

# Reliable WDM Multicast in Optical Burst-Switched Networks\*

Myoungki Jeong<sup>†</sup>, Chunming Qiao<sup>†</sup> and Yijun Xiong<sup>‡</sup>

Departments of EE and CSE<sup>†</sup>

State University of New York at Buffalo

Buffalo, NY 14260

email: {mjeong,qiao}@eng.buffalo.edu

Alcatel Corporate Research Center<sup>‡</sup>

Richardson, TX 75081

email: xionyi@usa.alcatel.com

**Abstract**– In this paper, we present a reliable WDM (Wavelength-Division Multiplexing) multicast protocol in optical burst-switched (OBS) networks. Since the burst dropping (loss) probability may be potentially high in a heavily loaded OBS backbone network, reliable multicast protocols that have developed for IP networks at the transport (or application) layer may incur heavy overheads such as a large number of duplicate retransmissions. In addition, it may take a longer time for an end host to detect and then recover from burst dropping (loss) occurred at the WDM layer. For efficiency reasons, we propose burst loss recovery within the OBS backbone (i.e., at the WDM link layer). The proposed protocol requires two additional functions to be performed by the WDM switch controller: subcasting and maintaining burst states, when the WDM switch has more than one downstream on the WDM multicast tree. We show that these additional functions are simple to implement and the overhead associated with them is manageable. Using the proposed protocol, member edge routers on the multicast tree is implicitly organized into a hierarchical structure dynamically. In addition, it leads to very efficient local loss recovery that results in no duplicate transmissions, no implosion and early detection of dropping. Simulations show that the proposed protocol results in a reasonable delivery latency and amount of bandwidth consumption under a various traffic load and group size.

**Keywords** : Optical Burst Switching, Reliable Multicast, Loss Recovery, WDM

## 1 Introduction

As traffic demand increases exponentially in the Internet, Wavelength Division Multiplexing (WDM) networks [1, 2, 3] become a natural choice for the backbone. Recently, IP over WDM networks (or so-called Optical Internet) have received a considerable amount of attention (e.g. [4, 5, 6]). Meanwhile, multicasting (i.e. one-to-many or many-to-many communications) is becoming more and more popular and important in the Internet. Most previous work for multicasting has been done in the broadcast-and-select WDM networks (e.g. [7, 8, 9]). Here we focus on multicasting in backbone WDM networks. Multicasting in IP over WDM networks (see Fig. 1) can be done via IP multicast, multiple WDM unicast, or WDM multicast [10]. In this paper, we will concentrate mainly on WDM multicast, where a signal from a source edge router is multicast to multiple destinations (edge IP routers) without going through O/E/O conversions.

WDM multicasting has several potential advantages over the other two. First, with the knowledge of the physical (i.e. WDM layer) topology, which may differ from what is seen at the upper electronic (e.g. IP) layer, more efficient (in terms of bandwidth and/or latency) multicast trees can be constructed at the WDM layer. Secondly, some WDM switches uses power-splitting components, and power-splitting is more efficient than copying (by IP) for multicasting purposes. Finally, multicasting at the WDM layer provides a higher degree of data transparency (in terms of bit-rate and coding format). In particular, it has been shown that the tremendous bandwidth saving and reduction in wavelengths can be achieved using WDM multicast [11].

---

\*This research is supported in part by a grant from NSF under contract number ANIR-9801778.

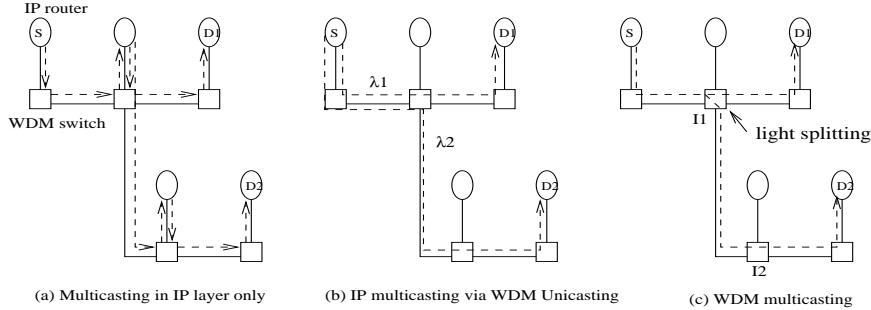


Figure 1: Multicasting on IP over WDM networks.

There are two WDM multicasting approaches, one based on wavelength-routing as in [11, 12, 13, 14], and the other based on optical burst switching (OBS) [6, 15] as in [10, 16]. In the former, a wavelength needs to be dedicated to each branch of a multicast tree and multicast data will be switched to one or more outgoing wavelengths according to the incoming wavelength that carries the data (as in wavelength-routing). This scheme is suitable for high bandwidth multicast applications having a relatively long duration, such as video (HDTV) distribution. In the latter, no wavelengths need to be dedicated to a multicast tree, and the multicast data is transported via OBS where data is sent as optical bursts at the rate of each wavelength. Given the bandwidth of each wavelength is much higher (e.g. OC-48 or even OC-192), than the sustained rate required by most of the prevailing multicast applications (e.g. news, content distribution), WDM multicast based on OBS will be more bandwidth efficient than wavelength routing for bursty traffic.

In OBS networks, a burst may be blocked at intermediate nodes due to contention for a limited number of wavelengths in an outgoing fiber since OBS uses one-way reservation (as in Tell-and-Go). A blocked burst is dropped (lost) when there is no buffer (in a form of fiber-delay lines (FDLs)) at the blocking node (which is usually the case in bufferless OBS networks). Note that although FDLs may exist in OBS networks, those are very scarce resources and in addition, can delay a burst only for a limited amount of time. Hence, it is highly likely that a multicasted burst will be dropped (lost) on its way to one of the multicast receivers. On the other hand, some applications (e.g., software distribution) may need reliable delivery to all receivers, and for those applications the lost bursts have to be recovered.

The rest of this paper is organized as follows. In Section 2, we describe the background and our motivations for burst loss recovery at the WDM link layer. In Section 3, we present the proposed burst loss recovery protocol in detail. Section 4 presents the performance results of the proposed protocol. In Section 5, we describe related work in reliable multicast at the transport (or application) layer. Section 6 concludes the paper.

## 2 Background and Motivation

As in reliable multicast in IP networks, loss recovery in OBS networks should also possess the following properties:

1. Avoidance of implosion - implosion of signaling packets (ACK or NAK) occurs when a large number of receivers requests for retransmission simultaneously.
2. Local recovery - burst loss has to be recovered in a localized way which results in reduced recovery latency and bandwidth saving.
3. No duplicate transmission - a retransmitted burst may be delivered to receivers which have already received the multicast burst. To save bandwidth and reduce processing load, the duplicate retransmission should be kept low. This is related to local recovery.
4. Closest responder - reducing the number of hops between the responder and the requester of burst retransmission is particularly important in OBS networks since loss probability of a retransmitted burst in OBS networks depends heavily on the hop count.
5. Adaptability - a loss recovery mechanism should be adaptable to membership changes.

In this paper, we propose a burst loss recovery protocol at the WDM link layer in bufferless OBS networks under the MPLS framework. We consider burst loss recovery in a backbone OBS network where WDM links and optical burst switches are used to connect ingress/egress edge routers as shown in Fig. 2. For each multicast group  $(S, G)$ , where  $S$  is a multicast source address and  $G$  is a multicast group address, we assume that an IP multicast tree has been set up. As a part of an IP multicast tree, one edge router (e.g., E1 in Fig. 2) serves as an upstream for some other edge routers (e.g., E2 and E3 in Fig. 2). These edge routers on the multicast tree for a group  $(S, G)$  are referred to as *member routers*. Not all edge routers have to be member routers (for example, E4 and E5 are not member routers for  $(S, G)$  as illustrated in Fig. 2). Within the OBS backbone, we assume that each WDM switch is a labeled optical burst switch (or LOBS), and its corresponding controller has unicast routing information. In addition, a label-switched multicast tree (or subtree) has been set up between the member edge routers as in [10, 17].

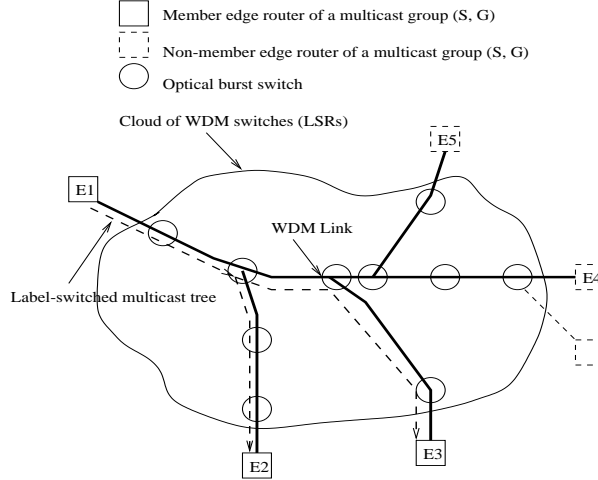


Figure 2: A backbone OBS network as WDM links.

To send a burst from one upstream edge router (e.g., E1) to downstream edge routers (e.g., E2 and E3), a control packet carrying a (multicast) label along with any other information is sent first to reserve bandwidth for the following burst. This control packet is processed by every intermediate WDM switch controller along the label-switched multicast tree, which determines the downstream WDM switch (or switches) to forward, and the new label(s) to be used, all based on the incoming label. After processing the control packet, the WDM switch controller will also reserve bandwidth on the outgoing link(s) for the following burst, and set the corresponding WDM switch so that the burst will cut-through the switch when it arrives (i.e., there will be no O/E/O conversions of the burst).

As to be described in more detail in later sections, our burst loss recovery approach is basically a WDM link layer approach as opposed to transport or application layer that deal with end hosts. Taking care of burst loss recovery at the WDM link layer in an OBS network will increase efficiency because of the potentially high blocking probability of the burst within a heavily loaded OBS network. More specifically, the loss recovery at the WDM layer can reduce the detection time of blocking (loss) and the recovery latency (compared to end-to-end mechanism at the transport layer).

### 3 Burst Loss Recovery Protocol (BLRP)

In this section, we present the proposed solution for burst loss recovery in OBS networks. First, we describe an overview of the solution, and followed by the details of the proposed protocol.

#### 3.1 Loss Recovery Overview

To recover a lost burst due to blocking in OBS networks, an efficient way is to request retransmission from a closest node on the multicast tree (e.g., an edge router reachable in a minimum number of hops) that has a copy of the burst. This will reduce

the delivery latency and bandwidth consumption in the network. In addition, it will reduce the blocking probability of the retransmitted burst.

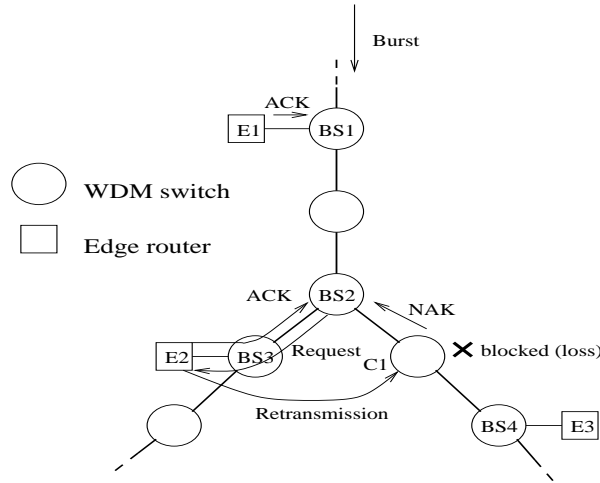


Figure 3: Burst loss and retransmission

Fig. 3 shows an example of a multicast tree where only edge routers that lead to members of a multicast group are shown (i.e., E1, E2 and E3). Suppose that a multicast burst is delivered to both E1 and E2, but dropped at C1. As a result, both E1 and E2 will send an ACK towards the multicast source, but the switch C1 will send a NAK. With local recovery, the switch BS2 will request a retransmission from E2 (rather than E1 or the multicast source) since E2 is the closest (in terms of the number of hops along the multicast tree). After receiving the request, E2 will transmit its copy of the burst to switch C1 by tunneling of the control packet (i.e., encapsulating the control packet). The encapsulated control packet can thus bypass intermediate nodes (including BS2 if it is on the unicast path to switch C1) as a unicast burst, but will be treated as a control packet for multicast at C1 (and beyond).

Note that the recovery procedure described above requires that topology of the multicast tree be maintained (not just the label information to reach downstreams) by the WDM switch controllers in order to forward signaling packets (ACK or NAK) upstream along the multicast tree. The WDM switch controllers also decide where to send the retransmission request based on the number of hops to the member edge routers that have sent ACKs.

To facilitate the following presentation, we introduce some concepts and definitions. We call a WDM switch (e.g., BS2 in Fig. 3 that has more than one downstream (along the multicast tree) a *branching switch* (BS). In addition, if one of the edge routers attached to a WDM switch is a member of a multicast group, we call the WDM switch (e.g., BS1, BS3 or BS4) a *member switch* (MS). An edge router (e.g., E2) that responds to the retransmission request is called *responder*. A *target switch* (e.g., C1) is a WDM switch that receives an encapsulated control packet and then multicasts the corresponding burst/control packet to its downstreams. Finally, *subcasting* is a means to multicast a lost burst to a subtree of the multicast tree, starting from a target switch.

The main idea of our protocol is as follows. An upstream edge router multicasts a burst with a control packet carrying a label for the multicast group ( $S, G$ ) and a sequence number. When a WDM switch receives a control packet for the multicast burst from its upstream, it extracts the label carried and determines downstreams to be forwarded. The control packet, (followed by the multicast burst), is forwarded to a downstream for which bandwidth (on a wavelength) is available for reservation by the control packet in the outgoing fibers. If the WDM switch is successful to forward the multicast burst to at least one of these downstreams and is a BS itself, the WDM switch creates a burst state for the multicast burst which maintains the status of the downstream/upstream (the upstream status is used to send an ACK upstream only once when there are multiple member edge routers at a BS) links for the transmitted burst of the multicast group ( $S, G$ ). After the BS receives ACK/NAK from its all downstreams for the burst, BS starts the recovery process if there is one or more NAKs. Although there are more than one NAK's, the BS will send only one request, and only one responder will retransmit the lost burst to the target switch (which could be the BS itself). The retransmitted burst will be multicast to downstreams of the target switch that have not sent an ACK based on the status of the burst state. The status of downstream/upstream links for the burst state is NULL, NAK or

ACK. A downstream link is in the NULL status if the WDM switch has not received either ACK or NAK after forwarding the multicast burst. The status of a blocked downstream link (on which no bandwidth is available for reservation at the BS or a NAK is received after forwarding the multicast burst) is set to NAK in the burst state. Note that if all downstream links are blocked (i.e., no wavelength available to any downstream link), the burst state is not created, and the WDM switch sends a NAK upstream. Since the burst loss recovery process can take place at every BS independently, the recovery of burst loss has an implicit hierarchical structure of responders, and it adapts to membership changes dynamically. In the next subsection, we describe detailed operations of the protocol.

## 3.2 BLRP Details

The previous subsection has described an overview of how the proposed protocol can achieve loss recovery. In this subsection, we present more detailed operations and some issues.

### 3.2.1 ACKs/NAKs

After a BS successfully forwards a received multicast burst to at least one of its downstreams along the multicast tree, it creates a state for the multicast burst, identified by  $\langle \text{burst id}, S, G \rangle$  where  $S$  and  $G$  are the multicast source address and group address, respectively. If the multicast burst is dropped at a WDM switch before reaching its destined downstreams, then the WDM switch (e.g., C1 in Fig. 3) generates a NAK and sends it upstream. The NAK contains the following information: the address of the WDM switch (label-switched router), the label associated with the multicast group and  $\langle \text{burst id}, S, G \rangle$ . A WDM switch that is not a BS simply forwards any NAK upstream (note that a MS is a BS if the MS is not a leaf WDM switch on the multicast tree). If a BS (e.g., BS2 in Fig. 3) receives the NAK from a downstream, it updates the status of the corresponding burst state (i.e., set the downstream link to NAK) and its related information. If the NAK is the first one received, the BS will store the address of the blocking WDM switch (e.g., C1 in Fig. 3) which generated the NAK and the label of the multicast group. This information will be used for retransmission (as to be described later).

Note that while a NAK can be sent by any non-BS switch which drops a burst on its outgoing link, only member edge routers can generate/send an ACK. More specifically, if a member edge router (e.g., E2 in Fig. 3) receives a multicast burst, it generates an ACK and sends it upstream with one of its fields, the hop count (HC) set to 0. Each WDM switch that is not BS will increment the HC by 1. After a BS (e.g., BS2 in Fig. 3) receives the ACK, it updates the status of the downstream link in the burst state. In addition, it stores the HC carried by the ACK, which is the number of hops to the member edge router that sent the ACK (e.g., E2 in Fig. 3). In addition, a BS switch (e.g., BS2 in Fig. 3) obtains the information on the number of hops to the closest upstream member edge router on the multicast tree by sending a “search packet” upstream at regular intervals. This hop count (from the search packet and the ACK) will be used for selecting an optimal responder to retransmission request (as described in the next subsection). Note that a downstream member edge router (or a closest upstream MS) that sends an ACK (or responds to the search packet) may be different as the membership changes. Hence, the hop count information maintained at the BS is adaptable to membership changes dynamically.

If a BS (e.g., BS1 in Fig. 3) is also a MS and receives an ACK from one of its attached member edge routers, it will send an ACK upstream immediately (if no ACK has been sent), and afterwards, set the upstream link to the ACK status (this will prevent sending multiple ACKs upstream if the BS has more than one member edge routers). The HC of the ACK will be set to 1. In addition, if the BS is not an MS, but receives ACKs from all its downstream links and those are all ACKs, it will send an ACK upstream. In this case, the HC of the ACK sent upstream is the minimum among those carried by all the ACKs. After sending the ACK upstream, the corresponding burst state is deleted.

### 3.2.2 Retransmission Request

After a BS receives ACKs/NAKs for a multicast burst from all its downstream links, the BS performs the following.

1. Check to see if there is any NAK. If not, it generates an ACK and sends it upstream (if the upstream status of the burst state is not set to ACK). The hop count of the ACK is set to the number of hops to the closest member edge router that has sent the ACK.
2. Otherwise (there are some NAKs), the BS selects the closest responder to forward the retransmission request, i.e., either upstream or one of its downstreams that have sent ACKs based on the number of hops.

3. Afterwards, the BS determines the target switch for sending an encapsulated control packet which is similar to IP-within-IP [18] based on which the target switch does subcasting. This decision is based on how many NAKs have been received from its downstream links. If that number (recorded in the burst state) is only one, the target switch is the downstream node that has sent the NAK. Otherwise, the target switch is the BS itself. In either case, the retransmission request carries the label to be used for the multicast group, the address of the BS and the address of the target switch. After sending a request, the BS sets a timer. If no acknowledgment for the request is received within the time-out period, it sends the request again until an ACK is received (with binary or random back-off period).

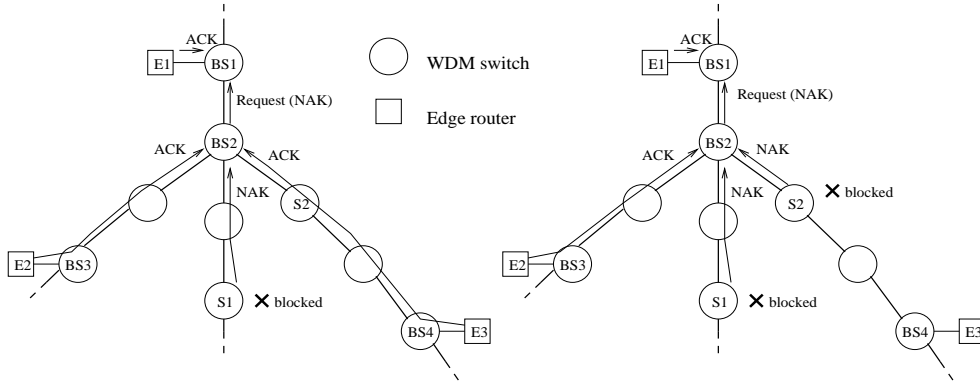


Figure 4: Target switch and retransmission request: (a) Single NAK (left) (b) Multiple NAKs (right)

As an example, let us consider Fig. 4. In Fig. 4 (left), BS2 has three downstream links for the multicast group (S, G). Assume that it has received two ACKs (from E2 and E3) and one NAK (from S1). In this case, BS2 cannot send an ACK upstream since it has a NAK from one of its downstreams. Instead, BS2 compares the number of hops to upstream switch E1, and downstream switches E2 and E3, and selects the closest member edge router. Note that if there are multiple member edge routers with the same minimum number of hops, and one of them is upstream, the BS sends the request upstream. Since the number of NAKs is only one in this case, the target switch is S1. On the other hand, when BS2 received two NAKs from its downstream links as in Fig. 4 (right), the target switch will be BS2 itself. Note that in Fig. 4, the request for retransmission is sent to upstream, as opposed to one of the downstreams as shown in Fig. 3.

By deciding where to request retransmission dynamically makes the protocol robust (in the presence of a single point of failure), and has an advantage of being able to distribute the multicast traffic over the network. In addition, by requesting the retransmission from a closest member edge router, the retransmitted burst will have a smaller blocking probability.

### 3.2.3 Examples

If a member edge router for the multicast group (S, G) receives a retransmission request for a multicast burst from its attached WDM switch (i.e., label-switched router), it performs the following.

1. First, the member edge router checks to see if it has the requested burst. If so, it sends an ACK for the request to the requester. Otherwise, it sends a NAK to the requester so that the sender can send another request either upstream or to other downstreams which have sent ACKs.
2. If the requested burst is found in the queue, the member edge router creates a copy of the multicast burst and generates a control packet that contains a stack of labels, i.e., a unicast label on top of a multicast label (extracted from the request packet). The number of hops to the target switch (from the unicast routing table) is set to the TTL value of the control packet generated (the TTL value decrements at every hop by 1). Then the member edge router sends the control packet and its associated multicast burst via unicasting to the target switch.
3. If the retransmitted burst reaches the target switch (the TTL value should be 0 at the target switch), the WDM switch pops up the label stack to use the multicast label for (S, G), and multicasts it to downstream links whose status is either

NAK or NULL. Otherwise, (if the retransmission is blocked at an intermediate node before reaching the target switch), a NAK is sent to the unicast sender (responder).

The subcasting and burst state at the target switch are essential elements to deliver the retransmitted burst only to a downstream that did not send an ACK for the multicast burst. The subcasting consists of two parts: a unicast from the responder to the target switch, and a subsequent multicast from the target switch to its downstreams. Let us consider Fig. 5 as an example for subcasting. Assume that node T has received one ACK (from S3) and two NAKs (from S4 and S5). After receiving the retransmission request, E1 makes a copy of the requested burst and creates its associated control packet. The control packet contains a label stack consisting of a unicast label (used for unicast to the target switch) and a multicast label (used for multicast from the target switch), and some other information such as the target switch, sender and  $\langle S, G, \text{burst id} \rangle$ . If the retransmitted burst is blocked at an intermediate node (e.g., S1 or S2 in Fig. 5), the WDM switch will send a NAK to the sender (i.e., E1), and E1 will transmit the burst again. Once the multicast burst is delivered to the target switch (T) via unicast (the TTL value of the control packet should be 0), the target switch pops up the label stack and extracts the multicast label at T for (S, G). Then T checks the burst state and forwards the retransmitted burst to S4 and S5 only. Afterwards, if both links to S4 and S5 are blocked, then T will send a NAK to the sender (E1). However, if at least one retransmission (to either S4 or S5) is successful, T does not send a NAK to the sender (E1). Note that if the target switch (e.g., C1 in Fig. 3) is not BS, there is no burst state, and hence the multicast burst is just transmitted to the only downstream (BS4 in Fig. 3).

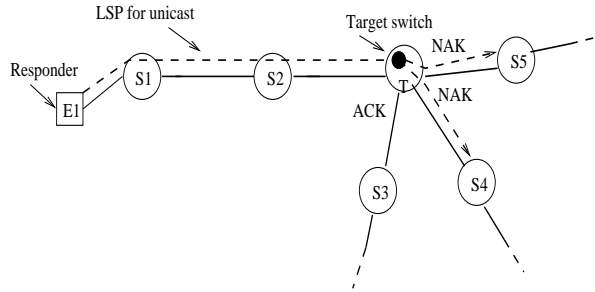


Figure 5: Retransmission and subcasting

### 3.2.4 Burst Stability at Edge Router

The source edge router (i.e., the ingress edge router to the OBS network) and other member edge routers on the multicast tree have to delete received multicast bursts after a certain interval. The source edge router can delete a received burst  $B_i$  after receiving an explicit ACK for the burst from its attached WDM switch. Other member edge (egress) routers on the multicast tree may need membership information for all other edge routers on the multicast tree to make sure that all other edge routers have received  $B_i$  so there will be no retransmission request for  $B_i$  from any one of them. Without that information, member edge routers (except the source edge router) may adopt a simple timeout mechanism with the timeout value set to several times of the maximum round-trip times in an OBS network. If a member edge router receives a retransmission request after timeout, it sends a NAK so that a requester (a BS) can try another member (upstream or downstreams). In the worst case, the recovery is done by the source host.

### 3.2.5 Scalability at BS

The proposed BLRP requires to maintain a burst state at all BSs along the multicast tree for each multicast burst. The burst state is created at a BS after the burst for (S, G) is successfully forwarded to at least one downstream link, and is deleted after the BS receives ACKs from all of its downstream links. Maintaining burst states at BS's represents an overhead involved in achieving reliable multicast at the WDM layer with the desirable properties stated in Section 2. Here we calculate an upper bound for the number of burst states at a BS.

Let the average burst length be  $B_L$  and the transmission speed be  $R$  (where  $R$  could be 2.5 Gbps or 10 Gbps). Accordingly, the transmission time  $T_{tr}$  for a burst is given by  $T_{tr} = \frac{B_L}{R}$ . Assume that  $T_{ack}$  is the average time for a BS to receive all ACKs

from its downstreams after creating the burst state. To derive an upper bound on the number of burst states, consider the worst case where a continuous stream of multicast bursts is arriving at an input fiber of a BS on a wavelength  $\lambda_i$ ,  $i=1, \dots, W$ , (i.e., there is no gap between bursts and in addition, there is no unicast traffic), and all bursts are successfully forwarded to at least one downstream. In such a case, the maximum number of burst states to be created at the BS during  $T_{ack}$  for bursts arriving on  $\lambda_i$  is calculated as  $\frac{T_{ack}}{T_{tr}}$ . If there are  $W$  wavelengths per fiber, and the nodal degree of the BS is  $D$  (i.e.,  $D$  input fibers to BS), the maximum number of burst states  $N_{states}$  to be maintained at the BS at any given time is

$$N_{states} = D \cdot W \cdot \frac{T_{ack}}{T_{tr}}$$

Note that the upper bound calculated above is independent of the number of multicast groups that are active in the network.

As a numerical example, assume that there are 32 wavelengths per fiber and the nodal degree of a BS is 5. In addition, the average multicast burst length is 100 Kbytes and  $T_{ack}$  is 48 msec (which is about the round-trip propagation time across the US continent). Given that the transmission time for the multicast burst is 80  $\mu$ s when R is 10 Gbps, the maximum number of burst states is 96,000, which is quite manageable. Of course, the actual number of burst states created at the BS should be much smaller than the upper bound.

### 3.2.6 Improvements of Retransmission

In this subsection, we consider two possible ways to reduce the loss recovery time and the blocking probability of retransmitted burst: namely, early retransmissions and optimized routing of retransmitting bursts, each with its own disadvantage as a trade-off.

First, let us consider early retransmissions using the multicast tree in Fig. 6 (left) as an example. Normally, a BS waits for all downstreams to respond and then sends a retransmission request if there are some NAKs. To expedite the retransmission, if a BS (e.g. BS2 in Fig. 6) receives a NAK, it may request the retransmission without waiting for all other downstreams to respond, (it still requests the retransmission from a closest member edge router among its upstream and downstreams that have already sent ACKs.). For example, if BS2 is not successful to forward the multicast burst to a downstream (E2) on link D1. Then BS2 sets the status of D1 to NAK and sends a retransmission request (in this case NAK) upstream. After receiving the NAK from BS2, BS1 sends a retransmission request to E1, which will send the requested burst to BS2 for subcasting. When the retransmitted burst arrives at BS2, the status of its downstream links is as follows : (D1, D2, D3) = (NAK, NULL, NULL). Suppose that BS2 is successful to transmit the retransmitted burst to all downstreams (D1, D2, D3), and that a NAK is being forwarded towards BS2 from D2. In this case, the early retransmission may expedite the loss recovery for the downstream D2. On the other hand, BS3 may have sent an ACK upstream along D3 before receiving the retransmitted burst. Since the retransmission is no burst state for the burst at BS3 (because BS3 has deleted the burst state after sending the ACK for the burst upstream), the retransmitted burst may be treated as a new burst by BS3, and as a duplicate burst by the member edge router. In this case, early retransmission will waste bandwidth and incur additional processing overheads at the WDM switches.

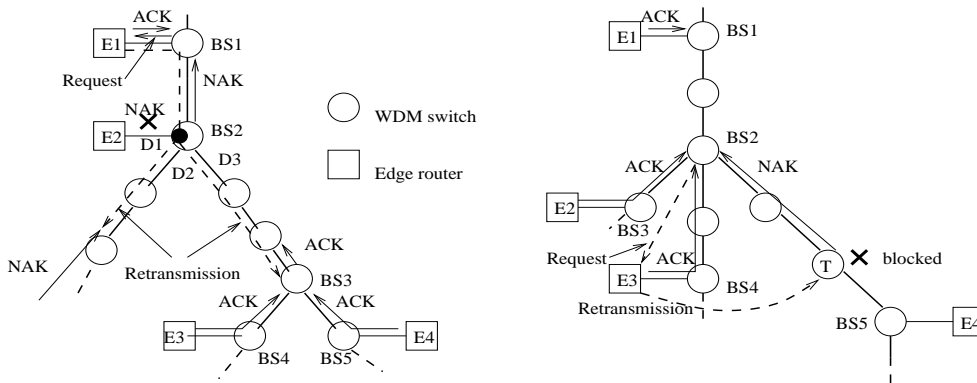


Figure 6: Early retransmission (left) and Optimized request (right).

Recall that in BLRP, we considered that an ACK packet carries only the number of hops from the member edge router that sent it. If the ACK packet carries the IP address of the member edge router that sent it, an optimized route for the retransmitted

bursts based on the actual distance (hop count) between the target switch and the responder of the retransmission request. For example, in Fig. 6 (right), assume that BS2 received two ACKs and one NAK. Then BS2 will decide the retransmission to E2 (which is the closest along the multicast tree). However, if the IP addresses of the member edge routers member edge routers that sent ACKs are available, BS2 can select the closest responder to the retransmission request by calculating the number of hops from each member edge router (that has sent an ACK, including upstream E1), i.e.,  $\min\{\text{hops}(E1 \rightarrow T), \text{hops}(E2 \rightarrow T), \text{hops}(E3 \rightarrow T)\}$  in the OBS network. Note that the target switch is T since there is only one NAK at BS2. If the number of hops from E3 to T is the minimum, then BS2 will send the retransmission request to E3 instead of E2 (which would be selected using the hop information only). However, in this case, BS2 has to store more information and more importantly, a routing table is needed at BS2 to calculate the number of hops between any two nodes.

### 3.2.7 Inter-operation with End-to-End Protocol

The BLRP is designed to work for burst loss recovery between edge routers in an OBS backbone network. In BLRP, the attached edge routers do not perform the operation of error (loss) detection in the received bursts. Therefore, end hosts are free to build their own loss recovery mechanism at the transport layer. Here we briefly describe the inter-operation of BLRP with end-to-end recovery protocol at the transport layer.

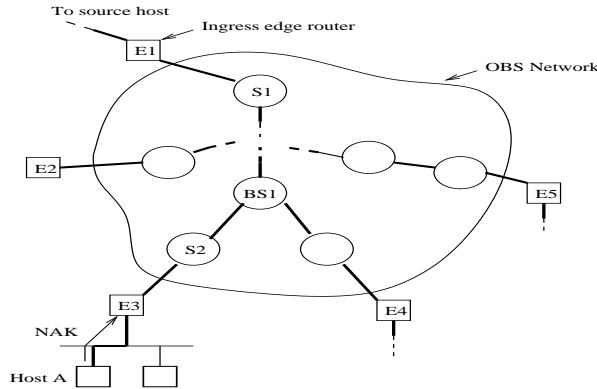


Figure 7: Inter-operation with end-to-end protocol.

Let us consider Fig. 7 as an example, where E1 is assumed to be the source edge router, and there are four member edge routers for a multicast group (S, G). Assume that Host A has detected a gap in the sequence numbers of bursts that it has received, as a result generates a NAK, which is sent towards the multicast source upstream (along the multicast tree). If edge router E3 receives the NAK, and it has the requested burst, it retransmit the burst to Host A. Otherwise, it forwards the NAK upstream (to S2). S2 will just forward the received NAK upstream (BS1) since it is not BS. When BS1 receives the NAK, it checks to see if the burst state is already set to the NAK, and if so, the NAK is ignored. If the status of the downstream is NULL, or the burst state is not found, the NAK is forwarded upstream. In the worst case, the NAK will be forwarded to source host.

## 4 Simulation Results

To verify the feasibility of the BLRP, simulations are carried out at the burst level. First, we describe the network model and assumptions and then the numerical results are given.

### 4.1 Network Model and Assumptions

The network model used consists of edge routers and WDM switches. Each edge router is an access point to a backbone network consisting of WDM switches. It is assumed that each WDM switch is connected to a number of edge routers  $E$  in the range  $1 \leq E \leq 3$  and the number of wavelengths on a WDM link is  $W$ . For simulation, a random backbone network is

generated (similar to [19]). Each of the  $N$  WDM switches in the backbone network is distributed across a Cartesian coordinate plane with coordinates between  $(0,0)$  and  $(2N, 2N)$ . In our simulation, the values of  $W$  and  $N$  are set to 8 and 40, respectively. The constraint for the minimum spacing between nodes is given by a Euclidean distance  $d_{min}$ . The edges are added to the random network by considering all possible pairs  $(x, y)$  of WDM switches and using the probability function

$$P(x, y) = \beta e^{-\frac{d_{x,y}}{\alpha L}},$$

where  $d_{x,y}$  is the Euclidean distance between the two WDM switches,  $L$  is the maximum possible distance between the two WDM switches, and  $\alpha$  and  $\beta$  are parameters in the range  $0 < \alpha, \beta < 1$ . To resemble a real network topology, the constraint that the minimum and maximum nodal degree of each WDM switch to other WDM switches are 2 and 6, respectively, are imposed when generating edges. A large value of  $\alpha$  increases the number of connections to nodes far away, while increasing  $\beta$  increases the number of edges from each WDM switch (to other WDM switches). After all edges are generated, the connectivity is checked to see if it meets the requirement for the nodal degree. The unit distance in the Cartesian coordinate plane is set to 60 Km in the generated random network. The distance between an edge router and its connected WDM switch is set to 10 Km in the simulation. After some experiments, the values for  $\alpha$  and  $\beta$  are set to 0.15 and 0.55, respectively.

To simulate contention at the WDM switch, we generate background unicast burst traffic according to a Poisson process. The average burst length  $B_{unicast}$  of unicast bursts is 500 Kbytes and the arrival rate  $\mu_{unicast}$  of the unicast bursts is 1600 bursts per second (which is 0.08 bursts during the transmission time of an average unicast burst per wavelength). It is assumed that the unicast burst traffic is uniformly distributed to all destination edge routers. On the other hand, the multicast burst traffic is also generated according to a Poisson process with the average burst length  $B_{multicast}$  that is set to 100 Kbytes. The arrival rate  $\mu_{multicast}$  of multicast bursts is 100 bursts per second, and the total number of multicast bursts generated during the simulation is 500. It is assumed that there is only one active multicast group  $(S, G)$  in the network during the simulation, and the multicast group has a pre-determined membership. The transmission speed is 10 Gbps. In the simulation, we consider two cases where the multicast source router is attached to a WDM switch along the perimeter of the network generated i.e., at the edge of the network topology and inside the network topology (i.e., close to the center), respectively. Note that a router attached to any WDM switch is called an edge router. Every edge router in the network has a probability  $p$  of being a member, so the average number of members (edge routers) for  $(S, G)$  is  $Np$  (excluding the multicast source). The multicast tree for  $(S, G)$  in the simulation is constructed by sending an explicit join message via a shortest path from each member edge router towards the multicast source.

## 4.2 Numerical Results

In this subsection, we present the numerical results on the proposed protocol. As to be shown, the location of the multicast source router (near the edge or the center of the OBS network) affects the performance such as the delivery latency and bandwidth consumption due to retransmission.

### 4.2.1 Delivery Latency

The delivery latency of a multicast burst is measured from the time the multicast burst is transmitted to the time the multicast source router receives ACKs from all member edge routers. For the purpose of performance measure, each member edge router sends an ACK (which is different from an ACK that is forwarded upstream along the multicast tree in BLRP) to the multicast source router via unicast. Fig. 8 (left) shows the average delivery latency as the group size increases from 10 percent to 100 percent. As the group size increases, the average delivery latency decreases since burst loss can be recovered from a closer responder. As can be seen, the delivery latency is shorter when the multicast source is located close to the center of the backbone network. Fig. 8 (right) shows the effect of the unicast traffic load in the network. As the traffic load increases in the network, more contention occurs at each WDM switch. Hence the delivery latency increases as the unicast burst traffic increases from 0 bursts to 1900 bursts generated per second.

### 4.2.2 Retransmission Bandwidth

We measured the bandwidth consumption due to retransmission and the number of retransmissions needed for a burst on average. The bandwidth consumption is measured by the number of times retransmitted multicast bursts will be forwarded. Fig. 9 (right) shows the bandwidth consumed due to retransmission as the group size increases. The number of forwards for

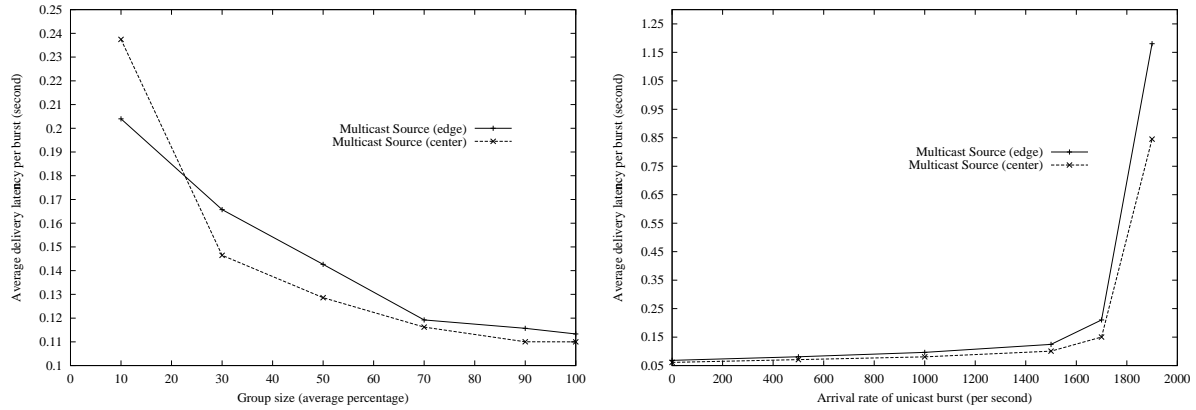


Figure 8: Average delivery latency: group size (left) and unicast traffic load (right).

retransmission reduces as the group size increases. This is because as there are more member edge routers in the multicast group, a closer member edge router can serve as a responder which results in a smaller hop count to a target switch.

Fig. 9 (left) shows the average number of retransmissions per member edge router in the multicast group. As the group size increases, the number of retransmissions decreases drastically for the same reason cited above, and the fact that the closer responder is, the better the chance the retransmission will be successful (since the burst loss probability is proportional to the number of hops the burst must travel).

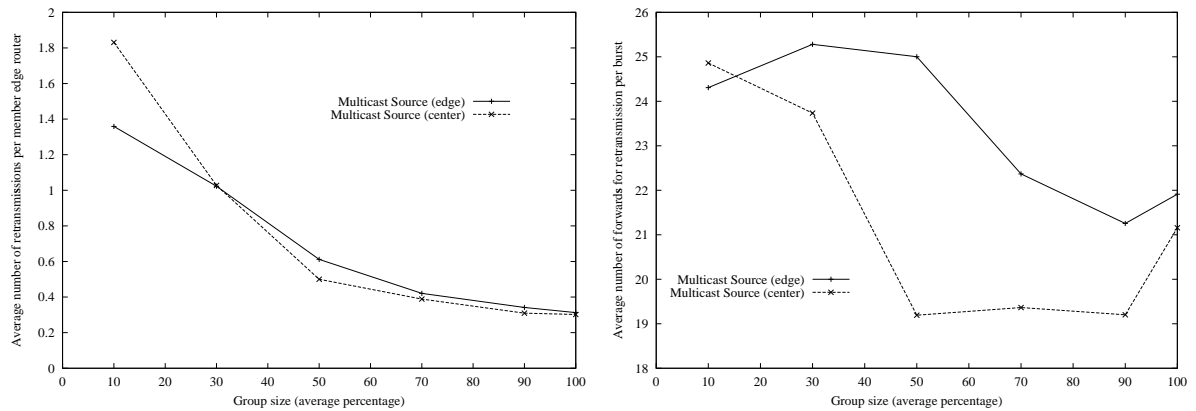


Figure 9: Bandwidth consumption (left) and the average number of retransmissions per member edge router (right).

### 4.2.3 Burst State Holding Time

The burst state holding time at a BS is measured from the time a burst state is created to the time the burst state is deleted after the BS receives all ACKs from its downstreams on the multicast tree. Fig. 10 shows the burst state holding time at a BS as the group size increases. We can observe that for a given traffic load in the network, the burst holding time decreases with the number of member edge routers in the multicast group. This is because the loss recovery can be achieved faster when a closer responder can be found. As an example, the average burst holding time at the BS is 9.89 msec when all edge routers are members and the multicast source is located close to the center of the network topology. Note that the state holding time is longer when the multicast source is located at the perimeter of the backbone network.

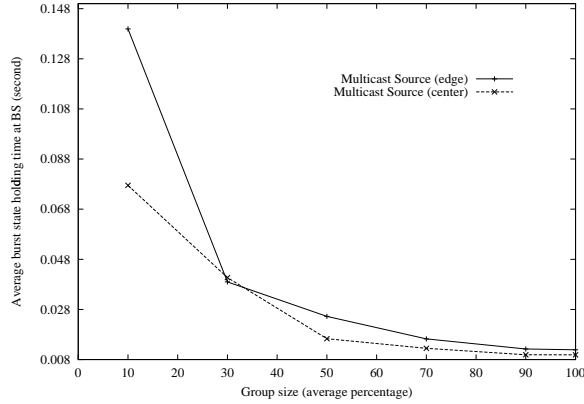


Figure 10: Burst state holding time at a BS.

## 5 Related Work

Several representative solutions for reliable multicast at the transport layer have been proposed in the literature to recover from data loss. In SRM [20], loss recovery is based on two mechanisms: back-off timers and duplicate request suppression. When receiver(s) detect loss (e.g., by detecting a gap in sequence numbers of packets received), they wait for a random time (determined by their distance from the original source of data). If the request timer expires, then the receiver multicasts a request for the missing data to the entire group. When other receivers that are also having the same loss hear that request, they suppress their own request (this prevents a request implosion.). Any host that has a copy of the request data can reply a request. It will set a repair timer to a random value and multicasts the repair message when the timer goes off. When other hosts that had the request data and scheduled repairs hear the multicast from the first host, they will cancel their repair timers (this prevents a response implosion.). Those approaches described above are global mechanisms to solve local loss problem. To achieve local recovery, SRM limits the scope of requests and repair messages by setting an appropriate TTL values in the TTL field of the IP header. Even with limited scope of requests/repair messages in SRM, SRM will have a large number of duplicate retransmission to unwanted receivers (which results in bandwidth consumption) and the timer operations are quite complex.

In RMTP [21] and TMTP [22], local recovery is achieved by using a hierarchical structure (or tree-based approach) to recover data loss. All receivers are explicitly organized into a hierarchical recovery tree at the transport layer. Each subtree (or domain) has one designated receiver (called DR or domain manager). Each DR is responsible for error recovery for its own subtree (or domain). However, RMTP provides only a static hierarchy while TMTP supports a dynamic hierarchy of domain managers. These approaches incur a high signaling overhead to construct a recovery tree. In contrast, our scheme supports a dynamic (implicit) configuration of responders supported by signaling packets (for downstream with no extra overhead) and search packet (for upstream). RMTP applies subcasting from DR to the subtree of the multicast tree rooted at the sender. In TMTP, to limit the scope of the retransmission request and the retransmitted data TMTP uses the TTL to restrict the transmission radius of the message. These approaches still have a substantial amount of duplicate delivery of the same data to receivers that have already received it.

Other related approaches include LMS [23], Search Party [24] and OTERS [25]. LMS uses the multicast tree itself to send request and retransmit the missing data (which is similar to our approach). Each router in a multicast tree selects one of its downstream links as its replier link. A request packet received from a non-replier link is forwarded to the replier link, while a request packet received from the replier link is forwarded upstream (parent). As all other routers on the multicast tree repeat the same process when a receiver detects loss in packets, LMS constructs an implicit hierarchy for loss recovery, (rather than an explicit (separate) recovery tree as in RMTP and TMTP). Another concept used in LMS is “turning point” where turning point is the router that bounces an upward-moving request back downward. LMS also uses directed multicast (subcast) for retransmission where directed multicast forwards the retransmitted packet to a link at the turning point from which the request is received. Only one request is forwarded upstream. In Search Party, instead of using a simple deterministic forwarding as in LMS, requests are forwarded randomly called randomcast. This may provide robustness in the presence of a single point of failure, but may increase the retransmission latency. On the other hand, our scheme provides a controlled forwarding based on the state information maintained at BS’s. In addition, how a retransmission request is forwarded may differ from burst to burst

according to the ACKs received for each burst. In OTERS, it uses multicast tree backtracing (based on a function provided by IGMP traceroute (mtrace)) and subcasting to provide local recovery. Using backtracing, OTERS determines subroot of a subtree so that a hierarchical recovery tree can be constructed. For each subtree, there is a designated receiver (DR), and DR retransmits the requested packet to the subroot of a subtree to which it belongs. All three approaches still result in duplicate transmissions to some receivers while our scheme has no duplicate transmission with only overhead being associated with maintaining burst states at BS's.

## 6 Conclusions

In this paper, we have presented a reliable WDM multicast protocol in OBS (optical burst-switched) networks. The proposed protocol operates at the WDM link layer and requires some additional functions to be performed at the WDM switch such as subcasting and maintaining burst states (at selected switches on the multicast tree). The overhead of maintaining the burst states is manageable (as described in Section 3.2.5), the proposed protocol can reduce the bandwidth consumption and processing overheads associated with duplicated retransmissions. This is a big advantage over other reliable protocols developed at the transport (or application) layer that incur a substantial amount of duplicate retransmissions. In addition, by operating at the WDM link layer, the proposed protocol provides early detection of burst dropping.

The proposed protocol also has a number of features. First, it does not incur implosion problem at the multicast source (i.e., ingress edge router on the multicast tree to the OBS backbone network) or at a responder. Only one request is sent upstream or to one of the downstream member edge routers, and only one member edge router responds to the request. The loss recovery is based on an implicit hierarchical structure of responders, and the recovery tree at the WDM link layer is dynamically adaptable to membership changes. In addition, the proposed protocol is very robust in the presence of a single point of failure since the retransmission request will be dynamically sent to a member edge router according to the received signaling packets. This also helps distribute the multicast traffic over the network.

## References

- [1] C. Brackett, "Dense Wavelength Division Multiplexing Networks: Principles and applications," *IEEE J. on Selected Areas in Communications*, pp. 373–380, Aug. 1990.
- [2] P. Green, *Fiber-Optic Communication Networks*. Prentice-hall, 1992.
- [3] R. Ramaswami, "Multi-wavelength Lightwave Networks for Computer Communication," *IEEE Communications Magazine*, no. 31, pp. 78–88, 1993.
- [4] F. Callegati, H. C. Cankaya, Y. Xiong and M. Vandenhoude, "Design Issues of Optical IP Routers for Internet Backbone Applications," *IEEE Communications Magazine*, pp. 124–128, Dec. 1999.
- [5] B. S. Arnaud, "Architectural and Engineering Issues for Building an Optical Internet," *Proc. of SPIE, All-optical Networking*, vol. 3531, pp. 358–377, Nov. 1998.
- [6] C. Qiao and M. Yoo, "Optical Burst Switching (OBS) - A New Paradigm for an Optical Internet," *J. of High Speed Networks*, vol. 8, no. 1, pp. 69–84, 1999.
- [7] Eytan Modiano, "Random algorithms for scheduling multicast traffic in WDM broadcast-and-select networks," *IEEE/ACM Trans. on Networking*, vol. 7, no. 3, pp. 425–434, 1999.
- [8] Zeydy Ortiz, George N. Rouskas and Harry G. Perros, "Maximizing Multicast Throughput in WDM Networks with Tuning Latencies Using the Virtual Receiver Concept," *European Trans. on Telecommunications*, vol. 11, pp. 63–72, Jan.-Feb. 2000.
- [9] Zeydy Ortiz, George N. Rouskas and Harry G. Perros, "Scheduling Combined Unicast and Multicast Traffic Broadcast WDM Networks," *J. of Photonic Network Communications*, vol. 2, pp. 135–154, May 2000.
- [10] C. Qiao, M. Jeong, A. Guha, X. Zhang and J. Wei, "WDM Multicasting In IP over WDM Networks," *Proc. of IEEE ICNP '99 Proceedings*, pp. 89–96, Nov. 1999.

- [11] R. Malli, X. Zhang and C. Qiao, "Benefits of Multicasting In All-Optical Networks," *Proc. of SPIE, All-Optical Networking*, pp. 209–220, Nov. 1998.
- [12] G. Sahin and M. Azizoglu, "Multicasting Routing and Wavelength Assignment in Wide-Area Networks," *Proc. of SPIE, All-Optical Networking*, vol. 3531, pp. 196–208, Nov. 1998.
- [13] L. H. Sahasrabudde and B. Mukherjee, "Light-Trees: Optical Multicasting for Improved Performance in Wavelength-Routed Networks," *IEEE Communications Magazine*, vol. 3, pp. 67–73, Feb. 1999.
- [14] X. Zhang, J. Wei and C. Qiao, "Constrained Multicast Routing in WDM Networks with Sparse Light Splitting," *Proc. of IEEE INFOCOM 2000*, vol. 3, pp. 1781–1790, Mar. 2000.
- [15] M. Yoo, M. Jeong and C. Qiao, "High-Speed Protocol for Bursty Traffic in Optical Networks," *Proc. of SPIE, All-Optical Communication Systems*, vol. 3230, pp. 79–90, Nov. 1997.
- [16] X. Zhang, J. Wei and C. Qiao, "On Fundamental Issues in IP Over WDM Multicasting," *Proc. of IEEE Int'l Conf on Comp. Comm. and Networks (IC3N)*, pp. 84–90, Oct. 1999.
- [17] R. Callon, P. Doolan, N. Feldman, A. Fredette, G. Swallow and A. Viswanathan, "A Framework for MPLS," *IETF draft, internet-drafts/draft-ietf-mpls-framework-05.txt*, Mar. 2000.
- [18] C. Perkins, "IP Encapsulation within IP," *RFC 2003*, Oct. 1996.
- [19] M. Doar and I. Leslie, "How Bad is Naive Multicast Routing," *Proc. of IEEE INFOCOM '93*, vol. 1, pp. 82–89, Apr. 1993.
- [20] S. Floyd, V. Jacobson, S. McCanne, C. Liu and L. Zhang, "A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing," *Proc. of ACM SIGCOMM '95*, pp. 342–356, Sept. 1995.
- [21] J. C. Lin and S. Paul, "RMTP: A Reliable Multicast Transport Protocol," *Proc. of IEEE INFOCOM '96*, vol. 3, pp. 1414–1424, Apr. 1996.
- [22] R. Yavatkar, J. Griffioen and M. Sudan, "A Reliable Dissemination Protocol for Interactive Collaborative Applications," *Proc. of ACM Multimedia '95*, 1995.
- [23] C. Papadopoulos, G. Parulkar and G. Varghese, "An Error Control Scheme for Large-Scale Multicast Applications," *Proc. of IEEE INFOCOM '98*, vol. 3, pp. 1188–1196, Mar. 1998.
- [24] A. M. Costello and S. McCanne, "Search Party: Using Randomcast for Reliable Multicast with Local Recovery," *Proc. of IEEE INFOCOM '99*, Mar. 1999.
- [25] D. Li and D. R. Cheriton, "OTERS (On-Tree Efficient Recovery using Subcasting): A Reliable Multicast Protocol," *Proc. of IEEE ICNP '98*, pp. 237–245, Oct. 1998.