Constrained Multicast Routing in WDM Networks with Sparse Light Splitting

Xijun Zhang, John Y. Wei, *Member, IEEE*, and Chunming Qiao

***Abstract*—As wavelength division multiplexing (WDM) technology matures and multicast applications become increasingly popular, supporting multicast at the WDM layer becomes an important and yet challenging topic. In this paper, we study constrained multicast routing in WDM networks with sparse light splitting, i.e., where some switches are incapable of splitting light (or copying data in the optical domain) due to evolutional and/or economical reasons. Specifically, we propose four WDM multicast routing algorithms, namely, Re-route-to-Source, Re-route-to-Any, Member-First, and Member-Only. Given the network topology, multicast membership information, and light splitting capability of the switches, these algorithms construct a source-based multicast "light-forest" (consisting one or more multicast trees) for each multicast session. While the first two algorithms can build on a multicast tree constructed by IP (which does not take into consideration the splitting capability of the WDM switches), the last two algorithms attempt to address the joint problem of optimal multicast routing and sparse splitting in WDM networks. The performance of these algorithms are compared in terms of the average number of wavelengths used per forest (or multicast session), average number of branches involved (bandwidth) per forest as well as average number of hops encountered (delay) from a multicast source to a multicast member. The results obtained from this research should present new and exciting opportunities for further theoretical as well as experimental work.**

***Index Terms*—Internet protocol (IP), light forest, light splitting, multicast routing, wavelength division multiplexing (WDM).**

I. Introduction

As the internet traffic continues to increase exponentially, a wavelength division multiplexing (WDM) network with terabits per second bandwidth per fiber becomes a natural choice as a backbone in the next generation optical internet. Given that multicast (for one-to-many or many-to-many communications) is important and increasingly popular on the Internet, issues concerning supporting multicast in internet protocol (IP) over WDM networks need to be studied.

A. Multicast in IP over WDM Networks

There are several schemes for multicasting data in IP over WDM networks. As shown in Fig. 1(a), a source IP router (s), as well as each IP router on a multicast tree constructed by the IP layer, can make copies of a data packet and transmit a copy to each of its child (i.e., immediate downstream router). However, this requires O/E/O conversions of every data packet at every router on the multicast tree, which may be undesirable and inefficient because routers may be over-loaded, IP layer forwarding may introduce a long latency, and data will lose bit-rate and format transparency, and so on.

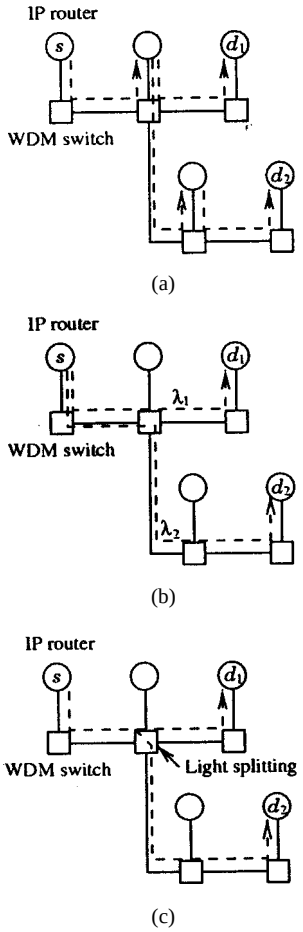

Fig. 1. Multicast in IP over WDM networks. (a) IP multicast. (b) PIP multicase via WDM unicast. (c) WDM multicast.

The above mentioned O/E/O conversions can be avoided by constructing a virtual topology consisting of a set of lightpaths (i.e., wavelength-routed paths) from the multicast source to each destination (which will be used interchangeably with the term "member" hereafter) as in Fig. 1(b). However, this is equivalent to having multiple unicasts (i.e., one-to-one connections) in a WDM network. In a previous study [11], we have found that using the second scheme, the network bandwidth consumed by

Manuscript received April 18, 2000; revised October 3, 2000. This work was supported in part by DARPA under Contract F30602-98-C-0202 and in part by the NSF under Contract ANIR-9801778. The work of X. Zhang was done while he worked at Telcordia (formerly Bellcore).

X. Zhang is with the Quantum Bridge Communications, Andover, MA 01810 USA (e-mail: xzhang@quantumbridge.com).

J. Y. Wei is with the Telcordia Technologies, Inc., Navesink Research and Engineering Center, Red Bank, NJ 07701 USA (e-mail: wei@research.telcordia.com).

C. Qiao is with the Department of Computer Science and Engineering, State University of New York at Buffalo, Buffalo, NY 14260 USA (e-mail: qiao@computer.org).

Publisher Item Identifier S 0733-8724(00)10995-8.

a large multicast session (i.e., containing many members) may become unacceptable.

In this paper, we focus on the third scheme [see Fig. 1(c)], where multicast is supported at the WDM layer by letting WDM switches make copies of data packets in the optical domain via *light splitting*. This scheme is more desirable since transmissions to different destinations can now share bandwidth on common links (resulting in significant bandwidth savings over the second scheme), while multicasting data all-optically. We assume that a multicast tree formed at the WDM layer (called a light-tree in [17]) uses a dedicated wavelength on each branch, or in other words, is wavelength-routed. Such a wavelength-routed light-tree is useful to support high-bandwidth multicast applications such as HDTV program distribution. Note that, as proposed in [21], an alternative is to establish a label switched path (LSP) for each branch of a light-tree, and use optical burst/label/packet switching (see [5], [16], [20] for example) to support multicast applications requiring low bandwidth or having bursty traffic.

In general, supporting multicast at the WDM layer has several potential advantages. First, with the knowledge of the physical (i.e., optical layer) topology, which may not be the same as that seen at the upper electronic (e.g., IP) layer, more efficient multicast routing is possible. Second, some optical switches are inherently capable of light splitting, which is more efficient than copying packets in electronics. Third, WDM multicast can alleviate the electronic processing bottleneck just as WDM unicast does. Last but not least, performing multicast optically provides consistent support of coding format and bit-rate transparency across both unicast and multicast. In fact, it makes little sense not to perform multicast in WDM while performing unicast in WDM.

B. Related Work

It has been shown that finding a minimum Steiner tree for a multicast session, whose members are only a subset of the nodes in a network with an arbitrary topology, is an NP-complete problem [8]. Accordingly, heuristics are often used to obtain a near-minimum cost multicast tree. Many multicast tree formation algorithms, which construct a source-based tree given the full knowledge of network topology and multicast session membership, have been proposed and their performance evaluated in the literature [2]–[4], [7], [9], [18]. These heuristic algorithms can roughly be classified into two categories. The first one contains algorithms based on the shortest path heuristic (SPH) which minimizes the cost of the path from a multicast source to each of the members, while the second one contains algorithms based on the minimum Steiner tree, which attempt to minimize the total cost of a multicast tree.

For WDM multicast, (optical) switches need to have the light splitting capability in order to be able to multicast (i.e., forward multiple copies of) data in the optical domain. Note that switches with the (light) splitting capability are usually more expensive to build than those without (see Figs. 2 and 3 and related discussion). Due to this and other (e.g., evolutional) reasons, one must consider the constraints on the splitting capability of the switches in a practical network. One of the constraints considered in this paper is *sparse splitting* [11], which means that only

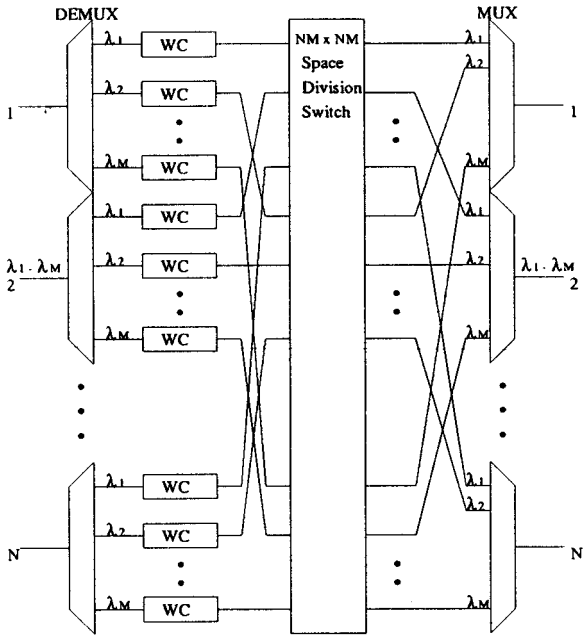

Fig. 2. An example architecture of multicast-incapable switches.

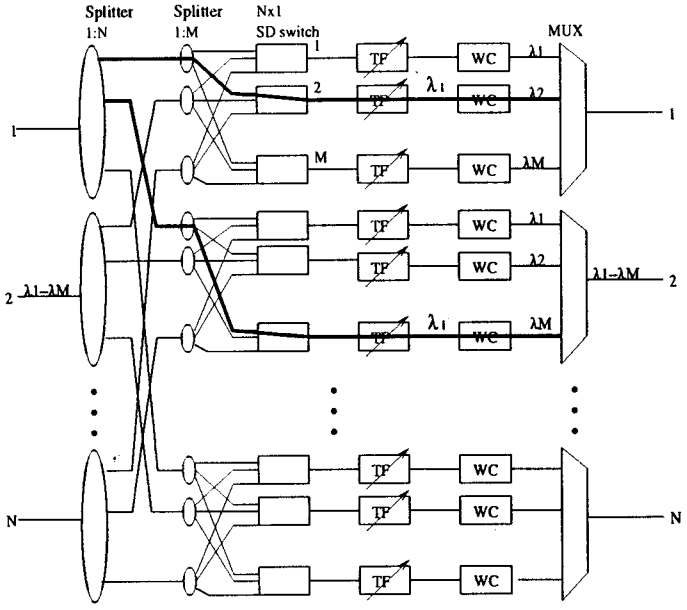

Fig. 3. An example architecture of multicast-capable switches.

a subset of the switches in a WDM network supports light splitting. Such a constraint invalidates the basic assumption made by previously proposed multicast tree formation algorithms in the literature (including [17]) that any node can be a branching point of a multicast tree and can have as many children as needed, or in other words, every node is *fully* multicast capable.[1] . Note that the problem of efficient multicast routing in a WDM network is already a complicated one due to the fact that one needs to consider wavelength assignment in a WDM network that may have

[1]In case some nodes are not fully multicast capable, no previous algorithms can ensure that all the multicast members in a session receive multicast data [1]

no, sparse, or limited wavelength conversion [13], [19]. Sparse splitting certainly makes the problem even more challenging.

In this paper, we study constrained multicast routing in WDM networks with sparse splitting (and sparse wavelength conversion). We propose that a new multicast medium called *light-forest*, consisting of one or more light-trees (rooted at a multicast source), be used to deliver multicast traffic to all intended destinations efficiently. Although a similar subject of supporting multicast when only a subset of nodes is multicast capable (MC) has been treated in IP multicast, the definition of "multicast incapable" (or MI) and MC, as well as the approaches taken in IP multicast are significantly different from those in WDM multicast. More specifically, in IP, MI means that a router, even though it can copy packets, does not run/understand the same multicast routing protocol as other routers (while in WDM, we assume that all the switches run the same multicast protocol/algorithm). Two solutions have been proposed in IP to deal with MI routers. One is to simply ignore MI routers (i.e., as if they do not exist) when constructing a multicast tree (as in MOSPF [12]), resulting in possible failures to deliver multicast traffic to *all* intended destinations. The other is to use IP-in-IP encapsulation or unicast tunneling to bypass MI routers as in MBone [10] (which implements DVMRP [14]). However, encapsulation in WDM networks implies that data needs to be processed (at the MC switches), and thus is not suitable for WDM multicast (based on wavelength-routing). In addition, although wavelength-routed paths may be established between MC switches to bypass all MI switches (just as tunnels are used to bypass MI routers), such an approach will still be inefficient in terms of bandwidth (or wavelength) usage, similar to the approach shown in Fig. 1(b) (IP multicast via WDM unicast). Intuitively, this is because in WDM multicast, it is no longer necessary to bypass an MI switch when constructing multicast trees (as long as the MI switch is not used as a branching node on a multicast tree). In fact, an MI switch can still be used as an intermediate node along a multicast tree to forward every incoming multicast traffic stream to one downstream switch (or destination).

The rest of this paper is organized as follows. In Section II, we describe the basic assumptions, and formally define the problem. We propose four new light-forest construction algorithms in Section III in sparse splitting WDM networks. Two of which modify multicast trees that have already been constructed (e.g., by IP) without taking into consideration the existence of MI switches, while the other two construct light-forests from scratch. In Section IV, we address how wavelengths are assigned in a light-forest and define the performance metrics we use when comparing the proposed algorithms. Simulation is described and performance results are presented in Section V under various assumptions (on the splitting capability, wavelength conversion capability, multicast session size, etc.). Finally, Section VI concludes this paper.

II. Constrained WDM Multicast Routing

As mentioned earlier, in a WDM network with sparse splitting, only a subset of the WDM switches (or nodes) has the multicast capability. We assume that every switch (even if it is MI) can support "drop and continue" as follows. It can be set to "drop only" (when the locally attached router is a destination, and there is no need to forward a copy to any downstream switch), "continue only" (when the locally attached router is not a destination and there is a downstream member) or "drop and continue" (when the locally attached router is a destination and there is a downstream member). This assumption is different from the assumption made in MBone where tunneling is used to span MI routers [10].

The rationale behind the "drop and continue" assumption is that, even at an MI switch, it is not difficult to tap a small amount of optical power from a wavelength channel for use by the local router while forwarding the data on that channel to an output. Alternately, one may use a wavelength add-drop multiplexer (WADM) which enables the local router to receive the data (through O/E conversion) and forward a copy (through E/O conversion). Note that, using a WADM may also allow a different wavelength to be used when forwarding the data. However, we will not consider such an approach in this study.

More formally, let $M(v)$ denote the splitting degree of switch v in terms of the number of copies v can forward to other switches (excluding the copy that may need to be dropped to the local router). Then, $M(v) = 1$ if switch v is MI, and $M(v) = N \cdot W$ if switch v is MC, where N is the number of neighboring switches that v has, and W is the number of wavelengths on each link between v and its neighbors. Note that although our work is based on the above assumption, we may extend it to cases where an MI switch is "drop *or* continue," or where switch v may have "limited" splitting capability, i.e., $1 < M(v) < N \cdot W$.

A. MI and MC Switch Architectures

Fig. 2 shows an example architecture of MI switches. It is assumed that the space-division (or SD) switching fabric is made of electro-optic directional couplers, and is thus multicast incapable (the "drop and continue" feature, which requires the use of a larger switching fabric, is not shown). In Fig. 2, each input WDM signal (or fiber) is demultiplexed first, and each channel may then be converted to a different wavelength using a wavelength converter (WC) in order to avoid conflicts. Each channel is then routed to a desired output port by an $NM \times NM$ SD switch. We will also consider switches, MI or MC, that do not have the wavelength conversion capability.

An example architecture of MC switches is shown in Fig. 3, where an input signal is split into NM signals, one for each $N \times 1$ SD switch through two stages of splitters. Each SD switch then selects one of the N input signals, out of which one wavelength is extracted using a tunable filter (TF). Wavelength conversion may then be performed to avoid conflicts. To support multicast, the same input signal needs to be selected by multiple SD switches that are connected to various outputs. For instance, it is shown in Fig. 3 how λ_1 of input port 1 multicasts to output 1 using λ_2 and output 2 using λ_M. Note that it is also possible to send multiple (up to M) "copies" to the same output using different wavelengths by letting multiple SD switches connected to the same output select the same input signal.

To compensate for the power loss due to splitting, power amplification/equalization is needed (though not shown in Fig. 3) in MC switches. Alternate architectures using for

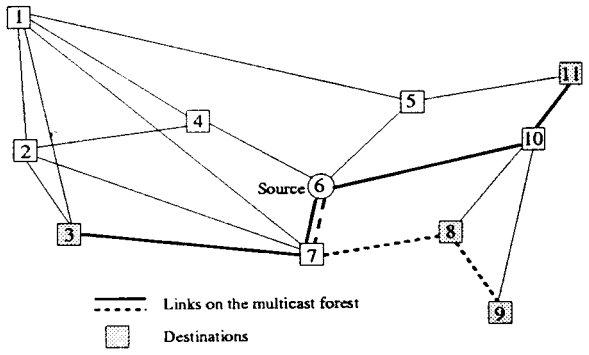

Fig. 4. An example of multicast forest in an 11-node random network with sparse splitting.

example, semiconductor optical amplifiers (SOAs) to turn each of the output (split) signals "on" and "off" may also be used. Because of their higher hardware/control complexity and/or dollar costs, MC switches will be, in general, more difficult and/or expensive to build than MI switches. This is, as mentioned earlier, one of the reasons why in practice we may have a sparse splitting WDM network.

Note that we will assume that the source of a multicast session has multiple transmitters (or a tunable one), and hence can transmit to as many children as needed when constructing a multicast tree (rooted at itself) even if the source switch is MI. Similarly, a source can transmit to its children on different wavelengths using different transmitters even if the source switch has no wavelength conversion capability.

B. Problem Description

A key observation is that, due to sparse splitting, a single light-tree may not be sufficient for multicasting data to all the destinations in a multicast session. An example is shown in Fig. 4, where in a random WDM network with 11 switches (or nodes hereafter), node 6 is the source of a multicast session, and nodes 3, 8, 9, and 11 are the destinations. It is assumed that none of the nodes, indicated by a square box, is MC (the source, being an MC node, is indicated by a circle). When using the shortest path heuristic, for example, to construct a light-tree, it is possible that nodes 7 and 10 are used to forward data to nodes 3 and 11, respectively. As a result, nodes 8 and 9 cannot be included in the light-tree (represented in solid lines). In this case, a second light-tree (dashed lines) which overlaps on link 6-7 with the first one has to be constructed, resulting in a "light-forest." Given that node 6 needs to send out two "copies" on link 6-7, two wavelengths (or branches) are needed on link 6-7 in a wavelength-routed network.

Note that, for the same multicast session, using different heuristics will likely result in light-forests with different costs in terms of the number of wavelengths (representing the amount of resources), total number of branches (representing the bandwidth consumed), and average number of hops from its source to a destination (representing the delay). In this paper, we propose light-forest construction algorithms for sparse-splitting networks, which construct a light-forest (consisting of one or more source-based light-trees) for a given multicast session so that multicast data can be delivered to all the members of the session. We also evaluate the performance of each proposed algorithm in terms of the costs associated with the forests it constructs.

In the following presentation, we assume that a pair of fibers is used to connect two nodes (i.e., switches), one for each direction. Accordingly, a WDM network can be represented as a *directed* graph (V, E), where V is a set of nodes (vertices), and E is a set of directed links (edges).[2] Given a graph, the multicast capability of each node in V, and a multicast session (s, D) where s is the source and $D = \{d_1, d_2, \ldots, d_n\} \subset V$ is the set of $n(= |D| \leq |V| - 1)$ destinations, each of the proposed forest construction algorithms will construct a forest, denoted by $F(s, D)$, on which an MI node does not need to multicast (i.e., split light). Such a forest consists of $t \geq 1$ source-based multicast trees $T_i(s, D_i)$ (without using any MI node as a branching point) such that $\bigcup_{i=1}^{t} D_i = D$ and $D_j \bigcap D_k = \phi$ for $1 \leq j \neq k \leq t$. Note that, although $T_i(s, D_i) \subset E$, $F(s, D) \not\subset E$ whenever $t > 1$. For example, as in Fig. 4 where $t = 2$, there are two copies of the link from node 6 to node 7 in $F(s, D)$ while only one exists in E.

III. New Multicast Forest Construction Algorithms

In this section, we describe four new light-forest construction algorithms, namely, Reroute-to-Source, Reroute-to-Any, Member-First, and Member-Only. In the following description, we use *hop-count* as the measure of path length (that is, a shortest-path is the one with a minimum number of hops), although other measures (such as geographical distance) may also be used. The performance of these algorithms will be compared in Section V.

A. Reroute-to-Source and Reroute-to-Any

A straight-forward way to construct a light-forest is to modify a multicast tree $T'(s, D)$, *where an MI node may be used as a branching point*, constructed using any existing algorithm (e.g., by pruning a spanning tree formed by Dijkstra's algorithm).

More specifically, the forest F can be obtained by checking every node (and in particular, branching node) on T' one by one in the breadth-first (or depth-first) order, and modifying tree T' accordingly. A node on tree T' is considered a *forest node* (or on the forest F being constructed) only after it has been checked to ensure that its splitting capability is not exceeded. More specifically, let $m(v)$ be the number of children that node v has on the tree T'. If $m(v) > 1$ and node v is MI, all but one downstream branches from node v will be cut (certain heuristics may be used to choose which branch to keep). Node v is now considered as a forest node. Each of the affected children can then "join" F at a forest node in one of two ways. In Reroute-to-Source, a cutoff child can join at an MC node j along the route from the source to v, including the source itself; In Re-route-to-Any, it can join at a forest node u along any route as long as u is either an MC or a *leaf* MI node (i.e., $m(u) = 0$).

Two examples of rerouting are shown in Fig. 5, where node v is assumed to be MI, and thus only one of the branches leading to its $m(v)$ children on tree T' (represented in thick solid lines),

[2]Hereafter, a directed link from node u to node v will be denoted by $e(u, v)$.

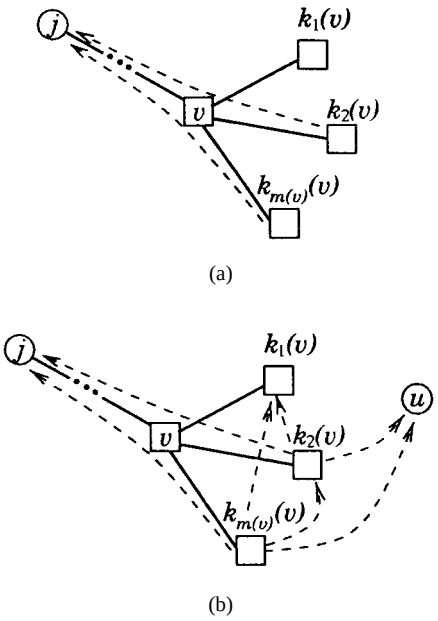

Fig. 5. Rerouting for the children of node v. (a) Reroute-to-Source. (b) Re-route-to-Any.

say $k_1(v)$ (kid 1 of node v), can be kept initially. The other nodes $k_2(v), \ldots, k_{m(v)}(v)$ will be cut and have to join the multicast forest by establishing $m(v)-1$ wavelength paths, one for each child, from an appropriate node (or nodes). In Reroute-to-Source, the algorithm traces, in the reverse direction, the route from the source s to node v used by tree T', denoted by $P(s,v)$, and finds the *first* MC node j. There are two rationales for using this algorithm, one being that in the worse case, source s can serve as node j, and the other being that should it become necessary to establish a wavelength path from s to a cut-off child, such a wavelength path will likely be on a shortest path (given that $P(s,v)$ was chosen to be a part of T' by an existing multicast tree formation algorithm).

Assume that node j is found and $j \neq s$, the cutoff children of v can join at node j as shown in Fig. 5(a), provided that j is capable of full wavelength conversion (see Fig. 3 for an example architecture), and there are at least $m(v)-1$ wavelengths available between j and v. When such is the case, only one (partial) wavelength path from s to j is needed, although $m(v)$ wavelength paths need to originate from j (and through v), one for each child of v. If, however, j is incapable of wavelength conversion, the cutoff children of v have to join at another (MC) node closer to the source s, and in the worse case, can join at source s. In either of these two cases, $m(v)$ wavelength paths will pass through node j, one for each child of v. We will not discuss the case where j is capable of only limited wavelength conversion but suffice it to say that some, instead of all, of the cutoff children may have to join at a node closer to source s (including s itself).

On the other hand, in Re-route-to-Any, the cutoff children of v can join at different nodes (e.g., a forest node u, or even $k_1(v)$) as shown in Fig. 5(b). This should facilitate load balancing as the wavelength paths to cutoff children can be established along different routes, thus reducing the number of wavelengths needed on each link. For example, assume $k_1(v)$ is a destination, then under the "drop-and-continue" model, even when both v and $k_1(v)$ are MI, one may establish a wavelength path (or rather extend the existing wavelength path) from $k_1(v)$ to $k_2(v)$ via v using the same wavelength on links $e(k_1(v),v)$ and $e(v,k_2(v))$ as that used on link $e(v,k_1(v))$. If any cutoff child $k_i(v)$ can choose from multiple nodes at which to join, the closest one (in terms of the path length from $k_i(v)$) may be selected. Detailed description of these two rerouting based algorithms is omitted, but suffice it to say that both algorithms have a polynomial-time complexity, and are amendable to distributed implementations in a way similar to that described in [15].

B. Member-First

In this subsection, we describe an algorithm whose aim is to construct a near-optimal light-forest from scratch, instead of based on an existing multicast tree. It combines the best of the shortest-path heuristics and the minimum Steiner tree-based heuristics, while taking into consideration the existence of MI switches.

More specifically, when every node has the splitting capability, and both the network topology and membership information are given, one may compute the shortest path from the source to every member, then eliminate common links among these shortest paths to obtain a shortest path tree. However, the distance among members is not considered and hence the total cost of the tree is usually not minimized. An alternative is to compute the minimum spanning tree to include all the nodes, and then prune the branches that do not lead to any member. Here, the membership information is not used during the spanning tree construction phase, and may also result in a multicast tree that consumes more bandwidth than necessary. The proposed Member-First algorithm considers both the membership information and the distance among members when constructing a forest F (or trees T_i). In addition, it avoids branching at nodes that do not have the splitting capability.

The basic idea of the Member-First algorithm is to construct a multicast forest one tree at a time, and each tree is constructed link by link as in Dijkstra's algorithm for constructing a spanning tree. However, it differs from the Dijkstra's algorithm as follows. First, the set of links being considered for possible inclusion in the current tree, denoted by L and called *fringe link* list, is organized as a *priority queue* where a link leading to a member has a higher priority than a link leading to a nonmember (when the two paths from the source s to the member and the nonmember have the same length). When expanding the tree, the link in L having the highest priority is used. Second, as soon as all the members are included, the Member-First algorithm stops expanding the tree and instead, starts pruning those branches that do not lead to any member. Last, perhaps the most important difference is that, immediately after expanding the tree by adding a link $e(v_1,u_1)$, if (and only if) u_1 is a member and v_1 is MI, the path $P(v_1,s)$ (which is the reverse path of $P(s,v_1)$ on the tree) is searched node by node until the first MC node on $P(v_1,s)$, say x, is reached (in the worst case, source s is reached). If the algorithm finds an MI node along the path, say y, all the links from y, except the one leading to u_1, are *cut* (so that they cannot be used to expand the current tree). If one of these cut links, say $e(y,z)$, is already on the tree, then a partial

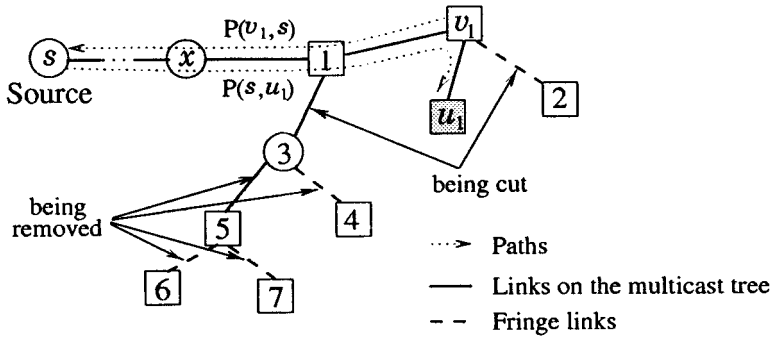

Fig. 6. Cut and remove branches in Member-First.

tree rooted at z is disconnected from the current tree, and all the links on the partial tree and the associated fringe links are *removed* (which may be used to expand the current tree later). At a certain point, either no fringe link is available in L, or all the members have been included in the existing tree(s). Note that if an additional tree needs to be constructed to include the remaining members, all the links in E (cut, removed, or used in the existing trees) may be used.

An example is shown in Fig. 6, where the solid and dashed lines denote the links on the tree T being constructed and the links in L (i.e., fringe links), respectively. It is assumed that link $e(v_1, u_1)$ has just been added to T, and u_1 is a member that has just been included on the forest (but neither node v_1, nor node 3 or 5 is). In addition, when searching along path $P(v_1, s)$, the first MC node is x, and neither node 1 nor v_1 is MC. Note that node 1 may have nodes v_1 and 3 as its children before $e(v_1, u_1)$ is added to T. However, after $e(v_1, u_1)$ is added to T, links $e(v_1, 2)$ and $e(1, 3)$ need to be cut since they can no longer be supported by nodes v_1 and 1, respectively. Consequently, the partial tree rooted at node 3 is disconnected, link $e(3, 5)$ needs to be removed from T, and links $e(3, 4), e(5, 6)$ and $e(5, 7)$ need to be removed from the fringe link set L. Note that, after u_1 is included on the tree, L will be updated by adding outgoing links from node u_1 (not shown in the figure). In addition, if node 3 becomes a tree node again later via a path from an MC node x or s (or some other MI/MC nodes), those removed links may be used again. A more detailed description of the algorithm, whose time complexity is also polynomial of the number of nodes in the network, is provided in the Appendix.

C. Member-Only

Similar to Member-First, the Member-Only heuristic builds a light-forest from scratch, one tree at a time. However, unlike Member-First, a multicast tree is constructed by including members one at a time (the closest member first) in Member-Only, and thus eliminates the need for pruning after all the members are included. The basic idea of Member-Only is similar to that of the shortest-path heuristic for constructing a near-minimum multicast tree [11] with the main feature being that, as long as an MI node y on a tree is a nonleaf node, other members will not join the tree at y. The detailed algorithm for Member-Only is given in the Appendix.

D. Comparison

Note that the four heuristics proposed above will likely construct different forests for the same multicast session. To help understand how they differ from each other, we use Fig. 7 as an example, where node 10 is assumed to be the source (and the only MC node), and nodes 1, 5, 6, 7, 8, 12, 13, 16, and 19 are members (i.e., destinations) in a 19-node random network.

- Re-route-to-Source: After a spanning tree has been constructed using Dijkstra's algorithm and pruned to remove branches that do not lead to any destinations, it is examined starting from the source (i.e., node 10). Since node 9 is MI but has two children, one of its children (node 8) is rerouted to the source node (via node 9). Similarly, node 6, 12, and 13 (children of node 11) are rerouted to the source via node 11.
- Re-route-to-Any: Similar to Re-route-to-Source, a pruned spanning tree is examined starting from the source. However, since a node can be rerouted to any other node on the tree, node 8 is rerouted to node 19, node 6 to node 1, node 12 to node 5, and (then) node 13 to node 12.
- Member-First: The multicast tree/forest is constructed link by link starting from the source considering members first. When node 9 becomes a node on the tree, links 9-7 and 9-8 are fringe links. After nodes 11, 15, and 17 are added to the tree, the fringe link list includes links 9-7, 9-8, 11-5, 11-6, 11-12, 11-13, 15-16, 17-18, and 17-19 in that order. Then, node 7 becomes a node on the multicast tree, link 9-8 is cut since node 9 is MI, and link 17-8 is added at the end of the fringe link list. Similarly, after node 5 is added to the tree, links 11-6, 11-12, and 11-13 are cut, and links 7-6, 5-12, and 15-13 are added at the end of the fringe link list in that order. This procedure continues until all the members have been added on the tree.
- Member-Only: The multicast tree/forest is constructed one member at a time. Node 5 becomes a node on the tree first since it has the shortest hop-count from the source (and among all the nodes that have the same hop-count, node 5 has the lowest node index). Then, node 12 becomes a node on the tree because it is only one hop away from node 5. The remaining members are added to the tree in the order of 13, 6, 7, 8, 19, 16, and 1.

IV. Wavelength Assignment and Performance Metrics

In this section, we first describe how wavelengths are assigned for a given source-based light-forest, assuming no, sparse and full wavelength conversion, respectively. Then, we define

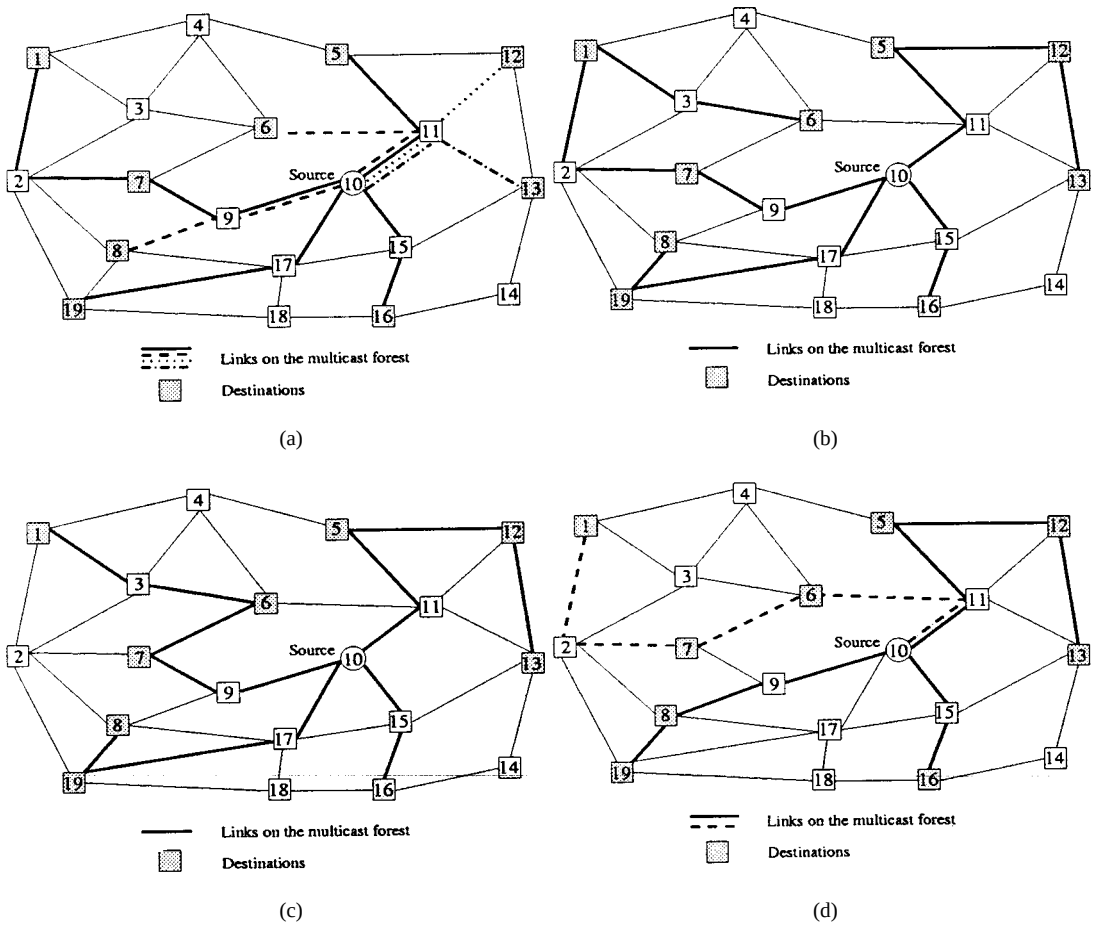

Fig. 7. Different multicast forests constructed using the proposed algorithms in a 19-node random network. (a) Re-route-to-Source. (b) Re-route-to-Any. (c) Member-First. (d) Member-Only.

the performance metrics we use when comparing the performance of these algorithms.

In the case of full wavelength conversion, any wavelength can be assigned on each link, while in the case of no wavelength conversion, the same wavelength has to be assigned to each subtree, which is a tree rooted at the source and contains one and only one child of the source. We call a collection of links on which the same wavelength has to be assigned a *segment*, which thus corresponds to a link in the case of full wavelength conversion and a subtree in the case of no wavelength conversion.

In the more general case of sparse wavelength conversion, a *segment* is determined as follows. In each subtree, we remove all the intermediate (i.e., nonleaf) nodes which have the wavelength conversion capability but keep the associated links. In this way, a subtree is partitioned into possible several segments, each of which requires the same wavelength to be assigned on all its links since there is no wavelength conversion capability within a segment. For example, in the multicast forest shown in Fig. 4, there are three subtrees (which contain leaf nodes 3, 9, and 11, respectively). If node 7 is the only node capable of wavelength conversion, then after it is removed, we will have two segments (6-7 and 7-3) from the first subtree, two segments (6-7, 7-8-9) from the second subtree, and only one segment (6-10-11) from the third subtree.

To facilitate performance comparison among the proposed forest-construction algorithms, we assume each link has a sufficient number of wavelengths to avoid blocking. In addition, the First-Fit algorithm [6] will be used (although other heuristics may also be used) to perform wavelength assignment after the light-forests are constructed and partitioned into segments. We will determine the maximum number of wavelengths needed by a given forest (over all the links), and then use the average maximum number of wavelengths needed per forest (over many forests and simulation runs), denoted by W, which represents the amount of network resources required per forest, as the first performance metric.

In wavelength-routed WDM networks, one wavelength channel (or a unit of bandwidth) needs to be reserved on each branch of a light-forest. For simplicity, we assume that all wavelengths are equally expensive (or cheap), and in addition, the bandwidth consumed using a wavelength on different links is more or less the same as well. Accordingly, we will determine the (total) number of branches on a multicast forest, and then use the average number of branches per forest, denoted by B, which represents the average bandwidth consumed per forest, as the second performance metric.

Finally, for a given forest (and multicast session), we will determine the average number of hops from the multicast source to a destination (over all the destinations of the multicast session), and then use the value obtained by averaging over many forests and simulation runs, denoted by H, which represents the delay, as the third performance metric.

V. Simulation Study and Results

In this section, we compare the performance of the proposed light-forest construction algorithms. An 11-node random network is simulated along with three parameters. Let S and C be the average fraction (i.e., in the range of [0–1]) of nodes in the network that possesses the splitting capability and wavelength conversion capability, respectively. It is assumed that the nodes with the splitting and/or wavelength conversion capability are distributed independently and uniformly throughout the network. In addition, each multicast session has only one (randomly chosen) source (note that a many-to-many session can be treated as multiple one-to-many sessions). A parameter G, where $0 < G \leq 1$, is used to represent the average fraction of nodes that are destinations (or equivalently, the probability that a given node is a destination). Obviously, the larger the G, the more members in a multicast session.

Simulation Setup

1) Read in the network topology and related parameters, S, C and G;
2) Determine which node is multicast capable and which one has wavelength conversion;
3) Generate multicast sessions (i.e., determine the source and destinations for each session);
4) Construct a multicast forest for each session using the proposed algorithms;
5) Partition each multicast forest into segments, and then assign a wavelength to each segment;
6) Collect statistics and check for convergence, if converged, stop; otherwise, go to 3);

By default, we assume $G = 0.5, S = 0.2$, and $C = 0.2$. To show the individual effect of a parameter, we vary each of these three parameters (while fixing the other two) in our simulations to obtain corresponding sets of results. Each set of results contains the average maximum number of wavelengths (W), average bandwidth per forest (B) and average delay (H) using each of the four algorithms.

Fig. 8 shows the results when G (or the number of destinations per multicast session) varies. As can be seen, both W and B increase with G in the four algorithms. However, H behaves differently for different algorithms. Specifically, it remains flat for Reroute-to-Source. This is because destinations are uniformly distributed in the network, and a shortest path is always used in Reroute-to-Source. Increasing the number of destinations will not likely change the average distance from the source. On the other hand, H increases when other algorithms are used because more destinations imply that it is more likely to use nonshortest paths. For the comparison among the four heuristics algorithms, Member-Only and Member-First require the least (and almost the same) number of wavelengths, but Member-Only requires the least amount of bandwidth, while Member-First results in a shorter delay than Member-Only. Also, Re-route-to-Source results in the shortest delay but requires much more bandwidth than other algorithms. Re-route-to-Any results in moderate wavelength and bandwidth requirements as well as delay.

As shown in Fig. 9, when S (or the number of splitting capable nodes) increases, all three metrics (number of wavelengths, bandwidth, and delay) decrease. Re-route-to-Source is an interesting case, in which delay remains unchanged while the other two metrics decrease dramatically (the most among the four algorithms). However, as shown in Fig. 10, the wavelength conversion capability does not have an effect on the performance metrics as significant as the splitting capability. Specifically, when C (or the number of nodes capable of wavelength conversion) increases, only W decreases *slightly*, while the other two metrics remain unchanged. This is because the wavelength conversion capability does not notably affect how light-forests are constructed using the proposed algorithms except in the special cases discussed in Section III-A.

Overall, Re-route-to-Source results in the shortest delay, and is the simplest to implement. However, it requires the largest

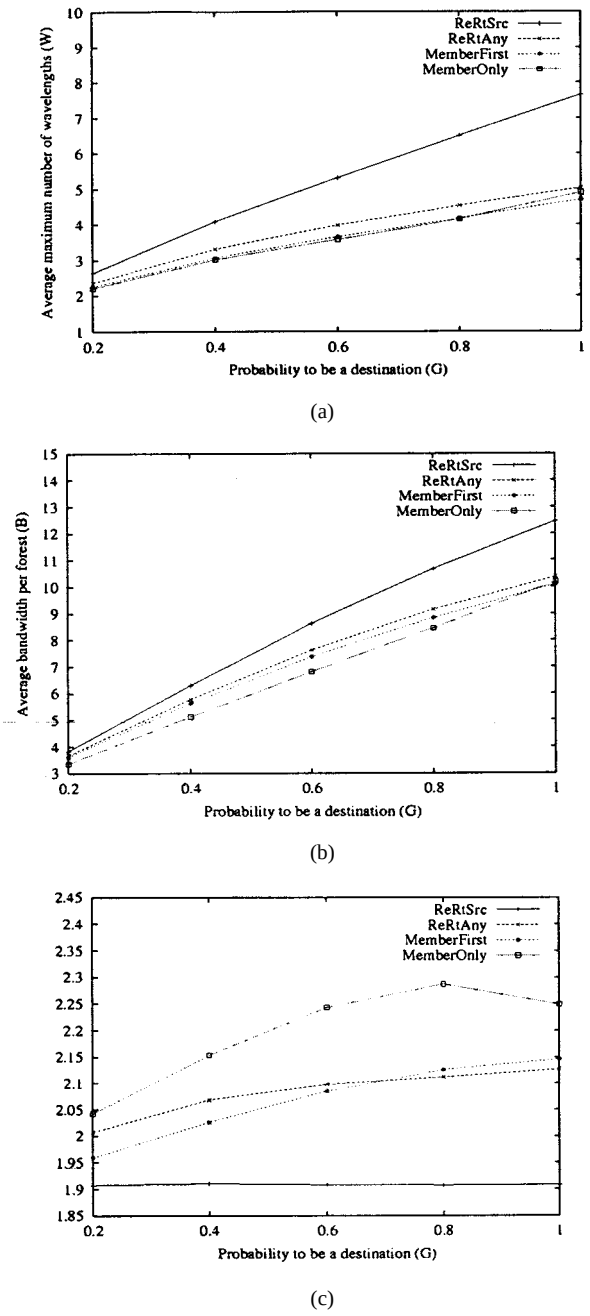

Fig. 8. The performance of multicast forest construction algorithms when G varies. (a) Average maximum number of wavelengths per forest. (b) Average bandwidths per forest. (c) Average delay.

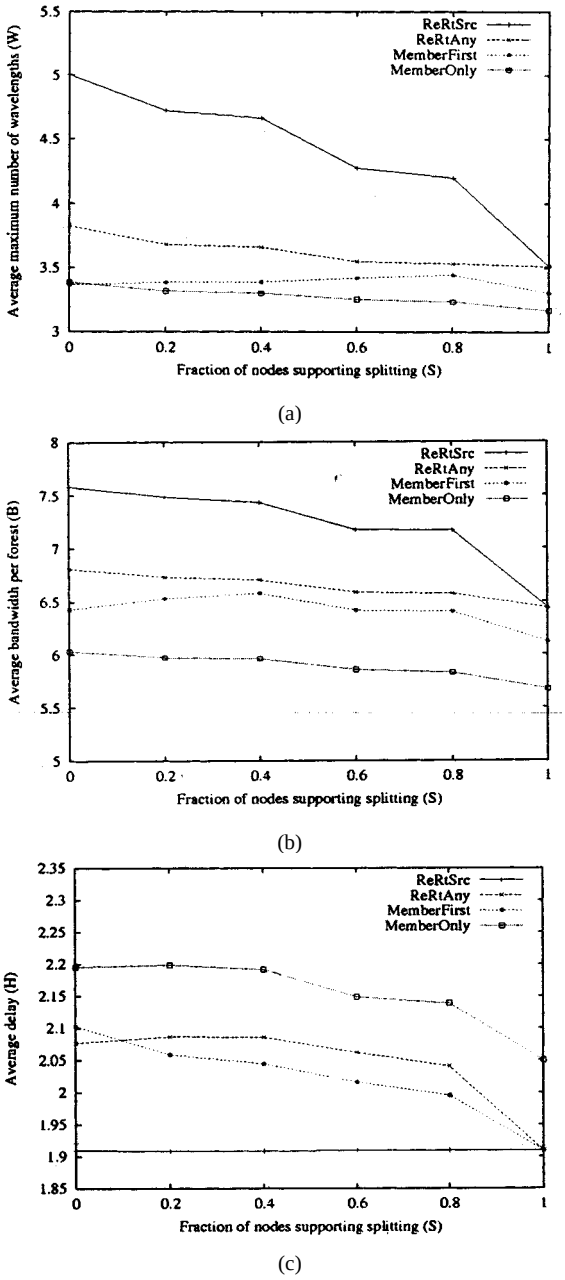

Fig. 9. The performance of multicast forest construction algorithms when S varies. (a) Average maximum number of wavelengths per forest. (b) Average bandwidths per forest. (c) Average delay.

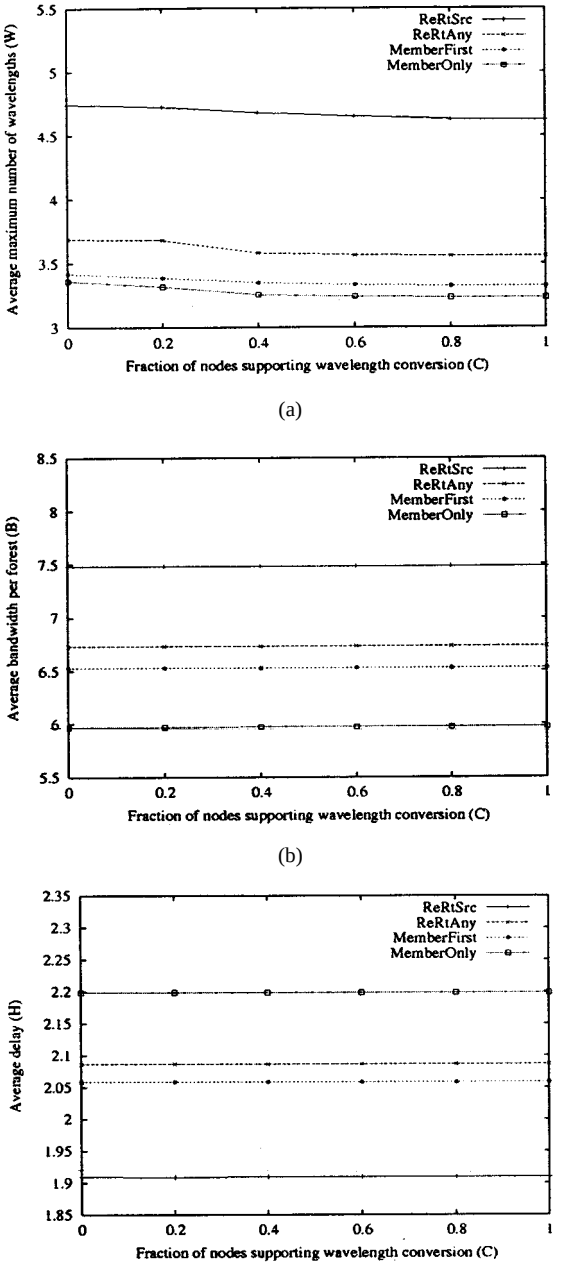

Fig. 10. The performance of multicast forest construction algorithms when C varies. (a) Average maximum number of wavelengths per forest. (b) Average bandwidths per forest. (c) Average delay.

amount of bandwidth and number of wavelengths. At the other extreme, Member-Only requires the least amount of bandwidth and number of wavelengths but results in the longest delay and has the greatest computational complexity (due to the need to compute all-pair shortest paths). We also note that Member-First requires almost the same number of wavelengths as Member-Only, results in a much lower delay, but requires a little more bandwidth than Member-Only. In addition, Member-First has a better overall performance than Re-route-to-Any, and hence, is the best choice if delay and bandwidth are to be balanced.

VI. Conclusion

Given the increasing popularity of multicast applications, and the inevitable evolution of WDM networks, supporting multicast in WDM networks will become an important subject. The constraints on the light splitting (or optical multicasting) capability of WDM switches invalidate assumptions made so far for electronic networks, and pose as a challenge for WDM multicast. In this paper, we have studied the problem of constrained multicast routing (and wavelength assignment) in wavelength-routed WDM networks, and proposed a new multicast medium called light-forest (consisting of one or more multicast trees) be used as a solution.

We have designed four light-forest construction algorithms, namely, Re-route-to-Source, Re-route-to-Any, Member-First, and Member-Only. These algorithms differ from all previously proposed multicast tree formation algorithms mainly in that

multicast traffic can now be delivered to all intended destinations (all-optically) using our algorithms even when some of the nodes in the network are not multicast capable (i.e., unable to split light). The performance of these algorithms has been compared in terms of the average maximum number of wavelengths per forest (amount of resources), average number of branches per forest (bandwidth), and average number of hops from a source to a destination (delay). We have found that: 1) Re-route-to-Source results in the shortest delay; 2) Member-Only requires the least bandwidth; 3) Member-First requires almost the same number of wavelengths as Member-Only, and achieves a slightly better trade-off between delay and bandwidth than Re-route-to-Any. We also note that although it has been implied that light-forests are constructed under centralized control, the proposed algorithms can also be used in a distributed way just as the Dijkstra's algorithm can be used in the IP multicast routing protocol MOSPF [12]. Finally, we note that a distributed protocol that can construct a light-forest without using the global knowledge of the network topology, multicast membership information, and light-splitting capability of the WDM switches, is also useful [15], and the work presented in this paper sheds light on the design of such distributed protocols based only on the local information.

APPENDIX
PSEUDO-CODE FOR MEMBER-ONLY AND MEMBER-FIRST

A. Member-Only

Let V_T be the set of nodes that are currently on a multicast tree and are either MC nodes or leaf MI nodes, and V_T' be the set of nonleaf MI nodes that are currently on the multicast tree (which cannot support any new branch). In addition, let D^* be the set of *members* that have not been included in the forest. The Member-Only algorithm is shown below.

Member-Only

(1) $F(s,D)=\phi, D^*=D$;
(2) $V_T=s, V_T'=\phi$, and $T=\phi$;
(3) try to find the shortest path $P(v,u)$, where $v\in V_T, u\in D^*$, which does not involve any node in V_T';
(4) if such a path $P(v,u)$ is found {
  add every link $e\in P(v,u)$ to the multicast tree T;
  if v (which just became a nonleaf node) is MI, move v from V_T to V_T';
  for any node y on $P(v,u)$, where $y\neq v$ and $y\neq u$ (y is a nonleaf node)
    if y is MI, move y to V_T';
    otherwise, move y to V_T;
  move u from D^* to V_T;
  if $D^*=\phi$, stop; otherwise, go back to step (3);
 }
 else (i.e., no such a path $P(v,u)$ can be found)
  move the branches in T to $F(s,D)$ and go to step (2) to construct another tree;

B. Member-First

Let V_T be the set of nodes that are currently on a multicast tree and are either MC nodes or leaf MI nodes, and V_T' be the set of nonleaf MI nodes that are currently on the multicast tree (which cannot support any new branch). In addition, let U_T be the set of remaining nodes that are not on the multicast tree, and $h(i)$ be the number of hops from source s to node i along a shortest path $P(s,i)$. The Member-First algorithm is shown below.

Member-First

(1) $F(s,D)=\phi, D^*=D$; // D^* is the set of members yet to be included;
(2) $V_T=\{s\}, V_T'=\phi, U_T=V-\{s\}$, and $T=\phi$; // T denotes the tree being constructed;
(3) $L=\phi$, UpdateFL (s); // initialize and update the fringe link set L;
(4) add the fringe link with the highest priority, say $e(v_1,u_1)\in L$, to T;
 if $u_1\in D^*$ {
  $D^*=D^*-\{u_1\}$;
  if v_1 is MI {
   trace along path $P(v_1,s)$ until a multicast capable node x is reached;
   for any MI node y on $P(x,v_1)$ (including v_1) {
    cut every branch/link $e(y,z)$ as long as z is not on $P(x,u_1)$;
    if z is on the tree (i.e., $e(y,z)$ is a branch in T),
     remove from T every branch (if any) in the partial tree rooted at z and
     from L every fringe link associated with the nodes on the partial tree;
   }
  }
 }
(5) UpdateFL (u_1); // update fringe link set L for u_1;
 also update V_T, V_T', U_T accordingly (e.g., move u_1 from U_T to V_T);
(6) if $D^*=\phi$, prune those branches that do not lead to any member, then stop;
 otherwise, go back to step (4) if $L\neq\phi$;
(7) if $L=\phi$ (and $D^*\neq\phi$), add the branches in T to $F(s,D)$, restore all removed and cut links, and go to step (2) to construct another tree;
UpdateFL (node v) { //update the fringe link set L;
 for every *uncut* link $e(v,u)$ {
  if u is not already on the tree and there is no $e(v',u)$ exists in L,
  or the existing $e(v',u)\in L$ has a lower priority
   add $e(v,u)$ to L (and remove $e(v',u)$ if any);
 }
}

REFERENCES

[1] F. Bauer and A. Varma, "Degree-constrained multicasting in point-to-point networks," in *INFOCOM'95*, Apr. 1995, pp. 369–376.

[2] J. Beasly, "An SST-based algorithm for the Steiner problem in graphs," *Networks*, vol. 19, pp. 1–16, 1989.
[3] L. Berry, "Graph theoretic models for multicast communications," in *Traffic Theories for New Telecommunications Services ITC Specialists Seminar*, Adelaide, Australia, Sept. 1989, pp. 95–99.
[4] K. Bharath-Kumar and J. M. Jaffe, "Routing to multiple destinations in computer networks," *IEEE Transactions Commun.*, vol. COM-31, no. 3, pp. 343–351, Mar. 1983.
[5] Y. Lin, W. Way, and G. K. Chang, "A novel optical label swapping technique using erasable optical single-sideband subcarrier labels," *IEEE Photon. Technol. Lett.*, vol. 12, pp. 1088–1090, Aug. 2000.
[6] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath communications: An approach to high-bandwidth optical WANs," *IEEE Trans. Commun.*, vol. 40, no. 6, pp. 1171–1182, July 1992.
[7] F. Hwang and D. Richards, "Steiner tree problems," *Networks*, vol. 22, pp. 55–89, 1992.
[8] R. Karp, Reducibility among combinatorial problems, in Complexity of Computer Computations, 1972.
[9] V. Kompella, J. Pasquale, and G. Polyzos, "Multicasting for multimedia applications," in *INFOCOM'92*, May 1992, pp. 2078–2085.
[10] V. Kumar, *MBone: Interactive Multimedia on the Internet*, IN: New Riders, 1996.
[11] R. Malli, X. Zhang, and C. Qiao, "Benefits of multicasting in all-optical networks," in *SPIE Proceedings, All Optical Networking*, Nov. 1998, pp. 209–220.
[12] J. Moy, Multicast extensions to OSPF, in IETF Document, Mar. 1994. RFC 1584.
[13] S. Subramaniam, M. Azizoglu, and A. K. Somani, "Connectivity and sparse wavelength conversion in wavelength-routing networks," in *INFOCOM'96*, 1996, pp. 148–155.
[14] T. Pusateri, Work in progress: Distance vector routing protocol, in IETF Draft, Sept. 1996. draft-ietf-idmr-dvmrp-v3-03.txt.
[15] C. Qiao, M. Jeong, A. Guha, X. Zhang, and J. Wei, "WDM multicasting in IP over WDM networks," in *Proc. Int. Conf. Network Protocols (ICNP)*, Nov. 1999, pp. 89–96.
[16] C. Qiao and M. Yoo, "Optical burst switching (OBS)—A new paradigm for an optical internet," *J. High Speed Networks (JHSN)*, vol. 8, no. 1, pp. 69–84, Jan. 1999.
[17] L. H. Sahasrabuddhe and B. Mukherjee, "Light-trees: Optical multicasting for improved performance in wavelength-routed networks," *IEEE Commun. Mag.*, vol. 37, no. 2, pp. 67–73, Feb. 1999.
[18] P. Winter, "Steiner problem in networks: A survey," *Networks*, vol. 17, no. 2, pp. 129–167, 1987.
[19] J. Yates *et al.*, "Limited-range wavelength translation in all-optical networks," in *Proc. INFOCOM*, 1996, pp. 954–961.
[20] M. Yoo and C. Qiao, "Just-enough-time (JET): A high speed protocol for bursty traffic in optical networks," in *Digest of IEEE/LEOS Summer Topical Meetings on Technologies for a Global Information Infrastructure*, Aug. 1997, pp. 26–27.
[21] X. Zhang, J. Wei, and C. Qiao, "On fundamental issues in IP over WDM multicast," in *Int. Conf. Computer Communications and Networks (IC3N)*, Oct. 1999, pp. 84–90.

Xijun Zhang received the B.S. degree from University of Science and Technology of China (USTC) in 1992, the M.S. degree from Institue of Automation, Chinese Academy of Sciences in 1995, and the Ph.D. degree from University at Buffalo (SUNY), Buffalo, NY, in 1999, all in electrical engineering.

Since summer 1998, he had worked as a cooperative at Alcatel Network Systems, Inc., Richardson, TX, and Telcordia Technologies, Inc. (formerly Bellcore), Red Bank, NJ, and worked as an employee at Lucent INS division, Westford, MA. Recently, he joined Quantum Bridge Communications, Andover, MA. In the past several years, he has published nearly 20 journal and conference papers. He has also served as a session chair at IC3N'99. His research interests include optical networks and internetworking with focus on network architecture, control and management, design and planning, network optimization and performance evaluation.

John Y. Wei (M'00) received the Ph.D. degree in computer science from the California Institute of Technology, Pasadena.

He is the Director of the Broadband Network Management Research Group at Telcordia Technologies, Red Bank, NJ. He is leading a research group in developing the next-generation network operations and management systems architecture and prototype for broad-band networks. His research group is responsible for the network management research tasks in the DARPA funded projects of ATDNet, MONET, and NGI, which focus on the network control and management issues of broad-band networks covering technologies from ATM, SONET, optical WDM, to IP over WDM. He was the task leader for the NC&M work in MONET, and the NGI Optical Label Switching Project, as well as the current Principal Investigator of the NGI SuperNet NC&M Project. He joined Bellcore (now Telcordia Technologies) in 1989 and has worked on network management, distributed systems, and fault-tolerant computing.

Dr. Wei is a member of the ACM.

Chunming Qiao received the Ph.D. degree in computer science from the University of Pittsburgh, PA, in 1993. Prior to that, he was admitted to the top-ranked University of Science and Technology (USTC) of China at age 15, and received the B.S. degree in computer engineering.

He is currently an tenured Professor at the Computer and Science Engineering Department. He is also an adjunct professor at the Electrical Engineering Department, and the director of the Lab for Advanced Network Design, Evaluation and Research (LANDER), which conducts cutting-edge research related to optical networks, wireless/mobile networks and the Internet. He has over 10 years of research experience in optical networks, covering the areas of photonic switching devices, WDM communications and optical packet-switching. He pioneered research on the next-generation Optical Internet, and in particular, optical burst switching (OBS). He has published more than six dozens of papers in leading technical journals and conferences, and given several invited talks and keynote speeches.

Dr. Qiao received the Andrew-Mellon Distinguished doctoral fellowship award from University of Pittsburgh. He is the Chair of the program section on IP over WDM at APOC'2001. He has served as the chair of the program on Optical Layer and Internetworking Technology in 2000, a Co-Chair for the annual All-optical Networking Conference since 1997, a Program Vice Co-Chair for the 1998 International Conference on Computer Communications and Networks (IC3N), expert panelists, technical program committee members, and session organizers/chairs in several other conferences and workshops. He is also the Chair of the recently established Technical Group on Optical Networks (TGON) sponsored by SPIE, the IEEE Communication Society's Editor-at-Large for optical networking, an editor of the *Journal on High-Speed Networks* (JHSN) and the new *Optical Networks Magazine*, and a member of IEEE Computer Society and IEEE Communications Society.