd}$, 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]. Overflow models have also been used in the study of blocking probabilities in wavelength routing networks in [15, 20].

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 $\mathcal{R}$ denote the set of primary and alternate paths for all node pairs, with $\mid \mathcal{R} \mid = 2N(N-1)$. From $\mathcal{R}$ we construct the set of path sub-systems $\mathcal{R}'$ 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 Figure 5.2 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

$$P_{sd} = P_{sd}^{(1)} \times P_{sd}^{(2)} \tag{5.7}$$

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 $P_{sd}$ in (5.7) converge for all $s, d$.

Numerical results to be presented in Section 5.2 will demonstrate that, despite the assumption that overflow traffic is Poisson, this iterative path decomposition approach is quite accurate for both regular and irregular 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 significantly higher than the blocking probabilities that can be tolerated in such an environment.

## 5.2 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 $5 \times 5$ 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.

Before we proceed, a discussion on the number of path sub-systems into which the original network is decomposed is warranted, since the number of these sub-systems determine the overall running time of the decomposition algorithm. For the $5 \times 5$ torus and the NSFNET topologies considered here, there is a total of 600 and 240 paths, respectively, when fixed routing is used. The technique described in Section 5.1.2, however, produces a decomposition of 100 and 94 sub-systems for the torus and NSFNET networks, respectively. Both these numbers are significantly smaller than the total number of paths in the respective topology. When alternate routing with one alternate path is used, our decomposition consists of 400 (out of 1200) sub-systems for the torus network and 160 (out of 480) for the NSFNET. Although it is clear that the number of sub-systems in a given decomposition will depend on the actual topology and routing algorithm used (e.g., shortest path or other), these results indicate that by selecting longer paths as sub-systems, we can significantly reduce the size of the decomposition and, consequently, the running time of our algorithm.

### 5.2.1 The $5 \times 5$ Torus Network

Let us first consider the $5 \times 5$ torus network shown in Figure 5.3 with $W = 10$ wavelengths per link. Since there are 600 source-destination pairs in this network, it is 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 5.1 along with the corresponding labels used in Figures 5.4 and 5.5. 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 $5 \times 5$ 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 5.1 are a representative sample of the various source-destination pairs.

Figure 5.4 plots the blocking probability for the 24 source-destination pairs of Table 5.1 for the $5 \times 5$ torus network under fixed routing, where the path assigned to each pair $(s, d)$ is the shortest path from $s$ to $d$. The call arrival rates were selected such that

$$\lambda_{sd} = \begin{cases} 0.4, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 1 \\ 0.3, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 2 \\ 0.2, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 3 \\ 0.1, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 4 \end{cases} \tag{5.8}$$

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 utilization can be explained by the fact that both the topology and the traffic load are symmetric.

As we can see in Figure 5.4, 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 5.5, 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 5.5 is very similar to that in Figure 5.4. The slight differences in call blocking probabilities among calls with a shortest path of length 2 can be explained by the fact that for some of these calls the length of the alternate path is shorter than for others. For example, the length of the alternate path from node 1 to node 7 is two (the same as the primary path), but the alternate path from node 1 to node 3 is of length three. Consequently, the blocking probability of the former calls (label 7 in Figure 5.5) is lower than that of the latter (label 5 in Figure 5.5). 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 burster 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 5.4 – 5.5, 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.

In Table 5.2 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 5.2 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 5.3 presents similar results for the same network with alternate routing (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.

Our model can also calculate a mesh network with more than one alternate paths. Figure 5.6 and Figure 5.7 show the results for two and three alternate paths. These results are not as accurate as those with one alternate route, because the assumption that overflow traffic is Poisson arrival causes larger errors on the second or third alternate route.

### 5.2.2 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 5.8. Since we will be using traffic data reported in [6], following that study, we have also augmented the 14-node NSFNET topology by adding two fictitious nodes, nodes 1 and 16 in Figure 5.8, 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 5.9 through 5.12 are shown in Table 5.4.

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 $\lambda_{sd}$ for a source-destination pair $(s, d)$ is given by (note that no shortest path is longer than

4 hops):

$$\lambda_{sd} = \begin{cases} 0.5, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 1 \\ 0.4, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 2 \\ 0.3, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 3 \\ 0.2, & \text{if the length of the shortest path from } s \text{ to } d \text{ is } 4 \end{cases} \tag{5.9}$$

The second traffic pattern was designed to reflect actual traffic statistics collected on the NSFNET backbone network, as reported in the traffic matrix in [6, 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 [6, Figure 6] by the link capacity (T3 links) to obtain the "offered load" $\rho_{sd}$ 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 $\lambda_{sd}$ 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 [6].

Figures 5.9 and 5.10 present the call blocking probabilities for the selected pairs of Table 5.4 and the first traffic pattern. Figure 5.9 shows results for fixed routing, while Figure 5.10 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.

Comparing Figure 5.9 to Figure 5.4 we observe that, in the NSFNET topology, as in the regular $5 \times 5$ 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 5.9 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 5.10 and 5.5. 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 5.9 and 5.10. 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 5.11 and 5.12 are similar to Figures 5.9 and 5.10, respectively, except that they present results for the second traffic pattern derived from the traffic statistics presented in [6]. The utilization under this traffic pattern is in the range [0.015, 8.059] with an average of 3.976 for fixed routing, and in the range of [0.014, 9.231] with an average of 4.168 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 5.11 and 5.12 is very similar to that in Figures 5.9 and 5.10, 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.

Another interesting observation from Figures 5.11 and 5.12 is that, with the exception of a few source-destination pairs, using an alternate path does not have a significant impact on the call blocking performance. Consider, for instance, the source-destination pairs (6,7) and (12,16) (labels 3 and 11 in the figures). The primary path for pair (6,7) is short (one hop) but the alternate path is quite long (five hops), thus, using an alternate path does not improve the blocking probability experienced by this pair. On the other hand, both the primary and the alternate paths for pair (12,16) are two hops long, thus, using an alternate path reduces the overall blocking probability for this pair by about four orders of magnitude! In general, source-destination pairs for which the alternate paths are longer than the primary paths experience only a slight drop (if any) in blocking probability under alternate routing. Since the traffic load offered to the network is high and the blocking probabilities also high, adding more traffic to the network through alternate paths does not improve the performance except for pairs for which the alternate path is short and not highly utilized, as is the case with pair (12,16).

Finally, Tables 5.5 to 5.8 present a summary comparison of analytical and simu-

lation 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 5.8, 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 5.11 and 5.12 (i.e., pair (15,16) in Figure 5.8) 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.

**Decomposition Algorithm for Mesh Networks with Fixed Routing**

**Input:** Network topology, set $\mathcal{R}$ of paths for all source-destination pairs, and arrival rates $\lambda_{sd}$

**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 5.1.2
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
   // accountf 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 5.2: Path decomposition algorithm

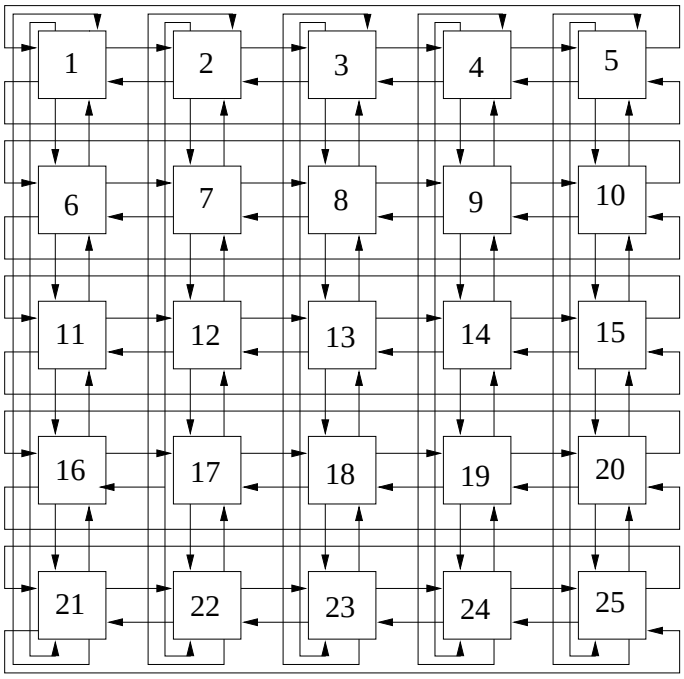


Figure 5.3: A $5 \times 5$ bidirectional mesh torus network

Table 5.1: 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 |

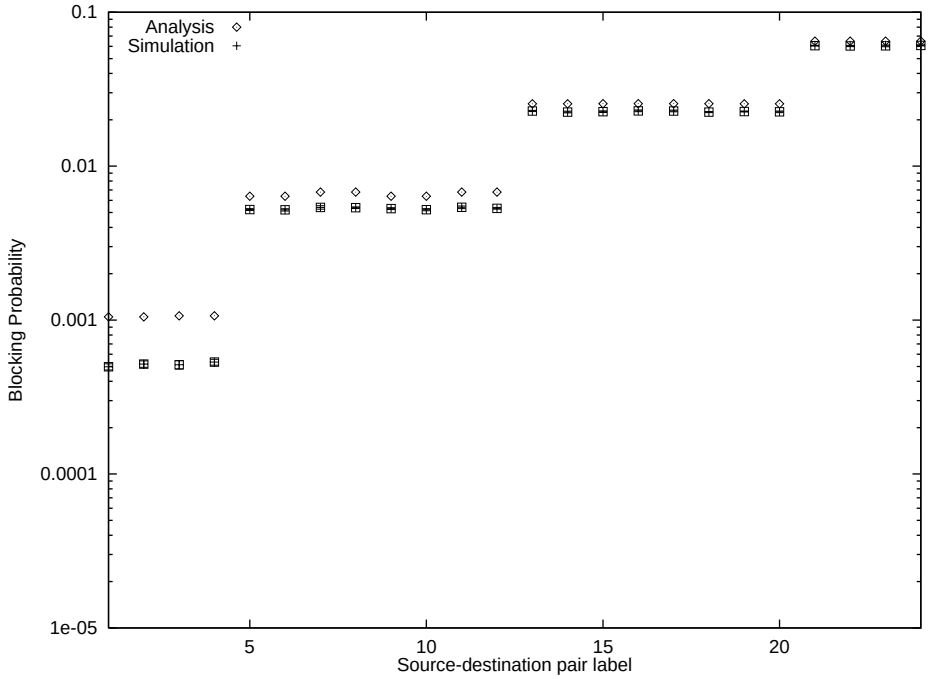


Figure 5.4: Blocking probability for selected source-destination pairs in the $5 \times 5$ torus network with $W = 10$ and fixed routing

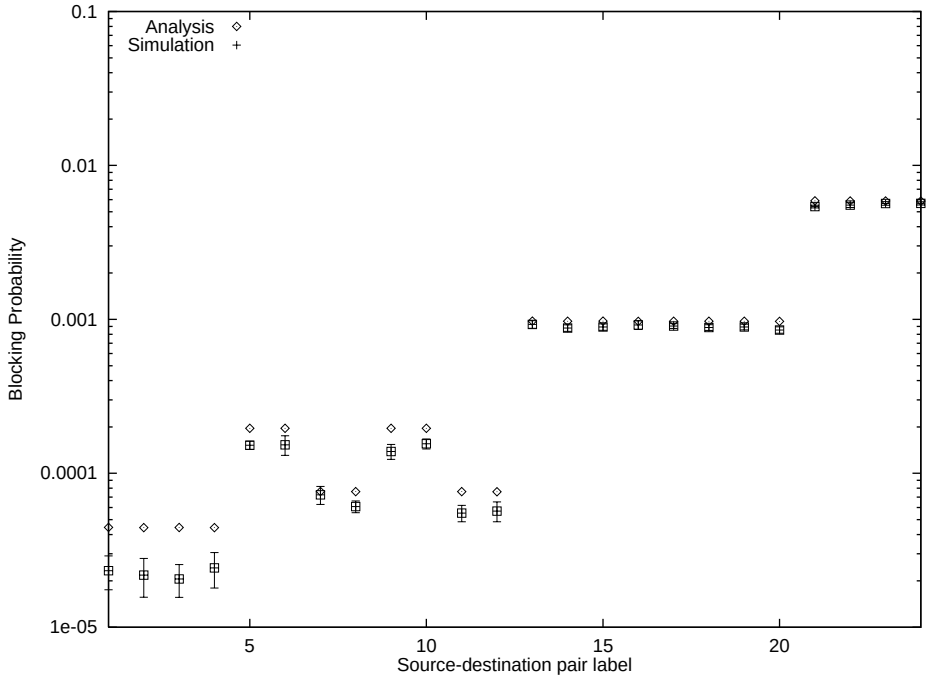


Figure 5.5: Blocking probability for selected source-destination pairs in the $5 \times 5$ torus network with $W = 10$ and alternate routing (one alternate path)

Table 5.2: Summary of results for the $5 \times 5$ 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 5.3: Summary of results for the $5 \times 5$ 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% |

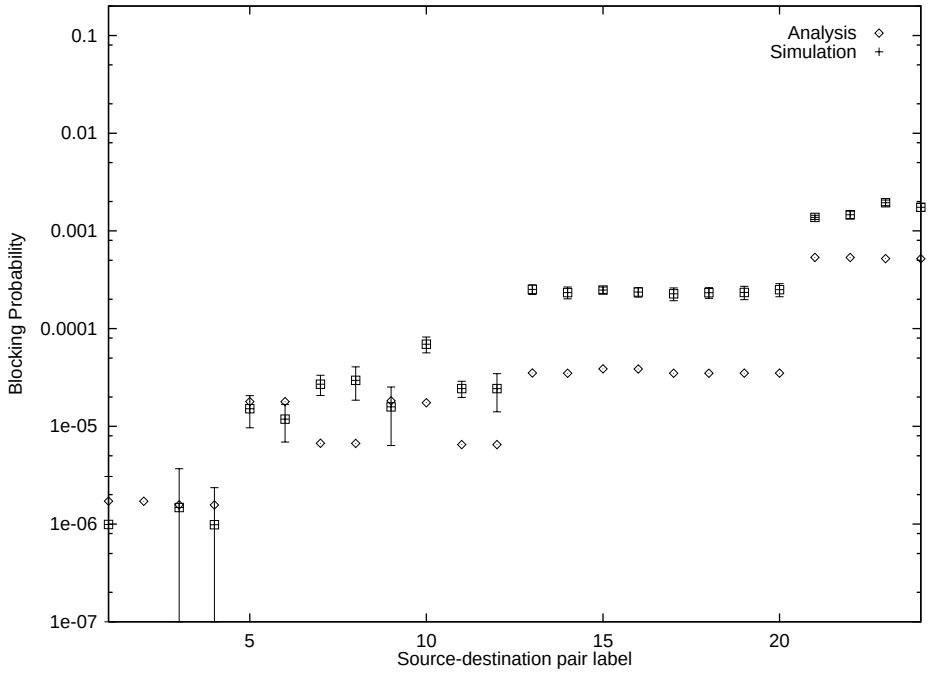


Figure 5.6: Blocking probability for selected source-destination pairs in the $5 \times 5$ torus network with $W = 10$ and alternate routing (two alternate paths)

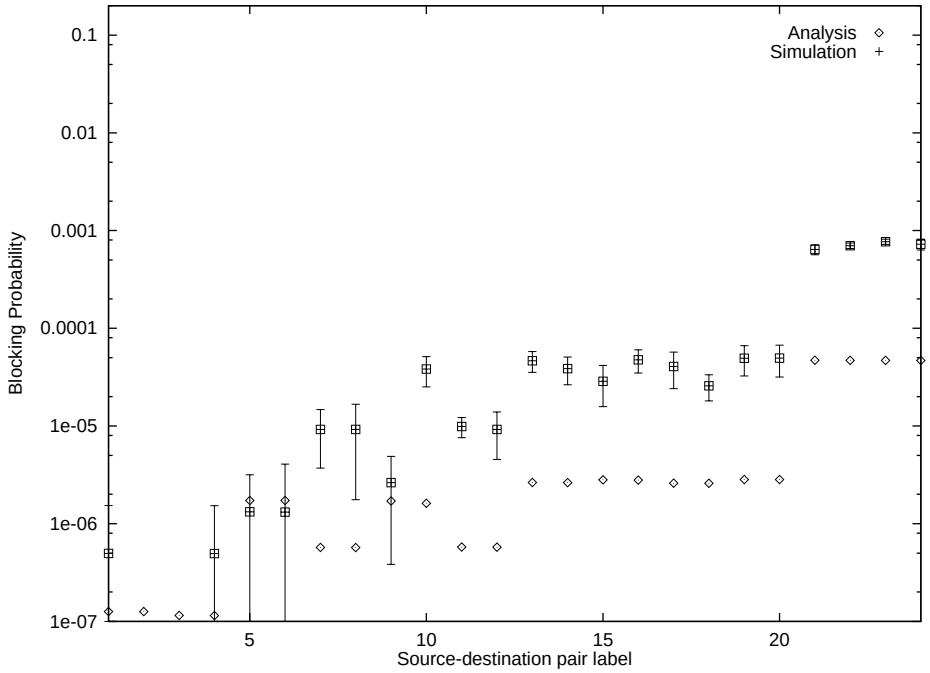


Figure 5.7: Blocking probability for selected source-destination pairs in the $5 \times 5$ torus network with $W = 10$ and alternate routing (three alternate paths)

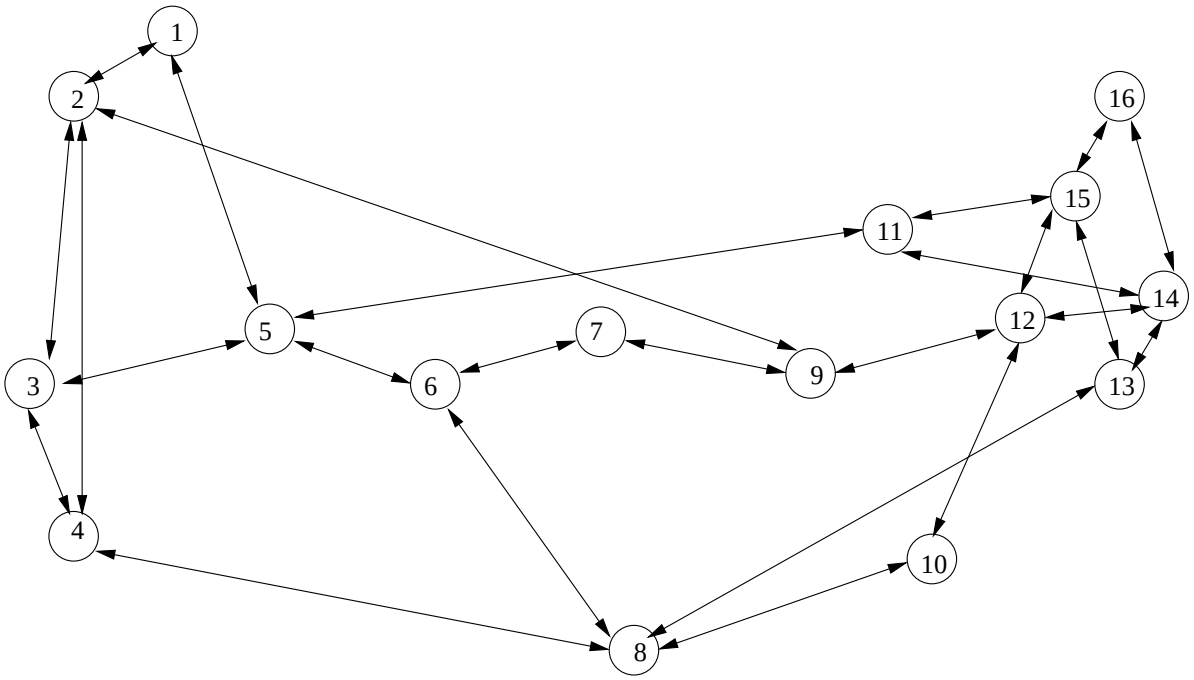


Figure 5.8: The NSFNET topology

Table 5.4: 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 | | |

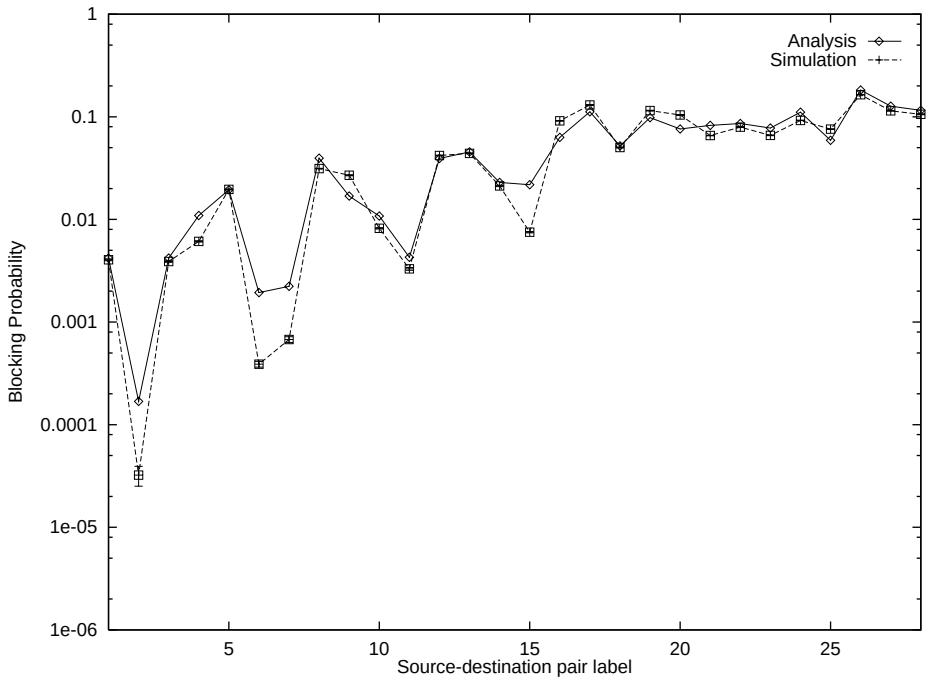


Figure 5.9: Blocking probability for selected source-destination pairs in the NSFNET with $W = 10$ and fixed routing (first traffic pattern)

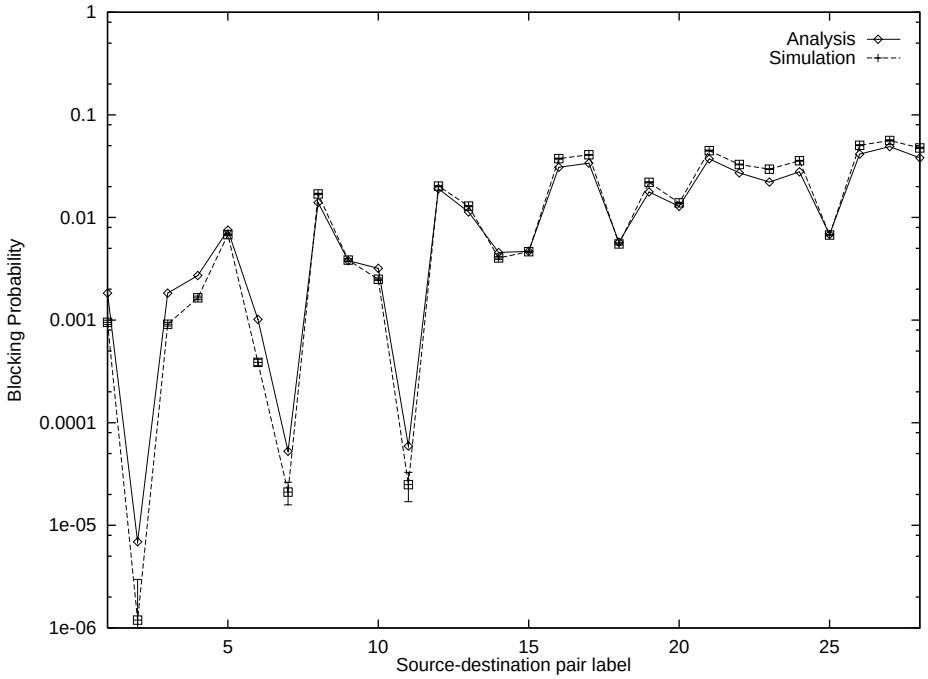


Figure 5.10: Blocking probability for selected source-destination pairs in the NSFNET with $W = 10$ and one alternate path (first traffic pattern)

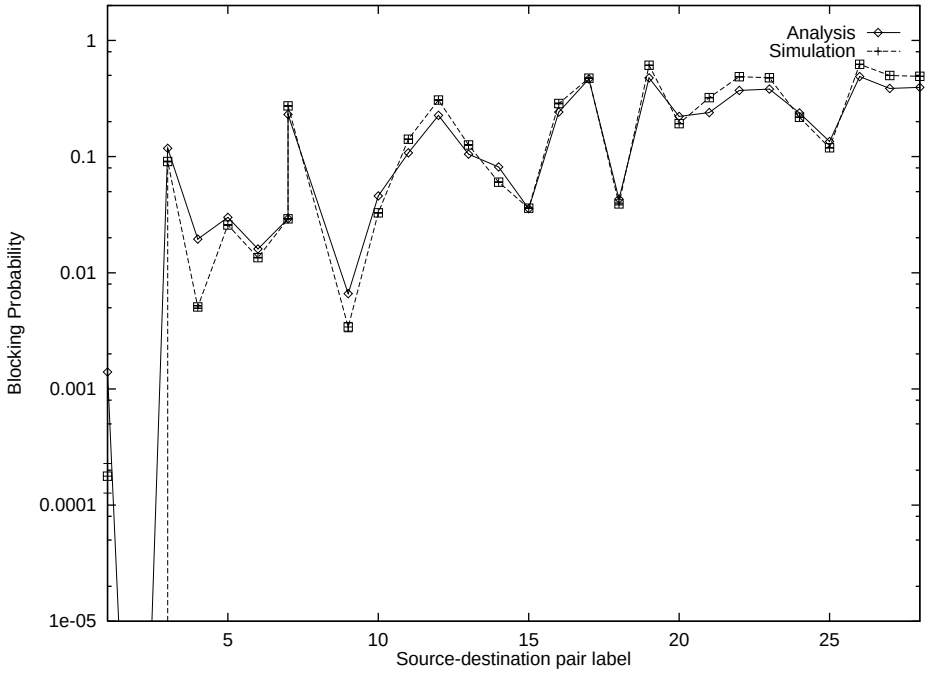


Figure 5.11: Blocking probability for selected source-destination pairs in the NSFNET with $W = 10$ and fixed routing (second traffic pattern)

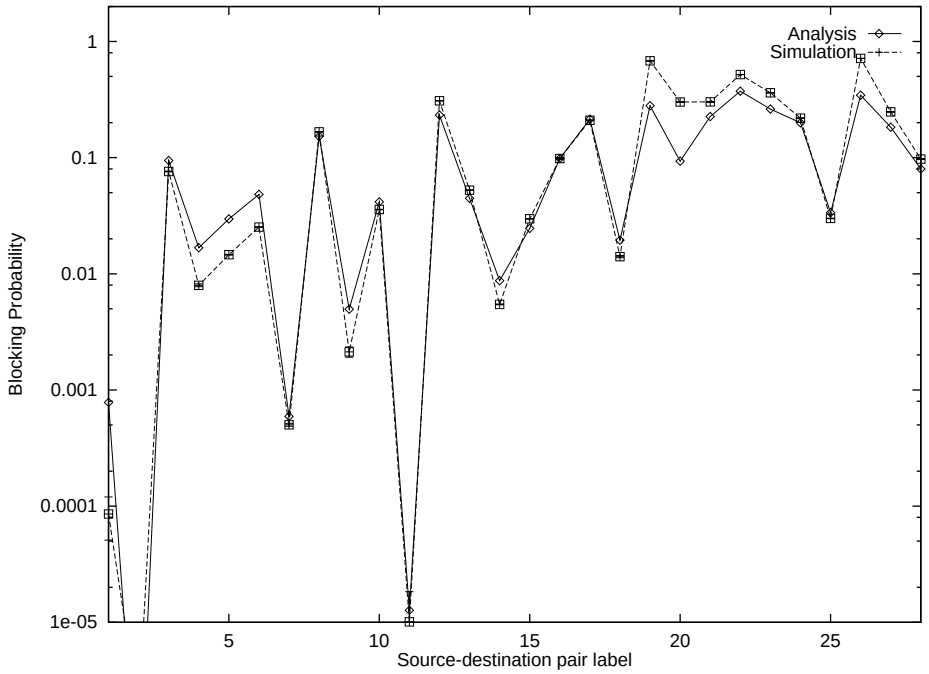


Figure 5.12: Blocking probability for selected source-destination pairs in the NSFNET with $W = 10$ and one alternate path (second traffic pattern)

### 5.2.3 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.

The issue of converter placement in a single path of a network has been studied in Chapter 4 and [29]. In Chapter 4, we used the single-path decomposition algorithm 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 [29], 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 $\begin{pmatrix} N \\ K \end{pmatrix}$ 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 [2] 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

[2] 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.

Table 5.5: 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 5.6: 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 5.7: 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% |

(for fixed or alternate routing) described in Section 5.1.

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 5.13 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 5.14 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 5.13 and 5.14.

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 [28, 29]. An important observation from Figures 5.13 and 5.14 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 0.645 when there are no converters to 0.492 with four converters, a decrease of only 24%.

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.

## 5.3 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 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.

Table 5.8: 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% |

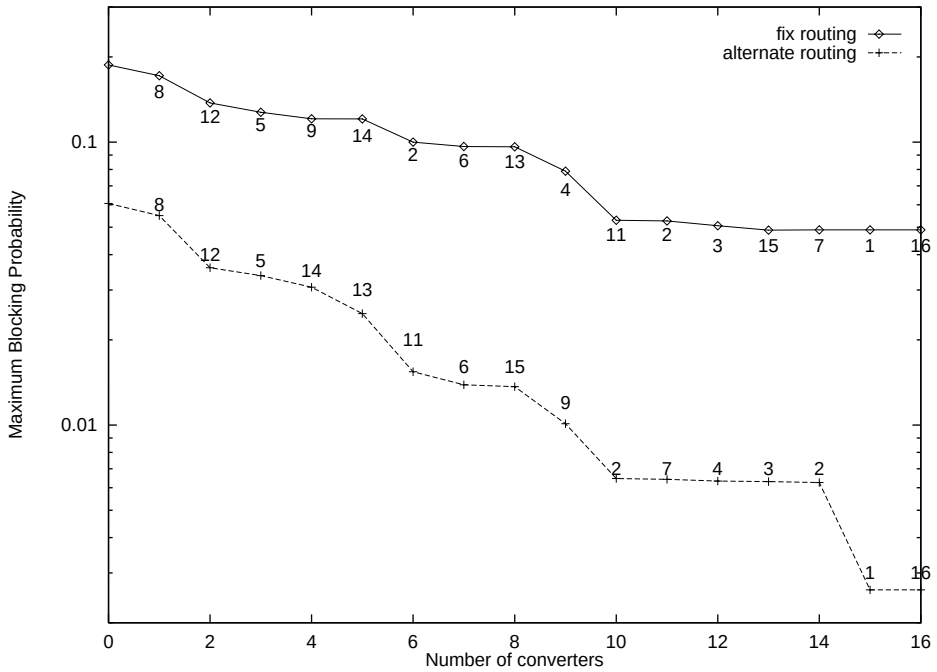


Figure 5.13: Maximum blocking probability in the NSFNET topology with converters (first traffic pattern)

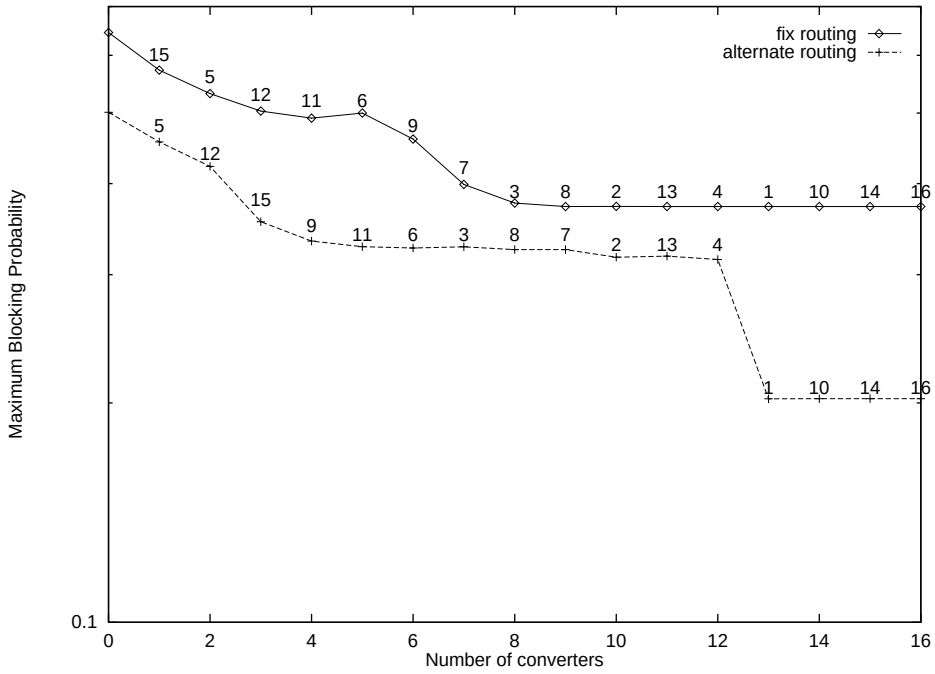


Figure 5.14: Maximum blocking probability in the NSFNET topology with converters (second traffic pattern)

# Chapter 6

# Comparison of Wavelength Allocation Policies

In the previous two chapters, we only used the random wavelength allocation policy. In this chapter, we study the blocking performance of several wavelength allocation policies for various network topologies and traffic patterns.

## 6.1 A Single Path of A Wavelength Routing Network

Let us first consider a single path of a circuit-switched wavelength routing network, such as the $k$-hop path shown in Figure 4.1. A $k$-hop path consists of $k+1$ nodes labeled $0, 1, \cdots, k$, and hop $i, i = 1, \cdots, k$, represents the link between nodes $i-1$ and $i$. (Unless noted otherwise, the terms "hop" and "link" will be used interchangeably.) Each link in the path supports exactly $W$ wavelengths, and each node is capable of transmitting and receiving on any of these $W$ wavelengths. We let $\lambda_{ij}, j \geq i$, denote the Poisson arrival rate of calls that use hops $i$ through $j$ of the path, i.e., calls that originate at node $i-1$ and terminate at node $j$. For instance, $\lambda_{22}$ is the arrival rate of calls that only use hop 2 (that is, those arriving at node 1 and leaving at node 2), while $\lambda_{12}$ is the arrival rate of calls using hops 1 and 2 (refer to Figure 4.1). If the request can be satisfied, an optical circuit is established between the source and destination for the duration of the call. Call holding times are exponentially distributed with mean $1/\mu$. Also, $\rho_{ij} = \lambda_{ij}/\mu$ is the offered load of calls using hops $i$ through $j$.

We define a "segment" of a $k$-hop path as a sub-path consisting of one or more

consecutive links of the original path. We let $n_{ij}, j \geq i$, be a random variable representing the number of calls using hops $i$ through $j$ that are currently active in the path. We also let $f_{ij}, j \geq i$, be a random variable representing the number of wavelengths that are free on all hops $i$ through $j$. As we shall shortly see, random variables $n_{ij}$ and $f_{ij}$ are part of the state description for the Markov process corresponding to the $k$-hop path.

In our model, we allow some of the nodes in the path to employ wavelength converters. These nodes can switch an incoming wavelength to an arbitrary outgoing wavelength. If no wavelength converters are employed in the path, a call can only be established if the *same* wavelength is free on all the links used by the call. This is known as the *wavelength continuity* requirement, and it increases the probability of blocking for calls using multiple hops. 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 assigned one of the wavelengths that are available on the links used by the call. If there multiple available wavelengths, a wavelength allocation policy must be employed to select a wavelength for the call. Different selection policies lead to different call blocking probabilities. In this chapter we investigate the following wavelength allocation policies:

- *Random allocation:* a call is randomly assigned one of the wavelengths that are available on the links used by the call. This policy has been extensively studied in the literature, and we have developed approximate analytical algorithms to evaluate their performance in Chapter 4 and 5.

- *Most-used allocation:* the wavelength that is used on the largest number of links in the path (other, of course, than those used by the call) is assigned to the call; ties are broken arbitrarily. The objective of the policy is to keep more wavelengths available on long paths.

- *Least-used allocation:* the call is assigned the wavelength used in the smallest number of links in the path, with ties broken arbitrarily. Intuitively, this policy results in "wavelength fragmentation," and will lead to higher blocking probability for calls traveling over long paths.

- *First-fit allocation:* the wavelength are given labels in a fixed order, and the call is assigned the wavelength with the smallest label that is available on the links it uses. The objective of this allocation is to minimize wavelength fragmentation. As we shall

show later, its performance is very close to that of the most-used policy, but it is easier to implement since there is no need to maintain information about global use of wavelengths.

In a path with wavelength converters, the corresponding allocation policy is used to assign a wavelength to the call within each segment of the path whose starting and ending nodes are equipped with converters. In addition to the above wavelength allocation policies, we will also study

- *Circuit-switched paths*: paths in which there are converters at all nodes. In circuit-switching, a call can be established as long as at least one wavelength (not necessarily the same one) is free on the links used by the call. Consequently, wavelength allocation is not an issue under a circuit-switching scenario, and all allocation policies, including the ones studied here, reduce to random allocation within each link.

In our study, we will use six different traffic load patterns to compare the wavelength allocation policies against each other and against circuit-switching. The six patterns are representative of the wide range of loading situations that one expects to encounter in practice. Figures 4.13 and 4.14 illustrate the six traffic patterns for a 10-hop path. Specifically, the figures plot the load $\rho_l$ of each hop $l$, $l = 1, \cdots, 10$, in the path, defined as the sum of the offered loads $\rho_{ij}, i \leq l \leq j$, for all calls that use hop $l$, for each load pattern. In the "uniform" pattern, all hops are equally loaded. The "bowl" (respectively, "inverted bowl") pattern is such that the load decreases (resp., increases) from hop 1 to hop 5, and then it increases (resp., decreases) from hop 6 to hop 10. These patterns are shown in Figure 4.13. The "ascending" and "descending" patterns are such that the load increases or decreases, respectively, from hop 1 to hop 10. Finally, in the oscillating pattern the load at each hop alternates between a low and a high value. The last three load patterns are shown in Figure 4.14. Similar load patterns were used for shorter paths. To ensure that the results are comparable across the different patterns, the load values were chosen so that the total load (or, equivalently, the average load per hop) is the same for all patterns.

### 6.1.1 Policy Comparison for 2-hop Paths

We will first study the performance of the various wavelength allocation policies for 2-hop paths as in Figure 4.2. The state space of these systems is small enough to

obtain exact numerical solutions for the call blocking probabilities, providing insight into the relative performance of the different policies.

### Exact and Approximate Markov Processes

We have shown in Chapter 4 that a 2-hop path with random wavelength allocation can be characterized by the four-dimensional Markov process $(n_{11}, n_{12}, n_{22}, f_{12})$. The first three random variables in the state description provide the number of active calls between the three source-destination pairs in the path, and the last random variable gives the number of wavelengths that are free on both links of the path. The state transition diagram of this Markov process is shown in Figure 4.3 for $W = 2$ wavelengths, and it is straightforward to see that the process is not time-reversible in Chapter 4. By modifying a few of the transition rates of this process, we have been able to derive an approximate time-reversible Markov process with the same state space, which has a product-form solution. If we let $G(W)$ denote the normalizing constant for a 2-hop path with $W$ wavelengths per link, the solution of the approximate Markov process is given by Equation 4.14 in Chapter 4. We have demonstrated in Chapter 4 that the blocking probabilities obtained through the product-form solution to the approximate Markov process are very close to the blocking probabilities obtained through the numerical solution to the original Markov process for a wide range of traffic loads.

Let us now consider the same 2-hop path with the most-used wavelength allocation policy. This policy can be modeled by a Markov process with the state description as that for the random policy: $(n_{11}, n_{12}, n_{22}, f_{12})$. The key difference is that, under the most-used policy, if, say, $n_{11} > n_{22}$, we know that there is at least one wavelength that is used on hop 1 but not used on hop 2. Thus, an incoming call that uses the second hop only will be assigned a wavelength that is already in use on the first hop, and will cause a transition to state $(n_{11}, n_{12}, n_{22} + 1, f_{12})$; similarly for $n_{22} > n_{11}$ and incoming calls using only the first hop. (Under the random wavelength allocation policy, the transition could be to either state $(n_{11}, n_{12}, n_{22} + 1, f_{12})$ or to state $(n_{11}, n_{12}, n_{22} + 1, f_{12} - 1)$ if the number of free wavelengths on both hops $f_{12} > 0$ and one of these wavelengths is assigned to the call.)

The state transition diagram of the Markov process for the most-used allocation policy is shown in Figure 6.2 for a 2-hop path with $W = 2$ wavelengths. Again, it is straightforward to verify that this Markov process is not time-reversible. Comparing to

Figure 6.1, we note that despite having the same state space, the two processes differ in two ways. First, some of the transition rates are different; for instance the transition rate from state (0,0,1,1) to state (1,0,1,1) is equal to $\lambda_{11}/2$ for the random allocation, but $\lambda_{11}$ for the most-used allocation. Second, some of the transitions are missing in the new Markov process. For example, there is a transition from state (0,0,1,1) to state (1,0,1,0) under random allocation in Figure 6.1, but there is no such transition in Figure 6.2. Furthermore, since there is a transition from state (1,0,1,0) to state (0,0,1,1) in Figure 6.2, but no transition in the reverse direction, it is not possible to obtain an approximate time-reversible process by simply modifying some of the transition rates, as we were able to do for the random wavelength allocation policy. Although we do not have an approximate product-form solution for the most-used allocation policy, the state space for a 2-hop path is small enough that the solution to the Markov process can be obtained numerically for up to $W = 20$ wavelengths.

Based on similar arguments, it can be determined that the least-used wavelength allocation policy can also be modeled by a Markov process with the state description $(n_{11}, n_{12}, n_{22}, f_{12})$. The state transition diagram for this process is shown in Figure 6.3, and it can be easily verified that the process is not time-reversible.

If a converter is placed at node 1 of the 2-hop path in Figure 4.2 (the only interesting possibility in this case), the system becomes equivalent to a 2-hop circuit-switched path, and it can be described by the three-dimensional Markov process $(n_{11}, n_{12}, n_{22})$. Random variable $f_{12}$ becomes redundant because calls continuing on both hops can now use *any* of the $(W - n_{12} - n_{22})$ available wavelengths on the second hop. It is well-known that this Markov process has the closed-form solution:

$$\pi_{cs}(n_{11}, n_{12}, n_{22}) \quad = \quad \frac{1}{G(W)} \, \frac{\rho_{11}^{n_{11}}}{n_{11}!} \, \frac{\rho_{12}^{n_{12}}}{n_{12}!} \, \frac{\rho_{22}^{n_{22}}}{n_{22}!} \tag{6.1}$$

It is also interesting to note that the normalizing constant in (6.1) is the same as that in (**??**). In Figure 6.4 we show the state space of a 2-hop circuit switched path with two wavelengths. Although this path is described by the above 3-dimensional Markov process, we include in the state description of Figure 6.4 the random variable $f_{12}$ to make it easier to compare to Figures 6.1–6.3. For instance, the fact that there are no transitions into state (1,0,1,0) in the figure can be explained by recalling that $f_{12} = 0$ (i.e., that no wavelength is free on both links of the path) implies that calls traversing both hops are blocked. However, since exactly one wavelength is free on each hop (even if it is not the same one), calls traversing both hops cannot be blocked in the circuit-switched path, and the system will never enter

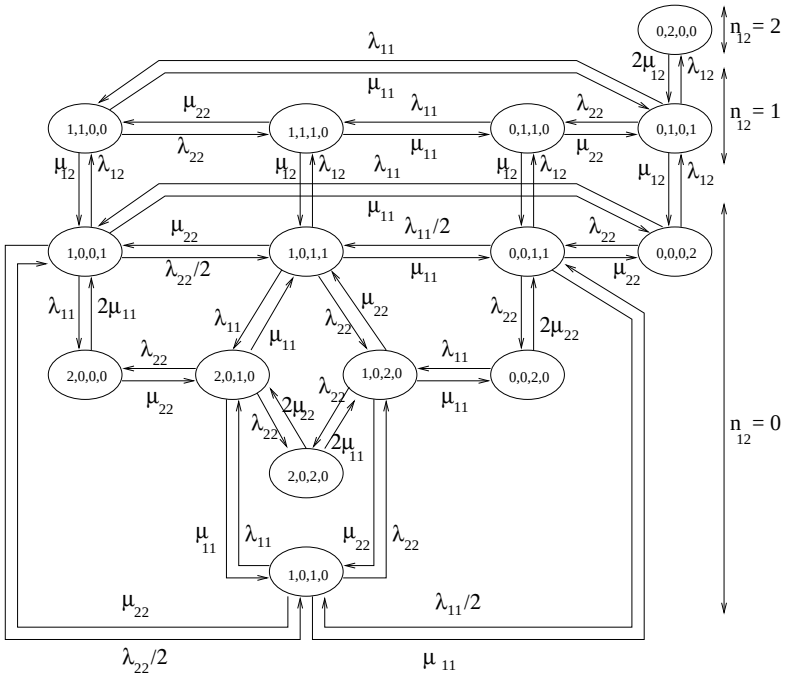


Figure 6.1: State space $(n_{11}, n_{12}, n_{22}, f_{12})$ of a 2-hop path with $W = 2$ wavelengths (random allocation)

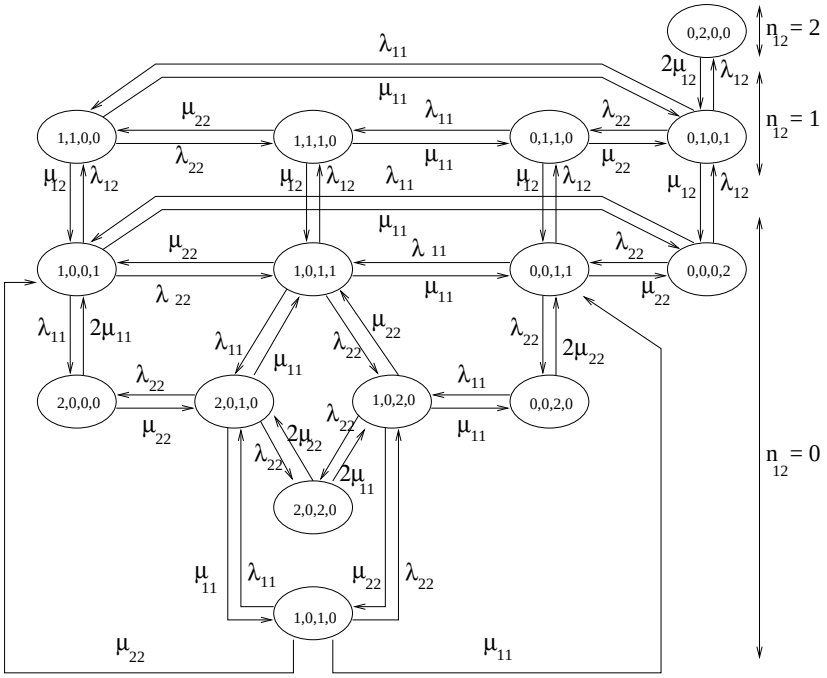


Figure 6.2: State space $(n_{11}, n_{12}, n_{22}, f_{12})$ of a 2-hop path with $W = 2$ wavelengths (most-used allocation)

state (1,0,1,0), but only state (1,0,1,1).

The first-fit wavelength allocation policy can be modeled by a Markov process with $W$ state variables $(l_1, \cdots, l_W)$. Each random variable $l_i$ corresponds to one of the $W$ wavelengths, and can take one of five values representing the status of wavelength $i$ on the two links of the path: 0, if the wavelength is free on both links, 1, if it is free on the first link and busy on the second, 2, if it is busy on the first link and free on the second, 3, if the wavelength is used by two different calls on each link, and 4, if it is used by a call traversing both links of the path. The state space of this Markov process is quite different than that in Figures 6.1–6.4, and no direct comparisons can be made. Furthermore, the size of the state space is in the order of $W^5$, too large to obtain a numerical solution even for relatively small values of $W$. The main motivation for considering the first-fit policy in this work is that simulation results indicate that its performance is very similar to that of the most-used policy, while it is easier to implement.

### Numerical Comparison

Let us first consider the blocking performance of the random, most-used, least-used, and circuit-switched systems for calls traversing both links of the 2-hop path. In Figures 6.1–6.4, the blocking states for these calls are those with the last random variable $f_{12} = 0$, i.e., those states in which neither of the two wavelengths is free on both links. We also observe that, except for state (1,0,1,0) at the bottom of the four figures, the transitions (and transition rates) in and out of all other blocking states are exactly the same for all four wavelength allocation policies. Consequently, we expect that the difference in the blocking probability experienced by calls traversing both links of the path under the different policies will be mainly due to the steady-state probability of blocking state (1,0,1,0).

Referring to Figure 6.4, we note that the corresponding Markov process never enters state (1,0,1,0). Thus, we expect that calls traversing both hops will experience the least blocking probability in a circuit-switched path. In Figure 6.2 (most-used policy) we note that there are two transitions into state (1,0,1,0), and four transitions out of it. The blocking probability will be higher under this policy compared to the circuit-switched case. The Markov process in Figure 6.1 (random policy) has two additional transitions into state (1,0,1,0) from states (0,0,1,1) and (1,0,0,1) with rates $\lambda_{11}/2$ and $\lambda_{22}/2$, respectively. Therefore, the blocking probability of these calls with the random policy will be higher than

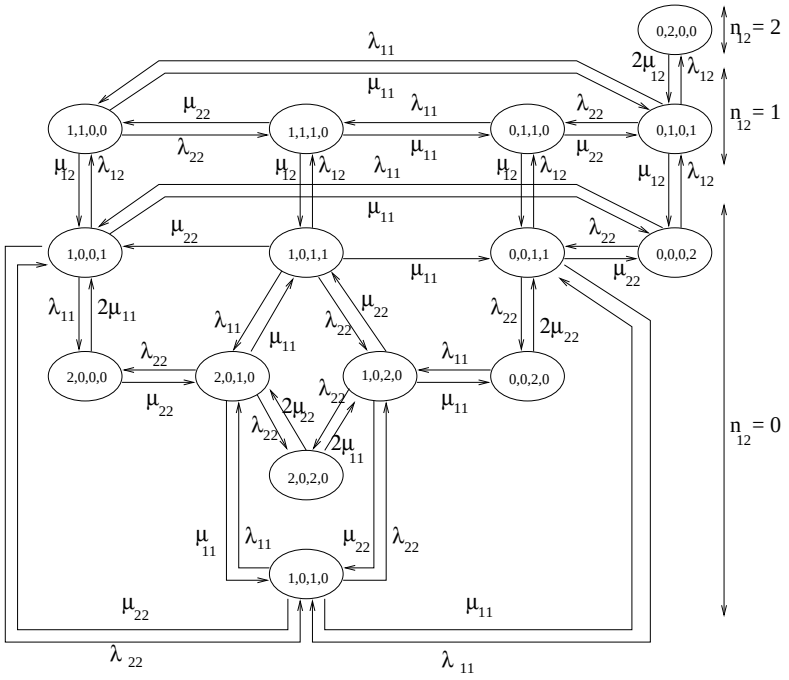


Figure 6.3: State space $(n_{11}, n_{12}, n_{22}, f_{12})$ of a 2-hop path with $W = 2$ wavelengths (least-used allocation)

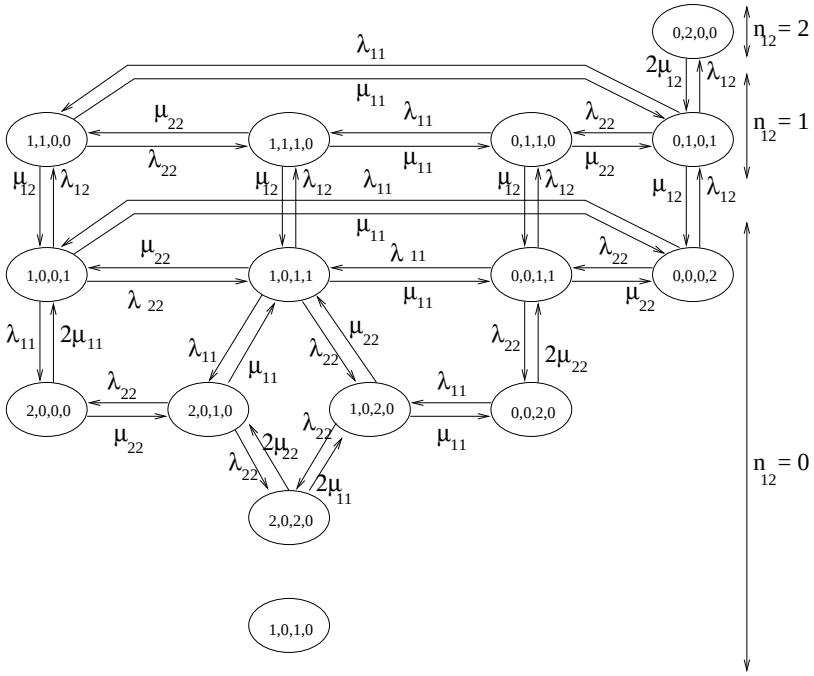


Figure 6.4: State space $(n_{11}, n_{12}, n_{22})$ of a 2-hop path with $W = 2$ wavelengths (circuit-switched)

with the most-used policy. Finally, the Markov process in Figure 6.3 (least-used policy) has the same transitions as the one in Figure 6.1, but the transition rates into state (1,0,1,0) from states (0,0,1,1) and (1,0,0,1) are $\lambda_{11}$ and $\lambda_{22}$, respectively. Therefore, we expect that these calls will experience the highest blocking probability under the least-used policy.

We now note that the lower the blocking probability for calls traversing both hops, the larger the number of such calls accepted, and the larger the number of wavelengths they occupy, leaving fewer wavelengths available for calls using a single link (either the first or the second) of the path. Hence, we expect that the behavior of the four policies in terms of the blocking probability of calls using a single link of the path will be exactly the opposite of what was discussed above. Specifically, we expect the least-used policy to provide the lowest blocking probability for these calls, followed by the random, the most-used, and the circuit-switched policies, in that order.

The above conclusions, derived by direct comparison of the Markov processes, are in agreement with intuition. We have confirmed these conclusions by numerically comparing the blocking probabilities of the various policies for 128 different load values. Figures 6.5 and 6.6 show results for two cases corresponding to a uniform and descending load pattern, respectively, similar to the corresponding patterns shown in Figures 4.13 and 4.14, and for $W = 10$ wavelengths. More specifically, the arrival rates (refer also to Figure 4.2) used in Figure 6.5 were $\lambda_{11} = 0.2, \lambda_{12} = 0.2, \lambda_{22} = 0.1$, while for the results in Figure 6.6 we used $\lambda_{11} = 3.0, \lambda_{12} = 2.0, \lambda_{22} = 2.0$. The two figures plot the blocking probability for the three types of calls, namely, calls using the first hop only (label "hop 1" in the $x$-axis of the figures), calls using the second hop only (label "hop 2"), and calls using both hops (label "both hops"). We first note that the results are consistent with the corresponding traffic pattern. For instance, under uniform loading (Figure 6.5), calls using the first hop only experience the same blocking probability as hops using the second hop only, while in the descending pattern (Figure 6.6), due to the lower load offered to the second hop, the latter calls experience much lower blocking probability for all four policies. More importantly, the relative values of the blocking probabilities for the four policies are also consistent with our discussion above. Very similar results have been obtained for all 128 different load values that we have studied.

Finally, in Figure 6.7 we compare the most-used and first-fit policies for the same arrival rates as those used for Figure 6.6. As before, the blocking probabilities of the most-used policy were obtained through a numerical solution to the corresponding Markov

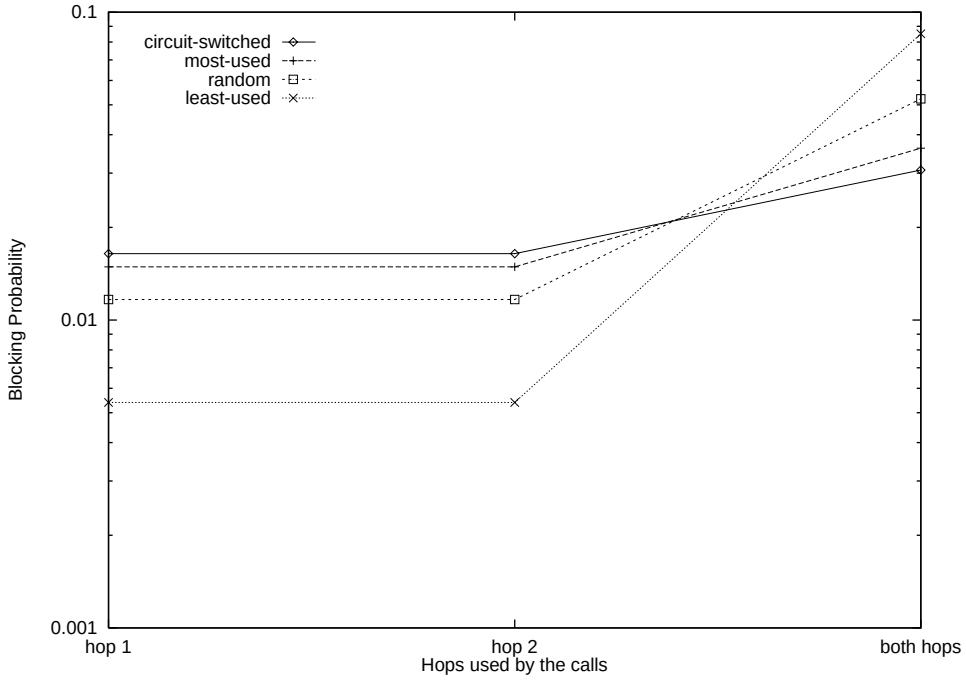


Figure 6.5: Policy comparison, 2-hop path, uniform traffic pattern

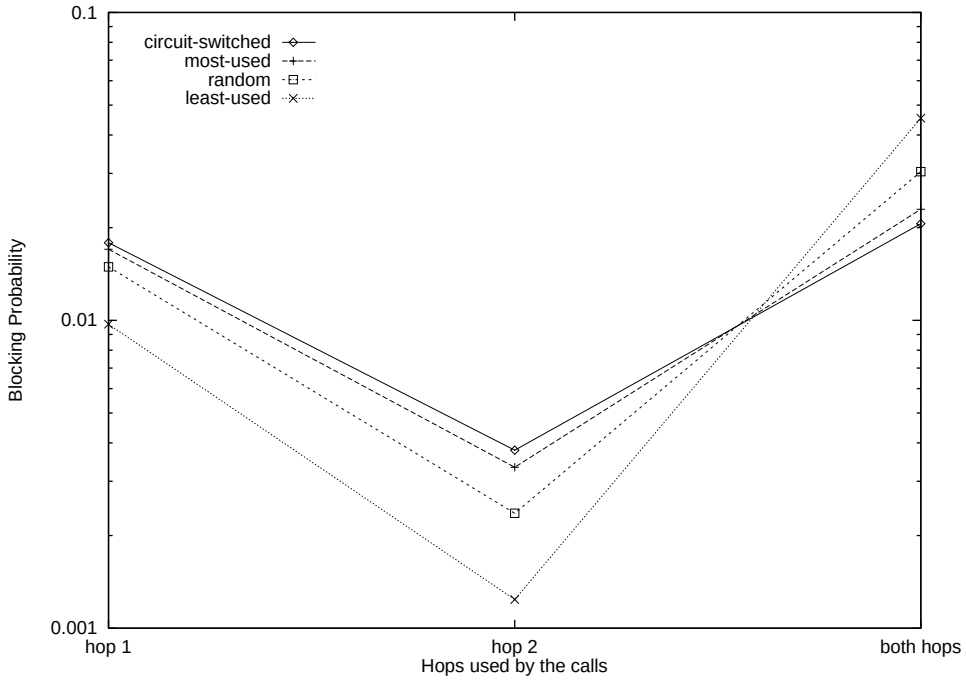


Figure 6.6: Policy comparison, 2-hop path, descending traffic pattern

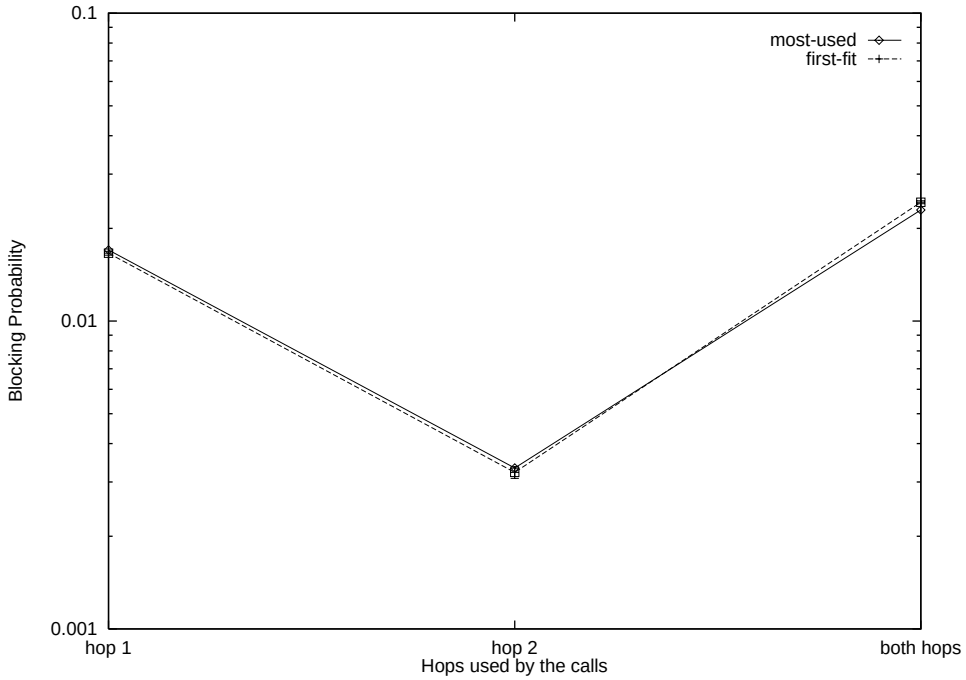


Figure 6.7: Most-used vs. first-fit allocation, 2-hop path, descending traffic pattern

process, while the values for the first-fit policy were obtained through simulation. We observe that the blocking probabilities values of the first-fit policy are almost identical to those of the most-used policy for all three types of calls. This result can be explained by noting that both policies attempt to maximize the number of wavelengths that are available for calls that use both hops of the 2-hop path by reducing the "fragmentation" of the set of wavelengths. The most-used policy assigns to an incoming call that uses a single hop of the path a wavelength that is already used on the other hop, if such a wavelength exists. On the other hand, the first-fit policy attempts to achieve the same goal by searching the set of wavelengths in a fixed order, thus increasing the chances that a wavelength used on a single hop will be assigned to an incoming call using the other hop. As we can see from Figure 6.7, the most-used policy is slightly more successful, but overall the blocking probability values of the two policies are very close. Similar results have been obtained for all 128 traffic loads that we have studied.

### 6.1.2 Policy Comparison for Longer Paths

Consider a $k$-hop path, $k > 2$, with the random wavelength allocation policy. Paths consisting of four links or less can be analyzed by solving the corresponding approximate time-reversible Markov process. For instance, a 3-hop path can be modeled by the 9-dimensional Markov process $(n_{11}, n_{12}, n_{13}, n_{22}, n_{23}, n_{33}, f_{12}, f_{13}, f_{23})$ whose solution can be written down as a straightforward generalization of expression (**??**). Paths longer than four hops are analyzed using the iterative decomposition algorithm in Chapter 4 to obtain the call blocking probabilities. The analytical techniques developed in Chapter 4 are both accurate and efficient, and can be used when the path employs converters whose location is fixed but arbitrary. When all nodes in a $k$-hop path employ converters (the circuit-switched case), the call blocking probabilities can also be obtained by solving the corresponding $k$-dimensional Markov process through a straightforward generalization of expression (6.1). For very large $k$, when the computation of the normalizing constant becomes computationally expensive, a decomposition algorithm similar to the one in Chapter 4 can be used.

Let us now consider the most-used wavelength allocation policy. It is not difficult to derive an exact Markov process to model a $k$-hop path, $k > 2$. However, unlike the 2-hop path case, the state description of this process for a $k$-hop path, $k > 2$, contains a number of random variables that is larger than the $k^2$ random variables in the state description of the process for the same path with random allocation. For a 3-hop path, for example, the state of the Markov process for most-used allocation is described by 12 random variables, the nine used to describe the corresponding process for random allocation plus three additional ones. The larger state space, combined with the fact that an approximate closed-form solution is not possible for this process, makes it difficult to numerically obtain the call blocking probabilities under this policy for paths longer than two hops by directly solving the exact Markov process. Furthermore, developing an iterative algorithm for analyzing long paths by decomposing them into 2-hop path sub-systems which can be solved in isolation, similar to the algorithm developed for the random policy in Chapter 4, has turned out to be a difficult task. For such an algorithm, it is crucial to have an accurate estimate of the blocking probability due to the wavelength continuity requirement for calls traversing more than one sub-system. Since the different wavelengths are not equally utilized, as under random allocation, it is difficult to derive an approximate expression for blocking due to the wavelength continuity requirement that is accurate for a wide range of loads.

Using similar reasoning, it can be seen that developing approximate analytical techniques for the least-used and first-fit policies is also a rather hard task. Instead, we have used simulation to obtain the call blocking probabilities for paths with the most-used, least-used, and first-fit wavelength allocation policies [1].

### Numerical Comparisons

In this section we present results for 6-hop and 10-hop paths, since the length of these paths (in hops) is representative of future backbone wavelength routing networks. Figures 6.8-6.11 correspond to a 6-hop path, while Figures 6.12-6.16 are for a 10-hop path.

In Figures 6.8 and 6.9 we compare the blocking probabilities for the four policies (random, most-used, least-used, and circuit-switched) and for the uniform and bowl traffic patterns, respectively. The two figures plot the blocking probability for the twenty one (21) different types of calls in a 6-hop path, numbered 1 through 21 in the $x$-axis. It is important to emphasize that the calls have been numbered such that numbers 1 through 6 correspond to the calls traversing a single hop in the path (i.e., hops 1 through 6, respectively), numbers 7 to 11 correspond to calls that traverse exactly two hops in the path, and so on.

From Figures 6.8 and 6.9 we observe that the relative behavior of the four policies is similar to that shown in Figures 6.5 and 6.6 despite the fact that the traffic patterns in these figures are very different. Specifically, for calls using only one or two hop (calls one through eleven in the figures), the least-used policy provides the lowest blocking probability, followed by the random policy, the most-used policy, and the circuit-switched case. However, for calls traversing multiple hops, the situation is reversed. The same behavior has been observed for other traffic patterns and paths of different length. We also note that, under the least-used policy, the blocking probability of calls using multiple hops increases significantly, and that the average blocking probability over all calls is higher than other policies. Therefore, we will not consider the least-used policy any further.

In Figure 6.10 we compare the most-used and first-fit policies for a 6-hop path with an inverted bowl traffic pattern. Again, as in Figure 6.7, we find that the two policies exhibit almost identical blocking performance, not only for the end-to-end call, but for all calls,

[1] Approximate analytical techniques based on overflow traffic have been developed for the first-fit policy in [15, 20]. However, these techniques are limited in that link blocking events are taken to be independent, an assumption that is not accurate over a wide range of traffic loads.

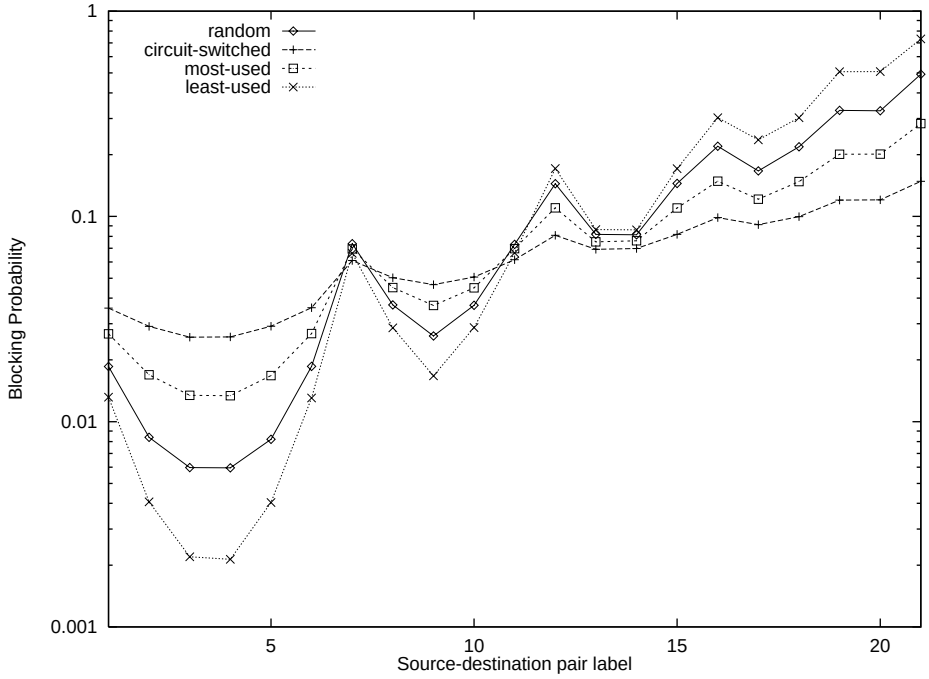


Figure 6.8: Policy comparison, 6-hop path, uniform traffic pattern

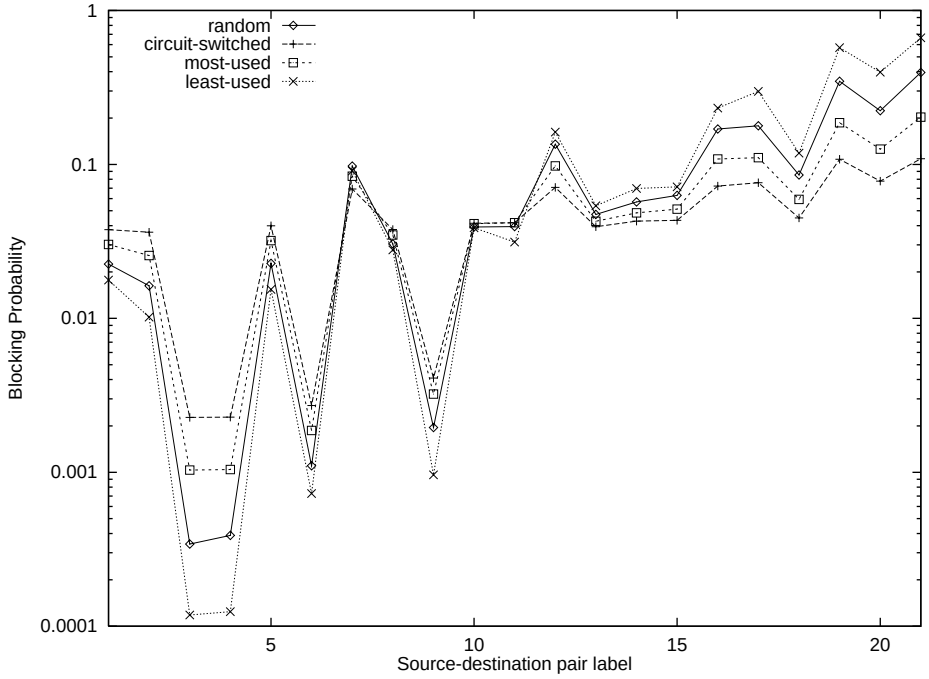


Figure 6.9: Policy comparison, 6-hop path, bowl traffic pattern

regardless of the number of hops used by the calls; very similar results have been obtained for all traffic patterns studied. Therefore, for the rest of the paper we will concentrate on the first-fit policy, since its implementation does not require that the network nodes maintain information about the global use of wavelengths. Specifically, our objective is two-fold. First, we will show that the values of the blocking probabilities obtained with this policy are bounded by the blocking probability values obtained by the random policy without converters and the circuit-switched case (equivalently, the random policy with converters at all nodes of a path). Second, we will demonstrate that, for calls traversing multiple hops, the gain (in terms of a reduction in blocking probability) obtained by employing the most-used over the random policy is roughly equivalent to using the random policy and deploying converters in the path or network.

In Figure 6.11 we compare the blocking probabilities of a 6-hop path obtained by the first-fit policy to three other systems: a path with the random policy and no converters, a path with the random policy and one converter at node 3, and a circuit-switched path (i.e., a path with the random policy and converters at all nodes). As we can see, the first-fit policy has an effect similar to that of using the random policy and employing a converter in the path. This is a general result that has been observed for a wide range of traffic loads, and will be discussed in more detail shortly.

Figures 6.12-6.16 present results for a 10-hop path and various traffic patterns. In Figures 6.12-6.14 we compare the first-fit policy to the random (no converters) and circuit-switched cases for the inverter bowl, oscillating, and ascending traffic patterns, respectively. Two interesting observations can be made from the three figures. First, the blocking probability values of the first-fit policy are always between the corresponding values of the random and circuit-switched cases. In other words, the blocking probability values under the random and circuit-switched cases provide lower and upper bounds for the blocking performance of the first-fit policy [2]. Second, it appears that the first-fit policy is quite effective in reducing the blocking probability of calls traveling over multiple hops (which are the ones experience the highest blocking probability under the random policy) close to the level of the circuit-switched case.

[2]Note that the random policy provides a *lower* bound for calls using a single hop, and an *upper* bound for calls traversing multiple hops; the reverse is true for the circuit-switched path.

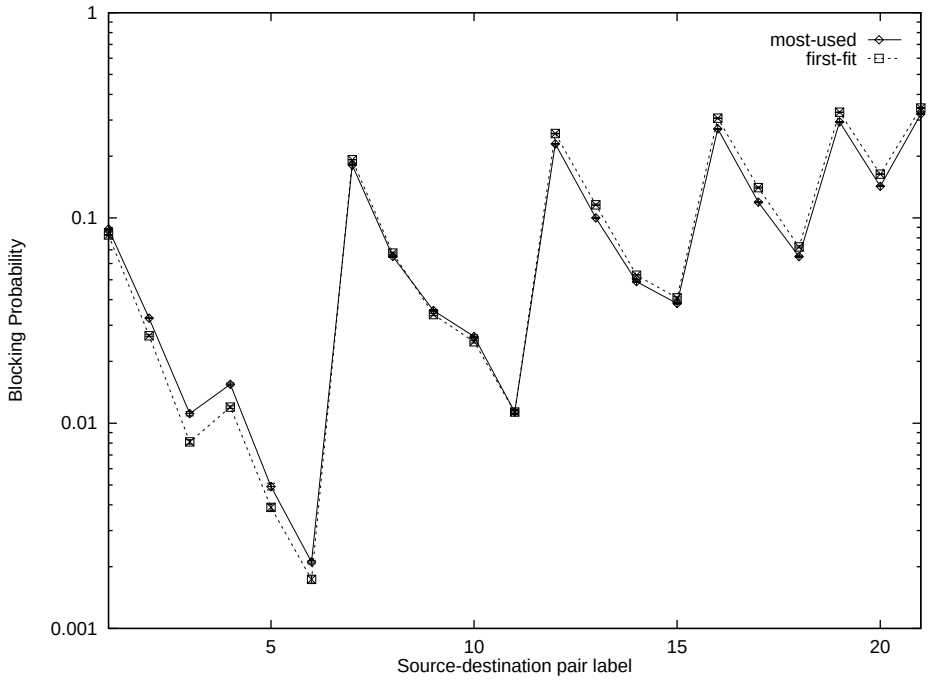


Figure 6.10: Most-used vs. first-fit allocation, 6-hop path, descending traffic pattern

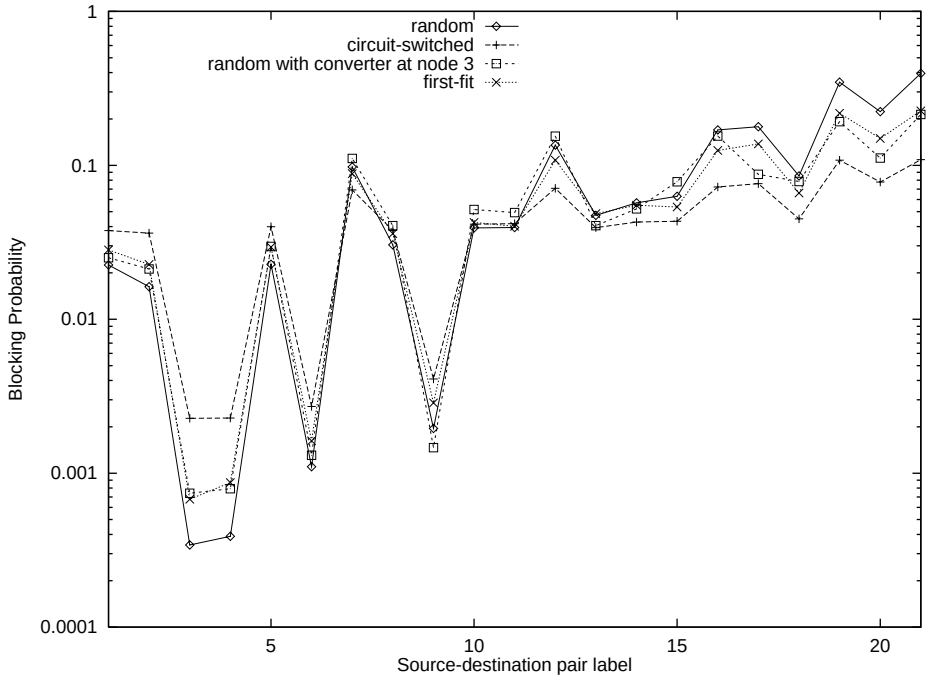


Figure 6.11: First-fit policy vs. random policy with converters, 6-hop path, bowl traffic pattern

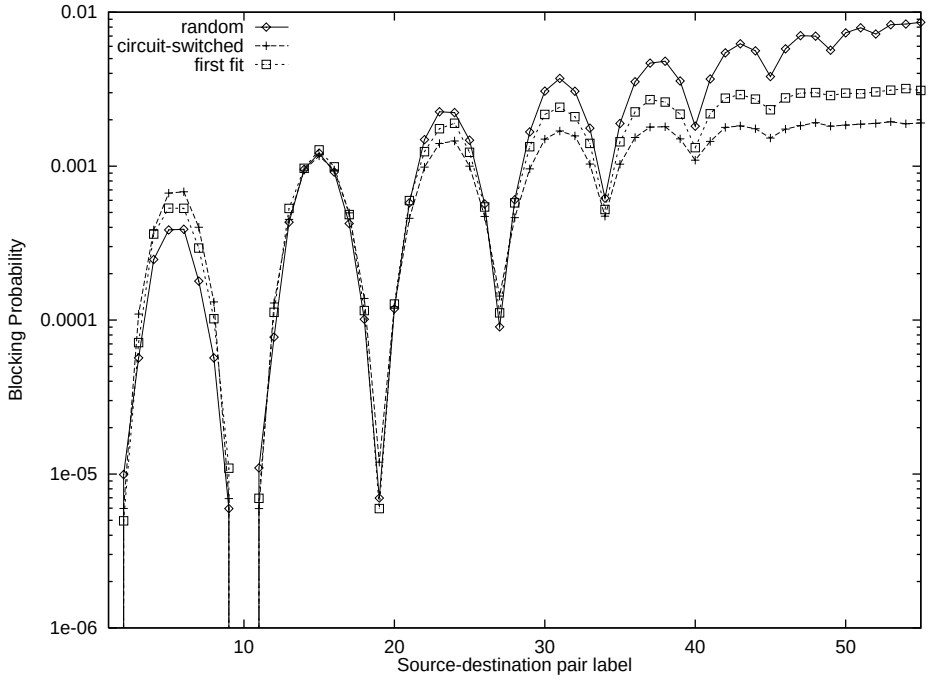


Figure 6.12: Policy comparison, 10-hop path, inverted bowl traffic pattern

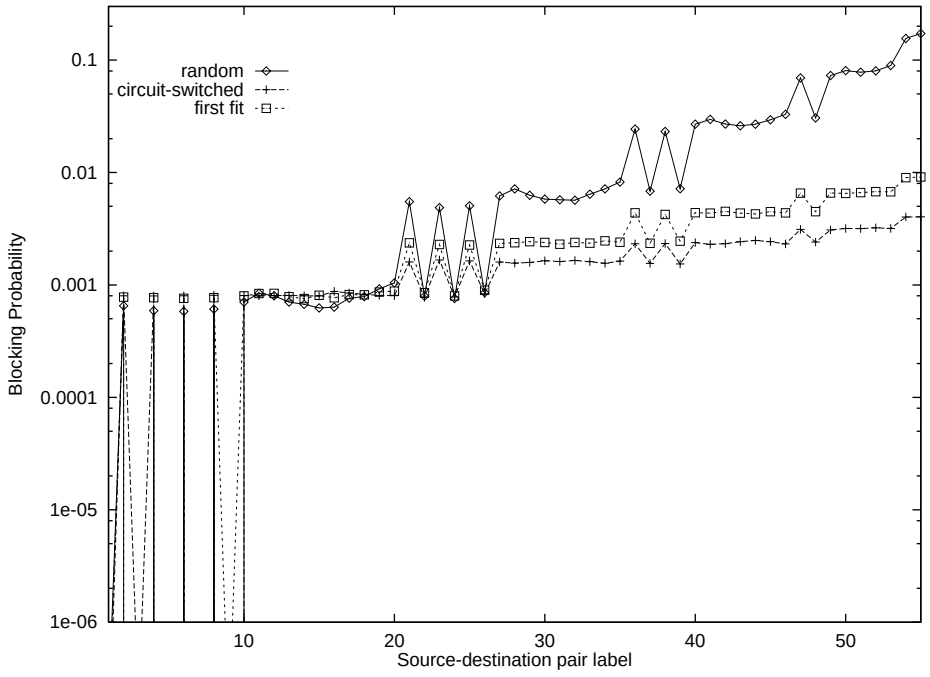


Figure 6.13: Policy comparison, 10-hop path, oscillating traffic pattern

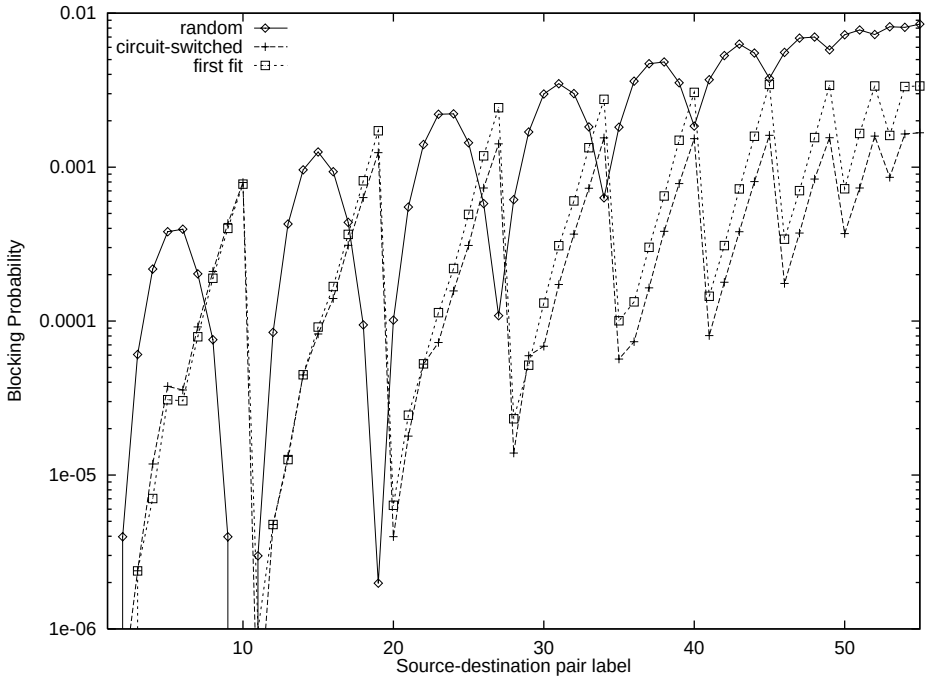


Figure 6.14: Policy comparison, 10-hop path, ascending traffic pattern

In Figure 6.15 we compare the first-fit to the most-used policy for the descending traffic pattern. As before, the blocking probability values of the two policies match for all types of calls. Finally, in Figure 6.16 we attempt to quantify the effect of the first-fit policy in terms of "number of converters." Specifically, we plot the blocking probability values for the first-fit policy as well as those of a random policy with either three or five converters. The converters are placed at nodes in a way that minimizes the blocking probability of calls traveling over all 10-hops, using the techniques developed in Chapter 4. Note that, by employing converters at some of the nodes, the blocking probability of calls traversing multiple hops improves, since converters reduce the requirement that the same wavelength be used on all hops of the path taken by the call. However, this improvement is at the expense of calls using a single hop, which now experience higher blocking probability. As we can see, the effect of using the first-fit policy in place of the random policy has an effect similar to employing converters.

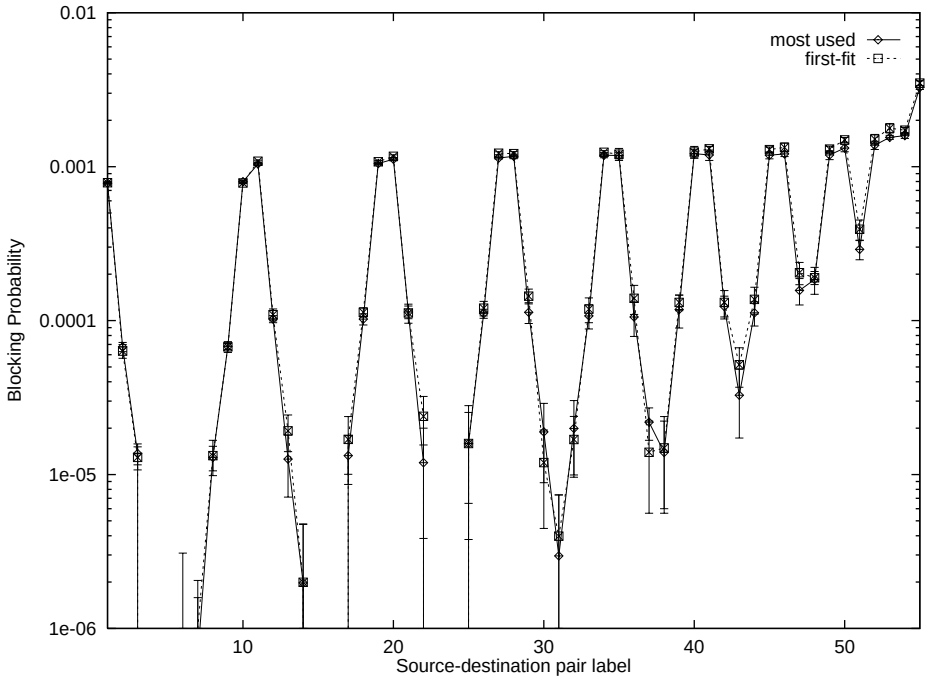


Figure 6.15: Most-used vs. first-fit allocation, 10-hop path, bowl traffic pattern

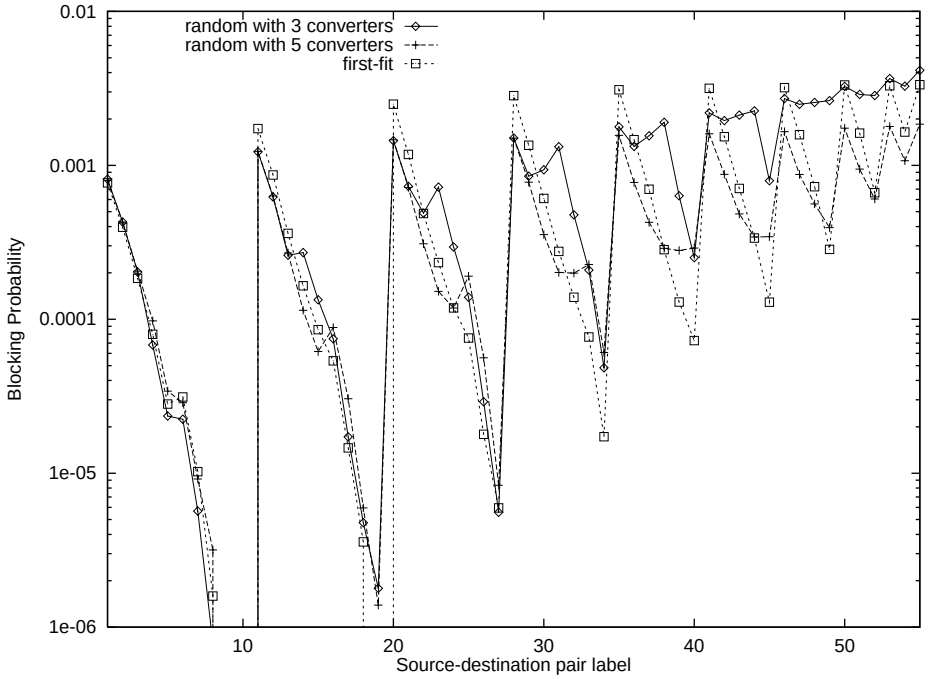


Figure 6.16: First-fit policy vs. random policy with converters, 10-hop path, descending traffic pattern

## 6.2 Mesh Wavelength Routing Networks

In this section we further investigate the effects of the most-used and first-fit policies on the call blocking probabilities compared to the effect of converters, by studying two network topologies: the regular $5 \times 5$ torus network in Figure 5.3, and the NSFNET irregular topology in Figure **??**.

### 6.2.1 The $5 \times 5$ Torus Network

Let us first consider the $5 \times 5$ torus network shown in Figure 5.3, with $W = 10$ wavelengths per link. Since there are 600 source-destination pairs in this network, it is impossible to present numerical results for all of them. We present, therefore, the call blocking probabilities for only 24 different source-destination pairs, namely, those with node 1 as the source. Because of the regular topology, the selected pairs are a representative sample of the various source-destination pairs. Similar to previous figures, the source-destination pairs have been labeled such that numbers 1 through 4 correspond to pairs for which a 1-hop path is used, numbers 5 to 12 correspond to pairs for which a 2-hop path is used, and so on.

In our study we have used two traffic patterns. The first pattern is listed in Equation 5.8. For the second pattern (which we will refer to as the random pattern), each arrival rate $\lambda_{sd}$ was selected from a uniform distribution in the range (0.1,0.4).

In Figure 6.17 we compare the blocking probabilities obtained through the most-used and first-fit policies for the pattern based on locality of traffic. From the figure, we observe that calls using a single hop (labels 1 to 4 in the figure) experience the lowest blocking probability, calls traveling over two hops have the next lowest blocking probability, and so on. (The fact that the blocking probability values are almost the same for all calls using the same number of hops is due to the symmetry of both the topology and the traffic pattern.) We also see that the two policies result in almost identical blocking probability values for all calls, further confirming our claim that the (simpler) first-fit policy can be used as a quite accurate approximation of the most-used policy. Similar results, not shown here, have been obtained for the random traffic pattern.

In Figures 6.18 and 6.19, we compare the first-fit policy to the random policy and the circuit-switched case, under the two traffic patterns. It is clear from both figures that the blocking probability values under the first-fit policy are between those of the other

two cases, a behavior which is consistent with our earlier results on single paths. However, there is also an important difference in the two figures. While the effect of the first-fit policy appears to have a significant effect for the traffic pattern based on locality (Figure 6.18), in that the blocking probability values for calls using multiple hops drops significantly from the corresponding values under the random policy, this effect is less pronounced in Figure 6.19 for the random traffic pattern. This difference can be explained by noting that the values of the blocking probability for calls 13 and higher in Figure 6.19 are more than 0.1, about an order of magnitude higher than the values for the corresponding calls in Figure 6.18. At such high values, not many wavelengths are available for these calls, thus, the actual wavelength allocation policy used will have little effect on the blocking probability. It is at these high blocking probability values that having converters at all nodes (circuit-switched case) will help. However, it is unlikely that realistic networks will be designed to operate in this region.

In Figures 6.20 and 6.21 we compare the first-fit policy to the random policy with 4, 8, and 12 converters employed in the torus network (note that these values correspond to 16%, 32% and 48%, respectively, of the network nodes having converters). As we can see, using the first-fit policy is roughly equivalent to employing a significant number of converters in the network. Another important observation form these figures is that employing converters has a rather uneven effect on the blocking probabilities of the various calls. Specifically, calls whose path includes a converter experience a rather dramatic drop in blocking probability, but the effect of converters on the blocking probability of other calls is considerably smaller. The first-fit policy, on the other hand, evenly decreases the blocking probability for calls traveling over long hops. This result is evident not only in Figures 6.20 and 6.21 but in the results for the 6-hop and 10-hop paths, as well as for the NSFNET discussed next.

### 6.2.2 The NSFNET Topology

We have also considered a realistic example of a backbone network, namely, the NSFNET irregular topology shown in Figure 5.8. Since we will be using traffic data reported in [6], following that study, we have also augmented the 14-node NSFNET topology by adding two fictitious nodes, nodes 1 and 16 in Figure 5.8, to capture the effect of NSFNET's connections to Canada's communication network, CA*net. The resulting topology consists

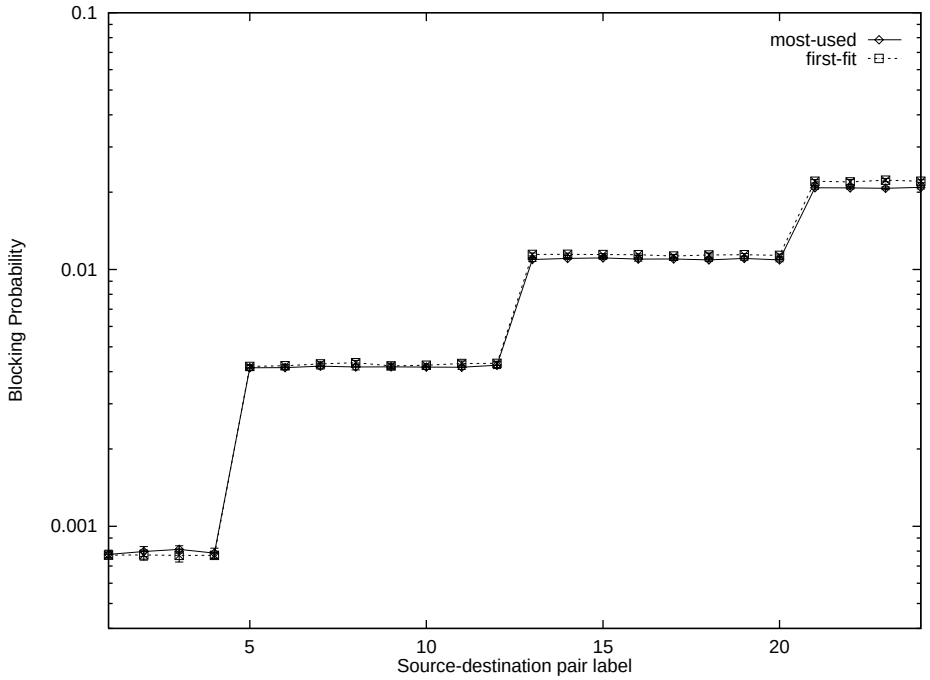


Figure 6.17: Most-used vs. first-fit allocation, $5 \times 5$ torus network, traffic pattern based on locality

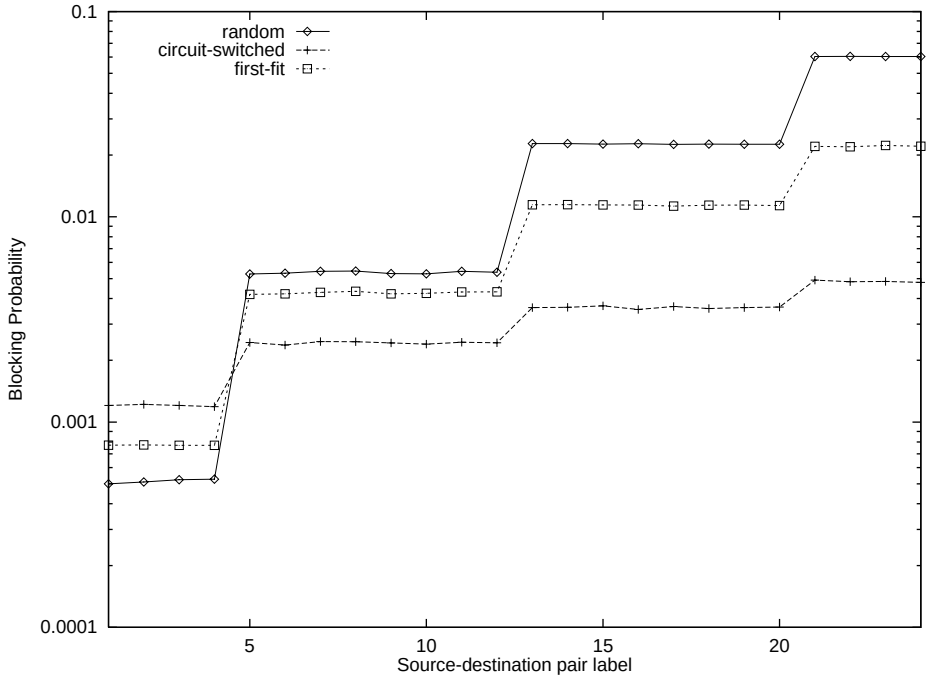


Figure 6.18: Policy comparison, $5 \times 5$ torus network, traffic pattern based on locality

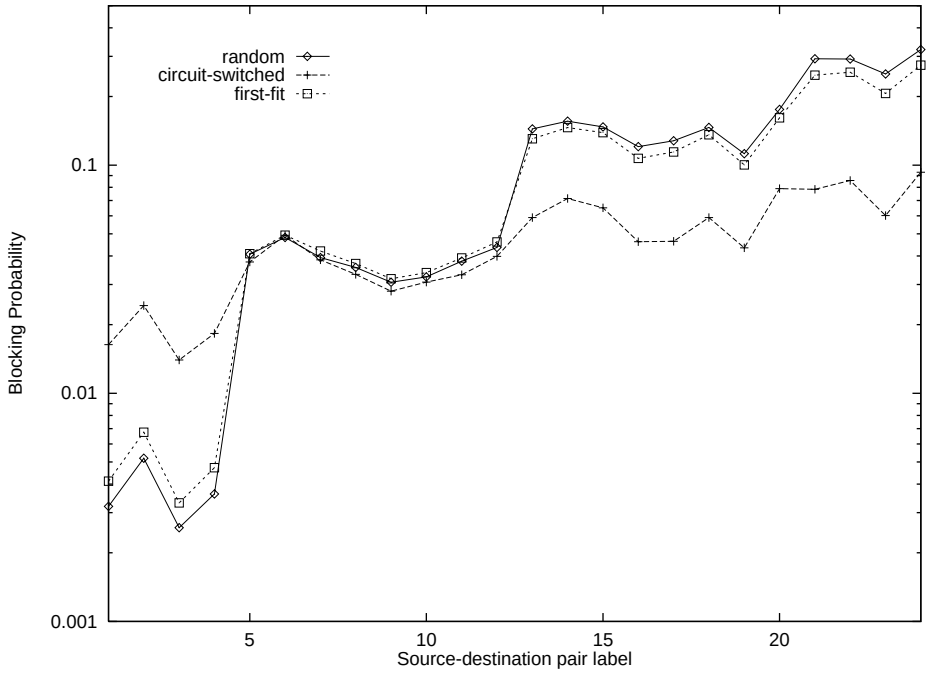


Figure 6.19: Policy comparison, $5 \times 5$ torus network, random traffic pattern

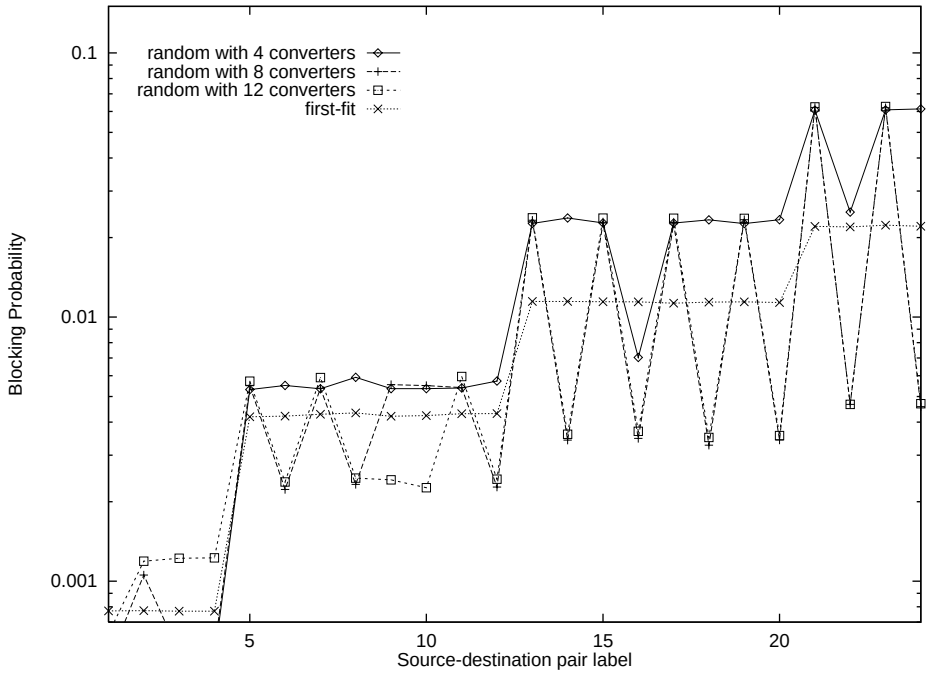


Figure 6.20: First-fit policy vs. random policy with converters, $5 \times 5$ torus network, pattern based on locality

of 16 nodes and a total of 240 source-destination pairs. As in the previous subsection, we only present detailed results for the call blocking probabilities of only a small number of calls, those 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).) There are 28 source-destination pairs in this path, and in Figures 6.22 to 6.26 they have been labeled such that numbers 1 to 7 refer to pairs for which the shortest path is one hop long, numbers 8 to 15 correspond to pairs with a shortest path of two hops, etc.

We have used the same traffic patterns as used in Chapter 5 with the NSFNET topology. The first pattern is listed in Equation 5.9. The second traffic pattern used the real traffic data in [6, Figure 6]. All values used here are exactly half of those used in Chapter 5.

Our results are presented in Figures 6.22-6.26. Figure 6.22 compares the first-fit to the most-used policies. Figures 6.23 and 6.24 demonstrate that the random and circuit-switched cases provide upper and lower bounds on the performance of the first-fit policy, while Figures 6.25 and 6.26 attempt to quantify the effect of the first-fit policy in terms of number of converters. The converters were placed in the network using the optimization techniques in Chapter 5. The overall behavior of the graphs shown in these figures is very similar to that discussed earlier for the torus network and the single path cases, indicating that our observations and conclusions are valid for a wide range of network topologies and traffic patterns.

## 6.3 Concluding Remarks

We have studied the blocking performance of various wavelength allocation policies for various single path and network topologies and under various traffic patterns. Our conclusions can be summarized as follows:

- We have shown that the most-used and first-fit policies have very similar performance for all calls in a network, regardless of the number of hops used by the calls. Specifically, the two policies tend to favor calls using multiple paths at the expense of calls using a single path. This is a desirable feature, since calls traversing multiple paths experience the highest blocking probability. However, the most-used policy requires

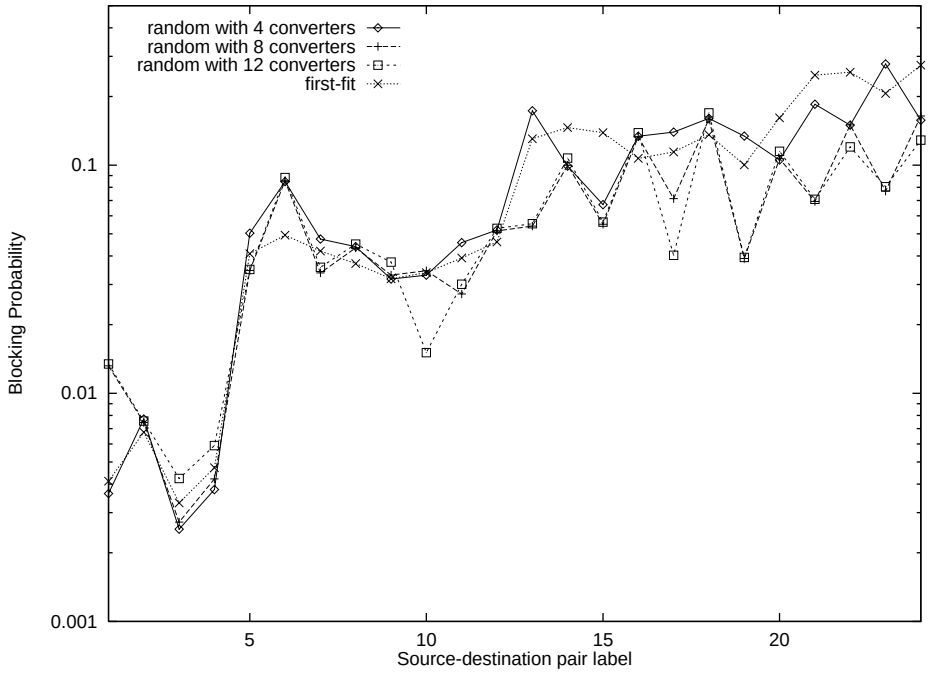


Figure 6.21: First-fit policy vs. random policy with converters, $5 \times 5$ torus network, random traffic pattern

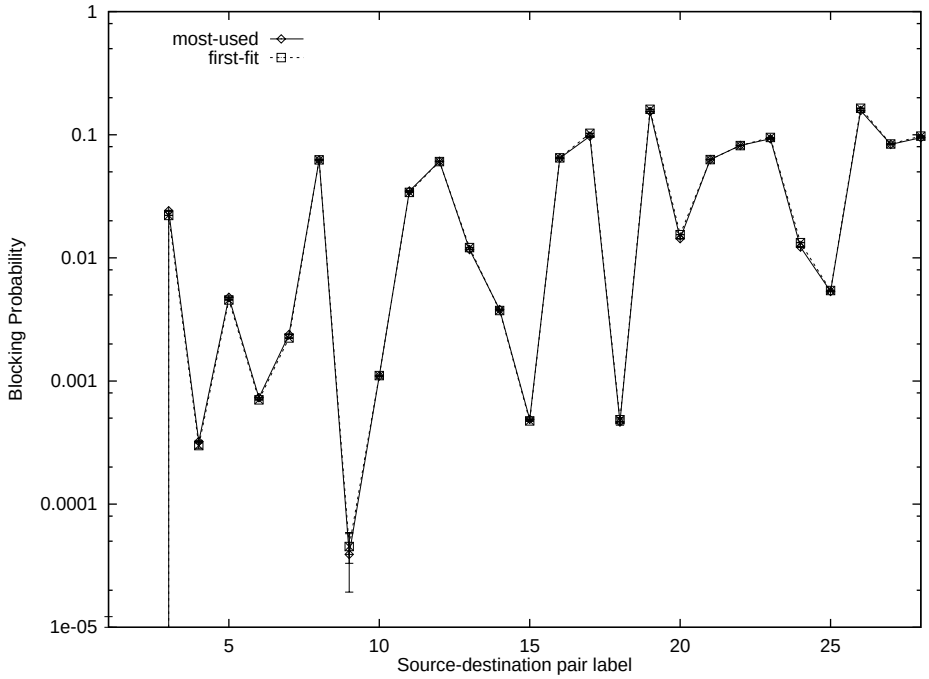


Figure 6.22: Most-used vs. first-fit allocation, NSFNET, pattern based on actual traffic

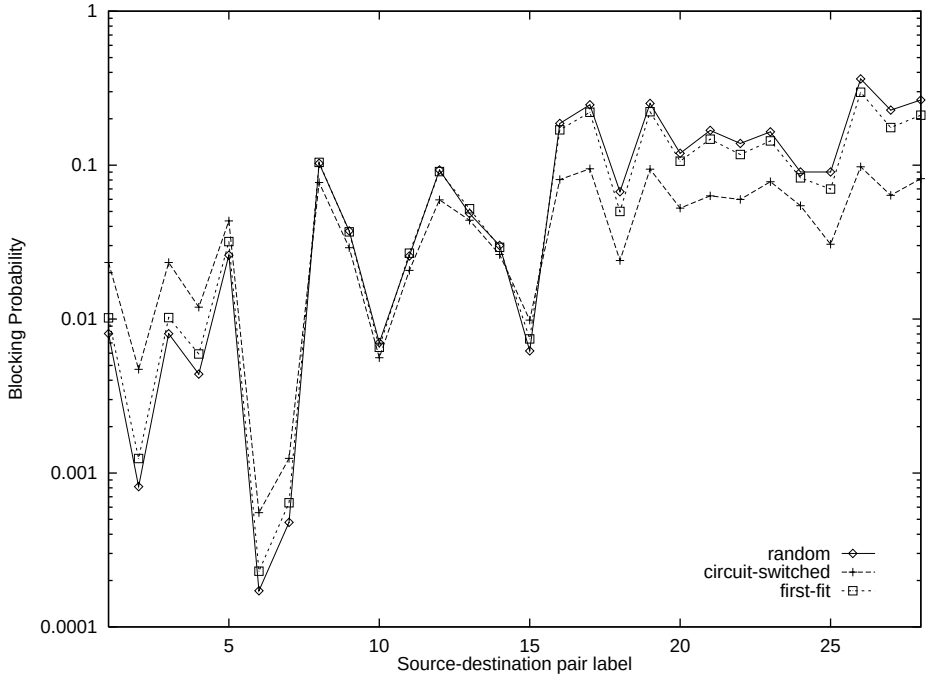


Figure 6.23: Policy comparison, NSFNET, traffic pattern based on locality

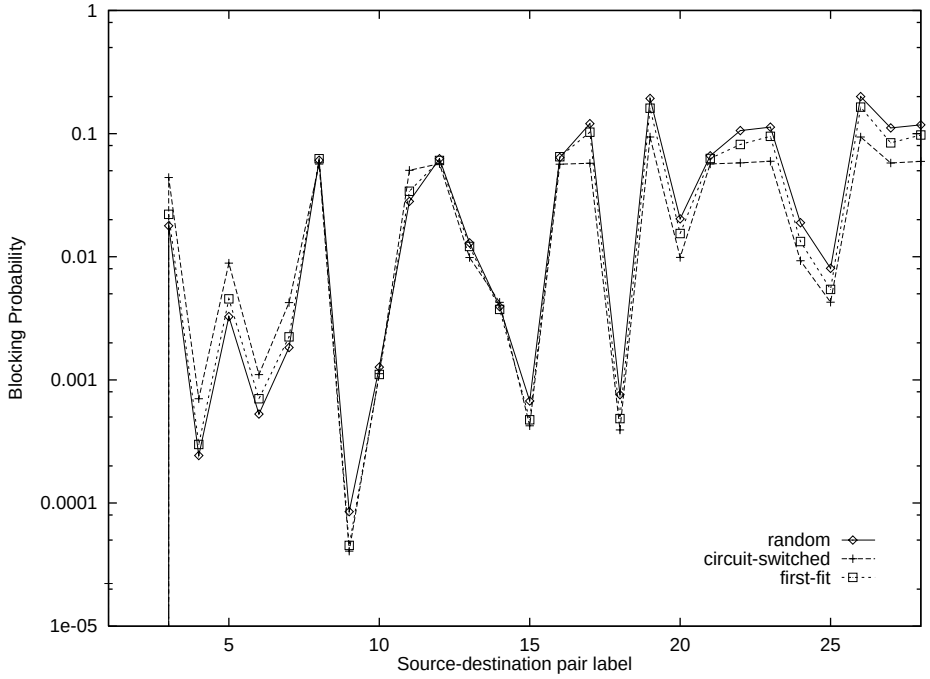


Figure 6.24: Policy comparison, NSFNET, pattern based on actual traffic

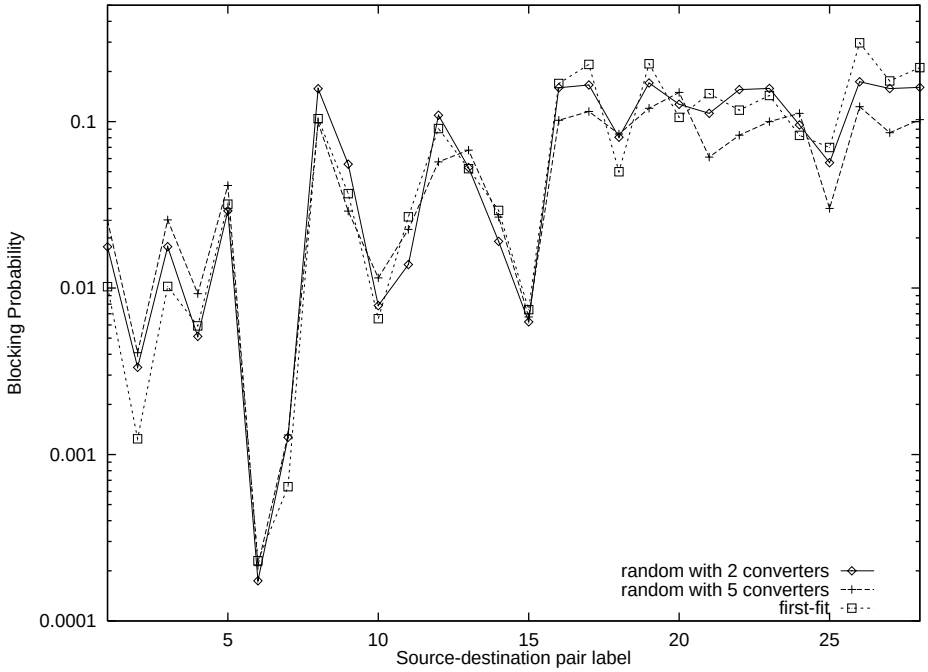


Figure 6.25: First-fit policy vs. random policy with converters, NSFNET, traffic pattern based on locality

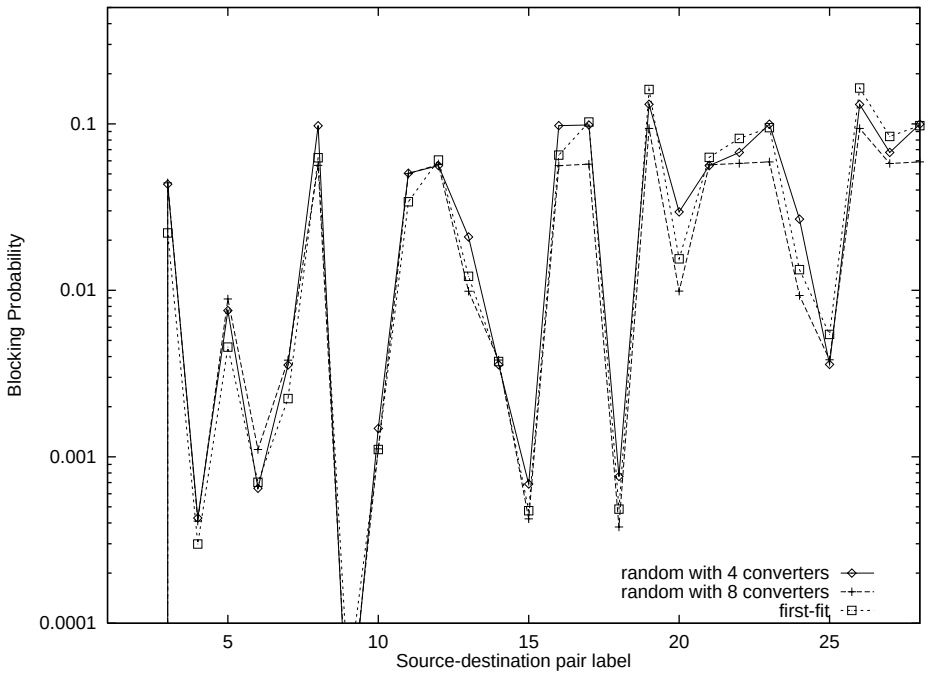


Figure 6.26: First-fit policy vs. random policy with converters, NSFNET, random traffic pattern

that the network nodes exchange information about the network-wide usage of wavelengths, while the first-fit policy only relies on a fixed ordering of wavelengths, making it significantly easier to implement.

- We have also demonstrated that the random policy and the circuit-switched case, for which analytical (exact or approximate) solutions exist for systems of large size, provide lower and upper bounds on the call blocking probability under the first-fit (or most-used) policy. Specifically, for calls using one or two hops, the random policy provides a lower bound and the circuit-switched case provides an upper bound, while for calls using longer paths the bounds are reversed.

- We have presented results which indicate that the effect of using the first-fit policy is "equivalent" to using the random policy but employing a number of converters (between 15% to 50% of the number of nodes) in the network. More importantly, in most cases, introducing the first-fit policy results in a decrease in the blocking probability of calls traveling over multiple hops to a level very close to the blocking probability experienced under the circuit-switched case. Note that, in terms of implementation, there is no significant difference between the first-fit and random policies. Consequently, the gains obtained by employing specialized (and expensive) hardware can be realized by making more intelligent choices in software.

- It also appears that the benefits of the first-fit policy diminish at high loads (blocking probability values of 0.1 or more). It is a these situations that employing converters would benefit calls traversing a large number of hops. However, the number of converters to be employed in this case must be very large, close to the number of nodes in the network, and even if all nodes contain converters the blocking probability will remain at (reduced but) high levels. Since it is unlikely that future wavelength routing networks will be designed to operate at such high call blocking probability values, attempting to reduce these values may not be of practical importance.

Overall, our results appear to contradict previous studies which have indicated that "sparse" wavelength conversion capabilities (i.e., selective placement of converters at a subset of network nodes) will be beneficial to wavelength routing networks. Those studies measured the improvement obtained by employing converters over the random wavelength allocation policy only, while we have shown that an equivalent improvement can be achieved

merely by using appropriate allocation policies such as first-fit or most-used. While further investigation of the value of converters is warranted, our results can have a major impact on the direction of research and development in the area of wavelength converters.

# Chapter 7

# Conclusions and Future Work

## 7.1 Conclusions and Future Work

The analytical framework presented in this thesis can be used evaluate blocking probabilities of wavelength routing networks accurately and efficiently. We first presented an exact Markov process for a single path, and constructed an approximate Markov process which has a closed-form solution. We then developed and evaluated iterative decomposition algorithms for long path and mesh networks. Moreover, our algorithms can calculate the blocking probabilities of networks with fixed or alternate routing network with any number of wavelength converters. We show that the blocking probabilities for the random wavelength allocation and the circuit-switched case provide upper and lower bounds on the blocking probabilities for first-fit wavelength allocation policies that are most likely to be used in practice. Compared with simulation, our analytical results are accurate and the computation time is much improved. The average relative error is around 15% for moderate blocking probabilities under fixed routing. The computation time is improved from more than ten hours for simulation to less than one hour for analytical model.

Several interesting issues may be investigated in the future:

- In our models, it was assumed that the arrival rate was Poisson. It may be possible to extend the analysis to non-Poisson traffic. This will give a more accurate model for the overflow traffic when modeling alternate routing. However, the improvement in accuracy must be weighted against the complexity of the non-Poisson model.

- Dynamic routing is widely adopted in telecommunication networks today, and is likely

to be adopted in optical network too. Extending the static routing model to dynamic routing is an interesting research topic.

- Optimal wavelength converter placement for arbitrary network topologies should be investigated. A quantitative comparison of the performance gains by introducing wavelength converters against using the first-fit wavelength allocation policy should also carry out.

- Multiservice loss models have been studied extensively in circuit-switched networks. This work should be extended to optical networks.

- Finally, with the increase of multicast applications on the Internet, multicast protocols and routing algorithms in optical networks should be studied and developed.

# Bibliography

[1] D. Banerjee and B. Mukherjee. A practical approach to routing and wavelength assignment in large wavelength-routed optical networks. *IEEE Journal Selected Areas in Communications*, 14:903–908, June 1996.

[2] 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.

[3] 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.

[4] A. Birman and A. Kershenbaum. Routing and wavelength assignment methods in single-hop all-optical networks with blocking. In *Proceedings of INFOCOM'95, Boston, MA*, pages 431–438. IEEE, April 1995.

[5] 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.

[6] 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.

[7] O. Gerstel and S. Kutten. Dynamic wavelength allocation in wdm ring networks. Technical Report RC 20462, IBM Research, May 1996.

[8] O. Gerstel, Rajiv Ramaswami, and G. Sasaki. Fault-tolerant multiwavelength optical networks. In *Proceedings of INFOCOM'97*. IEEE, To appear 1997.

[9] O. Gerstel, G. Sasaki, and R. Ramaswami. Dyanmic channel assignment for wdm optical networks with little or no wavelength conversion. In *Proceedings of 1996 Allerton Conference, Monticello, IL*, to appear 1996.

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

[11] P. E. Green. Optical networking update. *IEEE Journal Selected Areas in Communications*, 14(5):764–779, June 1996.

[12] 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.

[13] H. Harai, M. Murata, and H. Miyahara. Performance analysis of wavelength assignment policies in all-optical networks with limited-range wavelength conversion. In *Proceedings of INFOCOM'98*, pages 1051–1060. IEEE, 1998.

[14] G. Jeong and E. Ayanoglu. Comparison of wavelength-interchanging and wavelength-selective cross-connects in multiwavelength all-optical networks. In *Proceedings of INFOCOM'96, San Francisco, CA*, pages 156–163. IEEE, March 1996.

[15] 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.

[16] F. P. Kelly. *Reversibility and Stochastic Networks*. John Wiley & Sons, New York, 1979.

[17] L. Kleinrock. *Queueing Systems, Volume 1: Theory*. John Wiley & Sons, New York, 1975.

[18] 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.

[19] K.-C. Lee and V.K. Li. A wavelength-convertible optical network. *Journal of Lightwave Technology*, 11:962–970, May/June 1993.

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

[21] B. Mukherjee. *Optical Communication Networking*. McGraw-Hill, 1997.

[22] H. Perros. *Queueing Networks with Blocking: Exact and Approximate Solutions*. Oxford University Press, 1994.

[23] R. Ramaswami and G. H. Sasaki. Multiwavelength optical networks with limited wavelength conversion. In *Proceedings of INFOCOM '97*, pages 490–498. IEEE, April 1998.

[24] R. Ramaswami and K. N. Sivarajan. Routing and wavelength assignment in all-optical networks. *IEEE/ACM Transaction on Networking*, 3:489–500, October 1995.

[25] R. Ramaswami and K. N. Sivarajan. *Optical Networks*. Morgan Kaufmann Publishers, San Francisco, California, 1998.

[26] J. P. Ryan. Wdm: North american deployment trends. *IEEE Communications Magazine*, pages 40–42, February 1998.

[27] V. Sharma and E.A. Varvarigos. Limited wavelength translation in all-optical wdm mesh networks. In *Proceedings of INFOCOM'98*, pages 893–901. IEEE, 1998.

[28] 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.

[29] 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.

[30] S. Subramaniam and R. Barry. Wavelength assignment in fixed routing wdm networks. In *Proceedings of IEEE ICC, Montreal, P.Q., Canada*, pages 406–415. IEEE, November 1997.

[31] 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.

[32] J. Yates, J. Lacey, D. Everitt, and M. Summerfield. Limited-range wvelength translation in all-optical networks. In *Proceedings of INFOCOM'96, San Francisco, CA*, pages 954–961. IEEE, March 1996.

[33] Z. Zhang and A. Acampora. A heuristic wavelength assignment algorithm for multihop wdm networks with wavelength routing and wavelength re-use. *IEEE/ACM Transaction on Networking*, 3:281–288, June 1995.

# Appendix A

## A.1 Proof of Lemma 3.1

**Proof.** We will show that the state descriptor in (4.2) provides all the information necessary to predict the future evolution of a $k$-hop system. Let us first consider the arrival of a call that uses hops $i$ through $j \geq i$, inclusive, and which finds the system in state $\underline{n}$. If $i < j$, the number of wavelengths that are free on hops $i$ through $j$ is given by parameter $f_{ij}$. If $i = j$, the number $f_{ii}$ of wavelengths that are free on hop $i$, is not included in the state description of MC $\mathcal{M}_k$, but it can be easily derived as

$$f_{ii} \;=\; W \;-\; \sum_{1 \leq l \leq i \leq m \leq k} n_{lm} \tag{A.1}$$

If $f_{ij} = 0$, the call is blocked, and the system remains at state $\underline{n}$. If $f_{ij} > 0$, however, the call will be randomly assigned one of the $f_{ij}$ free wavelengths, and the system will make a transition to state $\underline{n'}$ such that:

$$n'_{lm} \;=\; \begin{cases} n_{lm} + 1, & l = 1 \text{ and } j = m \\ n_{lm}, & \text{otherwise} \end{cases} \quad , \quad f'_{ij} \;=\; f_{ij} - 1 \tag{A.2}$$

In addition, the values of some other parameters $f_{lm}$ may be affected, depending on the particular wavelength assigned to the call. For instance, let us assume that $i = j$, and consider a parameter $f_{lm}, l \leq i \leq m$, in state $\underline{n}$ that has a non-zero value. Obviously, $f_{lm} \leq f_{ii}$, since the set of free wavelengths on hops $l$ through $m$ is a subset of the set of free wavelengths on hop $i$ alone. Because of the random wavelength assignment policy, with probability $\frac{f_{lm}}{f_{ii}}$ the new state $\underline{n'}$ will have $f'_{lm} = f_{lm} - 1$, while with probability $1 - \frac{f_{lm}}{f_{ii}}$ it will have $f'_{lm} = f_{lm}$. Similarly when $i < j$. Thus, upon the arrival of a call request, the information in the descriptor of the current state $\underline{n}$ is sufficient to determine all possible new states $\underline{n'}$, as well as the corresponding transition rates.

Let us now consider the completion of a call that uses hops $i$ through $j \geq i$, inclusive, when the system is in state $\underline{n}$ with $n_{ij} > 0$. The system will make a transition to a new state $\underline{n'}$ such that:

$$n'_{lm} = \begin{cases} n_{lm} - 1, & l = 1 \text{ and } j = m \\ n_{lm}, & \text{otherwise} \end{cases} \quad , \quad f'_{ij} = f_{ij} + 1 \qquad \text{(A.3)}$$

As with a call arrival, the values of some other parameters $f_{lm}$ may be affected. For simplicity, let us again assume that $i = j$; similar arguments apply when $i < j$. Because of the memoryless property, the completed call is equally likely to be any of the $n_{ii}$ active calls on hop $i$. Consider parameter $f_{lm}, l \leq i \leq m$, and parameters $f_{l,i-1}$ and $f_{i+1,m}$. Obviously, $f_{lm} \leq f_{l,i-1}$ and $f_{lm} \leq f_{i+1,m}$. If $f_{lm} = f_{l,i-1}$ or $f_{lm} = f_{i+1,m}$, the transition due to the completion of the call will not affect the value of $f_{lm}$. But if $f_{lm} < f_{l,i-1}$ and $f_{lm} < f_{i+1,m}$, if the newly freed wavelength is the same as one of the free wavelengths on hops $l$ through $i-1$ *and* on hops $i+1$ through $m$, then the new state will have $f'_{lm} = f_{lm} + 1$. The probability of the last event can be computed as follows. Of the $f_{l,i-1}$ wavelengths, $f_{l,i}$ are also free on hop $i$, so $f_{l,i-1} - f_{l,i}$ are not free on hop $i$. Similarly, of the $f_{i+1,m}$ wavelengths, $f_{i+1,m} - fi, m$ are not free on hop $i$. The probability that the newly freed wavelength is one of $f_{l,i-1} - f_{l,i}$ is simply $\frac{f_{l,i-1} - f_{l,i}}{n_{ii}}$, and the probability that it is one of $f_{i+1,m} - f_{i,m}$ is $\frac{f_{i+1,m} - f_{i,m}}{n_{ii}}$. Therefore, the probability that the newly freed wavelength will increase the number $f_{lm}$ of wavelengths that are free on hops $l$ is $\frac{f_{l,i-1} - f_{l,i}}{n_{ii}} \frac{f_{i+1,m} - f_{i,m}}{n_{ii}}$. Again, we have sufficient information to determine the set of possible new states $\underline{n'}$ and the transition rates upon the termination of an active call. □