

Connection management for survivable wavelength-routed WDM mesh networks

Hui Zang

Sprint Advanced Technology Laboratories
Burlingame, California USA

Biswanath Mukherjee

Department of Computer Science
University of California, Davis
Davis, California USA

ABSTRACT

In a wavelength-routed optical network, lightpaths are set up between node pairs to satisfy each connection request. Under a dynamic traffic pattern, a lightpath is taken down after a period of time, called the connection-holding time. Lightpaths are high-capacity all-optical channels. Hence, when there is a link failure, e.g., an optical fiber cable cut, the data loss may be very large if the traffic is not re-routed quickly. In a mesh network, end-to-end path-protection schemes can be employed to achieve efficient resource utilization. We develop an on-line network-control mechanism to manage the connections in such a network using path-protection schemes. The goal is to protect each connection from single-link failures, as well as to minimize the overall blocking probability and end-to-end delays. We compare dedicated-path protection and shared-path protection over several performance metrics. Simulation results show that, with shared-path protection, we can achieve lower call-blocking probability with fast fault-recovery.

1 Introduction

With the rapid growth in popularity of the Internet and the WWW, the data network is evolving to include more and more bandwidth-intensive network applications. Fiber optics and wavelength-division multiplexing (WDM) are being researched as well as commercially deployed as technologies that can satisfy the bandwidth requirements of the Internet today and the foreseeable future. In a wavelength-routed WDM network, end users may communicate with one another via all-optical WDM channels which may span multiple fiber links. Such all-optical channels are referred to as *lightpaths*¹ [1, 2]. Without wavelength conversion, a lightpath must occupy the same wavelength on all the links it traverses. This is called the *wavelength-continuity constraint*, which is assumed in this study.

**This work has been supported in part by National Science Foundation (NSF) Grant Nos. NCR-95-08239 and ANI-98-05285; ST Microelectronics; and State of California UC MICRO Grant No. 00-069.*

¹*A lightpath can be unidirectional or bidirectional; in this study, all lightpaths are assumed to be unidirectional.*

Once a lightpath is set up, a node or a link failure may lead to the failure of all the lightpaths that traverse the failed node or link. Note that transmitter/receiver failures can also occur. In this study, we develop and analyze survivability approaches to combat link failures as well as transmitter/receiver failures.

In an optical network, the high capacity of a link has the problem that a link failure can potentially lead to the loss of a large amount of data. So, we need to develop appropriate protection and restoration schemes which minimize the data loss when a link failure occurs. Upper layers of protocols (such as ATM, IP, and MPLS) have their own procedures to recover from link failures [3, 4, 5]. However, the recovery time for upper layers is significantly large (in the order of seconds), whereas we prefer that the fault-recovery times at the optical layer should be on the order of milliseconds in order to minimize data losses. Furthermore, it is beneficial to consider fault-recovery mechanisms in the optical layer for the following reasons [6]: (a) the optical layer can efficiently multiplex protection resources (such as spare wavelengths and fibers) among several higher-layer network applications, and (b) survivability at the optical layer provides protection to higher-layer protocols that may not have built-in fault recovery.

Essentially, there are two types of fault-recovery mechanisms [6, 7, 8]. If backup resources (routes and wavelengths) are pre-computed and reserved in advance, we call it a *protection* scheme [9, 10]. Otherwise, when a failure occurs, if another route and a free wavelength have to be discovered dynamically for each interrupted connection, we call it a *restoration* scheme [11, 12]. A restoration scheme is usually more resource-efficient [12], while a protection scheme has a faster recovery time and provides guaranteed recovery ability. We consider protection schemes in this study. Restoration schemes are of interest for future research. From the network-topology perspective, protection schemes can be classified as ring protection and mesh protection. Ring-protection schemes include Automatic Protection Switching (APS) and Self-Healing Rings (SHR) [2]. Both ring protection and mesh protection can be further divided into two groups: path protection and link protection. In path protection, the traffic is rerouted through a link-disjoint backup route once a link failure occurs on its working path. In link protection, the traffic is rerouted only around the failed link. Path protection usually has less resource requirement [9] and lower end-to-end propagation delay for the recovered route. In this study, we consider path protection in networks with mesh topology.

In path protection, for each lightpath that is set up, there are two link (and node) disjoint paths: a *primary* path and a *backup* path. The lightpath is set up on the primary path. In case of a link failure on the primary path, the lightpath is switched to the pre-reserved or pre-set-up backup path. The primary and the backup paths

are link-disjoint, while the backup paths of different connections may or may not share common wavelengths on common links. If we do not allow sharing among backup paths, then we have a *dedicated-path protection* scheme. The switches on backup paths can be configured at the beginning, i.e., when the lightpath is set up on the primary path. Then, no switch configuration is necessary when the failure occurs. This type of recovery can be very fast but the resources are not utilized very efficiently. There are two types of dedicated-protection: 1 + 1 protection and 1 : 1 protection.

- In 1 + 1 protection, traffic is transmitted on both paths from the source to the destination. The destination receives data from the primary path first. If there is a failure on the primary path, the destination switches over to the backup path and continues to receive data. In order to avoid data loss, the source node should delay transmitting data on the backup path for some amount of time ε , depending on the propagation delay difference between the primary path and the backup path, as well as the failure-detection time, i.e., if the k th bit of data reaches the destination at time t_1 on the primary path, the same k th bit should reach the destination at time $t_2 \geq t_1 + \varepsilon$ on the backup path. If the destination receives the $(k-1)$ th bit, detects there is a failure, and switches to the backup path, it should not miss the k th bit.
- In 1 : 1 protection, data is normally not transmitted on the backup path. Thus, we can use the backup path to carry some low-priority preemptable traffic. If there is a failure on the primary path, the source node is notified (by some protocol) and it switches over to transmit on the backup path. So, some data may be lost in the network and the source must be able to retransmit those data.

If sharing among backup paths is allowed as long as they satisfy certain constraints, the switches on the backup paths cannot be configured until the failure occurs. The recovery time in this scheme is longer, but the overall resource utilization is much better than the dedicated-path protection. Of course, more signaling is required to recover from the failure. We call this scheme the *shared-path protection* (or M : N) scheme. In this study, we consider both dedicated-path protection and shared-path protection. There have been some studies applying ring-protection schemes into a mesh-topology network. One such approach is to map a planar meshed graph into directed cycles and each directed link is protected by a directed cycle [13]. So, basically, this approach is a ring-based, link-protection scheme. Another approach is also ring-based link protection but works differently [10]. In [13], each fiber is unidirectional and is part of only one ring. In [10], rings can share fibers. A “ring cover” is first decided for the meshed network and shared-link protection is used within rings whereas protection wavelengths are not shared on different rings. This way, the switches

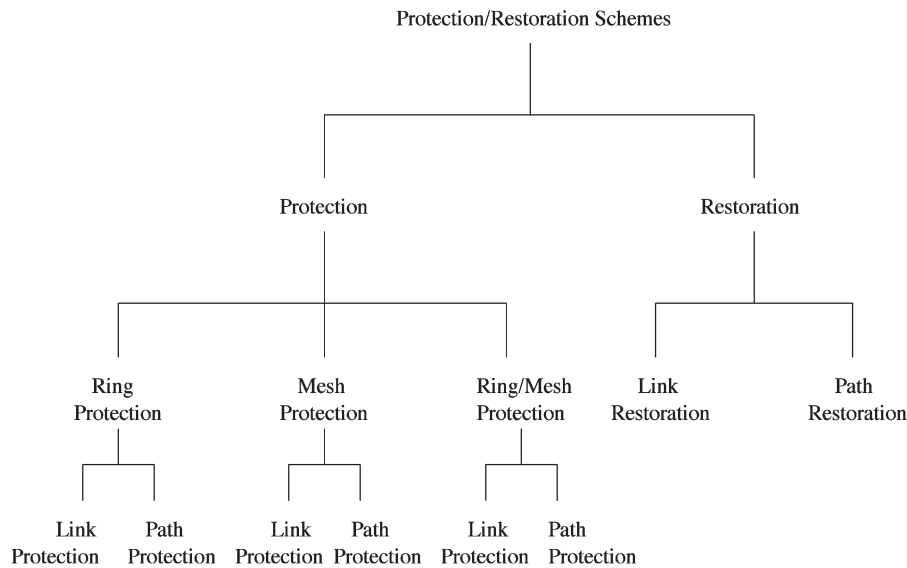


Figure 1: Different protection and restoration schemes.

can be pre-configured and the approach in [10] encourages a certain degree of sharing among protection wavelengths.

Figure 1 summarizes the classification of protection and restoration schemes.

In a wavelength-routed network, the traffic can be either static or dynamic. Under a static traffic pattern, the connection requests are available all at once. We call this the *static lightpath establishment* (SLE) problem which was studied in [9]. This problem can be solved by *Integer Linear Programming* (ILP). The constraints of this problem are the number of wavelengths on each link, the number of transmitters and receivers at each node, and the wavelength-continuity constraint (if no wavelength conversion is used). The objective is to minimize the total number of wavelengths used on all the links in the network, which is denoted by a quantity called *wavelength-mileage* in [10]. An alternative objective is to maximize the carried load, i.e., block the fewest number of connection requests.

Under dynamic traffic, connection requests come in one at a time and each connection exists for only a finite duration, called the *connection-holding time*. Given a fixed number of wavelengths on each fiber link, and a fixed number of transmitters and receivers at each node, our objective is to minimize the overall call-blocking probability. Also, we would like to achieve small end-to-end propagation delays of the connections which are set up. A *control and management protocol* is required to set up and take down lightpaths. This protocol must have the following capabilities:

- *Routing and Wavelength-Assignment* (RWA): upon the arrival of a connection request, the protocol must select two link-disjoint routes from the source to the destina-

tion,² and assign a wavelength to each route; if this process is not successful, the connection request is blocked;

- *Signaling*: after the routing and wavelength assignment process is completed, the protocol signals the appropriate nodes to reserve the wavelength on requested links and/or configure their switches;
- *Fault detection*: if a link failure occurs, the end nodes of the failed link (which is two unidirectional fibers, going on opposite directions) must be able to detect the failure; and those which detect the failure must notify the end nodes of the connections which are going through the failed link that a failure has occurred;
- *Fault recovery*: the end nodes of connections involved in a failure must be able to signal the nodes in the backup paths (if shared-protection is used), and switch their transmission or reception to the backup paths;
- *Reverting/Non-reverting*: once the fiber link is fixed (signaled by some higher-layer messages), the end nodes will/will not switch the connections back to their primary paths. Reverting is preferred in shared-path protection scheme because new failures can be handled;
- *Network state update*: the mechanism must also be able to provide updates to reflect which wavelengths are currently being used on each link so that nodes may make routing decisions based on up-to-date information.

In this study, we develop a control protocol with the above capabilities.

²We do not consider node disjointness in the primary and backup paths because “carrier-class” switching equipment should have redundancy in the switch fabric and control plane at each node.

The rest of the document is organized as follows. In Section 2, we describe the architecture of our network model. In Section 3, we propose our control and management protocol, and an algorithm to solve the transmitter/receiver sharing problem in shared-path protection. We present numerical examples in Section 4. Section 5 concludes the study and discusses areas for future research.

2 Network Architecture

Figure 2 shows the architecture of a wavelength-routed WDM network consisting of six *wavelength-routing switches* (WRS). Associated with each WRS, there is an *access station*. A WRS is an intelligent switch and is re-configured upon new connection requests. Our control and management protocol is executed in each WRS. An access station has one or more fixed transmitter arrays (a transmitter array is a set of transmitters, one on each wavelength) and one or more fixed receiver arrays. The link connecting an access station and a WRS is one or more fibers. In this study, we assume that there are M transmitter arrays and M receiver arrays at each access station. Also there are M fibers connecting an access station and its WRS. We will vary the value of M and study its relationship with blocking probability. Note that the maximum number of M , $\max(M) = \Delta(G)$, where $\Delta(G)$ is the maximum nodal degree of the topology graph G of the network. For simplicity of later discussion, we combine a WRS with its associated access station as an integrated unit, which we refer to as a *node*. In the network in Figure 2, we show two lightpaths: one from Node 1 (combination of Access Station A and WRS 1) to Node 3 (combination of Access Station C and WRS 3) on wavelength λ_1 , and the other from Node 1 to Node 3 on wavelength λ_2 . Since the two paths are link-disjoint, one can serve as a primary path and the other can serve as a backup path for a connection from Node 1 to Node 3.

This wavelength-routed network (Figure 2) can be modeled as a layered graph as in Figure 3. Each layer represents a wavelength, and a physical fiber has a corresponding link in each layer. Without wavelength conversion,

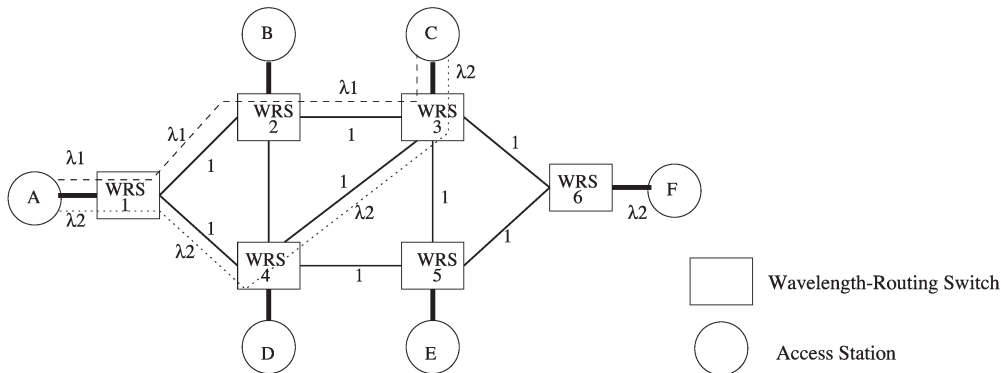


Figure 2: Architecture of a wavelength-routed WDM network.

each lightpath is routed inside one single layer. We use a single layer (e.g., λ_0 in this network) for control messages and connections are set up in other layers (e.g., λ_1 and λ_2 layers). The control layer is a packet-switched network and all packets are routed by shortest paths. Note that, whenever there is a link failure, the corresponding logical link in the control layer also fails. The control packets that are generated after a link failure must be routed on paths that do not go through the failed link.

3 Protocol Descriptions

3.1 Link-state protocol

Two approaches for connection management were compared in [14]: the *link-state* approach and the *distributed-routing* approach. The distributed-routing approach works well in terms of the amount of information stored at each node and the connection-setup delay. However, it is not very suitable for a path-protection network because the source node does not have enough knowledge to find two link-disjoint paths. Hence, we adopt the link-state approach for updating network information and signaling. Note that, because of the advantages of the distributed-routing approach [14], we can combine link-state's updating network state information protocol and distributed-routing's signaling protocol for connection management. This is a future research topic. In this study, we consider link-state protocol only.

The link-state approach is proposed in [15]. It is different from other approaches (such as the distributed-routing approach) in how it updates network state information as well as its signaling protocol. We describe the basic link-state protocol first and then propose modifications to make it work for path-protection networks.

In the original link-state approach, each node maintains the complete network topology, including information about the wavelengths that are in use on each link. Upon the arrival of a connection request, a node utilizes the topology information to select a route and a wavelength. Once the route and the wavelength are selected, the node attempts to reserve the selected wavelength along

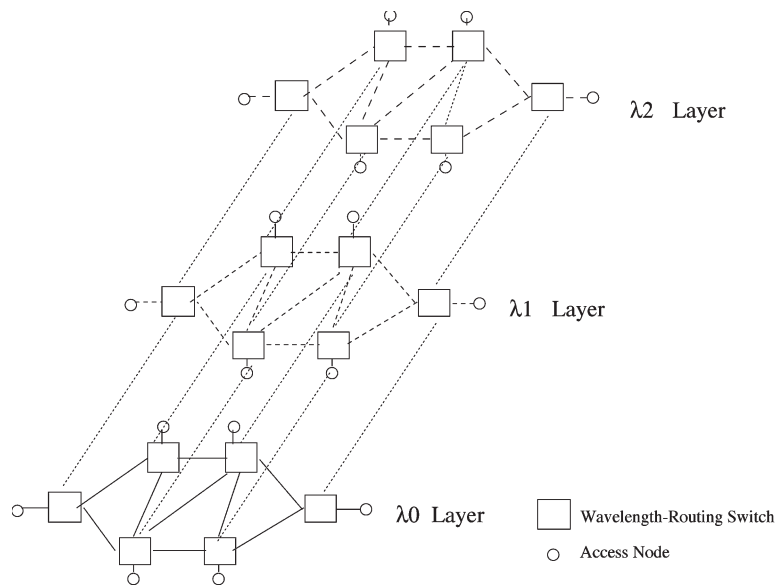


Figure 3: Layered-graph model of the wavelength-routed WDM network with three wavelengths.

each link on the route by sending simultaneous reservation requests to each node on the route. If an intermediate node is able to reserve the wavelength on the appropriate link, it sends an acknowledgement directly back to the source node. If all of the reservations are successful, the source sends a SETUP message to each of the nodes. The appropriate switches are then configured at each node, and the connection is established. If even one of the reservations is not successful, then the call is blocked and the source node sends a TAKEDOWN message to each node on the route in order to release the reserved resources. When a connection is established or taken down, each node involved in the connection broadcasts a topology-update message which indicates any changes in the status of wavelengths being used on the node's outgoing links.

To keep a record of local connections' state information, each node has a *Connection Switch Table* (CST). In CST, each entry is the information about a connection path, including connection ID (a unique identifier in the network for a connection), incoming port, outgoing port, and state of the path (*reserved*, which means the connection on this path is reserved at this node but the switch is not configured yet, or *up*, which means the connection is set up on this path at this node, i.e., the switch has been configured). In a path-protected network, each connection has two paths: a primary path and a backup path,

routed link-disjointly. They will have the same connection ID, but going through (not necessarily) different node(s) in the network. We add one more bit to each entry to indicate whether it is a primary path or a backup path. We call this bit the *type* field. This information is required for routing and wavelength assignment (RWA) for future connection requests as well as for failure recovery if this connection is involved in a link failure. Table 1 shows the CST at Node 3 (combination of Access Station C and WRS 3, which we refer to as Node 3) in the network illustrated in Figure 2. Note that the incoming port is a {previous-hop, wavelength} pair, and the outgoing port is a {next-hop, wavelength} pair. For a connection source, the previous-hop in incoming port is -1 . For a connection destination, the next-hop in outgoing port is -1 .

The next subsection addresses the signaling algorithms in different path-protection schemes as well as other aspects of each scheme.

3.2 Path-protection schemes in mesh networks

Here, we describe three path-protection schemes in mesh networks: 1 + 1 dedicated-path protection, 1:1 dedicated-path protection, and M:N shared-path protection. We will describe each scheme from the desirable capabilities listed in Section 1.

Connection-ID	Incoming-Port	Outgoing-Port	State	Type
1,3,0	2, λ_1	$-1, \lambda_1$	<i>up</i>	primary
1,3,0	4, λ_2	$-1, \lambda_2$	<i>reserved</i>	backup

Table 1: CST at Node 3.

3.2.1 1 + 1 dedicated-path protection and 1:1 dedicated-path protection

The 1 + 1 dedicated-path protection and 1:1 dedicated-path protection schemes work very similarly; they only differ in the fault detection and recovery stages.

3.2.1.1 RWA

The source node computes the shortest path on each wavelength, on which the source node has at least one free transmitter and the destination node has at least one free receiver. Then, it selects the wavelength leading to the “shortest” path for the primary path. If there is a tie, the wavelength with the lowest index is selected. This wavelength-assignment approach is called *first-fit* (FF) [16]. Then, the source node removes every link that appears in the selected primary path (i.e., it removes the corresponding logical link in each of the data layers in Figure 3) and repeats the computation again for the backup path. If the source node cannot find two link-disjoint paths, the connection is blocked.

3.2.1.2 Signaling

After successfully finding a route and a wavelength for each of the two link-disjoint paths, the source node uses the link-state protocol to set up both lightpaths. Note that, each time the node sends out a request (RESERVE or SETUP), it sends the information to $H_1 + H_2$ nodes, where H_1 is the number of hops on the primary path and H_2 is the number of hops on the backup path. When all $H_1 + H_2$ RESERVE-ACK's are received, the source node sends out the SETUP requests. If there is a RESERVE-NACK, the source node takes down the connection by sending TAKEDOWN messages to the $H_1 + H_2$ nodes.

3.2.1.3 Fault detection and recovery

Each link is bidirectional with a pair of uni-directional fibers going on opposite directions. Usually, these two fibers reside in the same cable and they get cut at the same time. To accommodate scenarios in which only a single fiber is cut (in one direction), we develop the following detection scheme which works for both single-fiber-cut and fiber-pair-cut scenarios. Obviously, this approach also works for single bidirectional fiber cut. Before a link is cut, there are either some traffic going through that link or probing data in some special patterns just for keeping the line “alive” instead of “idle”. When there is a cut, the downstream node (both end nodes will be downstream as well as upstream if the link is bidirectional) of this link will detect the failure. If 1 + 1 protection is used, each connection destination simply switches to the other path to receive data once it detects that there is no signal coming from the primary path. No signaling is required for the recovery. However, in 1:1 protection, no data is transmitted on the backup path normally. When

there is a failure on fiber $x \rightarrow y$, Node y looks up its CST, finds all the entries with incoming port $(x, *)$ and type *primary*, and notifies the source of every such connection about this failure; the source nodes will subsequently switch to transmit on their backup paths.

As an example, consider one connection. The source node that is notified about the failure may have transmitted some data which is lost in the network. So, it needs to retransmit the lost data. By the time the retransmitted data reaches the destination, the destination has already detected the failure and is waiting for data to come in from the backup path. So, the connection is recovered. If the link $x \leftrightarrow y$ is cut, both Node x and Node y will do the notification job. Also, they must reflect this information in their state-update messages so that other nodes will not attempt to use this link in their routing algorithm until the failure is fixed. In previous work [9], the upstream node of a failed link is usually assumed to take the role of notifying connection sources, and the downstream node of the failed link is assumed to notify connection destinations. Then, to combat single-direction link failures, we must have the downstream node of a failed link to notify the upstream node first, which can subsequently notify the connection sources. Compared to this approach, our approach is more efficient and gives better performance in recovery time. In Figure 4, a lightpath from Node 0 to Node 3 goes through link $1 \rightarrow 2$. When link $1 \rightarrow 2$ breaks, Node 2 detects the failure, and notifies Node 0 via Node 4 and Node 5 (note that this message is delivered in a store-and-forward manner since the control layer is a packet-switched network). Node 0 can then switch to the backup path $0 \rightarrow 5 \rightarrow 4 \rightarrow 3$.

3.2.1.4 Reverting/non-reverting

Once the link is fixed, the source node and the destination node can switch back to their primary path because the primary path usually has fewer hops than the backup path. But reverting is not necessary in dedicated-path protection.

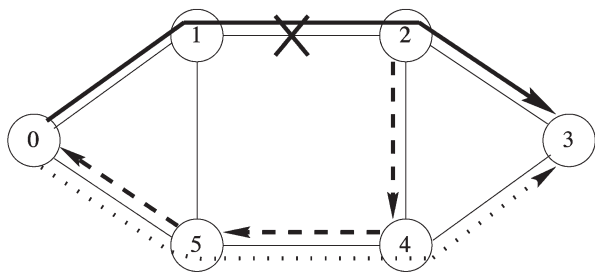


Figure 4: An example of the downstream node of a failed link notifying the connection source: the solid line shows the primary path, the dotted line shows the backup path, and the dashed line shows the failure notification to the source node of the connection.

3.2.2 M: N shared-path protection

3.2.2.1 RWA

Shared-path protection is more complex in the RWA algorithm because we must decide:

- which primary paths can share wavelength/link(s) in their backup paths, and
- how many transmitters/receivers are required for backup paths.

To solve these two problems, each node must not only have knowledge of the state of each link in the network, but also have knowledge of which connection is using which wavelength/link, and whether on its primary path or backup path. So, in each network-state update message, we let each node broadcast its CST instead of the wavelength states on each of its outgoing links.

The first problem is easier to solve than the second. If we know the routes of the two paths, we can easily determine whether or not they are link-disjoint. If the answer is “yes”, we allow them to share wavelength/link(s); otherwise we do not. However, when we select a wavelength, we must consider whether the source and the destination still have transmitter/receiver on this wavelength. We do not allocate transmitter/receiver for backup paths until a failure occurs but we have to make sure that there is free transmitter/receiver that can be used for each of the connections affected if a failure occurs. One simple way to do this is to have the primary and backup paths on the same wavelength so that the transmitter/receiver allocated for the primary can be used for its backup if a failure occurs. This is not an ideal situation because it limits the degree of sharing among protection resources. Also, it makes the network vulnerable to transmitter/receiver failures. So, we allocate different transmitters/receivers to primary and backup paths.

If, on a wavelength, all the transmitters at source or all the receivers at destination are occupied, this wavelength is out of our consideration. We address how to decide the minimum number of transmitters/receivers on a certain wavelength at the source/destination node in Section 3.2.3.

Below, we introduce the RWA algorithms for primary paths and backup paths separately.

- 1) RWA for a primary path.
 - a) In each wavelength layer, remove the logical links³ that appear in either a primary or a backup path.
 - b) Initialize our candidate wavelength set to be all the wavelengths: $\{\lambda_1, \lambda_2, \dots, \lambda_w\}$, where w is the total number of wavelengths used for data connections in the network, i.e., we have $w + 1$ wavelengths in the network and λ_0 is used for control messages.

³A logical link is a link in the layered graph of Figure 3, i.e., a wavelength on a physical link.

- c) For each wavelength λ_p let the set of primary paths originating at the source node on wavelength λ_i be P_i and the set of backup paths originating at the source node on wavelength λ_i be B_i . Apply the method described in Section 3.2.3 to B_i and decide the number of transmitters required T_i . If $|P_i| + T_i = M$ (recall that M is the number of transmitter arrays and receiver arrays at each access station), λ_i is removed from our candidate wavelength set. Otherwise, repeat this procedure for the set of primary paths terminating at the destination node on wavelength λ_i (P_i'), and the set of backup paths terminating at the destination node on wavelength λ_i (B_i'). Let R_i be the number of receivers required at the destination node for B_i' . If $|P_i'| + R_i = M$, λ_i is removed from our candidate wavelength set. Otherwise, λ_i stays in the candidate wavelength set.
 - d) For each wavelength remaining in the candidate wavelength set, compute the shortest-path. For the wavelengths giving the best shortest paths, we break the tie by First-Fit. Assume we get path p_l (which is a set of links) and wavelength λ_l for the primary path.
- 2) RWA for a backup path.

We name the backup path by b^* . Note that we do not know b^* yet but we know its primary path p_l .
 - a) Eliminate all the logical links used in either a primary path, or a backup path if its primary path shares common physical link(s) with p_l . Also eliminate all the corresponding logical links in all wavelength layers of the links appearing in p_l .
 - b) Initialize our candidate wavelength set to be all the wavelengths: $\{\lambda_1, \lambda_2, \dots, \lambda_w\}$.
 - c) For each wavelength λ_p let the set of primary paths originating at the source node on wavelength λ_i be P_i and the set of backup paths originating at the source node on wavelength λ_i be B_i . Note that B_i does not contain b^* and P_i does not contain p_l since they have not been reserved yet and are not contained in any CSTs. Now set $B_i = B_i + \{b^*\}$. Also, for $i = 1$ only, set $P_i = P_i + \{p_l\}$ because λ_1 is selected for p_l . Apply the method described in Section 3.2.3 to B_i and decide the number of transmitters T_i required. If $|P_i| + T_i > M$, λ_i is removed from our candidate wavelength set. Otherwise, repeat this procedure for the set of primary paths terminating at the destination node on wavelength λ_i (P_i'), and the set of backup paths terminating at the destination node on wavelength λ_i (B_i'). Set $B_i' = B_i' + \{b^*\}$ and $P_i' = P_i' + \{p_l\}$. Let R_i be the number of receivers required at the destination node for B_i' . If $|P_i'| + R_i > M$, λ_i is removed from our candidate wavelength set. Otherwise, λ_i stays in the candidate wavelength set. Note that, by setting $P_l = P_l + \{p_l\}$, $B_i = B_i$

+ $\{b^*\}$, $P_i' = P_i' + \{p_i\}$, and $B_i' = B_i' + \{b^*\}$, we do not allow the sharing of transmitter/receiver between the primary path and the backup path for a given connection. By avoiding the sharing of transmitter/ receiver within the same connection, the network is immune to single-transmitter/receiver failures.

d) For each wavelength remaining in the candidate wavelength set, compute the shortest-path. For the wavelengths giving the best shortest paths, we break the tie by *First-Fit*, *Last-Fit*, or, *Max-Shared-First*. We already explained First-Fit in Section 3.2.1. The latter two are as follows.

- *Last-Fit* (LF): the highest-indexed wavelength is selected.
- *Max-Shared-First* (MSF): for each wavelength λ_p let the links used in the backup path be l_0, l_1, \dots, l_{k-1} if the backup path is of k hops. Suppose wavelength λ_i is shared by s_j backup paths on link l_j , $0 \leq j \leq k-1$. Then, compute the following function for λ_i :

$$MSF(i) = \sum_{j=0}^{k-1} s_j$$

The λ_i of the largest MSF(i) value is selected for the backup path. This is based on the idea of maximizing the sharing among protection resources. If there is a tie, either First-Fit or Last-Fit can be used to break the tie.

Since wavelength assignment for primary paths is using First-Fit, for wavelength assignment for backup paths, Last-Fit and Max-Shared-First with Last-Fit for tie-breaking should give better performances than First-Fit, because they “pack” the backup wavelengths at one end of the wavelength space. Thus, there can be more sharing among backup paths than the case where they are interleaved with wavelengths used by primary connections.

3.2.2.2 Signaling

When the source node gets a connection request, it performs RWA for both paths, then sends RESERVE request to each node on the two paths. Upon reception of all the RESERVE-ACKs, it sends out SETUP request to nodes on primary path only. When it gets all the SETUP-ACKs from the nodes to whom it sent SETUP request, the connection is set up and the source node starts transmitting its data.

3.2.2.3 Fault detection and recovery

When there is a link cut, the downstream node of the fiber detects the cut immediately. As in 1:1 protection, the downstream node will notify each source node whose connections are going through this fiber about this failure. Each source node, as in the SETUP stage, sends SETUP-BACKUP request to each node along the backup path and tells it to configure its switch. The nodes configure their switches, and send SETUP-BACKUP-

ACK back to the source node. When the source node receives all SETUP-BACKUP-ACKs, it starts transmitting data on the backup path. Of course, it needs to estimate how much data has been lost due to this failure and retransmit the lost data.

3.2.2.4 Reverting

Reverting should be done because, when a backup path b is being used, the other primary paths whose backup paths share wavelength/links with b are left unprotected. So, the connection should be immediately switched back to the primary path once the link failure is fixed. However, there is a *vulnerable period* for the connections whose backup paths share wavelength/links with b . The length of this vulnerable period is equal to the time required to fix a link. If there is a second link failure during this vulnerable period, the affected connections may not be restored.

3.2.3 Determining the number of transmitters/receivers required for backup paths at a source node in shared-path protection

Given a set of backup lightpaths originating at the same node n and on the same wavelength w , our objective here is to decide the minimum number of transmitters that should be reserved for them; or given a set of backup lightpaths destined for node n and on the same wavelength w , we need to decide the minimum number of receivers that should be reserved for them. This problem can be solved as follows:

- 1) Find the set of primary paths for which these backup paths are reserved; let it be P_w .
- 2) For each link l in the network, count the number of paths in P_w that traverse it. Note that each link consists of two fibers going on opposite directions. Hence, traversing from both directions should be counted. Let this number for link l be b_{wl} . b_{wl} denotes the number of broken paths we will have to restore on wavelength w if link l gets cut. In other words, it means the number of transmitters/receivers on wavelength w that will be working for backup paths once link l gets cut.
- 3) The maximum number of b_{wl} , among all links in the network, will be the number of transmitters (or receivers) required for the set of backup paths at source node n (or destination n) on wavelength w .

4 Illustrative Numerical Examples and Discussion

We study the performance of the three protection schemes using the sample network shown in Figure 5, which represents a typical metropolitan-area telecom network. The numbers on each link represent the length of the links in units of 10 km. Thus the propagation delay of a link with length 10 units, i.e., 100 km, is 500 μ s.

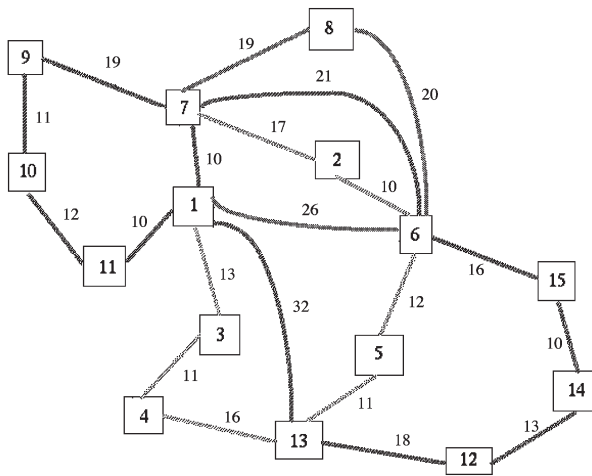


Figure 5: Sample network.

We assume the following parameters:

- Message processing time at a node, P , is $10 \mu\text{s}$.
- Time to configure, test and set up a cross-connect, C , is $500 \mu\text{s}$.
- Link-cuts occur at rate of $0.0015 \text{ cuts}/\text{ms}$ and it takes 20 ms to fix a link.⁴
- Number of wavelengths on each link (in each direction) is 8.
- Bit rate per wavelength channel (per lightpath) is 2.5 Gbps , i.e., OC-48.
- Connection requests arrive as a Poisson process with mean arrival rate $\in [0.01, 0.3] \text{ arrivals}/\text{ms}$ (given as a program parameter). Connection requests are uniformly distributed among all source-destination pairs.
- Connection-holding time is exponentially distributed with mean 100 ms . Note that, in the plots, load (Erlang) is calculated as: request arrival rate \times mean connection-holding time.

The following are some characteristics about the sample network:

- Total number of nodes in the network, $N = 15$.
- Number of links (bidirectional) = 21.
- Average nodal degree = 2.8.
- Average link length = 153 km .
- Average propagation delay between two nodes $D = 1.82381 \text{ ms}$.
- Average hop distance between two nodes $H = 2.40952$.

We simulate uniform traffic between each source-destination pair on this network and evaluate the performance of $1 + 1$, $1 : 1$, and shared-path ($M:N$) protection from the following metrics:

- Connection setup time—time required to establish a connection once a connection request arrives.

⁴Usually fiber cuts occur at rate of $4.39 \text{ cuts}/1000 \text{ sheath miles}/\text{year}$ and it takes around 12 hours to fix a fiber failure [17]. In order to simulate enough cuts in our system, we have to increase the cut rate and shrink the cut-fixing delay.

- Blocking probability—probability that a connection cannot be established due to failure in routing and wavelength assignment or resource contention along the desired route.
- Restoration time—time required to recover a connection when a failure occurs.
- Data loss per cut—number of bits lost due to a link cut, which is related to both restoration time and number of connections carried by a link.

It is obvious that $1 + 1$ and $1 : 1$ have the same performance as to blocking probability as well as connection setup time. There is neither restoration time nor data loss per cut in $1 + 1$, if we choose ε properly (Recall that ε is the delay from the time the source transmits a bit on the primary path till it transmits the same bit on the backup path. Usually, the backup path is longer than the primary path and the switching at the destination is very fast upon detection of failure on the primary path. In that case, ε can be 0.) So we present the results of $1 : 1$ only for dedicated-path protection.

Figure 6 shows blocking probability versus load when dedicated-path protection is employed. The case without link failures is also plotted. We observe that the performance when $M = 3$ transmitter/receiver arrays are used at each node is very close to the performance of six transmitter/receiver arrays. Our results indicate that we do not need to equip each node with the maximum nodal-degree number of transmitter/receiver arrays, i.e., six in this network.

Figures 7 through 10 plot the blocking probability versus load for shared-path protection when different wavelength-assignment schemes for backup paths are applied. We notice that, in some cases, a system with three transmitter/receiver arrays gives better performance than a system with six. This may due to the fact that, when only three transmitter/receiver arrays are existing at each node, intermediate nodes have fewer chances to

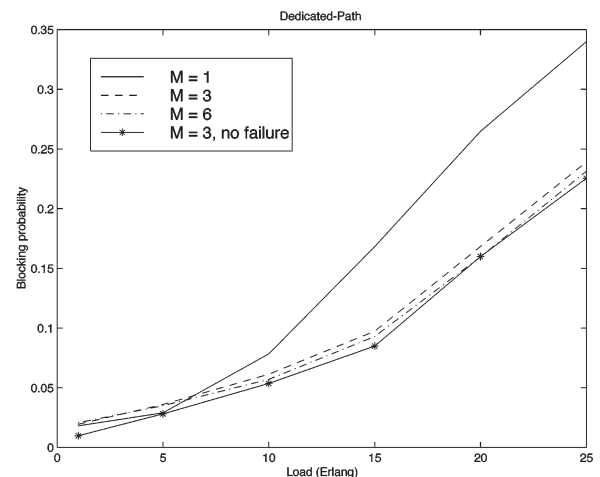


Figure 6: Blocking probability versus load for dedicated-path $1 : 1$ and $1 + 1$ protection, with $M = 1, 3, 6$.

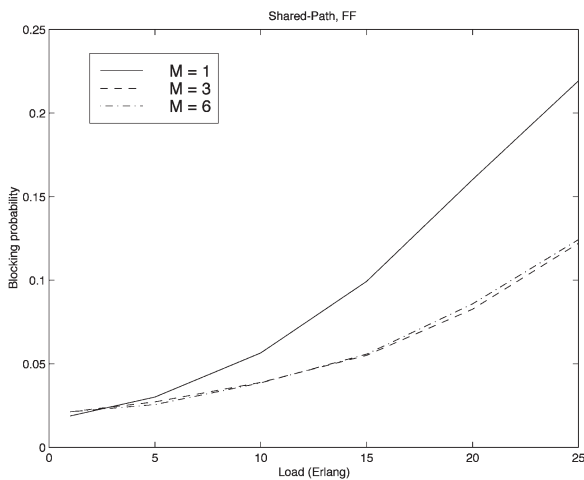


Figure 7: Blocking probability versus load for shared-path protection, FF for backup-path wavelength assignment, with $M = 1, 3, 6$.

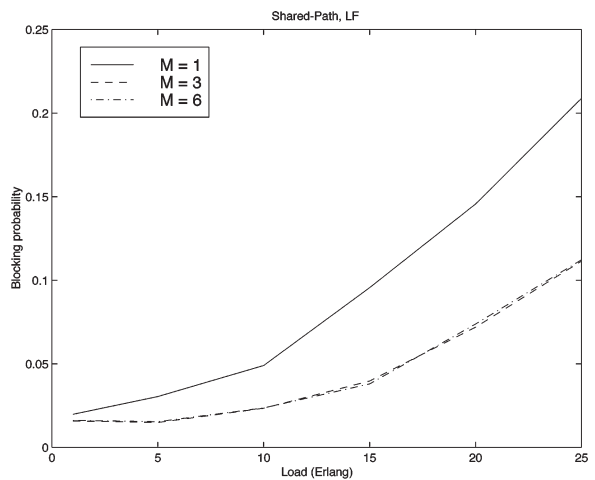


Figure 9: Blocking probability versus load for shared-path protection, LF for backup-path wavelength assignment, with $M = 1, 3, 6$.

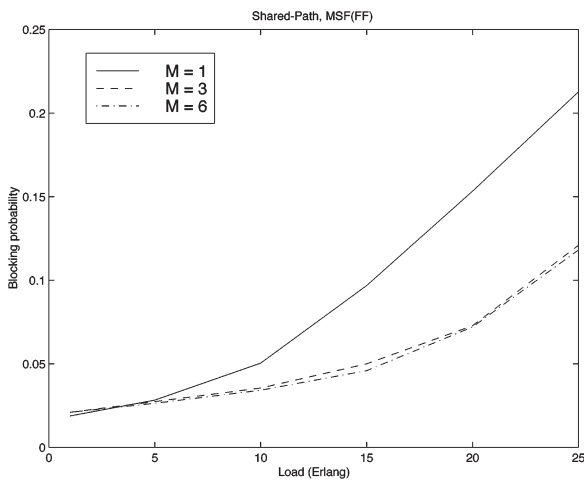


Figure 8: Blocking probability versus load for shared-path protection, MSF & FF for backup-path wavelength assignment, with $M = 1, 3, 6$.

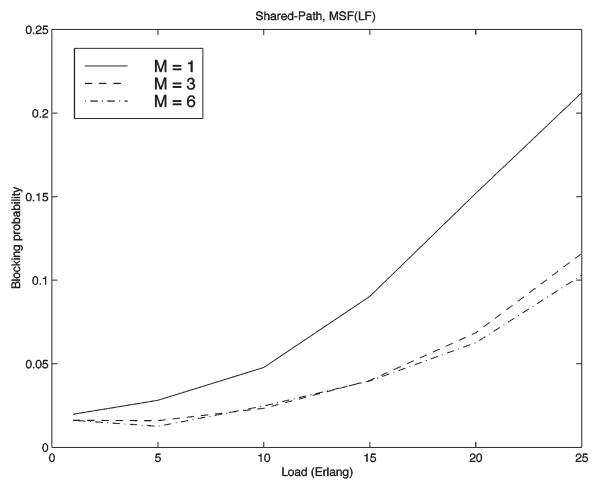


Figure 10: Blocking probability versus load for shared-path protection, MSF & LF for backup-path wavelength assignment, with $M = 1, 3, 6$.

source/sink connections than in the six transmitter/receiver-array case, which implies that more connections can be routed through those nodes. Hence, the overall blocking is decreased.

The four wavelength-assignment schemes for backup paths are compared for $M = 3$ in Figure 11. It is interesting that Last-Fit (LF) performs better than Max-Shared-First with Last-Fit for tie-breaking (MSF(LF)) in some cases. This is because, when the load is high, each node may have out-of-date information about existing connections; hence, they may not be able to make the best decision with MSF(LF). However, with LF, the wavelengths on the backup paths have better chances to be shared.

Thus, one may use LF for its simplicity and good performance.

Figure 12 shows the connection setup delay versus load for dedicated-path 1:1 protection and shared-path

($M:N$) protection with MSF(LF), when $M = 3$. The higher setup delay in 1:1 protection comes from the switch configuration time and propagation delay of signaling messages when setting up the backup paths right at the beginning. Also note that the connection setup time decreases as load increases. This is because longer connections are more likely to get blocked than shorter ones. It decreases faster in 1:1 protection due to its poorer resource utilization at heavy loads.

Figures 13 and 14 compare the restoration delay and data loss per cut in 1:1 and shared-path ($M:N$) protection with MSF(LF) ($M = 3$) schemes. Figure 13 also plots the 95% confidence intervals for the restoration delay in shared-path protection. With current assumptions of message processing speed and switch configuration time, it takes about 2 ms to recover a connection in 1:1 and about 9 ms in shared-path protection. Both are decent re-

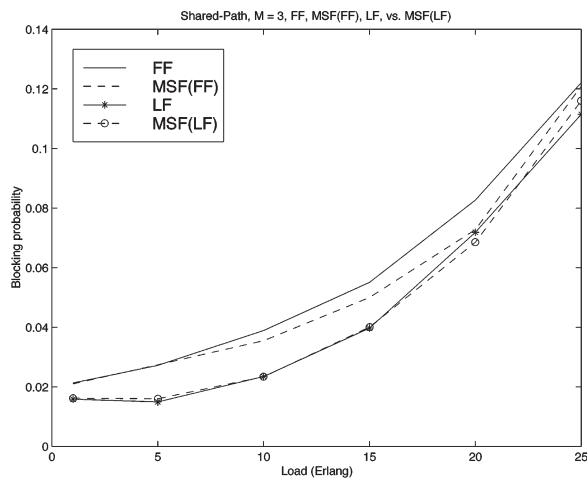


Figure 11: Blocking probability versus load for shared-path ($M:N$) protection, and different wavelength assignment schemes for backup path, with $M = 3$.

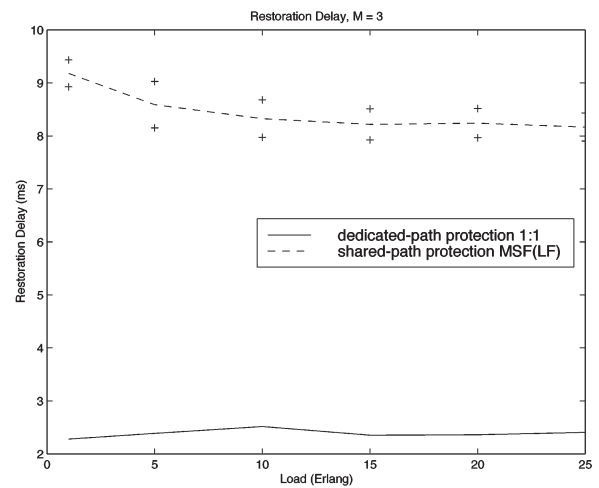


Figure 13: Restoration delay versus load for dedicated-path 1:1 protection and shared-path ($M:N$) protection, with $M = 3$.

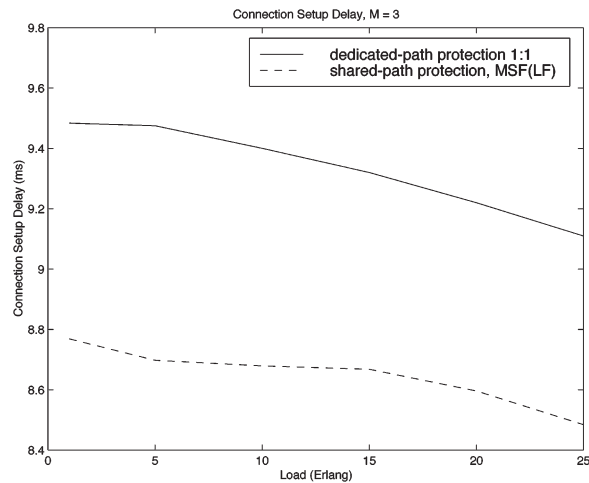


Figure 12: Connection setup delay versus load for dedicated-path 1:1 protection and shared-path ($M:N$) protection, with $M = 3$.

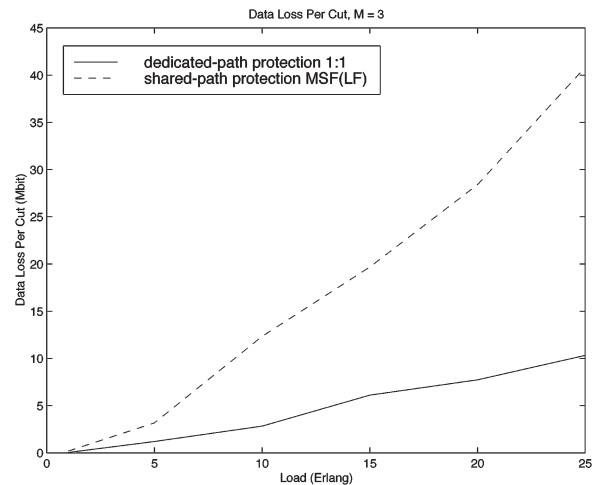


Figure 14: Data loss per cut versus load for dedicated-path 1:1 protection and shared-path ($M:N$) protection, with $M = 3$.

covery speed. Of course, the gap will grow if our assumed parameter values become larger. In Figure 14, the average data loss is found to grow with the network load in both schemes. Under a load of 25 Erlangs and $M = 3$, in which case 23% connections are blocked in 1:1 (Figure 6) and 11% connections are blocked in shared-path protection with MSF(LF) for backup-path wavelength-assignment (Figure 11), 1:1 loses around 10 Mbits of data per cut while shared-path loses around 40 Mbits of data per cut.

5 Conclusion

In this study, we proposed an on-line control and management protocol for setting up lightpaths with protection paths. The wavelength/links on the protection paths can be either dedicated to a certain connection, or

shared among multiple connections. Dedicated-path protection has better performance in terms of connection-recovery time. However, it is not very resource-efficient. It has higher blocking probability than the shared-path protection scheme. Under our current assumptions of message processing speed and switch configuration time, the connection-recovery time in the sample network (Figure 5) when applying shared-path protection is under 10 ms. That is an acceptable recovery time.

As ongoing research, we are applying these protocols in a network with different connectivity characteristics. Future research topics include developing an analytical model for each scheme, and combining link-state's updating-network-state-information protocol and distributed-routing's signaling protocol for connection management in a survivable WDM network.

6 References

- [1] B. Mukherjee, *Optical Communication Networks*. New York: McGraw-Hill, 1997.
- [2] R. Ramaswami and K. N. Sivarajan, *Optical Networks: A Practical Perspective*. San Francisco: Morgan Kaufmann, 1998.
- [3] J. Anderson, B. T. Doshi, S. Dravida, and P. Harshavardhana, "Fast restoration of ATM networks," *IEEE Journal on Selected Areas in Communications*, vol. 12, pp. 128-138, January 1994.
- [4] C. Huitema, *Routing in the Internet*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
- [5] S. Makam, V. Sharma, K. Owens, and C. Huang, "Protection/restoration of MPLS networks." *IETF draft*, October 1999. Expiration day April 2000.
- [6] O. Gerstel, "Opportunities for optical protection and restoration," *Proc. OFC '98*, vol. 2, (San Jose, CA), pp. 269-270, February 1998.
- [7] T. Wu, *Fiber Network Survivability*. Boston, MA: Artech House, 1992.
- [8] T. Wu, "Emerging technologies for fiber network survivability," *IEEE Communications Magazine*, pp. 58-74, February 1998.
- [9] S. Ramamurthy and B. Mukherjee, "Survivable WDM mesh networks, part I—protection," *Proc. IEEE INFOCOM '99*, vol. 2, (New York, NY), pp. 744-751, March 1999.
- [10] A. Fumagalli, I. Cerutti, F. Masetti, R. Jagannathan, and S. Alagar, "Survivable networks based on optimal routing and WDM self-healing rings," *Proc. IEEE INFOCOM '99*, vol. 2, (New York), pp. 726-733, March 1999.
- [11] S. Ramamurthy and B. Mukherjee, "Survivable WDM mesh networks, part II—restoration," *Proc. IEEE International Conference on Communications (ICC '99)*, vol. 3, (Vancouver, Canada), pp. 2023-2030, June 1999.
- [12] R. R. Iraschko and W. D. Grover, "A highly efficient path-restoration protocol for management of optical network transport integrity," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 779-794, May 2000.
- [13] G. Ellinas, S. Rong, A. Hailemariam, and T. E. Stern, "Protection cycle covers in optical networks with arbitrary mesh topologies," *Proc. OFC '00*, vol. Th, (Baltimore, Maryland), pp. 213-215, March 2000.
- [14] H. Zang, L. Sahasrabudde, J. P. Jue, S. Ramamurthy, and B. Mukherjee, "Connection management for wavelength-routed WDM networks," *Proc. IEEE Globecom '99*, vol. 2, (Rio de Janeiro, Brazil), pp. 1428-1432, December 1999.
- [15] R. Ramaswami and A. Segall, "Distributed network control for optical networks," *IEEE/ACM Transactions on Networking*, vol. 5, pp. 936-943, December 1997.
- [16] H. Zang, J. P. Jue, and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical [WDM] networks," *Optical Networks Magazine*, vol. 1, pp. 47-60, January 2000.
- [17] M. To and P. Neusy, "Unavailability analysis of long-haul networks," *IEEE Journal on Selected Areas in Communications*, vol. 12, pp. 100-109, January 1994.

Hui Zang
hzang@sprintlabs.com



Hui Zang received the B.S. degree from Tsinghua University, China in 1997, and the M.S. degree from the University of California, Davis in 1998. She is expected to receive the Ph.D. degree in computer science from the University of California, Davis in June 2001. From June 1999 to September 1999, she worked at IBM Almaden Research Center in San Jose, CA as a summer intern. From October 1999 to September 2000, she worked at Sprint Advanced Technology Laboratories, Burlingame, CA as a part-time intern, where she became a principal R&D engineer in October 2000. Her research interests include photonic packet switching, WDM network control and management, and fault-tolerance in WDM networks.

Biswanath Mukherjee
mukherje@cs.ucdavis.edu



Biswanath Mukherjee received the B.Tech. (Hons) degree from Indian Institute of Technology, Kharagpur (India) in 1980 and the Ph.D. degree from University of Washington, Seattle, in June 1987. At Washington, he held a GTE Teaching Fellowship and a General Electric Foundation Fellowship. In July 1987, he joined the University of California, Davis, where he has been Professor of Computer Science since July 1995, and Chairman of Computer Science since September 1997. He is co-winner of paper awards presented at the 1991 and the 1994 National Computer Security Conferences. He serves or has served on the editorial boards of the *IEEE/ACM Transactions on Networking*, *IEEE Network*, *ACM/Baltzer Wireless Information Networks (WINET)*, *Journal of High-Speed Networks*, *Photonic Network Communications*, and *Optical Network Magazine*. He also served as Editor-at-Large for optical networking and communications for the *IEEE Communications Society*. He served as the Technical Program Chair of the *IEEE INFOCOM '96* conference. He is author of the textbook "Optical Communication Networks" published by McGraw-Hill in 1997, a book which received the Association of American Publishers, Inc.'s 1997 Honorable Mention in Computer Science. His research interests include light-wave networks, network security, and wireless networks.