Multicast Routing
Under Optical Layer Constraints

Yufeng Xin
Advanced Networking Research
MCNC-RDI
RTP, NC, 27709
E-mail: yxin@anr.mcnc.org

George N. Rouskas
Department of Computer Science
North Carolina State University
Raleigh, NC 27695-7534
E-mail: rouskas@csc.ncsu.edu

***Abstract*— It has been widely recognized that physical layer impairments, including power losses, must be taken into account when routing optical connections in transparent networks. In this paper we study the problem of constructing light-trees under optical layer power budget constraints, with a focus on algorithms which can guarantee a certain level of quality for the signals received by the destination nodes. We define a new constrained light-tree routing problem by introducing a set of constraints on the source-destination paths to account for the power losses at the optical layer. We investigate a number of variants of this problem, we characterize their complexity, and we develop a suite of corresponding routing algorithms; one of the algorithms is appropriate for networks with sparse light splitting and/or limited splitting fanout. We find that, in order to guarantee an adequate signal quality and to scale to large destination sets, light-trees must be as balanced as possible. Numerical results demonstrate that existing algorithms tend to construct highly unbalanced trees, and are thus expected to perform poorly in an optical network setting. Our algorithms, on the other hand, are designed to construct balanced trees which, in addition to having good performance in terms of signal quality, they also ensure a certain degree of fairness among destination nodes. While we only consider power loss in this work, the algorithms we develop could be appropriately modified to account for other physical layer impairments, such as dispersion.**

I. Introduction

Over the last few years we have witnessed a wide deployment of point-to-point wavelength division multiplexing (WDM) transmission systems in the Internet infrastructure. The rapid advancement and evolution of optical technologies makes it possible to move beyond point-to-point WDM transmission systems to an all-optical backbone network that can take full advantage of the available bandwidth by eliminating the need for per-hop packet forwarding. Such a network will consist of a number of optical cross-connects (OXCs) arranged in some arbitrary topology, and its main function will be to provide interconnection to a number of client networks, e.g., IP subnetworks running Multi-Protocol Label Switching [21] (MPLS) . Each OXC can switch the optical signal coming in on a wavelength of an input fiber link to the same wavelength in an output fiber link. The OXC may also be equipped with converters that permit it to switch the optical signal on an incoming wavelength of an input fiber to some other wavelength on an output fiber link. The main mechanism of transport in such a network is the *lightpath*, an optical communication channel established over the network of OXCs which may span a number of fiber links (physical hops). If no wavelength converters are used, a lightpath is associated with the same wavelength on each hop. This is the well-known wavelength continuity constraint. Using converters, a different wavelength on each hop may be used to create a lightpath. Thus, a lightpath is an end-to-end optical connection established between two subnetworks attached to the optical backbone.

The concept of a lightpath can be generalized into that of a *light-tree* [24] which, like a lightpath, is a clear channel originating at a given source node and implemented with a single wavelength. But unlike a lightpath, a light-tree has multiple destination nodes, hence it is a point-to-multipoint channel. The physical links implementing a light-tree form a tree, rooted at the source node, rather than a path in the physical topology, hence the name. Light-trees may be implemented by employing optical devices known as *power splitters* at the OXCs. A power splitter has the ability to split an incoming signal, arriving at some wavelength λ, into up to m outgoing signals, $m \geq 2$; m is referred to as the *fanout* of the power splitter. Each of these m signals is then independently switched to a different output port of the OXC. Due to the splitting operation and associated losses, the optical signals resulting from the splitting of the original incoming signal must be amplified before leaving the OXC. Also, to ensure the quality of each outgoing signal, the maximum fanout m of the power splitter may have to be limited to a small integer. If the OXC is also capable of wavelength conversion, each of the m outgoing signals may be shifted, independently of the others, to a wavelength different than the incoming wavelength λ. Otherwise, all m outgoing signals will be on the same wavelength λ. Note that, just like with wavelength converter devices, incorporating power splitters within an OXC is expected to increase the network cost because of the large amount of power amplification and the difficulty of fabrication.

Light-trees have several applications [22] in optical networks, including:

- **Optical multicast.** An attractive feature of light-trees is the inherent capability for performing multicast in the

This work was supported by the NSF under grant ANI-9805016.

optical domain (as opposed to performing multicast at a higher layer, e.g., the network layer, which requires electro-optic conversion). Therefore, light-trees can be useful for transporting high-bandwidth, real-time applications such as high-definition TV (HDTV). We note that TV signals are currently carried over distribution networks having a tree-like *physical* topology; creating a *logical* tree topology (light-tree) over an arbitrary physical topology for the distribution of similar applications would be a natural next step. Because of the multicast property, we will refer to OXCs equipped with power splitters as *multicast-capable* OXCs (MC-OXCs).

- **Enhanced virtual connectivity.** In opaque networks, the virtual degree of connectivity of each node is not tied to the number of its interfaces: electronic routing creates the illusion that a node can reach any other node in the network. In transparent networks, on the other hand, the degree of connectivity of each client node (e.g., IP/MPLS router) connected to the optical core is limited by its physical degree, i.e., the number of its optical transceivers. A light-tree service would enable a client node to reach a large number of other client nodes independently of its physical degree, significantly enhancing the virtual connectivity of the network.
- **Traffic grooming.** Generalized MPLS (GMPLS) [3] makes it possible to tunnel a set of MPLS label-switched paths (LSPs) over a wavelength channel. Since switching at OXCs takes place at the granularity of a whole wavelength, a point-to-point lightpath allows the sharing of the wavelength bandwidth only between clients attached to the same ingress and egress OXCs. The light-tree concept offers a way to overcome this constraint, since it allows for the grooming and tunneling of a number of lower rate point-to-point LSPs to several destinations, regardless of the egress OXC to which these destinations attach.

In this paper we study the problem of light-tree routing in optical networks with light splitting capabilities. Since the effects of light splitting and power attenuation on optical signals are only partially mitigated by amplification, our focus is on algorithms which can guarantee a certain level of quality for the signals received by the destination nodes. The problem of signal quality does not arise in the context of multicast above the optical layer, and, to the best of our knowledge, it has not been directly addressed in the literature. We define a new constrained light-tree routing problem by introducing a set of constraints on the source-destination paths to account for the power losses at the optical layer. We investigate a number of variants of this problem, and we prove that they are all NP-complete. We also develop a suite of corresponding routing algorithms, one of which can be applied to networks with sparse light splitting and/or limited splitting fanout. One significant result of our study is that, in order to guarantee an adequate signal quality and to scale to large destination sets, light-trees must be balanced, or distance-weighted balanced (a term we define later). Numerical results demonstrate that existing algorithms tend to construct highly unbalanced trees, and are thus expected to perform poorly in an optical network setting. Our algorithms, on the other hand, are designed to construct balanced trees which, in addition to having good performance in terms of signal quality, they also ensure a certain degree of fairness among destination nodes.

The paper is organized as follows. In the remainder of this section we review related work on multicast routing, both in general and in the context of WDM networks. In Section II, we describe the multicast optical network under study, and we develop a model to account for optical signal losses in the network. In Section III we introduce the problem of constructing light-trees under constraints that ensure the quality of the optical signals received at destination nodes. We define three versions of the light-tree routing problem in Section IV, mainly differing on which type of power loss is the dominant factor for signal degradation. We characterize the complexity of all three versions of the problem, and we provide light-tree routing algorithms for each. We present numerical results in Section V, and we conclude the paper in Section VI.

A. Related Work

1) The Steiner Tree Problem: To make efficient use of bandwidth in point-to-point networks, the typical approach for multicast communication is to build a multicast tree rooted at the source and spanning all the destinations in a given multicast group. Usually, a cost is assigned to each link of the network, and the objective is to determine the tree of minimum cost. This is the famous Steiner tree problem [11] in graph theory, which is known to be NP-complete [9] when the multicast group has more than two members. Several heuristics and approximation schemes have been developed for the Steiner tree problem. These algorithms can be categorized roughly into the following three groups:

- **Shortest path-based heuristics (SPH).** This algorithm [6] initializes the Steiner tree to the shortest path from the source to an arbitrary multicast member. It then repeatedly includes a new member by adding the shortest path between this member to the current partial tree, until all members have joined the tree. Many variants of this algorithm have been developed to improve the quality of the final tree, such as including the members in the order determined by their distance to the multicast tree [29] instead of random inclusion, or growing the Steiner tree from the destinations [16] instead of from the source.
- **Spanning tree-based heuristics (STH).** This algorithm [15] first constructs a closure graph of the multicast nodes from the original graph using the cost of the shortest path between each pair of members. A minimum spanning tree of the closure graph is obtained (in polynomial time), and then the shortest paths in the original graph are used to replace the edges of this minimum spanning tree. Finally, the multicast tree is obtained by removing any cycles. This approach yields an approximation algorithm with a ratio of 2.

- **Metaheuristics**. Metaheuristics such as simulation annealing [7] , genetic algorithms [8] , and Tabu search [20] have been investigated to solve the Steiner tree problem and have been shown to perform well on average.

In practice, the nature of some multicast applications is such that the routing tree must satisfy certain constraints related to physical limitations (e.g., a limited fanout capability) or the desired quality of service (e.g., an upper bound on the end-to-end delay along any path of the tree). Constrained Steiner tree problems are at least as hard as the unconstrained one, and for certain constraints it has been shown that no polynomial-time approximation scheme exists. Several heuristics have been developed to compute constrained trees, most of which are based on the above heuristics for the unconstrained Steiner tree problem. For instance, the KPP algorithm [14] uses an approach similar to STH to compute an approximate Steiner tree in which the end-to-end delay along any path from the source to a destination node is bounded. The degree-constrained Steiner tree problem [4], [5] , in which it is assumed that some nodes may not support multicast (i.e., they cannot be used as branching points) or have a limited fanout capability has also been studied and appropriate heuristics have been proposed. A constrained multicast tree problem in which the objective is to to bound both the end-to-end delay and the delay variation among all source-destination paths has also been studied [23]; the total cost of the tree was not considered in that work, but it was shown that constructing such a constrained tree is an NP-complete problem.

2) Light-Tree Routing: With recent advances in MC-OXC technology [12] , it is now possible to envision a future backbone network environment that provides a practical multicast service at the optical layer. Such a service will be implemented by using GMPLS-related protocols to establish light-trees on demand [22] . While the problem of establishing a light-tree that spans a given source and a set of destination nodes bears some similarities to the Steiner tree problem, the nature of optical multicast introduces several new issues and complexities, as we discuss next.

Splitting an optical signal introduces losses, a problem not encountered in electronic packet- or circuit-switched networks, and thus, not addressed by existing routing tree algorithms. Even in the presence of optical amplifiers, this signal loss imposes a hard upper bound on the number of times a signal can be split, as well as on the number of hops that the signal can travel after every split operation. In the absence of wavelength conversion in the network (or even in networks with limited or sparse conversion capability), multicast routing is tightly coupled to wavelength allocation, an issue that does not arise in electronic networks. Also, optical networks may only have a sparse multicast switching capability, i.e., only a subset of the OXCs may be multicast capable. When only a few MC-OXCs are present in the network, a feasible multicast tree may not exist, and therefore the heuristics for degree-constrained multicast [4] are not applicable at all. Finally, the problems of capacity planning of MC-OXCs and multicast routing strongly depend on one another.

Several recent research efforts have aimed to address some of the problems associated with optical multicast and light-tree establishment, including studies of wavelength assignment in the presence of multicast [13], [18], [25] , and multicast routing algorithms for networks with a sparse light splitting capability [30], [32], [33] . To deal with the fact that a feasible multicast tree may not exist for a given source and destination set, the concept of a *light-forest* has been proposed [33] . In general, all the multicast routing algorithms for optical networks assume unlimited fanout capacity at MC-OXCs, and each tree of a given light-forest must be assigned a different wavelength. The problem of optimally placing a small number of MC-OXCs in a WDM network has been studied in [1]. Finally, two designs for MC-OXCs have been proposed. The first is based on the splitter-and-delivery architecture [12] , while the second is an enhancement of the former that results in better power efficiency [2] . The reader is also referred to a recent comprehensive survey [22] of the optical multicast problem by one of the authors.

II. The Multicast Optical Network

We consider an optical WDM network with N nodes interconnected by fiber links. Each of the links is capable of carrying W wavelengths, and each of the nodes is equipped with an OXC with P input ports and P output ports. The OXC at (some of) the nodes is multicast-capable (MC-OXC). A $P \times P$ MC-OXC consists of a set of W $P \times P$ splitter-and-delivery (SaD) switches, one for each wavelength; Figure 1 shows a 3×3 MC-OXC for $W = 2$ wavelengths. In addition to the W SaD switches, P demultiplexers (respectively, multiplexers) are used to extract (respectively, combine) individual wavelengths. The SaD switch design was first proposed [12] and was later modified [2] in order to reduce cost and improve power efficiency. A $P \times P$ SaD switch, as originally proposed [12] , is shown in Figure 2. It consists of P power splitters, P^2 optical gates (to reduce the excessive crosstalk), and P^2 2×1 photonic switches. We assume that the splitters are configurable, in that they can be instructed to split the incoming signal into m output signals, $m = 1, \cdots, P$; note that $m = 1$ corresponds to no power splitting, i.e., no multicast, while $m = P$ corresponds to a broadcast operation. By appropriately configuring the corresponding m 2×1 photonic switches, each of the m signals resulting from the splitting operation can be switched to the desired output ports.

In a transparent network, optical signals suffer losses as they travel from source to destination node. We distinguish two types of losses:

1) *Signal attenuation*. This is due to the propagation of light along the fibers between the source and destination nodes. Optical amplifiers (EDFAs) are used along the optical paths to boost the power of the information-carrying signals in order to compensate for the signal attenuation. However, optical power amplification is not perfect, and there is a limit on the number of times a signal may be amplified. Thus, it has been suggested [28] that power attenuation (along with other physical layer

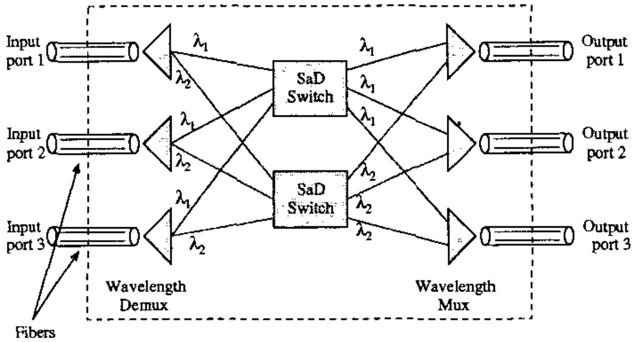

Fig. 1. A 3 × 3 MC-OXC based on the SaD switch architecture, $W = 2$

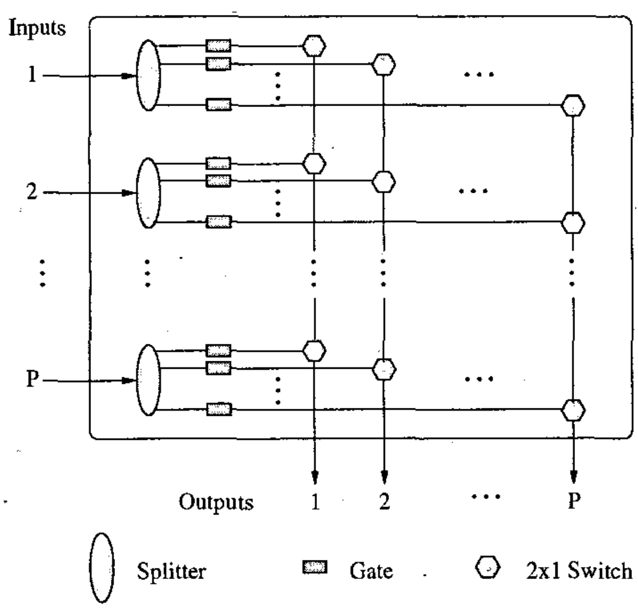

Fig. 2. A $P \times P$ SaD switch

impairments, such as dispersion) be taken into account when routing lightpaths in a transparent optical network.

2) *Splitting loss.* An m-way splitter (similar to those shown in Figure 2) is an optical device which splits an input signal among m outputs. For an ideal device, the power of each output is $(1/m)$-th of that of the original signal; in practice, the splitting operation introduces additional losses and the power of each output is lower than that of the ideal case. Splitting losses occur within MC-OXCs at the branch points of light-trees carrying point-to-multipoint signals. While amplification may partially compensate for the power loss due to light splitting, it is clear that this type of loss must be taken into account for light-tree routing.

The next two subsections discuss the two types of losses in more detail.

A. Power Attenuation Along a Fiber Link

The output power P_{out} at the end of a fiber of length L is related to the input power P_{in} by

$$P_{out} = P_{in}\ e^{-\alpha L} \tag{1}$$

where α is the fiber attenuation ratio [27]; near 1550 nm, we have that $4.34\alpha = 0.2 = \alpha_{dB}$. In general, distributed feedback (DFB) lasers put out about 50 mW (17 dB) of power after the output signal is boosted by an amplifier, while the sensitivity of avalanche photodiode (APD) receivers at 2.5 Gb/s is -34 dB [19] . Therefore, from (1), we obtain the maximum transmission distance in a fiber as $L_{max} = 255$ km. For any fiber link whose length is greater than L_{max}, a number of EDFA amplifiers must be added to compensate for the power attenuation, so that the receiving power at the end of the fiber is no less than -34 dB. When using optical amplifiers, other constraints must be considered, including the maximum permissible power on a fiber, the effects of fiber nonlinearities, and the receiver sensitivity. Consequently, in current practice, amplifier spacings range from 20 km to 100 km.

Suppose that the span of length between two consecutive amplifiers (EDFAs) in the optical network is S, and that the gain of each amplifier is denoted by G_{amp} (S and G_{amp} are assumed to be parameters which are fixed for a particular fiber system). Then, the power received at the end of a fiber link of length L is related to the input power as follows:

$$P_{out} = P_{in}\left(G_{amp}\ e^{-\alpha S}\right)^{\frac{L}{S}} = P_{in}\left(G_{amp}^{\frac{1}{S}}\ e^{-\alpha}\right)^{L} = P_{in}Q^{L} \tag{2}$$

where $Q = G_{amp}^{\frac{1}{S}}\ e^{-\alpha} < 1$ is a constant determined by the fiber system. Expression (2) describes the signal attenuation within a fiber link equipped with optical amplifiers, as a function of the link length L.

B. Power Loss due to Light Splitting at the MC-OXC

Let us now consider a signal that arrives at some input port of an MC-OXC such as the one shown in Figure 1. This signal is split into m output signals at the SaD switch corresponding to the input's wavelength. The m output signals are then switched to the appropriate output ports of the MC-OXC. We assume that the power splitters at the SaD switch (refer to Figure 2) are configurable, such that a multicast optical signal does not always need to be split P times, where P is the number of input/output ports of the SaD switch. Instead, the multicast signal is split into exactly m signals, $m = 1, \cdots, P$, where m is the out-degree of the node in the corresponding light-tree. Configurability is made possible by new devices such as the compact multimode interference couplers with tunable power splitting ratios that were reported recently [17] . We also assume that the tunable power splitting ratio can be controlled by the multicast signaling protocol, making it possible to realize MC-OXCs with any desirable fanout $m, m = 1, \cdots, P$.

Given these assumptions, the power loss (in dB) at an MC-OXC for an input signal that is split into m output signals is given by [2]:

$$Loss_{SaD} = 10\log_{10} m + \beta(P) \quad (3)$$

In the above expression, the term $\beta(P)$ captures losses due to the multiplexing and demultiplexing of signals, as well as the insertion and coupling losses at the 2×1 switching elements (refer to Figure 2). Since the number of switching elements in the signal path is equal to the number P of input/output ports of the SaD switch, then this term is a function of P.

From (3), we can now derive the output power of each of the m output signals as a function of the input power as follows:

$$P_{out} = \frac{10^{-\frac{\beta(P)}{10}}}{m} P_{in} \leq \frac{P_{in}}{m} \quad (4)$$

Expression (4) assumes that signals are not amplified as they leave the MC-OXC. To compensate for the power loss due to light splitting, optical amplifiers may be placed at the output ports of the MC-OXC. Let G_{amp} denote the gain of an amplifier, and define $R = 10^{-\frac{\beta(P)}{10}} G_{amp}$. R is a constant for a given SaD switch, and is determined by the number of ports P of the switch, the losses incurred at the various elements of the switch, and the amplifier gain. Then, the output power of a signal that has undergone m-way splitting is given by:

$$P_{out} = \frac{RP_{in}}{m} \leq P_{in} \quad (5)$$

III. THE LIGHT-TREE ROUTING PROBLEM

We represent a network of MC-OXCs by a simple graph $G = (V, A)$. V denotes the set of nodes (i.e., MC-OXCs), and A, the set of arcs, corresponds to the set of (unidirectional) fiber links connecting the nodes. We will also use $N = |V|$ to refer to the number of nodes in the network. We define a *distance function* $\mathcal{D} : A \rightarrow \mathcal{R}^+$ which assigns a non-negative weight to each fiber link in the network. More specifically, the value $\mathcal{D}(\ell)$ associated with link $\ell = (u, v) \in A, u, v \in V$, is the geographical distance that the optical signal travels along the link ℓ from node u to node v.

Under the light-tree routing scenario we are considering, an optical signal originating at some *source* node $s \in V$ in the network must be delivered to a set $M \subseteq V - \{s\}$ of destination nodes. In general, several point-to-multipoint sessions may proceed concurrently within the network, each characterized by a source node and a destination set. We assume that communication in the network is connection-oriented, and that point-to-multipoint connections are established by issuing a *connect request*; similarly, at the conclusion of a session a *disconnect request* is issued. In response to a connect request, and prior to any optical signal been transmitted from the source to the destinations, a connection establishment process is initiated. Central to the connection establishment is the determination of a light-tree, i.e., a set of paths between the source and the destinations, over which the optical signal will be carried for the duration of the point-to-multipoint session.

Let s and M be the source and destination set, respectively, of a certain point-to-multipoint session. We let $T = (V_T, A_T)$ denote the light-tree, rooted at s, for this session. The light-tree is a subgraph of G (i.e., $V_T \subseteq V$ and $A_T \subseteq A$) spanning s and the nodes in M (that is, $M \cup \{s\} \subseteq V_T$). In addition, V_T may contain *relay* nodes, that is, nodes intermediate to the path from the source to a destination. Relay nodes do not terminate the optical signal transmitted by the source node s; rather, they simply split and/or switch the signal towards the downstream links of the light-tree. We let $H_T(s, v)$ denote the unique path from source s to destination $v \in M$ in the light-tree T. We define $P_{in}(s)$ as the power of the optical signal injected into the network by the source node s, and $P_{out}(s, v)$ as the power of the optical signal received by destination $v \in M$. The output power $P_{out}(s, v)$ at destination v is related to the input power at the source s through the following expression:

$$P_{out}(s, v) = P_{in}(s) \times L^{(atten)}(s, v) \times L^{(split)}(s, v) \quad (6)$$

In the above expression, parameter $L^{(atten)}(s, v)$ (respectively, $L^{(split)}(s, v)$) accounts for the power loss due to attenuation (respectively, light splitting) along the path from s to v in the light-tree T; we assume that both parameters include the effects of amplification.

Recall that expression (2) relates the input and output signal power for a single fiber link. The expression can be generalized to a path from a source s to a destination v in a straightforward manner, allowing us to express $L^{(atten)}(s, v)$ as follows:

$$L^{(atten)}(s, v) = \prod_{\ell \in H_T(s,v)} Q^{\mathcal{D}(\ell)} = Q^{\sum_{\ell \in H_T(s,v)} \mathcal{D}(\ell)} < 1 \quad (7)$$

Similarly, we can obtain an expression for $L^{(split)}$ by considering all MC-OXCs in the path from s to v in the light-tree T, and applying expression (5). Let us define $F_T(u)$ as the fanout of the MC-OXC at node u of the light-tree T, with respect to the optical signal carried on this light-tree [1]. The fanout $F_T(u)$ corresponds to the quantity m in expression (5). As a result, we obtain:

$$L^{(split)}(s, v) = \prod_{u \in H_T(s,v)} \frac{R}{F_T(u)} < 1 \quad (8)$$

We note that, as we explained in the previous section, quantities Q and R in expressions (7) and (8) are constants for a given optical network.

A. Path Constraints to Ensure Optical Signal Quality

We now introduce two parameters that can be used to characterize the quality of the light-tree as perceived by the application making use of the point-to-multipoint optical communication. These parameters relate the end-to-end power loss along individual source-destination paths to the desired level of signal power at the receivers, as follows.

- *Source-destination loss tolerance*, Δ. Parameter Δ represents an upper bound on the acceptable end-to-end power loss along any path from the source to a destination

[1] Note that node v may be part of a different light-tree T', with a different source and destination set; its fanout with respect to T' may be different than its fanout with respect to T.

node. This parameter reflects the fact that if the optical signal power falls below the receiver sensitivity, then the information carried by the signal cannot be recovered.

- *Inter-destination loss variation tolerance*, δ. Parameter δ is the maximum difference between the end-to-end losses along the paths from the source to any two destination nodes that can be tolerated by the application. This parameter can be thought of as a measure of *fairness* among the destination nodes of the light-tree.

By supplying values for parameters Δ and δ, the application in effect imposes a set of constraints on the optical signal power at the receivers of the light-tree:

$$P_{out}(s,v) \geq \Delta P_{in}(s) \qquad \forall\, v \in M,\ \Delta \leq 1 \tag{9}$$

$$\frac{1}{\delta} \leq \frac{P_{out}(s,v)}{P_{out}(s,u)} \leq \delta \qquad \forall\, v,u \in M,\ \delta \geq 1 \tag{10}$$

We will refer to (9) as the *source-destination loss constraint*, while (10) will be called the *inter-destination loss variation constraint*. We will also say that tree T is a *feasible* light-tree for a point-to-multipoint session with source s and destination set M, if and only if T satisfies both (9) and (10). Note that, in order for the application to proceed, it is necessary and sufficient that a *single* feasible light-tree be constructed, since *any* feasible tree can meet the quality of service requirements as expressed by parameters Δ and δ.

IV. Optical Signal Power Constrained Light-Trees

Let Δ and δ be the loss and loss variation tolerances, respectively, as specified by a client application that wishes to initiate a point-to-multipoint session. Our objective is to determine a light-tree such that the power losses along all source-destination paths in the tree are within the two tolerances. This problem, which we will call the *Power Constrained Light-Tree* (PCLT) problem, can be formally expressed as follows.

Problem 4.1 (PCLT): Given a network $G = (V, A)$, a source node $s \in V$, a destination set $M \subseteq V - \{s\}$, a distance function $\mathcal{D} : A \rightarrow \mathcal{R}^+$, a loss tolerance Δ, and a loss variation tolerance δ, does there exist a light-tree $T = (V_T, A_T)$ spanning s and the nodes in M, that satisfies both constraints (9) and (10)?

In the next three subsections we study three variants of the PCLT problem. The variants mainly differ in the assumptions made regarding the degree to which each of the two types of power loss (i.e., loss due to attenuation or light splitting) affects the quality of the received signal. As we explain, the assumptions depend on the geographical span of the light-tree and the size of the destination set, and it is possible that different variants of the PCLT problem apply to different light-trees within the *same* optical network. Therefore, we characterize the complexity of, and provide light-tree algorithms for, all three variants of the PCLT problem.

A. The PCLT Problem Under Power Attenuation Only

Let us first consider the PCLT problem under the assumption that power attenuation is the dominant factor in determining the signal quality at the receivers of the light-tree. In other words, we assume that $L^{(split)}(s,v) \approx 1$ in expression (6), for all destinations v. This is a reasonable assumption when **(i)** the source of the point-to-multipoint session and the destination nodes are separated by large geographical distances, and/or **(ii)** there is a small number of destination nodes, thus, the optical signal only needs to undergo a small number of splitting operations. In this case, we can use (7) to rewrite the source-destination constraint (9) and the inter-destination loss variation constraint (10) as follows.

$$\begin{aligned} \forall\, v \in M: \quad & L^{(atten)}(s,v) \ \geq \ \Delta \\ \Rightarrow \quad & Q^{\sum_{\ell \in H_T(s,v)} \mathcal{D}(\ell)} \geq \Delta \\ \Rightarrow \quad & \sum_{\ell \in H_T(s,v)} \mathcal{D}(\ell) \leq \log_Q \Delta \end{aligned} \tag{11}$$

$$\begin{aligned} \forall v, u \in M: \quad & \frac{1}{\delta} \leq \frac{L^{(atten)}(s,v)}{L^{(atten)}(s,u)} \leq \delta \\ \Rightarrow\ & Q^{\left|\sum_{\ell \in H_T(s,v)} \mathcal{D}(\ell) - \sum_{\ell \in H_T(s,u)} \mathcal{D}(\ell)\right|} \leq \delta \\ \Rightarrow\ & \left| \sum_{\ell \in H_T(s,v)} \mathcal{D}(\ell) - \sum_{\ell \in H_T(s,u)} \mathcal{D}(\ell) \right| \leq \log_Q \delta \end{aligned} \tag{12}$$

Note that the last step in (11) is due to the fact that constants Q and Δ are such that $0 < \Delta, Q < 1$.

An interesting observation regarding constraints (11) and (12) is that they represent two conflicting objectives. Indeed, the loss constraint (11) dictates that short paths be used. But choosing the shortest paths may lead to a violation of the loss variation constraint (12) among nodes that are close to the source and nodes that are far away from it. Consequently, it may be necessary to select longer paths for some nodes in order to satisfy the latter constraint. Then, the problem of finding a feasible light-tree becomes one of selecting paths in a way that strikes a balance between these two objectives.

The PCLT problem with constraints (11) and (12) is equivalent to the *delay- and delay variation-bounded multicast tree* (DVBMT) problem [23], [26] . Specifically, the loss constraint (11) is equivalent to the delay constraint of DVBMT, while the loss variation constraint (12) is equivalent to the delay variation constraint of DVBMT. We have proved [23] that the DVBMT problem is NP-complete whenever the size of the destination set $\mid M \mid \geq 2$. Consequently, if we ignore the power loss due to the splitting of the optical signal at the branch nodes of the light-tree, the PCLT problem is also NP-complete. In this case, the heuristics developed for DVBMT [23], [26] can be applied directly to construct a light-tree that satisfies both constraints (11) and (12).

B. The PCLT Problem Under Splitting Losses Only

Let us now turn our attention to the case when signal attenuation is negligible (i.e., $L^{(atten)} \approx 1$ in expression (6)), and power loss due to light splitting is the dominant factor affecting signal quality at the receivers. This situation may arise when **(i)** the destination set includes a large number of nodes, and/or **(ii)** the source and destination nodes are located in close proximity to each other. We can then use expression (8) to rewrite constraints (9) and (10) as follows (recall that $F_T(w)$ is the fanout of node w with respect to light-tree T, in other words, it denotes the number of times the optical signal traveling along light-tree T is split at node w).

$$\prod_{w \in H_T(s,v)} \frac{R}{F_T(w)} \geq \Delta \qquad \forall\, v \in M \tag{13}$$

$$\frac{1}{\delta} \leq \frac{\prod_{w \in H_T(s,v)} \frac{R}{F_T(w)}}{\prod_{w \in H_T(s,u)} \frac{R}{F_T(w)}} \leq \delta \qquad \forall\, u, v \in M \tag{14}$$

Let us interpret constraints (13) and (14). Without loss of generality, let us assume that $R = 1$, i.e., that the power of each of the $F_T(w)$ output signals at node w is $(1/F_T(w))$-th of that of the input signal; our conclusions are valid even when $R > 1$. When $R = 1$, the denominator of the left hand side of (13) corresponds to the product $\prod_{w \in H_T(s,v)} F_T(w)$ along the path from the source s to destination v. We will call this product the *split ratio of node* v, and its inverse corresponds to the residual power of the optical signal received at node v after all the splits along the path. We can see that constraint (13) imposes an upper bound on the split ratio on the path to each destination node in set M.

Let us now turn our attention to constraint (14). When $R = 1$, it states that the split ratios of any two paths from the source to two destination nodes v and u should be within a tight range from each other, where the tightness of the range is determined by parameter δ. Therefore, this constraint suggests that light-trees must be as balanced as possible. To see why, suppose that a light-tree is constructed for a set of K destinations such that one destination node, say v, is directly connected to the root (source) while the remaining $K-1$ nodes are all in a different subtree connected to the root. It is clear that, even after amplification (i.e., $R > 1$), node v will receive a signal of better quality than the other $K-1$ destinations: the signal arriving at node v is of the same quality as the one traveling towards the other subtree, but the latter signal will have to be split several times (and thus, it will degrade further) before it reaches each of the $K-1$ destinations in the subtree. Such an unbalanced tree has two important disadvantages. First, it introduces unfairness, since receivers at small depth in the (logical) tree receive a signal of better quality than receivers at large depth, *independently* of their geographical distance to the source. Second, it is not scalable, since it may introduce excessive losses that make it impossible to deliver a signal to a given number of destinations. To see this, consider a worst case scenario where the tree is a binary one and is recursively constructed such that the left subtree consists of exactly one receiver, while the right subtree contains all remaining receivers and consists of left and right subtrees in a similar way. It is easy to see that the receiver at depth one (in the left subtree of the whole tree) receives a signal that has undergone one split and its power is one-half of that of the original signal. On the other hand, the receiver at depth K (the rightmost leaf of the tree) receives a signal that is the result of K splits, and its power is $(1/2^K)$-th of that of the original signal. While extreme, this scenario illustrates the pitfalls of unbalanced trees for the multicast of optical signals.

The requirement that the light-tree be as balanced as possible is a direct consequence of the fact that when an optical signal undergoes m-way splitting, its power is equally divided among the m output signals. Thus, this requirement is unique to optical layer multicast. To the best of our knowledge, the problem of constructing balanced multicast trees has not been studied in the literature, since it does not arise in the context of multicast above the optical layer. We now prove the problem of constructing balanced multicast trees to be NP-complete. In the next subsection we present a suite of heuristics to obtain balanced light-trees that satisfy constraints (13) and (14).

Our proof is by reduction from the Exact Cover by Three-Sets (X3C) problem [10] , a well-known NP-complete problem defined as:

Definition 4.1 (X3C): Given a set $S = \{S_i\}$ with $3k$ elements for some natural number k and a collection $Y = \{Y_j\}$ of subsets of the set, each of which contains exactly three elements, do there exist in the collection Y k subsets that together cover the set S?

Theorem 4.1: The PCLT problem under constraints (13) and (14) is NP-complete.

Proof. Clearly, PCLT belongs in the class NP, since a solution to the PCLT problem can be verified in polynomial time. We now transform the NP-complete X3C problem to PCLT. Consider an arbitrary instance of the X3C problem consisting of **(i)** a set $S = \{S_i\}$ of elements, where $|S| = 3k$ for some natural number k, and **(ii)** a collection $Y = \{Y_j\}$ of subsets of S, each subset containing exactly three elements of S. Let $m = |S|, n = |Y|$. We construct a corresponding instance of PCLT as follows. The graph $G = (V, A)$ has $n+m+1$ nodes, with $V = \{s, Y_1, Y_2, \cdots, Y_n, S_1, S_2, \cdots, S_m\}$, where s is the source node and $M = S = \{S_i\}$ is the destination set of the light-tree. The set A of links is:

$$\begin{aligned} A &= \{(s, Y_1), (s, Y_2), \cdots, (s, Y_n)\} \\ &\quad \cup \{(Y_j, S_i) | Y_j \in Y \wedge S_i \in Y_j\} \end{aligned} \tag{15}$$

In other words, there is a link from s to every node Y_i, and a link from every node Y_i to every node S_j which is a member of Y_i (see Figure 3). The distance function is defined as $\mathcal{D}(\ell) = 1,\ \forall\, \ell \in A$ (in fact, the distance function can be arbitrary; since this variant of PCLT neglects power attenuation, the constraints (13) and (14) do not depend on the link weights). Finally, the loss and loss variation tolerances are $\Delta = \frac{1}{3k}$ and $\delta = 1$, respectively.

It is obvious that this transformation can be performed in polynomial time. We now show that a feasible light-tree for the PCLT problem exists if and only if set S has an exact cover. If S has a cover $X = \{Y_{\pi_1}, Y_{\pi_2}, \cdots, Y_{\pi_k}\}$, the tree containing

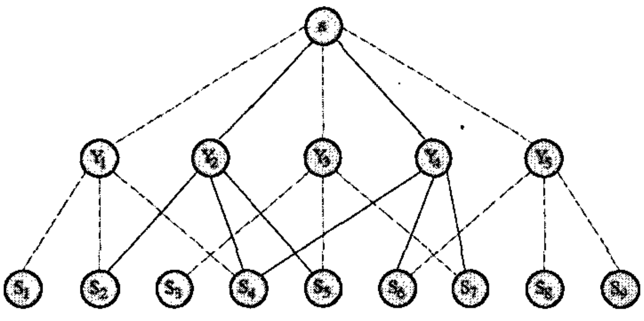

Fig. 3. Instance of PCLT corresponding to an instance of X3C with $k = 3, S = \{S_1, \cdots, S_9\}, Y_1 = \{S_1, S_2, S_4\}, Y_2 = \{S_2, S_4, S_5\}, Y_3 = \{S_3, S_5, S_7\}, Y_4 = \{S_4, S_6, S_7\}, Y_5 = \{S_6, S_8, S_9\}$, and exact cover $\{Y_1, Y_3, Y_5\}$; the light-tree T is denoted by dashed lines

the source s, the set of nodes $X = \{Y_{\pi_1}, Y_{\pi_2}, \cdots, Y_{\pi_k}\}$, and the set of nodes $S = \{S_i\}$ is a feasible solution for PCLT. This is because the split ratio of each destination node S_i is equal to $\frac{1}{3k}$, and the tree satisfies both constraints (13) and (14). Conversely, let T be a feasible light-tree for PCLT. Then, T must contain the source node s, all destination nodes S_i, and a subset X of $Y = \{Y_j\}$. Since $\delta = 1$, all destination nodes S_i have the same split ratio. By construction of the PCLT instance, each destination node S_i must have exactly one parent in the light-tree T: if some node S_i had more than one parents a loop would exist (contradicting the hypothesis that T is a tree), and if it had no parent, it would not be connected to the tree T (again contradicting the hypothesis that T is a solution to the PCLT problem, i.e., it spans all destination nodes). Therefore, the nodes in the subset X of Y contained in the light-tree T exactly cover the set S, implying that X is a solution to the instance of the X3C problem. ∎

1) Balanced Light-Tree Algorithm: We now present an algorithm for the version of the PCLT problem discussed above. The objective of any such algorithm would be to construct a feasible light-tree, i.e., one that satisfies both constraints (13) and (14). Note, however, that since the PCLT problem is NP-complete, any polynomial-time algorithm may fail to construct a feasible light-tree for a given problem instance, even if one exists. The algorithm we present can be used to search through the space of *candidate* trees (i.e., trees spanning s and the nodes in M) for a feasible solution to the PCLT problem. Our algorithm either returns a feasible tree, or, having failed to discover such a tree, it returns one for which **(i)** the maximum split ratio of any node in M, and **(ii)** the maximum difference between the split ratios of any pair of nodes in M, are minimum *over all trees considered by the algorithm*.

The *balanced light-tree* (BLT) algorithm, described in detail in Figure 4, takes as input an initial tree T_0 spanning the source s and destination nodes in M; the issue of constructing this initial tree is addressed shortly. In general, tree T_0 may be infeasible, i.e., it may violate (13) and/or (14). The key part of the BLT algorithm is the tree balancing procedure that is implemented by the **while** loop in Steps 4-18 of Figure 4. Consider an intermediate light-tree T, and let u (respectively, v) denote the leaf node with maximum (respectively, minimum) split ratio. The idea behind the BLT algorithm is to delete node u from T, and add it back to the tree by connecting it to some node y in the path from source s to v. Doing so reduces the split ratio of node u, but it also increases the split ratio of all nodes below node y in the tree; therefore, this pair of delete/add operations is performed only if it does not increase the split ratio of any node beyond that of node u (refer also to the **if** statement in Steps 14-17 of Figure 4). Thus, after each iteration of the algorithm, the split ratio of the node with the maximum value is decreased, in an attempt to satisfy constraint (13). While the split ratio of some other node(s) is increased, it does not increase beyond the previous maximum value. As a result, the difference between the maximum and minimum split ratio values also decreases with each iteration, as required by constraint (14). The algorithm terminates after a certain number of iterations, or if two successive iterations fail to reduce the maximum split ratio; the latter condition is not shown in Figure 4 in order to keep the pseudocode description simple.

In order to completely specify the BLT algorithm, we now explain how to select the node y in the path from s to v (the node with the minimum split ratio) to connect node u (the one with the maximum split ratio). Let Y denote the number of nodes in the path from source s to node v. We consider three different criteria for selecting a node $y \in Y$ to which to connect node u, resulting in three variants of the BLT algorithm.

1) **Shortest path (BLT-SP).** In this variant, we select node y such that the path from y to u is shortest among the paths from any node in Y to u.
2) **Minimum split ratio (BLT-MSR).** In this case, node y is one with the smallest split ratio among all the nodes in Y.
3) **Degree constraint (BLT-D).** This is similar to BLT-MSR, except that the node y selected must be such that its fanout is no more than a maximum value F. F may correspond to the maximum fanout capacity of the SaD switches at each MC-OXC. With this selection criterion, the resulting light-tree will have a bounded degree (fanout). Note that, if we use a different value of F for each node in the network, then the algorithm can be used in optical networks with sparse light splitting, since multicast-incapable OXCs can be accounted for by letting $F = 1$ for these nodes.

Finally, we use the SPH algorithm [29] to construct an initial tree T_0 that spans the source node s and the destination set M. The SPH algorithm is a fast algorithm which has been used successfully as a starting point for several constrained Steiner tree problems [4] . The algorithm starts with a partial tree consisting of the shortest path from the source s to some destination node. It then repeatedly extends the partial tree to another destination node u, until all destination nodes have

Balanced Light-Tree (BLT) Algorithm

Input: A graph $G = (V, A)$ representing the network of MC-OXCs, a source node $s \in V$, a destination set $M \subseteq V$, a loss tolerance Δ, a loss variation tolerance δ, and an initial light-tree T_0 spanning the set $\{s \cup M\}$

Output: A light-tree T_f spanning the set $\{s \cup M\}$, and such that either **(i)** T_f is feasible, or **(ii)** the difference between the maximum and minimum split ratio of any two nodes in M is minimum

1\. **begin**
2\. $T \leftarrow T_0$ // Initialize the light-tree
3\. $h \leftarrow 1$ // Number of iterations
4\. **while** ($h \leq$ MAX_ITER)
5\. Use DFS to calculate the split ratio of all nodes in M
6\. **if** (light-tree T is feasible) **then** return T
7\. $u \leftarrow$ leaf node with maximum split ratio
8\. $v \leftarrow$ leaf node with minimum split ratio
9\. $w \leftarrow$ the first node in the path from u to s in T such that $w \in M$ or w has a fanout > 1
10\. $Y \leftarrow$ set of nodes in the path from v to s in T
12\. In G, compute shortest paths from u to every node in Y
13\. $y \leftarrow$ a node in Y selected based on a predefined criterion // See Section IV-B.1
14\. **if** (max split ratio of T does not increase) **then**
15\. Delete the path from w to u in tree T
// Delete the node with the maximum split ratio
16\. Add the shortest path from y to u to T
// Add the node back to T on a different path
17\. **end if**
18\. **end while**
19\. return T
20\. **end algorithm**

Fig. 4. General balanced light-tree (BLT) algorithm

been included. A new destination node u is connected to the partial tree by including the shortest path from some node y of the tree to u. Therefore, the issue arises of selecting the node y of the partial tree to which to connect node u. For each variant of the BLT algorithm, we use the corresponding selection criterion to select node y of the partial tree.

Regarding the complexity of the BLT algorithm shown in Figure 4, it is straightforward to verify that the worst-case running time is $O(N^2 I)$, where N is the number of nodes in the network and I, an input parameter, is the number of iterations of the **while** loop in Steps 4-18. We note that the worst-case complexity is the same for all three variants of the BLT algorithm.

C. The General PCLT Problem

We now consider the most general version of the PCLT, which arises when both signal attenuation and light splitting contribute to the degradation of the quality of the signal as it travels through the optical network. In this case, the signal power received at each destination node is related to the signal power emitted by the source node through expressions (6)-(8), and the light-tree must be constructed such that constraints (9) and (10) be satisfied. Clearly, this version of the PCLT problem is also NP-complete, since it includes as special cases the two versions studied in Sections IV-A and IV-B, both of which are NP-complete.

An interesting observation regarding this general version of the PCLT problem is that there is a tradeoff between the number of times a signal may be split and the distance that the signal can travel. Signals that have been split multiple times may not be able to travel over large distances, even after amplification, and vice versa. This tradeoff, which is unique to optical networks, is not taken into account by existing multicast routing algorithms. In this case, it would be desirable to have receivers which are far away (in terms of distance traveled by the optical signal) from the source, be closer to the source in the (logical) light-tree. This way, the signal arriving to these receivers will have undergone a smaller number of splits. In this case, the resulting light-tree will not necessarily be balanced (in the traditional definition of the term), but rather it must be balanced in a manner that accounts for the geographical locations of the various receivers relative to the source. In other words, the number of signal splits for each receiver must be appropriately weighted by the distance to the receiver.

Based on the above observations, we modify the BLT algorithm shown in Figure 4 to construct *distance-weighted* balanced light-trees; we will call this algorithm *weighted* BLT (WBLT). The main idea is to consider the tree node with the largest *total* loss and attempt to reduce its splitting loss by moving it closer to the source in the logical light-tree. Doing so may increase the attenuation loss (since the node may be added to the tree on a longer path), but it will also decrease its splitting loss, possibly resulting in a smaller total loss. This weighted balancing procedure can be accomplished by making the following small changes in the algorithm of Figure 4: in Step 7 (respectively, Step 8), select the node with the maximum (respectively, minimum) total loss, and in the **if** statement in Step 14, check whether the maximum total loss at any node of the tree increases. Otherwise, the algorithm remains unchanged. Note that, since there are three variants of the BLT algorithm, we also have three variants of WBLT, namely, WBLT-SP, WBLT-MSR, and WBLT-D.

V. Numerical Results

We have used simulation to evaluate the average case performance of the light-tree routing algorithms on randomly generated graphs. The graphs were generated using Waxman's method [31] . The nodes of the graphs were placed in a grid of dimensions 5000×5000 km, an area roughly the size of the continental United States. The weight of each link was set to the Euclidean distance between the pair of nodes connected by the link. To test the performance of our algorithms, we randomly generated graphs with a number of nodes ranging from 50 to 110, and we varied the size of the destination set from 5-15% of the number of nodes in the graph. In all the results shown in this section, each point plotted represents the average over 300 graphs for the stated number of nodes. We have also computed 95% confidence intervals which are not shown, since they are very narrow and including them would affect the clarity of the figures. For algorithm BLT-D, we set the degree constraint as 4, a reasonable value for the maximum fanout of an MC-OXC.

We first study the performance of the three variants of the BLT algorithm (namely, BLT-SP, BLT-MSR, and BLT-D) for the PCLT problem under splitting losses only. We consider three performance measures:

1) *maximum split ratio*, which captures the quality of the signal at the destination node where it is worst,
2) *maximum-to-minimum split ratio*, which reflects the difference between the best and worst signal quality, and is a measure of inter-destination fairness, and
3) *number of links of the light-tree*, which captures the amount of resources (e.g., wavelengths) consumed by the point-to-multipoint session.

In Figures 5-7 we plot the behavior of the algorithms in terms of the three metrics as a function of the number N of nodes in the network, for light-trees with a number of destinations equal to 15% of the number of nodes; very similar results have been obtained when the number of destination nodes is equal to 5% or 10% of the number of nodes, but due to space constraints we cannot present them here. Each figure shows three pairs of plots, each pair corresponding to one of the variants of the BLT algorithm, BLT-SP, BLT-MSR, and BLT-D. The two plots within each pair correspond to two light-trees: the initial light-tree T_0, provided as input to the BLT algorithm [2], and the final light-tree returned by the algorithm after the tree balancing procedure (the **while** loop in Steps 4-18 of Figure 4).

Figure 5 shows the maximum split ratio for the three algorithms, before and after the tree balancing procedure. Let us first concentrate on the initial trees. As we can see, the initial tree for BLT-SP has the worst average performance, while the maximum split ratios of the initial trees for BLT-MSR and BLT-D are much smaller (especially for large networks), with BLT-D being slightly better than BLT-MSR. In particular, the maximum split ratio of the initial tree constructed by BLT-SP is significantly larger than the size of the destination set; for instance, for $N = 100$, the destination set has 15 nodes, but the maximum split ratio is around 48; in other words, without amplification, the corresponding destination node would have received (1/48)-th of the power of the signal transmitted by the source. Even after amplification, this signal will have undergone severe degradation due to splits. Note that BLT-SP corresponds to the pure SPH algorithm [29] , which has been used extensively in the literature for the Steiner tree problem. Naturally, the SPH algorithm does not take into account optical layer power constraints, and thus, it may produce very unbalanced trees. This result indicates that algorithms not specifically designed with these constraints in mind would have very poor performance in the context of optical layer multicast. On the other hand, BLT-MSR and BLT-D are variants of SPH that take the split ratio into account when building the initial tree. As we can see, such customization results in significant improvements in performance with respect to this metric.

Let us now turn our attention to the final trees produced by the three algorithms. We immediately see that the tree balancing procedure is successful in reducing significantly the maximum split ratio from that of the initial tree, for all three algorithms. Specifically, the improvement (decrease) in the maximum split ratio ranges from about 50% (for the BLT-MSR and the BLT-D algorithms) to 70% (for the BLT-SP algorithm). In other words, the signal quality at the destination where it is worst, is 50-70% better, depending on the algorithm, in the final, balanced tree compared to the initial tree. Furthermore, the maximum split ratio of the final trees increases more slowly with the number of nodes than that of the initial trees. We also observe that the BLT-SP algorithm shows the best improvement after the balancing operation, and its final trees SP have a maximum split ratio smaller than that of the corresponding final trees constructed by BLT-MSR and BLT-D. This result is due to the fact that the BLT-SP algorithm does not impose any constraints on the final tree (e.g., compared to the BLT-D algorithm), and thus, it is able to find better trees. Overall, the results of Figure 5 suggest that the suite of BLT algorithms can be used to construct light-trees with good performance in terms of signal power degradation. Consequently, light-trees can scale to large destination sets and networks sizes. Such scalability may not be possible with currently available algorithms, since the resulting light-trees (refer to the initial tree for BLT-SP in Figure 5) have a high maximum split ratio which also increases quickly with the number of network nodes.

Figure 6 plots the maximum-to-minimum split ratio for the initial and final trees of all three algorithms. This is a measure of the worst to best signal power at the destinations, i.e., a measure of fairness. As we can see, BLT-SP has the worst performance (both for the initial and final trees), while the performance of the initial trees constructed by BLT-MSR and BLT-D is better. More importantly, the final trees of BLT-MSR and BLT-D have a very low value (around 2.5), suggesting fair treatment of the destination nodes. Furthermore, this low value of the maximum-to-minimum split ratio remains almost constant across the range of network sizes considered, again indicating that the fairness property scales to networks and destination sets of realistic size.

Figure 7 plots the number of edges of the initial and final trees for the three algorithms. The trees constructed by BLT-SP have fewer edges than those by BLT-MSR and BLT-D. Also,

[2]Note that, while the SPH algorithm [29] is used to construct the initial light-tree T_0, a different criterion is used by each BLT variant to determine how a new destination node is connected to the partial tree, as we explained in Section IV-B.1. Therefore, the initial light-tree is different for each BLT variant.

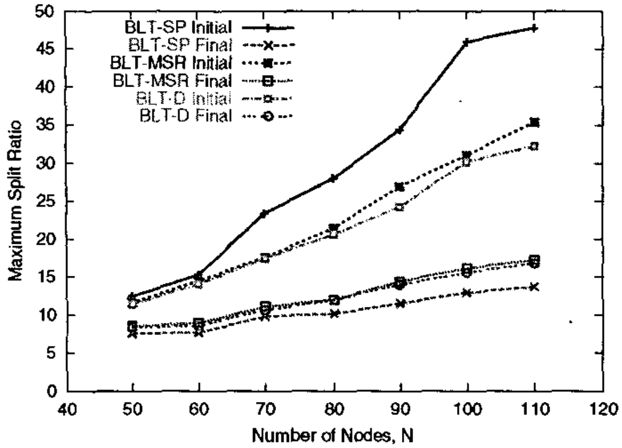

Fig. 5. Maximum split ratio, destination set size $= .15N$

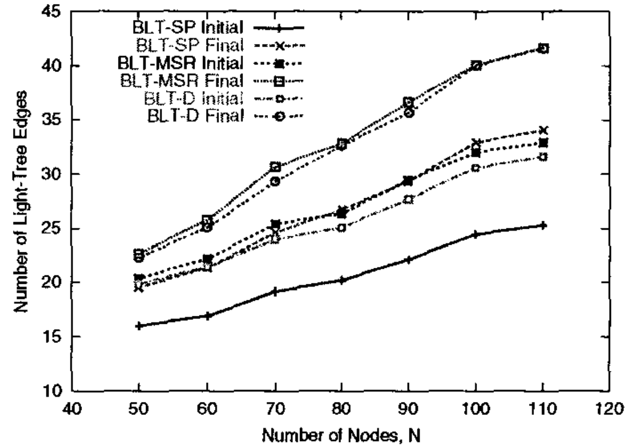

Fig. 7. Cost, destination set size $= .15N$

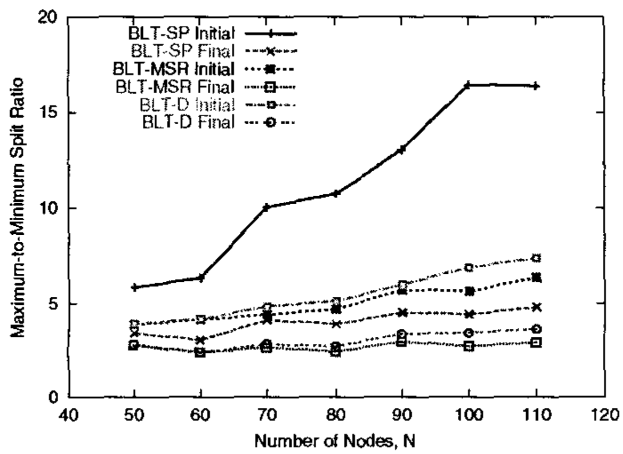

Fig. 6. Maximum-to-minimum splitting ratio, destination set size $= .15N$

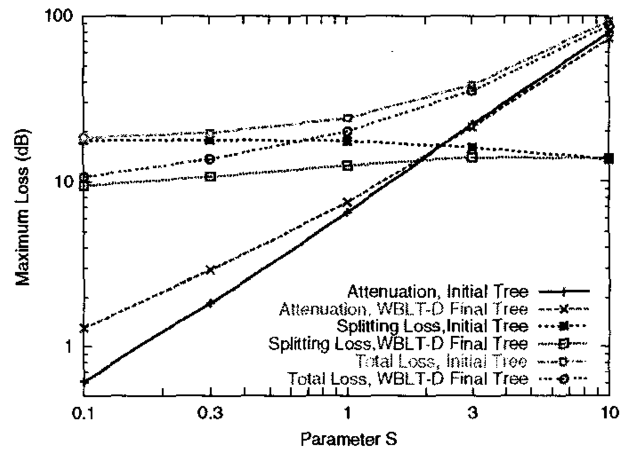

Fig. 8. Maximum loss for WBLT-D, destination set size $= .15N$

performing tree balancing increases the number of edges of the final tree, regardless of the algorithm employed. This result illustrates the penalty involved in balancing the tree to reduce the maximum split ratio and improve the signal quality at the destinations where it is worst. In order to balance the light-tree, destinations with high split ratios are added closer to the source by extending the tree and using additional relay nodes and edges. Consequently, balanced trees use additional network resources, including relay nodes, links, and wavelengths. Thus, there is a tradeoff between using resources efficiently and balancing the light-tree to accommodate optical layer power constraints.

We now demonstrate the operation of the WBLT algorithm which constructs distance-weighted light-trees by taking into account losses due to both attenuation and light-splitting. We define parameter $S, S > 0$, to capture the relative importance of loss due to attenuation and loss due to power splitting: when $S > 1$, loss due to attenuation is the dominant component of total loss, while when $S < 1$, splitting loss dominates. As we discussed above, the value of S (i.e., whether it is greater than or less than one) depends on several network parameters including the diameter of the network, the destination set size, the distance between amplifiers, and the technology of amplifiers, SaD switches, and power splitters. By varying the value of S we are able to investigate a wide range of relative values for the power splitting and attenuation losses.

Figure 8 plots the maximum loss (in dB), against parameter S. Due to space constraints, we only show results for the WBLT-D algorithm. The initial tree is constructed using Dijkstra's algorithm, and consists of the shortest paths from the source to all destinations; thus, this tree minimizes loss due to attenuation. The figure shows three pairs of plots: one for the total loss, one for the loss due to attenuation, and one for loss due to power splitting. Each pair consists of one plot corresponding to the initial tree, and one corresponding to the final tree after applying WBLT-D to the initial tree.

As we can see from Figure 8, the total loss tracks the dominant loss component (attenuation or power splitting). The total loss is smaller for the final tree, especially when loss due to power splitting dominates. The decrease in total loss can be more than 50% at low values of S (note that both axes are shown in log scale). The plots corresponding to loss due to attenuation and power splitting explain how the distance-weighted balancing operation of WBLT is successful in reducing the total loss. Specifically, WBLT moves nodes that are far away from the source (in geographical distance) closer to the source in the light-tree. Doing so increases the loss due to attenuation (compare the corresponding plots for the initial and final tree), but reduces the loss due to power splitting (again, compare the corresponding plots). This operation is particularly successful when loss due to power splitting is dominant or even roughly equivalent to loss due to attenuation (i.e., for values of S up to 3 in the figure). When loss due to attenuation is dominant (e.g., for $S = 10$), the WBLT algorithm has little effect on total loss. This result is expected, of course, since the initial tree is optimal with respect to attenuation, and any reduction in loss due to power splitting would have negligible effect on total loss.

VI. Concluding Remarks

We have studied the light-tree routing problem under optical layer power budget constraints. We considered both attenuation and splitting loss as factors affecting the quality of signals delivered to the destination nodes. We introduced a set of constraints on the end-to-end paths in order to guarantee an adequate signal quality and to ensure a measure of fairness among the destination nodes. These constraints require the light-tree to be balanced or distance-weighted balanced. We proved that constructing such a light-tree spanning a given source and destination node set is an NP-complete problem. We developed a number of algorithms for building balanced trees, and we investigated their performance through extensive simulation experiments on a large number of randomly generated network topologies.

References

[1] M. Ali and J. Deogun. Allocation of splitting nodes in wavelength-routed networks. *Photonic Network Communications*, 2(3):245–263, August 2000.

[2] M. Ali and J. Deogun. Power-efficient design of multicast wavelength-routed networks. *IEEE Journal on Selected Areas in Communications*, 18(10):1852–1862, 2000.

[3] P. Ashwood-Smith *et al.* Generalized MPLS – signaling functional description. IETF Draft <draft-ietf-mpls-generalized-signaling-06.txt>, April 2001. Work in progress.

[4] F. Bauer and A. Varma. Degree-constrained multicasting in point-to-point networks. In *Proceedings of INFOCOM '95*, pages 369–376. IEEE, April 1995.

[5] C. Chow. On multicast path finding algorithms. In *Proceedings of INFOCOM '91*, pages 1311–1317. IEEE, 1991.

[6] M. Doar and I. Leslie. How bad is naive multicast routing. In *Proceedings of INFOCOM '93*, pages 82–89. IEEE, April 1993.

[7] K. A. Dowsland. Hill-climbing simulated annealing and the steiner problem in graphs. *Eng. Opt.*, 17:91–107, 1991.

[8] H. Esbensen. Computing near-optimal solutions to the steiner problem in a graph using a genetic algorithm. *Networks*, 26:173–185, 1995.

[9] M. R. Garey, R. L. Graham, and D. S. Johnson. The complexity of computing steiner minimal trees. *SIAM Journal of Applied Mathematics*, 32(4):835–859, June 1977.

[10] M. R. Garey and D. S. Johnson. *Computers and Intractability*. W. H. Freeman and Co., New York, 1979.

[11] S. L. Hakimi. Steiner's problem in graphs and its implications. *Networks*, 1:113–133, 1971.

[12] W. S. Hu and Q. J. Zeng. Multicasting optical cross connects employing splitter-and-delivery switch. *IEEE Photonics Technology Letters*, 10:970–972, July 1998.

[13] X.-H. Jia, D.-Z. Du, X.-D. Hu, M.-K. Lee, and J. Gu. Optimization of wavelength assignment for QoS multicast in WDM networks. *IEEE Transactions on Communications*, 49(2):341–350, Feb. 2001.

[14] V. P. Kompella, J. C. Pasquale, and G. C. Polyzos. Multicast routing for multimedia communication. *IEEE/ACM Transactions on Networking*, 1(3):286–292, June 1993.

[15] L. Kou, G. Markowsky, and L. Berman. A fast algorithm for steiner trees. *Acta Informatica*, 15:141–145, 1981.

[16] J. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proc. American Mathematical Society*, 7:48–50, 1956.

[17] Juerg Leuthold and Charles H. Joyner. Multimode interference couplers with tunable power splitting ratios. *IEEE/OSA Journal of Lightwave Technology*, 19(5):700–706, May 2001.

[18] D. Li, X. Du, X. Hu, L. Ruan, and X. Jia. Minimizing number of wavelengths in multicast routing trees in WDM networks. *Networks*, 35(4):260–265, 2000.

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

[20] C. C. Ribeiro and M. C. De Souza. Tabu search for the steiner problem in graphs. *Networks*, 36(2):138–146, September 2000.

[21] E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol label switching architecture. RFC 3031, January 2001.

[22] G. N. Rouskas. Optical layer multicast: Rationale, building blocks, and challenges. *IEEE Network*, 17(1):60–65, January/February 2003.

[23] G. N. Rouskas and I. Baldine. Multicast routing with end-to-end delay and delay variation constraints. *IEEE Journal on Selected Areas in Communications*, 15(3):346–356, April 1997.

[24] L. H. Sahasrabuddhe and B. Mukherjee. Light-trees: Optical multicasting for improved performance in wavelength-routed networks. *IEEE Communications*, 37(2):67–73, February 1999.

[25] G. Sahin and M. Azizoglu. Multicast routing and wavelength assignment in wide-area networks. In *Proceedings of SPIE*, volume 3531, pages 196–208, November 1998.

[26] P-R. Sheu and S-T. Chen. A fast and efficient heuristic algorithm for the delay- and delay variation-bounded multicast tree problem. *Computer Communications*, 2001.

[27] T. E. Stern and K. Bala. *Multiwavelength Optical Networks*. Prentice Hall, Upper Saddle River, New Jersey, 2000.

[28] J. Strand, A. L. Chiu, and R. Tkach. Issues for routing in the optical layer. *IEEE Communications*, 39:81–87, February 2001.

[29] H. Takahashi and A. Matsuyama. An approximate solution for the steiner problem in graphs. *Math. Japonica*, 24(6):573–577, 1980.

[30] W.-Y. Tseng and S.-Y. Kuo. All-optical multicasting on wavelength-routed WDM networks with partial replication. In *ICIN-2001*, pages 813–818. IEEE, 2001.

[31] B. W. Waxman. Routing of multipoint connections. *IEEE Journal on Selected Areas in Communications*, 6(9):1617–1622, December 1988.

[32] K.-D. Wu, J.-C. Wu, and C.-S. Yang. Multicast routing with power consideration in sparse splitting WDM networks. In *Proceedings of IEEE ICC*, pages 513–517, 2001.

[33] X. Zhang, J. Y. Wei, and C. Qiao. Constrained multicast routing in WDM networks with sparse light splitting. *Journal of Lightwave Technology*, 18(12):1917–1927, December 2000.