Efficient Impairment-Constrained 3R Regenerator Placement for Light-Trees in Optical Networks

Yi Zhu, Xiaofeng Gao, Weili Wu, and Jason P. Jue

***Abstract*—Light-trees can efficiently guarantee point-to-multipoint connection in optical networks for many widely used multicast applications, such as Internet protocol television (IPTV). The establishment of a light-tree requires the placement of *3R regenerators* along the tree due to the *wavelength continuity constraint* and *physical impairments*. Thus, the problem is to establish a light-tree and to assign wavelengths such that the number of regenerators is minimized. We call this problem the *efficient 3R regenerator placement* (ERP) problem. If we fix the routing of the multicast tree, then how to place a minimum number of regenerators and assign wavelengths to links becomes a subproblem of ERP, which is named the *wavelength assignment and regenerator placement* (WARP) problem. We find that ERP is NP-hard, and then provide an approximation algorithm named SPT-ReWa, which has a subroutine named ReWa which can solve WARP optimally. We prove that ReWa can find an optimal solution for WARP, and we analyze the approximation ratio of SPT-ReWa for ERP. Finally, we illustrate several simulation scenarios to show the efficiency of SPT-ReWa.**

***Index Terms*—Light-tree; Multicast; 3R regenerator placement; Wavelength assignment.**

I. Introduction

Many emerging multicast applications, such as Internet protocol television (IPTV), live auction, and distributed games, require point-to-multipoint connections from a source to multiple destinations in the network. These applications also require guaranteed high-bandwidth transmission. For example, high definition TV without compression needs 6 Gbps bandwidth, ultra-high definition TV needs 72 Gbps, and 4k-cinema needs 6 Gbps [1]. In optical networks, these requirements can be achieved by utilizing *light-trees* [2].

A light-tree is a point-to-multipoint optical channel that enables simultaneous communication between a source and a set of destinations. Traditionally, there are two steps to set up a light-tree: *routing* and *wavelength assignment*. The routing problem is to construct a tree in the given optical network which is rooted at the source and connects all destinations; the wavelength assignment problem is to select an available wavelength on each edge of the generated routing tree [3].

However, in a practical setting, routing and wavelength assignment are not sufficient to guarantee successful communication between source and destinations. A destination may not receive the data from the source due to two reasons.

a) *Wavelength Continuity Constraints* (*WCC*): WCC require that the incoming and outgoing light signals at a node in the tree reserve the same wavelength in the absence of wavelength conversion. If there is no common wavelength available along the path from the source to a destination, the destination will not receive the data and will be blocked. The blocking probability is higher for multicast traffic than for unicast traffic, since unicast only considers a path, while multicast considers a tree.

To overcome the effect of WCC, an *optical wavelength converter* can be introduced to switch one wavelength to another [4]. However, such converters are not widely used because of high cost, limited capability to switch the wavelength, and additional impairments to the transmission.

b) *Physical Impairment* (*PI*): PI refers to physical degradation effects, such as noise, intra-crosstalk, linear impairments, and nonlinear fiber effects, which cause the signal quality to degrade in terms of power. When the receiving signal power is below a threshold, the receiver cannot receive the data correctly.

In a light-tree, a node may use a *passive power splitter* to split an incoming signal into m outgoing signals, which degrades the incoming signal power by a factor of $\frac{1}{m}$. Such a device significantly reduces the probability that a destination will receive data successfully. One possible method to compensate for the signal power loss is to place *amplifiers* [5]. However, amplifiers will also amplify the noise accumulated in the signal. Furthermore, amplifiers cannot convert the wavelength and may also introduce additional impairments.

To overcome WCC and PI and to avoid the shortcomings of converters and amplifiers, *3R regenerators* can be used in the optical network. 3R refers to *reamplification*, *reshaping*, and *retiming*, and involves converting the optical signal back to the electronic domain and regenerating a brand new signal (a.k.a. O–E–O conversion).

If 3R regenerators are placed at all nodes (i.e., *full placement*), the destinations are guaranteed to receive the data correctly, which ensures successful light-tree setup. However, Scheffel [6] argues that full placement will cause high capital expenditures for network providers due to

Manuscript received August 2, 2010; revised January 4, 2011; accepted January 5, 2011; published March 31, 2011 (Doc. ID 132424).

Yi Zhu (e-mail: yizhu@utdallas.edu), Weili Wu, and Jason P. Jue are with the Computer Science Department at the University of Texas at Dallas, Texas, USA.

Xiaofeng Gao is with the School of Science and Technology, Georgia Gwinnett College, Georgia, USA.

Digital Object Identifier 10.1364/JOCN.3.000359

the significant number of O–E–O conversions needed in the network. Furthermore, compared to full placement, the routing protocol and signaling overhead will be tremendously reduced with a limited number of 3R regenerators (i.e., *sparse placement*) [7–9]. Therefore, we wish to place as few 3R regenerators as possible to satisfy the light-tree setup. In this paper, our objective is to place the minimum number of regenerators for a given multicast request to guarantee successful communications. This problem is named the *efficient 3R regenerator placement* (ERP) problem.

In ERP, we place the regenerators either to relax WCC or to compensate PI, while in Chen's problem [10], wavelength conversion is needed only to relax WCC. Therefore, Chen's problem is a special case of ERP. Since Chen's problem is proved to be NP-hard, we will prove that ERP is also NP-hard in Section III. We solve ERP with an approximation algorithm named *SPT-ReWa*, which contains two steps. First, we use the *shortest path tree* (SPT) algorithm to construct a light-tree for one source and several destinations in a target network. We then design an efficient subroutine named *ReWa* to determine wavelength assignment and regenerator placement for the given SPT, which is a subproblem for ERP. We call this subproblem the *wavelength assignment and regenerator placement* (WARP) problem. ReWa consists of three phases:

1) mathematical transformation,
2) minimum regenerator calculation, and
3) wavelength assignment and regenerator placement.

We prove that ReWa finds the minimum number of 3R regenerators (optimal solution) for WARP, we analyze the approximation ratio of SPT-ReWa, and we evaluate the performance of ReWa and SPT-ReWa theoretically and practically. Our numerical results demonstrate the efficiency of SPT-ReWa.

The rest of this paper is organized as follows. We review previous works in Section II. In Section III, we model ERP and WARP formally, including switch architecture analysis, power function formulation, and ERP and WARP definition. We propose and illustrate SPT-ReWa in Section IV. In Section V, we analyze the performance of SPT-ReWa and ReWa. Finally, we report some numerical results in Section VI and conclude the paper in Section VII.

II. Previous Works

The research related to ERP can be divided into three categories.

(a) *Multi λ-light-tree*: Multi λ-light-tree [11] relaxes WCC by using wavelength converters. This problem has several optimization objectives. In [3], the authors tried to minimize the number of wavelengths in a multicast routing tree by formulating it as the *wavelength cover* (WC) problem. WC was proved to be NP-hard and solved by two-phase heuristics. Chen and Wang [10] tried to minimize the number of converters and argued that the wavelength assignment for a given tree was not NP-hard. Liang and Shen [12] balanced wavelength cost and conversion cost for light-tree construction and transferred it to the well-known *Steiner Tree* problem. Two extensions of [12] were made by considering transmission delay and conversion delay in [13] and [14], respectively.

Minimizing the number of wavelengths or the number of converters is not equivalent to minimizing the number of regenerators, which is our objective. An example can be found in Subsection VI.A. Moreover, we also consider PI constraint. Thus, ERP is different from Multi λ-light-tree.

(b) *Power-Constrained Light-tree*: Power-Constrained Light-Tree takes the power budget into account in different ways. In an optical network, a power budget is the allocation of available electrical power among the various functions that need to be performed [15]. In practice, an optical signal transmitted at a source node has a fixed amount of power. Each time the message is split at a router onto multiple output ports, a splitting loss is incurred, which reduces the power of the signal at each of the outputs. Thus, while the power budget may allow a message on a given wavelength to be dropped at more than one destination, it may not be possible to drop the message at an arbitrary number of destinations using a single light-path or light-tree [16].

In [17], the authors considered the power budget when constructing a light-tree and prove that *power aware light-tree* (PALT) construction is NP-hard. They formulated the power function for link attenuation and node splitting.

Another problem, named *multicast routing and wavelength assignment for multi-drop light-tree* (MC-RWA), allows the optical signal to be dropped at a maximum of k destinations [18]. The value k is determined by the power budget. It is proved that MC-RWA is NP-hard for a general graph. On the other hand, the lower bound of MC-RWA can be solvable in certain topologies, such as rings, hypercubes, and tori [18,19]. In [20], the authors argued that MC-RWA for a 2-drop light-tree can be reduced to the *minimum bipartite weighted matching* problem to obtain the optimal solution.

Hamad and Kamal [21] proposed routing and wavelength assignment for *power aware multicasting*, named RWA-PAM. They developed a mixed integer linear programming (MILP) to formulate RWA-PAM and proposed a heuristic, since RWA-PAM is NP-hard.

In our problem, ERP, we place regenerators where the power drops below some threshold, which is the main difference from PALT, MC-RWA, and RWA-PAM. We do not constrain the number of destinations in our topology, which is different from MC-RWA. Besides, RWA-PAM may fail if it cannot find a possible light-tree to meet a certain power level, but we can always obtain a feasible solution for ERP.

(c) *Traditional Regenerator Placement*: Traditional Regenerator Placement consists of two possible techniques. The first is to place 3R regenerators beforehand, and then to decide routing and wavelength assignment for the coming request based on the location of regenerators. Pachnicke *et al.* [22] placed the regenerators based on the estimated signal degradation along links and nodes, and set up paths according to the demands. Peng *et al.* [23] proposed a new switch architecture (tMC-OXC), which can support transparent unicast and opaque multicast. When the multicast traffic passes through this switch, it is equivalent to having 3R regenerators at all incoming and outgoing links. Based on this new architecture, they developed algorithms for multicast routing.

Another technique first finds the routing and wavelength assignment, and then places 3R regenerators when necessary [1]. The authors formulated the signal-to-noise ratio (SNR) function and developed the MILP to place the regenerators based on the assumption that the links on the tree can obtain the same wavelength.

Our work utilizes the second technique as the basic approach; however, we have different objectives compared to [1]. Additionally, we also provide the optimal wavelength assignment and regenerator placement simultaneously for a given tree.

III. Efficient 3R Regenerator Placement Problem

In this section, we formulate ERP and WARP formally. We first introduce the network models, notations, and other useful assumptions in Subsection III.A. Next, Subsection III.B introduces the switch architecture, which helps to develop the power loss function in Subsection III.C. Finally, Subsection III.D provides the problem definitions in detail and proves that ERP is NP-hard.

A. Models and Symbols

a) *Network Model*: We model the network as an undirected graph $G=(V,E)$, where each switch denotes a node $u \in V$, and each fiber denotes an edge $e \in E$. $N(u)$ is the neighbor set of node u. Define Λ as the set of wavelengths supported by each fiber. On each fiber $e=(u,v)$, there is a set of available wavelengths $\lambda(u,v) \subseteq \Lambda$. $\Lambda = \cup_{(u,v)\in E}\lambda(u,v)$. Let s denote the source node, and $D=\{d_1,d_2,\ldots,d_n\}$ denote the set of n destinations.

Similarly, we define a multicast tree as $T=(V',E')$ rooted at s, where $\{s\}\cup D \subseteq V' \subseteq V$, and $E' \subseteq E$. For a node $u \in V'$, let f_T^u be the parent of u on T, and C_T^u the children set of u. Specifically, $f_T^s=\phi$, and $C_T^u=\phi$ if u is a leaf. Let T_u be a subtree (a branch) cut from T and rooted at u.

To place the minimum number of regenerators and to assign a wavelength on each edge, we introduce some additional notation. A regenerator can only be placed before a node or after a node on a link (which will be illustrated in Subsection III.B), so we define a binary variable $r_v(u)$. If $r_v(u)=1$, it means that we place a regenerator on edge (u,v) at u's side. Otherwise $r_v(u)=0$. Next, let $R_u^{\lambda_i}$ denote the number of regenerators placed on T_u, where λ_i is the wavelength assigned to edge (f_T^u,u). $\mathbf{R_u}=\langle R_u^{\lambda_1},R_u^{\lambda_2},\ldots,R_u^{\lambda_{|\Lambda|}}\rangle$ is a vector containing the number of regenerators placed on T_u with every possible λ_i as u's input. Let $R(T)$ be one possible regenerator placement. Then $R(T)=[r_i(j)]_{|V|\times|V|}$ is a binary matrix. Here $|V|$ is the cardinality (size) of set V. Correspondingly, let $|uv|$ denote the Euclidean distance between u and v. Similarly, let $R(u)$ denote the number of regenerators placed at node u, which can be determined as $R(u)=\sum_{j\in T,j\neq u}\left[r_j(u)+r_u(j)\right]$.

TABLE I
Symbol Definitions for the Network

G	Given topology, $G=(V,E)$
s	Source in G
D	Destination set for multicast
Λ	Possible wavelength set
$\lambda(u,v)$	Available wavelength set for (u,v)
T	A multicast tree rooted at s
$N(u)$	Neighbor set of u
f_T^u	u's parent on T
C_T^u	u's children set on T
T_u	A subtree of T rooted at u
$r_v(u)$	Regenerator placed on edge (u,v) at u's side
$R_u^{\lambda_i}$	Number of regenerators on T_u with $w(f_T^u,u)=\lambda_i$
$\mathbf{R_u}$	Regenerator vector, $\langle R_u^{\lambda_1},R_u^{\lambda_2},\ldots,R_u^{\lambda_{\lvert\Lambda\rvert}}\rangle$
$R(T)$	Binary regenerator matrix
$R(u)$	Number of regenerators placed on node u's side
$w(u,v)$	Wavelength assignment on edge $e=(u,v)$
$W(T)$	Wavelength assignment for T
$\lvert uv\rvert$	Euclidean distance between u and v
$\lvert X\rvert$	Cardinality of set X

Each edge can choose only one possible wavelength, so let $w(u,v)$ denote the chosen wavelength for (u,v). Then $W(T)=[w(i,j)]_{|V|\times|V|}$ is a matrix denoting the wavelength assignment for every edge in T.

We summarize all symbols in Table I.

b) *Energy Assumption*: Let $P_{in}(u)$ be the incoming power of node u, $P_{out}(u)$ be the outgoing power of u, and H be the receiving power threshold for each destination. To make the problem easier, initially, let $P_{in}(s)=1$. According to Fig. 1, various equipment may reduce or increase the transmission power for a data flow. We define a power loss function $F(\cdot)$ to determine how much power is lost or gained when a flow passes through a node, a link, or a regenerator. Table II lists all needed symbols for power functions.

B. Switch Architecture

In this subsection, we introduce the architecture of switches (given by Sahasrabuddhe and Mukherjee in [24]) in the optical network, which helps us to abstract the power loss function $F(\cdot)$. Figure 1 is an example of a commonly used $M\times M$ *multicast-capable all-optical switch* (MC-OXC) which cross connects optical channels directly in the optical domain. (The traffic is bidirectional, and the reverse traffic will pass a group of symmetric elements.)

When the signal arrives on the wavelength λ_a (shown as the dotted line), it will bypass the switch on the same wavelength to the output port through the first stage *optical switch* (OSW) directly after demultiplexing (DEMUX). If the signal needs to go to multiple outputs, a splitter is needed in the MC-OXC. The duplicated signals are treated independently as the incoming signals to the second stage OSW and are switched into the proper outgoing MUX. When local traffic needs to be added, the traffic can be added through the first stage OSW. When the switch is a destination, the traffic can be dropped at the

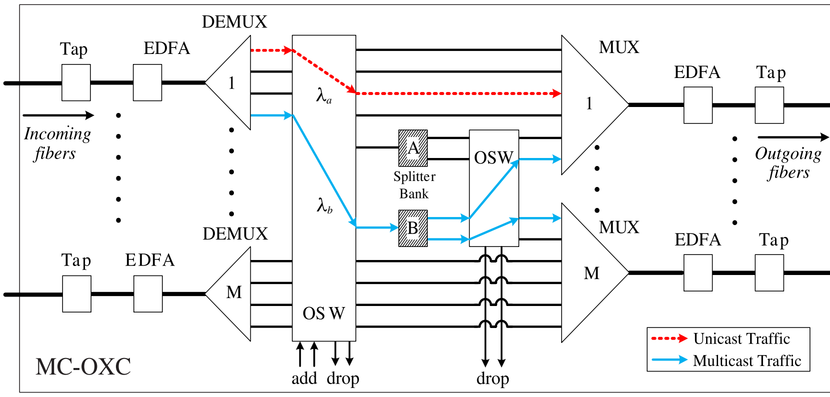

Fig. 1. (Color online) Switch architecture illustration.

TABLE II
SYMBOL DEFINITIONS FOR POWER ISSUES

$P_{in}(u)$	Incoming power of switch u
$P_{out}(u)$	Outgoing power of switch u
$F(u)$	Power loss function through u
$F(u,v)$	Power loss function between u and v
G_{amp}	30 dB, amplifier gain
α	0.3 dB/km, fiber attenuation constant
H	Receiving power threshold
S	Distance between two amplifiers
Q	Power loss coefficient on fibers
L_{OSW}	−8 dB, OSW power loss
L_{DEMUX}	−4 dB, DEMUX power loss
L_{MUX}	−4 dB, MUX power loss
L_{tap}	−1 dB, tap power loss
G_{in}	12 dB, EDFA (amplifier) power gain at input side
G_{out}	12 dB, EDFA (amplifier) power gain at output side
β	$\beta = L_{\text{OSW}} \cdot L_{\text{MUX}} \cdot G_{out} \cdot L_{tap}$
RE	$RE = \frac{1}{2}\log_{\beta} H$, reachability (in terms of hop counts) for the optical signal without regenerators and splitters

first stage OSW (if this switch is a leaf on a light-tree) or the second stage OSW (otherwise). In general, the OSWs may have different insertion loss due to different port counts; however, for simplicity, we assume that all OSWs have the same power loss L_{OSW}.

Due to PI and WCC, 3R regenerators are needed at the MC-OXC in the splitter bank. Similar to converter placement in [25], the regenerators can be placed at different positions. Figure 2 shows examples of possible regenerator positions along with the splitter. In detail, if PI is negligible and WCC are not needed, then no regenerators are needed, which is shown in Fig. 2(a). We place the regenerator before the splitter, shown in Fig. 2(b), if the outgoing wavelengths are the same (λ_b) but are different from the incoming wavelength (λ_a) or if the incoming signal power is not sufficient to guarantee the receiving power level of all subtrees rooted at this node. We only place regenerators after the splitter, shown in Fig. 2(c), if the outgoing wavelength (λ_b) is not the same as the incoming wavelength (λ_a) and some other outgoing wavelengths (λ_a), or if the outgoing signal power is not sufficient to cover all the destinations of the particular subtree rooted at this node. In Fig. 2(d), the regenerators may be placed before and after the splitter to meet the receiving power requirement.

C. Power Loss at the Fiber and the Switch

In this subsection, we discuss the power loss in two places: the fiber and the switch. For the power loss in the fiber (also known as power attenuation along the fiber), the power loss from switch u to switch v can be determined as follows (refer to [17] for details):

$$P_{in}(v) = P_{out}(u) \cdot (G_{amp} e^{-\alpha|uv|})^{\frac{|uv|}{S}} = P_{out}(u) \cdot Q^{|uv|}, \quad (1)$$

where $Q = G_{amp}^{\frac{1}{S}} e^{-\alpha} \leq 1$ is determined by the fiber system. Now we give the power loss function for the fiber:

$$F(u,v) = P_{in}(v) = P_{out}(u) \cdot Q^{|uv|}. \quad (2)$$

Usually, the gain of the amplifier is set to be $G_{amp} = e^{-\alpha S}$ and, therefore, $Q = 1$ and $F(u,v) = P_{in}(v) = P_{out}(u)$.

Now we consider the power loss at the MC-OXC. Figure 3 shows five different roles in which the switch u may act within the light-tree. Shaded nodes form the destination set, and dotted lines denote a virtual node, whose role will be explained as follows.

1) **Add only**: this role only refers to the source s in Fig. 3. If the signal is generated at source s and split into $m = |C_T^s|$ output ports, then, according to Fig. 1, the power loss for each output port, which sends out the multicast traffic, can be determined in two cases: if $|C_T^s| > 1$,

$$F(s) = \frac{L_{OSW_1} \cdot L_{OSW_2} \cdot L_{\text{MUX}} \cdot G_{out} \cdot L_{tap}}{|C_T^s|} P_{in}(s), \quad (3)$$

else if $|C_T^s| = 1$,

$$F(u) = L_{OSW_1} \cdot L_{\text{MUX}} \cdot G_{out} \cdot L_{tap} \cdot P_{in}(s). \quad (4)$$

We assume $L_{OSW_1} = L_{OSW_2} = L_{\text{OSW}}$, $\beta = L_{\text{OSW}} \cdot L_{\text{MUX}} \cdot G_{out} \cdot L_{tap}$ and ignore the impact of one OSW when the signal goes to multiple output ports, since it only has a small effect on $F(u)$ and we may make an additional check at the source node. Then the power loss function can be simplified

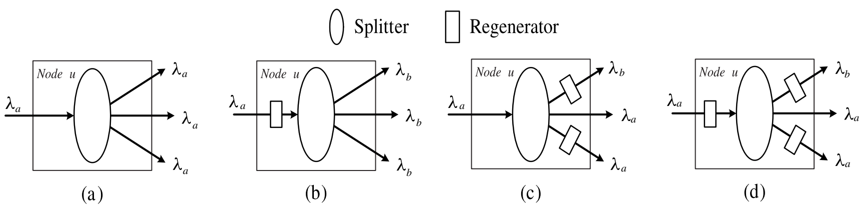

Fig. 2. (Color online) Example for regenerator placement.

as follows:

$$F(u) = P_{in}(s) \cdot \frac{\beta}{|C_T^s|}. \quad (5)$$

For simplicity, in Fig. 3, we use a node to represent a splitter bank at an MC-OXC, and one trapezoid to represent all devices, which include one OSW, one MUX, one EDFA, and one tap, after the splitter bank. When the signal passes through the trapezoid, the power will reduce by β.

2) **Split and pass through**: the second role is for a purely intermediate switch to split the incoming power, such as u_1 in Fig. 3. If the signal arrives at switch u_1 and departs to $m = |C_T^u| \geq 2$ output ports, then the power loss can be determined by

$$F(u) = P_{out}(u) = P_{in}(u) \cdot \frac{\beta^2}{|C_T^s|}, \quad (6)$$

where we assume $G_{in} = G_{out}$ and $L_{\mathrm{DEMUX}} = L_{\mathrm{MUX}}$. Thus we add another trapezoid in Fig. 3 to represent all devices, which include one OSW, one DEMUX, one EDFA, and one tap, before the splitter bank. The power will reduce by β when passing through a trapezoid.

3) **Pass through**: the third role is similar to unicast, such as u_2 in Fig. 3. Let $m = |C_T^u| = 1$, then the power loss function is

$$F(u) = P_{out}(u) = P_{in}(u) \cdot \beta^2. \quad (7)$$

4) **Drop only**: this role may exist only when the switch is the destination, such as u_4 in Fig. 3. The power loss function at the destination is

$$F(u) = P_{out}(u) = P_{in}(u) \cdot \beta. \quad (8)$$

5) **Drop and continue**: the last role refers to the switch which acts as both intermediate and destination, such as u_3 in Fig. 3. The signal will split into $m = |C_T^u| + 1$. Among the m separated signals, $|C_T^u|$ will go to the output ports and the remaining one will be dropped. The total power loss function can be determined as

$$F(u) = P_{out}(u) = P_{in}(u) \cdot \frac{\beta^2}{|C_T^u| + 1}. \quad (9)$$

In order to keep consistent with the previous four cases, we add a virtual switch which connects to the switch u. For example, in Fig. 3, we add virtual node u_3' and connect u_3' to u_3 through a virtual link.

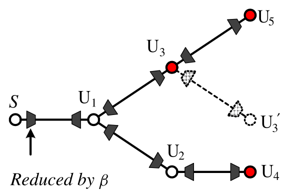

Fig. 3. (Color online) Example for u with different roles for tree T.

Since the signal power may not be sufficient for receivers to detect, we need to place regenerators to compensate for the power loss. Furthermore, if the outgoing wavelength is not the same as the incoming one, a regenerator is needed for wavelength conversion purposes. In detail, we give two rules for placing a regenerator.

- **Rule 1** (Power Sufficient Rule) When the signal arrives at the switch u with power $P_{in}(u)$ from root s along tree T, if $P_{in}(u) < H$, then $R(x) = R(x)+1$, where x is the intermediate switch along the path from s to u (including s and u) on T such that $P_{in}(u) \geq H$.
- **Rule 2** (Wavelength Conversion Rule) At the switch u, if $W(x,u) \neq W(u,y)$, where $e_1 = (x,u) \in T$ and $e_2 = (u,y) \in T$, then $R(x) = R(x) + 1$, which means that either $r_x(u) = 1$ or $r_y(u) = 1$.

If we place a regenerator at switch u, the outgoing signal power is no longer determined by the incoming signal power shown by Eqs. (3)–(9). We recalculate the power loss function according to the different regenerator placements in Fig. 2. Now, we assume the output power of the regenerator is $P_{in}(s)$, and the power loss function at the switch u can be determined based on the following two scenarios.

- If $r_x(u) = 1$ and $r_y(u) = 0$, then

$$F(u) = \frac{\beta}{|C_T^u|} P_{in}(s). \quad (10)$$

- If $r_y(u) = 1$, then

$$F(u) = \beta \cdot P_{in}(s). \quad (11)$$

From Eqs. (3)–(11), we determine the power loss function with and without regenerators and give the rules for regenerator placement.

D. Definition of ERP and WARP

Based on the previous introduction, we can now define the *efficient regenerator placement* (ERP) problem.

Definition 1 (*ERP*). Given a network topology $G=(V,E)$ with source s and destination set D, a wavelength family $\lambda(E)$, a power loss function $F(\cdot)$, and two regenerator placement rules, the *efficient regenerator placement* (ERP) problem is to find a multicast tree T with wavelength assignment $W(T)$ and regenerator allocation $R(T)$ with minimum $|R(T)|$.

If we find a multicast tree T, then it is easy to define WARP as follows.

Definition 2 (*WARP*). Given a multicast tree T with source s and destination set D, a wavelength family $\lambda(E)$, a power loss function $F(\cdot)$, and two regenerator placement rules, the WARP problem is to find wavelength assignment $W(T)$ and regenerator allocation $R(T)$ with minimum $|R(T)|$.

Theorem 1. *The efficient regenerator placement (ERP) problem is NP-hard.*

Proof. The decision form of the ERP problem can be defined as follows.

Given a network topology $G=(V,E)$ with source s and destination set D, a wavelength family $\lambda(E)$, a power loss function $F(\cdot)$, two regenerator placement rules, and an integer K, is there a tree T from source s to all destinations in D with wavelength assignment $W(T)$ and regenerator allocation $R(T)$ based on two placement rules such that $|R(T)| \le K$.

ERP belongs to the NP class since we can guess the tree T with wavelength assignment $W(T)$ and regenerator allocation $R(T)$ in polynomial time and then use the breadth first search (BFS) algorithm to check whether or not T satisfies the two rules and $|R(T)| \le K$. Therefore, ERP belongs to NP.

Now we prove ERP is NP-hard. In [10], the *multi λ-light-tree problem with the objective of minimizing the total number of wavelength converters for a single request* (MWC) is shown to be NP-hard. When we set $\beta=1$, Rule 1 is definitely satisfied and finding the tree T with $W(T)$ and $R(T)$ to minimize $|R(T)|$ is equivalent to MWC. Therefore, MWC is a special case of ERP.

Since ERP belongs to the NP class and MWC is proved to be NP-hard, our ERP problem is also NP-hard. □

IV. Optimal Wavelength Assignment and Regenerator Placement

In this section, we introduce our approximation algorithm for ERP, named SPT-ReWa. This algorithm contains two main subroutines. The first subroutine selects a multicast tree for s and D. Here we use the classical *shortest path heuristic* [26] to construct the tree T rooted at s with the objective of minimizing the total number of links in T. We then use a subroutine named ReWa to solve the WARP problem, which can select an optimal solution within polynomial time.

The main purpose of ReWa is to determine the placement of regenerators and to assign wavelengths on each tree edge when the routing of the tree is given. The first process of the algorithm is the decision process which starts from leaves and ends at the root of the tree. For each node along the tree, 1) the node determines whether or not to place the regenerators on the incoming and outgoing links if and only if all children of this node have already determined the wavelengths and the regenerator placement; 2) the node determines the number of regenerators needed for any available wavelength in the incoming link; 3) the node chooses the wavelengths and determines the placement of regenerators such that the total number of regenerators placed at this node is minimized; 4) if the regenerator is placed, the node relaxes both **Rule 1** and **Rule 2** simultaneously, which is different from either amplifier placement (only relaxing **Rule 1**) or wavelength converter placement (only relaxing **Rule 2**); and 5) the node reports the minimum number of regenerators needed and the wavelength(s) on the incoming link to the parent node along the tree. When the source node determines the regenerator placement, the algorithm will start the second process, which informs all the nodes from the root to the leaves to reserve the wavelengths and to place the regenerator based on their reported information in the first process.

We give the detailed algorithm to achieve the key idea of ReWa in the following subsections. In **ReWa—Phase I**, we make a mathematical transformation for every node; in **ReWa—Phase II**, we collect necessary information from leaf nodes to the root, and calculate the number of regenerators needed; in **ReWa—Phase III**, we assign a wavelength for each edge on the tree and then decide the position of regenerators.

A. Mathematical Transformation (ReWa—Phase I)

Recall that WARP is dealing with a given multicast tree. First, we need to insert some *virtual leaves* into the tree to make the algorithm consistent, say, to satisfy Eq. (9). $\forall u \in V$, if $u \in D$ and $C_T^u \neq \phi$, then set $V = V \cup \{u'\}$, $E = E \cup \{(u,u')\}$, $\lambda(u,u') = \Lambda$, $C_T^{u'} \neq \phi$, and $f_T^{u'} = u$. Let *VL* denote all these us, which is a virtual leaf set.

After inserting new sets into T, the input of ReWa is $T = (V,E)$, $D \subseteq E$, $\lambda(E)$, β, and H. Note that T is modified to include the set *VL*. The output of ReWa is a wavelength assignment $W(T)$ and a regenerator placement $R(T)$.

ReWa is an algorithm running at every node. Say, each node runs a copy of ReWa and sends corresponding information to other nodes. For simplicity, we only illustrate one copy. In Phase I, each node runs independently. We now give **ReWa—Phase I** shown in Algorithm 1.

If u is a leaf in T, then T_u is only one node, so the number of regenerators placed on edge (f_T^u,u) on u's side with available incoming wavelength $\lambda_i \in \lambda(f_T^u,u)$ is 0, and all other wavelengths, which are not available on edge (f_T^u,u), are set to ∞, shown as Line 2. This means that the destination can be connected only through an available wavelength along the edge (f_T^u,u). Moreover, u is either a destination with outgoing power threshold H, or a virtual node with incoming power threshold $H\cdot\beta^2$, to guarantee that the outgoing power of u's parent node, which is the real destination, is H. Thus we can assign corresponding power $P_{out}(u)$ and $P_{in}(u)$ as shown in Line 3 and 4, respectively. Finally, we set all $r_v(u)$ to 0.

Algorithm 1 ReWa—Phase I

if $C_T^u \neq \phi$ **then** // u is a leaf in T
  $R_u^{\lambda_i} = 0$, if $\lambda_i \in \lambda(f_T^u, u)$; otherwise $R_u^{\lambda_i} = \infty$.
  $P_{out}^{\phi}(u) = H \cdot \beta^2$, if $u \in VL$; otherwise $P_{out}^{\phi}(u) = H$.
  $P_{in}(u) = P_{out}^{\phi}(u)/\beta$.
  $r_{\phi}(u) = 0$, $r_{f_T^u}(u) = 0$.
else // u is not a leaf in T
  $R_u^{\lambda_i} = \infty$, $\forall \lambda_i \in \Lambda$.
  $P_{in}(u) = 0$, $P_{out}^{c_i}(u) = 0$, $\forall c_i \in C_T^u$.
  $r_{f_T^u}(u) = 0$, $r_{c_i}(u) = 0$, $\forall c_i \in C_T^u$.
end if

If u is not a leaf in T, then we simply set everything as "undetermined," and then go to the next phase to calculate the value.

B. Bottom-Up Traffic With Regenerator Calculation (ReWa—Phase II)

After the mathematical transformation, we calculate the minimum number of regenerators for T. Phase II is a bottom-up phase starting from leaf nodes, and ending at the source. Each node u will run a copy of Algorithm 2 to determine the number of regenerators for the subtree routed at itself (node u) once it receives necessary information from all its children. There are three types of nodes in the tree T: 1) leaf node, 2) intermediate node, and 3) source node. We discuss each type as follows.

1) Leaf node: the number of regenerators has already been determined by Algorithm 1. Any wavelength $\lambda_i \in \lambda(f_T^u, u)$ can be used, and minimum incoming energy should be larger than or equal to $P_{in}(u)$.

2) Intermediate node: when the node receives necessary information (including incoming wavelength(s) and power of the children) from all its children, it first determines whether a regenerator is needed due to Rule 1 (power sufficient rule). We define a temp set C_T^{u1} for each u, to record the set of u's children for which we put regenerators on link (u, c_i), $c_i \in C_T^u$, and we define set $C_T^{u2} = C_T^u - C_T^{u1}$ to record the other children. The node then calculates the number of regenerators needed for Rule 2 (wavelength conversion rule) similar to the strategy in [10]. The difference is that we assume any outgoing wavelength can be accepted for the children in C_T^{u1} since the regenerator can switch to any wavelength. Finally, the node will pick a set of wavelength(s) with minimum number of regenerators needed for the subtree rooted at node u and will report this set to its parent.

3) Source node: similar to the intermediate node without reporting information to a parent.

ReWa—Phase II can be described in Algorithm 2.

Algorithm 2 ReWa—Phase II

if $C_T^u \neq \phi$ **then** // u is a leaf in T
  send $\mathbf{R_u}$, $P_{in}(u)$ to f_T^u.
else if $f_T^u \neq \phi$ **then** // u is an intermediate node
  Step 1: Determine regenerator based on Rule 1
  for $c_i \in C_T^u$ **do**
    if $P_{in}(c_i) \cdot |C_T^u|/\beta > 1$ **then**
      $r_{c_i}(u) = 1$.
      $C_T^{u1} = C_T^{u1} \cup \{c_i\}$.
    else if $P_{in}(c_i) \cdot |C_T^u|/\beta^3 > 1$ **then**
      $r_{f_T^u}(u) = 1$.
    end if
  end for
  Step 2: Calculate regenerator numbers
  Let $x = \sum_{c_i \in C_T^{u1}} (\min_{1 \le j \le |\Lambda|} R_{c_i}^{\lambda_j} + 1)$.
  for $\lambda_j \in \Lambda$ **do**
    $I = 1$ if $R_{c_i}^{\lambda_j} \neq \infty$, otherwise $I = 0$.
    $y_j = \sum_{c_i \in C_T^{u2}} (R_{c_i}^{\lambda_j} \cdot I + (\min_{1 \le k \le |\Lambda|} R_{c_i}^{\lambda_k} + 1) \cdot \bar{I})$.
  end for
  $y_j^* = \min_{1 \le j \le |\Lambda|} y_j$, λ_j^* is such λ_j to achieve y_j^*.
  Step 3: Determine $P_{in}(u)$ and no. of regenerators
  if $r_{f_T^u} = 1$ **then**
    $P_{in}(u) = H/\beta$.
  else
    $\forall \lambda_i \in \lambda(f_T^u, u)$, $R_u^{\lambda_i} = \min\{1 + x + y_j^*, x + y_i\}$.
    Let $R_u^{\lambda_i^*} = \min R_u^{\lambda_i}$.
    $\forall \lambda_i \in \Lambda$, if $R_u^{\lambda_i} > R_u^{\lambda_i^*}$, then $R_u^{\lambda_i} = \infty$.
    if $\exists \lambda_i$, s.t. $R_u^{\lambda_i} = 1 + x + y_j^*$ **then**
      $P_{in}(u) = H/\beta$.
    else
      $P_{in}(u) = \max_{c_i \in C_T^{u2}} P_{in}(c_i) \cdot |C_T^u|/\beta^2$.
    end if
  end if
  Step 3: Send $\mathbf{R_u}$, $P_{in}(u)$ to f_T^u.
else // u is the source in T
  Step 1: Determine regenerator based on Rule 1
  for $c_i \in C_T^u$ **do**
    if $P_{in}(c_i) \cdot |C_T^u|/\beta > 1$ and $|C_T^u| = 1$**then**
      $r_{c_i}(u) = 1$.
    if $P_{in}(c_i) \cdot |C_T^u|/\beta \cdot L_{\mathrm{OSW}} > 1$ and $|C_T^u| > 1$**then**
      $r_{c_i}(u) = 1$.
      $C_T^{u1} = C_T^{u1} \cup \{c_i\}$.
    end if
  end for
  Step 2: Calculate x, y_j^* similarly
  Step 3: Determine final number of regenerators
  $R_u = x + y_j^*$ for all $\lambda_j \in \Lambda$.
  $\forall c_i \in C_T^u$, if $R_{c_i}^{\lambda_j^*} = \infty$, then $r_{c_i}(u) = 1$.
end if

C. Top-Down Regenerator Placement With Wavelength Assignment (ReWa—Phase III)

When the source determines the number of regenerators and outgoing wavelength, the algorithm starts ReWa—Phase III. In ReWa—Phase II, the regenerators placed based on Rule 1 can be determined. However, we still need to assign

Algorithm 3 ReWa—Phase III

if $f_T^u = \phi$ **then** // u is the source in T
  for $c_i \in C_T^u$ **do**
    if $r_{c_i}(u) = 0$ **then**
      $w(u, c_i) = \lambda_j^*$.
    else
      $w(u, c_i) = \min_{1 \le j \le |\Lambda|} R_{c_i}^{\lambda_j}$.
    end if
    send $w(u, c_i)$ to c_i.
  end for
else if $C_T^u \neq \phi$ **then** // u is an intermediate node in T
  Determine whether to place a regenerator before u
  if $r_{f_T^u}(u) = 0$ & $P_{in}(u) = H/\beta$, **then** $r_{f_T^u}(u) = 1$.
  for $c_i \in C_T^u$ **do**
    if $r_{f_T^u}(u) = 1$ **then**
      $w(u, c_i) = \lambda_j^*$.
    else
      $w(u, c_i) = w(f_T^u, u)$.
    end if
  end for
  Determine whether to place a regenerator after u
  if $r_{c_i}(u) = 0$ & $R_{c_i}^{\lambda_j^*} = \infty$, **then** $r_{c_i}(u) = 1$.
  for $c_i \in C_T^u$ **do**
    if $r_{f_T^u}(u) = 1$ **then** $w(u, c_i) = \min_{1 \le j \le |\Lambda|} R_{c_i}^{\lambda_j}$.
    send $w(u, c_i)$ to c_i.
  end for
end if

a wavelength on each edge and decide other regenerator positions based on Rule 2. Therefore, we design ReWa—Phase III, which is a top-down phase from the source to all leaves. Each node will run a copy of this phase once it receives the information sent from its parent and determine the outgoing wavelength and regenerator position at this node. The detailed algorithm description can be seen in Algorithm 3.

V. Theoretical Analysis

In this section, we first prove the correctness of ReWa by showing that ReWa achieves the minimum number of regenerators placed along the given tree. In order to show that ReWa achieves an optimal solution, we need to prove that 1) ReWa achieves a feasible solution which meets both power constraint (**Rule 1**) and wavelength constraint (**Rule 2**) and 2) ReWa achieves an optimal solution in terms of number of regenerators for any subtree, which can be proved by induction. We then analyze the approximation ratio for SPT-ReWa and provide an example to demonstrate that there exists a case which requires $O(RE) \cdot OPT$ regenerators if we apply SPT-ReWa. Finally, we analyze the time complexity of SPT-ReWa.

Now let us discuss the performance of ReWa.

Theorem 2. *ReWa outputs a feasible solution to set up a light-tree.*

Proof. ReWa—Phase I guarantees that the input power for every destination is at least H. Next, in ReWa—Phase II, the bottom-up calculation ensures successful communication for every intermediate node. Finally, ReWa—Phase III chooses a suitable wavelength on each edge. All three phases together calculate a feasible solution for a tree T to overcome WCC and PI. □

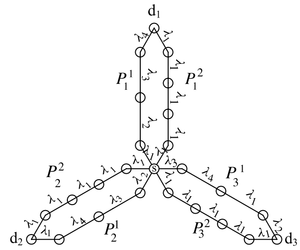

Fig. 4. Special network topology $G(V, E)$.

Theorem 3. *ReWa determines the minimum number of regenerators for T.*

Proof. Proof in Appendix A. □

Next, we analyze the approximation ratio for SPT-ReWa.

Theorem 4. *SPT-ReWa has approximation ratio* $2\log_\beta H = 4RE$.

Proof. Proof in Appendix B. □

Theorem 5. *There exists a case which requires* $O(RE) \cdot OPT$ *regenerators.*

Proof. Consider the following graph G. In the graph G, there are two paths from source s to each destination d_i: one is the path P_i^1 with total length *mRE-1* and with every wavelength different on each link, and the other is the path P_i^2 with total length *mRE* and with all links having the same wavelength. Figure 4 shows an example with $m = 3$ and $RE = 2$.

Obviously, the optimal solution should choose path P_i^2 from source s to destination d_i, which requires m regenerators due to Rule 1. The optimal solution requires $OPT = m \cdot |D|$ regenerators. However, a total of our algorithm will always choose P_i^1 from source s to destination d_i, due to the shortest path heuristic, and the path requires *mRE-1* regenerators. Our approximation algorithm requires a total of $m \cdot |D| \cdot RE - |D| < OPT \cdot RE = O(RE) \cdot OPT$. Therefore, we can create a case for which the approximation algorithm requires $O(RE) \cdot OPT$ regenerators. □

Theorem 6. *The total time complexity for SPT-ReWa is* $O(|D| \cdot |V|^2 + |\Lambda| \cdot |T|)$.

Proof. The time complexity for *SPT* requires at most $O(|D| \cdot |V|^2)$ time, since the minimum hop count path can be determined by BFS [27,28] or Dijkstra's algorithm [29] in at most $O(|V|^2)$. Now we analyze the time complexity for ReWa. Phase I takes $O(|\Lambda| \cdot |T|)$ time to initialize all nodes in the tree T. Phase II needs $O(|\Lambda| \cdot |C_T^u|)$ for each node u on the tree T, and, therefore, the total time complexity for Phase II is $\sum_{u \in T} O(|\Lambda| \cdot |C_T^u|) = O(|\Lambda| \cdot |T|)$. As for Phase III, the time complexity is also bounded by $O(|\Lambda| \cdot |T|)$. Thus, the total time complexity for ReWa is $O(|\Lambda| \cdot |T|)$. The time complexity for SPT-ReWa is $O(|D| \cdot |V|^2 + |\Lambda| \cdot |T|)$. □

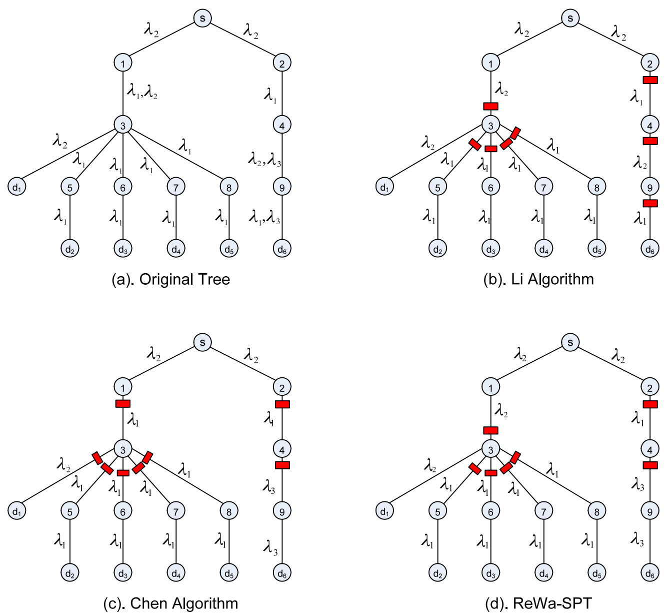

Fig. 5. (Color online) Example of algorithm comparison.

VI. Numerical Results

In this section, we first show an example to demonstrate that ERP is different from problems in which the objective is to minimize the total number of wavelengths or to minimize the number of wavelength converters. We then present some numerical examples to show that the total number of regenerators obtained by SPT-ReWa is close to the optimal solution in a small scale network. We also check the performance of ReWa for a large scale tree with tens to hundreds of nodes and demonstrate that both PI and WCC greatly affect the regenerator placement.

A. Comparison Example

In this subsection, we compare ReWa with Li's [3] and Chen's [10] algorithms. The comparison is illustrated in Fig. 5.

Figure 5(a) gives an original light-tree with the available wavelength set for each edge. Figure 5(b) is the result for Li's algorithm, with 8 regenerators placed as rectangles, and with the wavelength assignment shown on edges. From Fig. 5(b), we find that Li's algorithm reserves a minimum number of wavelengths (2 wavelengths) for the light-tree, but requires 8 regenerators, 7 of which are used as wavelength converters. The remaining regenerator acts as a power compensator.

Figure 5(c) is the result for Chen's algorithm, also with 8 regenerators and a corresponding wavelength assignment. From Fig. 5(c), we find that only 4 regenerators are used as wavelength converters, and another 4 are used to compensate power loss. The system requires 3 different wavelengths.

Figure 5(d) is the result for ReWa, with only 7 regenerators and a corresponding wavelength assignment. From Fig. 5(d), we can see that, due to the two functions of regenerators (wavelength conversion and power compensation), 4 regenerators are used for both functions, 1 only for power compensation, and 2 only for wavelength conversion. Thus fewer regenerators are placed compared to Li's and Chen's algorithms. The system needs 3 different wavelengths.

B. Small Scale Example

In this example, we check the performance of SPT-ReWa and show that it is close to the optimal placement in a small scale network. We use a six-node network, and the topology and the available wavelength set on each link are shown in Fig. 6. Since the network topology is symmetric, we test three cases, and choose Node 1, Node 2, and Node 3 as the source for each case. We also set the number of destinations $|D|$ from 2 to 4. After the source and $|D|$ are determined, we test all combinations (cases) of non-source nodes as the $|D|$ destinations. We set the reachability $RE = \frac{1}{2}\log_\beta H = 2$. For each case, we obtain the optimal solution by emulating all possible routings, applying ReWa to the given tree, and selecting the tree with the minimum number of regenerators. SPT-ReWa determines the routing based on the *shortest path heuristic*. If two paths from the branch node to the destination have the same cost (say, hop count), SPT-ReWa breaks ties with the number of available wavelengths along the path.

We also study the performance of another routing algorithm. We first assign the cost to each link (u,v) as $\frac{1}{\lambda(u,v)}$, which means that a link with a larger number of available wavelengths will have lower cost. We find the shortest path from each destination to the source with minimum cost and form a tree. We name this tree the minimum cost path (MCP) tree. After finding a tree, we apply ReWa to determine the wavelength assignment and regenerator placement. We name this algorithm MCP-ReWa.

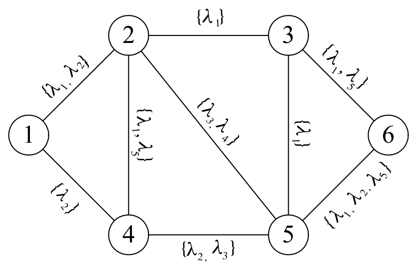

Fig. 6. Topology for simulation.

TABLE III
COMPARISON BETWEEN SPT-REWA, MCP-REWA, AND OPTIMAL SOLUTION

					No. of cases −△		Ratio of △	
s	$\|D\|$	OPT	SPT-ReWa	MCP-ReWa	0	1	0	1
1	2	0.9	1.3	1.7	6	4	64%	36%
	3	1.2	1.6	2.7	6	4		
	4	1.8	2	3.2	4	1		
2	2	0.7	0.9	1.0	8	2	64%	36%
	3	0.9	1.2	1.4	7	3		
	4	1	1.6	1.8	2	3		
3	2	0.7	0.9	1.0	8	2	84%	16%
	3	1.1	1.2	1.4	9	1		
	4	1.4	1.6	2.0	4	1		

Table III shows the average number of regenerators obtained by the optimal algorithm and compares the results of SPT-ReWa ($|R(T)|$) and the optimal solution ($|R(T')|_{opt}$). We define the difference ratio ($\triangle = |R(T)| - |R(T')|_{opt}$) as the number of cases with $\triangle$ over the total cases we test for the given source. Here column *OPT* denotes the average number of regenerators obtained by the optimal algorithm. Columns SPT-ReWa and S-ReWa are the average numbers of regenerators obtained by SPT-ReWa and S-ReWa, respectively. From the table, we can find that the number of regenerators obtained by the optimal algorithm is always smaller than or equal to the solution achieved by SPT-ReWa. As the number of destinations increases, we need more regenerators to guarantee the light-tree establishment due to more power splitting and more links. From the optimal solution, the source Node 2 always requires fewer regenerators than the other two source nodes, since Node 2 has fewer hops to all other nodes (average hop count is 1.2).

We find that SPT-ReWa achieves good results, which are within the approximation ratio, in a small scale network. In more than 60% of the total cases, the number of regenerators obtained by SPT-ReWa is the same as the optimal solution. The rest of the cases only require one more regenerator because SPT-ReWa may choose the routing with minimum hop count but with more regenerators for wavelength conversion purposes. Furthermore, we also find that SPT-ReWa always achieves better results than MCP-ReWa because SPT-ReWa finds the tree with minimum links to reduce the regenerator placement due to **Rule 1**.

C. Large Scale Example

Since we give the approximation ratio in Section V, the total performance of SPT-ReWa for the large scale network can be guaranteed within the approximation ratio. Thus, in this large scale example, we only check the performance of ReWa under two regenerator placement rules for the given large scale tree in 1). We then calculate the number of regenerators for different tree shapes which are determined by the total number of nodes (N) and the splitter capability (B) in 2).

We randomly generate a connected graph (such as the topology studied in our simulation) with 205 nodes and with an average node degree of 4. We randomly distribute the available wavelengths on the links to meet the different available wavelength ratios (AWR). AWR is the average fraction of available wavelengths on each link of the topology. AWR reflects the effect on the current regenerator placement by other existing traffic, such as other multicast demands (light-trees) and unicast demands (light-paths) that reserve wavelengths.

1) *Effect of placement rules on the number of regenerators for a given tree*

We randomly select nodes from the topology to form a tree with N total nodes ($N = 120$). We randomly choose one node as a source. We check each node along the tree to guarantee that each node has a maximum of B ($B = 3$) children, which is the maximum capability for the splitter to split the incoming signal. All leaves of the tree form the destination set D. From D, we randomly select nodes as virtual destination nodes. We apply ReWa for this generated tree and take the result as one experiment.

Figure 7 shows the results obtained by ReWa for generated trees. Each point in the figure is the average of 100,000 experiments. We also show the range with 95% confidence level in the figure. From Fig. 7, we can find that when the reachability is small ($RE = \frac{1}{2}\log_\beta H < 7$), we need more regenerators than when the reachability is large. The reason is that the optical signal degrades quickly when reachability is small and, therefore, the regenerators are used to compensate PI (Rule 1) as well as to relax WCC (Rule 2). When reachability (RE) increases, PI does not significantly affect the transmission, and most regenerators are used to relax WCC. Therefore, when reachability is large, the number of regenerators remains fixed for a given AWR, and, as AWR increases, we need fewer regenerators.

2) *Effect of tree shape on the number of regenerators*

In this example, we set AWR = 40% and test two scenarios: small reachability ($RE = \frac{1}{2}\log_\beta H = 4$) and large reachability ($RE = \frac{1}{2}\log_\beta H = 8$).

Figures 8 and 9 show the results we obtain from ReWa. From Fig. 8, we find that as N increases, more regenerators are needed. Based on the same reason given in 1), these regenerators are used to compensate PI. Since we need to regenerate the signal at most intermediate nodes, the splitter capability (B) does not have a significant effect on the number of regenerators placed on the tree. When the signal can be transmitted further, the splitter capability has a greater effect on the number of regenerators. For example, in Fig. 9, the case

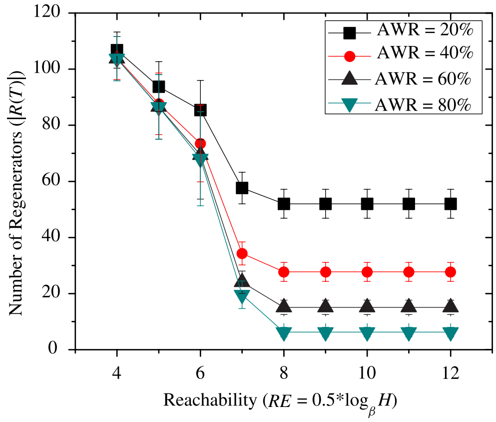

Fig. 7. (Color online) Number of regenerators ($|R(T)|$) versus reachability ($RE = \frac{1}{2}\log_{\beta}H$).

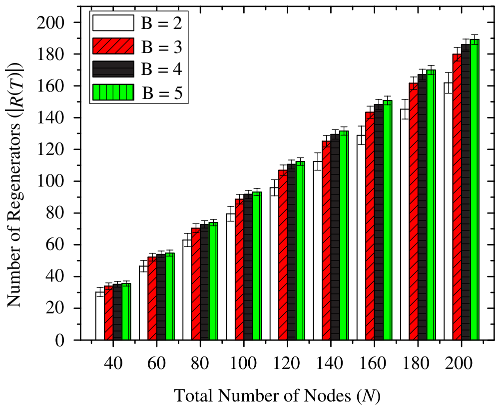

Fig. 8. (Color online) Number of regenerators ($|R(T)|$) versus number of nodes (N): $RE = \frac{1}{2}\log_{\beta}H = 4$.

of $B = 5$ requires almost 100% more regenerators than the case of $B = 2$. The reason is that, although the tree height is smaller when B is larger, splitting the signal at the intermediate node will degrade the signal quality significantly, which requires more regenerators to compensate PI. Therefore, from Figs. 8 and 9, longer reachability ($RE = \frac{1}{2}\log_{\beta}H$) and less splitting (e.g., $B = 3$ and 4) reduces the total number of regenerators.

VII. Conclusion

In this paper, we consider an optimization problem in optical networks for a multicast request with one source and a set of destinations. The goal is to select a multicast tree from a given topology, and then place the minimum number of 3R regenerators and assign an available wavelength on each edge of this tree to set up a light-tree. We name this problem the ERP problem. We found that ERP is NP-hard, and then provided a heuristic named SPT-ReWa, which contains a subroutine ReWa to specifically deal with regenerator placement and wavelength assignment for a given tree. We proved that ReWa can obtain the optimal solution for a given tree, and then provided the approximation ratio for SPT-ReWa, which is $4RE$. We also illustrated several numerical results to exhibit the efficiency of our algorithm, comparing it with other solutions in the literature dealing with similar problems.

One possible future direction is to place the regenerators under different physical impairment measurements, such as Q-factor, BER, or OSNR.

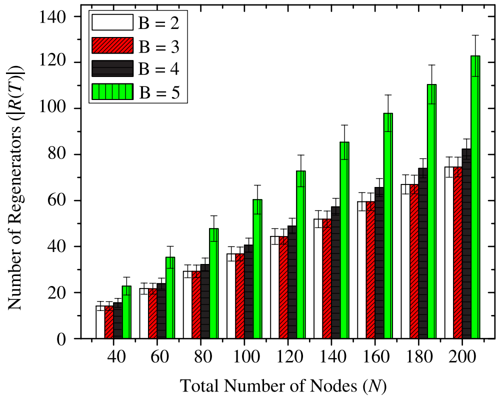

Fig. 9. (Color online) Number of regenerators ($|R(T)|$) versus number of nodes (N): $RE = \frac{1}{2}\log_{\beta}H = 8$.

Appendix A

Appendix A.1 Proof of Theorem 3

In this appendix, we prove that ReWa achieves an optimal solution for a given tree (shown in Theorem 3). We first provide two lemmas which are used to prove the theorem.

Lemma 1. *If $r_{f_T^u}(u) = 1$, then T can be separated into two subtrees, T_s and T_u, where $T_s = T \setminus T_u$. Then we have* $\min R(T) = \min R(T_s) + \min R(T_u) + 1$.

Proof. We know that a regenerator will generate a new signal with power $P_{in}(s)$. This signal can use any available wavelength. Thus, the regenerator allocation for T_s will not influence that of T_u. Then the multicast tree can be divided into two independent subtrees, which reduces the size of our problem. Therefore, the only thing needed is to insert one regenerator on edge (f_T^u, u) at u's side. Then $\min R(T) = \min R(T_s) + \min R(T_u) + 1$. □

Lemma 2. *If $r_v(u) = 1$, where $v \in C_T^u$, then T can be separated as two subtrees T_s and T_{uv}, where $T_{uv} = T_v \cup (u,v)$, and $T_s = T \setminus T_{uv}$. Then we have* $\min R(T) = \min R(T) = \min R(T_s) + \min R(T_u) + 1$.

Proof. Similar to Lemma 1, T can be divided into two subtrees with independent regenerator placement. The only difference is that $P_{out}(u) = \frac{\beta}{|C_T^u|} \cdot P_{in}(s)$. □

Now we prove Theorem 3.

Proof. We prove this theorem by induction on the height of T_u, denoted as $H(T_u)$.

The base case is $H(T_u) = 1$, which means that u is a leaf for T. Based on ReWa, $\forall v \in N(u)$, $r_v(u) = 0$, which is definitely optimal.

Suppose when $H(T_u) = k$, $R(T_u)$ is minimum. For $H(T_u) = k+1$, if we travel bottom-up, then at level $k+1$ there may exist several independent subtrees. We only need to consider one subtree, and the others can be proved similarly. Assume we have a subtree T_u with u at level $k+1$, then each of u's children $c_i \in C_T^u$ forms a tree T_{c_i} with height less than or equal to k. Based on Lemma 1, if $r_u(c_i) = 1$, we can remove T_{c_i} from T_u, which guarantees the optimal property for $R(T_u)$. Similarly, based on Lemma 2, if $r_{c_i}(u) = 1$, we can also remove $T_{c_i} \cup (u,v)$ from T_u. Next, we need to consider the rest of tree T_u, where there are no regenerators placed on each remaining (u,c_i). If ReWa does not output an optimal solution for T_u, then according to the induction hypothesis, the only edges where we can place some redundant regenerators are (u,c_i) (since each T_{c_i} outputs the minimum number of regenerators). However, we already cut all edges that have regenerators, so we cannot place any more regenerators.

Therefore, our algorithm outputs an equivalent result as the optimal solution for level $k+1$, which implies the correctness of ReWa. □

Appendix B

Appendix B.1 Proof of Theorem 4

In this appendix, we prove that ReWa achieves an optimal solution for a given tree (which is shown in Theorem 4). We first provide two lemmas which are used to prove the theorem.

Lemma 3. *Given a network topology $G = (V,E)$, a source s and a destination set D, the lower bound on the number of regenerators can be determined by $|R(T)|_{low} = n \cdot \log_H \beta$, where n is the minimum number of links of the Steiner tree from s to D.*

Proof. If we consider both Rule 1 and 2, we will place more regenerators than if we only consider Rule 1 or 2. Therefore, if we place regenerators just based on Rule 1, it should be the lower bound of ERP.

Any tree T spanning from s to D can be separated into $|D|$ link-disjoint paths represented as $\{P_{B_0,d_1}, P_{B_1,d_2}, \ldots, P_{B_i,d_{i+1}}, \ldots, P_{B_{|D|-1},d_{|D|}}\}$, where $s = B_0, B_1, \ldots, B_i, \ldots, B_{|D|}$ represents the branching nodes of the tree. For each path $P_{B_i,d_{i+1}}$, the power loss without placing any regenerator is at least $\beta^{|P_{B_i,d_{i+1}}|}$. Consider the effect of regenerators. In order to meet threshold H, $|R(P_{B_i,d_{i+1}})| \geq \log_H \beta^{|P_{B_i,d_{i+1}}|} = |P_{B_i,d_{i+1}}| \cdot \log_H \beta$. Therefore, $|R(T)| \geq |T| \cdot \log_H \beta \geq n \cdot \log_H \beta$, where $|T|$ is the total number of links of tree T. □

Lemma 4. *The upper bound of regenerators placed on a shortest path heuristic tree SPT can be determined by $|SPT|$, where $|SPT|$ is the total number of links of SPT.*

Proof. We know that full regenerator placement guarantees the light-tree establishment. Thus it is trivial that $|R(SPT)|_{up} \leq |SPT|$. □

Now we prove Theorem 4.

Proof. Let $|R(T^*)|$ be the number of regenerators placed on an optimal multicast tree T^*, and $|R(SPT)|$ is the number of regenerators placed on a shortest path tree by our algorithm. According to Theorem 3, $|R(SPT)| \leq |R(SPT)|_{up}$. Thus,

$$\frac{|R(SPT)|}{|R(T^*)|} \leq \frac{|R(SPT)|_{up}}{|R(T)|_{low}} \leq \frac{|SPT|}{n \log_H \beta} \leq \frac{|SPT|}{n \log_H \beta}. \quad (12)$$

According to [26], we know that $\frac{|SPT|}{n} \leq 2$. Thus,

$$\gamma = \frac{|R(SPT)|}{|R(T^*)|} \leq \frac{|SPT|}{n \log_H \beta} \leq 2 \cdot \log_\beta H = 4 \cdot RE, \quad (13)$$

which is the approximation ratio for SPT-ReWa. □

Acknowledgment

This work is supported by the National Science Foundation under grants Nos. CCF-0514796 and CNS-0627128.

References

[1] Y. Morita, H. Hasegawa, K. Sato, Y. Sone, K. Yamada, and M. Jinno, "Optical multicast tree construction algorithm considering SNR constraint and 3R regeneration," in *Proc. of ECOC*, 2008, pp. 1–2.

[2] S. Sankaranarayanan and S. Subramaniam, "Comprehensive performance modeling and analysis of multicasting in optical networks," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 9, pp. 1399–1413, 2003.

[3] D. Li, X. Du, X. Hu, L. Ruan, and X. Jia, "Minimizing number of wavelengths in multicast routing trees in WDM networks," *Networks*, vol. 35, no. 4, pp. 260–265, 2000.

[4] T. Durhuus, C. Joergensen, B. Mikkelsen, R. J. S. Pedersen, and K. E. Stubkjaer, "All optical wavelength conversion by SOA's in a Mach–Zehnder Configuration," *IEEE Photon. Technol. Lett.*, vol. 6, no. 1, pp. 53–55, 1994.

[5] M. Hamad and A. E. Kamal, "Efficient power-aware network provisioning for all-optical multicasting in WDM mesh networks," in *Proc. of IEEE GLOBECOM*, 2008, pp. 1–5.

[6] M. Scheffel, "Regenerator allocation strategies for optical transparency domains considering transmission limitations," in *Proc. of IEEE ICC*, 2005, pp. 1771–1776.

[7] S. W. Kim, W. W. Seo, and S. C. Kim, "Regenerator placement algorithms for connection establishment in all-optical networks," in *Proc. of IEEE GLOBECOM*, 2000, pp. 1205–1209.

[8] G. Shen, W. D. Grover, T. H. Chen, and S. K. Bose, "Sparse placement of electronic switching nodes for low blocking in translucent optical networks," *J. Opt. Netw.*, vol. 1, no. 12, pp. 424–441, 2002.

[9] X. Yang and B. Ramamurthy, "Dynamic routing in translucent WDM optical networks: the intradomain case," *J. Lightwave Technol.*, vol. 23, no. 3, pp. 955–971, 2005.

[10] B. Chen and J. Wang, "Efficient routing and wavelength assignment for multicast in WDM networks," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 1, pp. 97–109, 2002.

[11] I. Chlamtac, A. Farago, and T. Zhang, "Lightpath (wavelength) routing in large WDM networks," *IEEE J. Sel. Areas Commun.*, vol. 14, no. 5, pp. 909–913, 2003.

[12] W. Liang and H. Shen, "Multicasting and broadcasting in large WDM networks," in *Proc. of IPPS/SPDP*, 1998, pp. 516–523.

[13] H. Shen, F. Chin, and Y. Pan, "Efficient fault-tolerant routing in multihop optical WDM networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 10, no. 10, pp. 1012–1025, 1999.

[14] T. F. Znati, T. Alrabiah, and R. Melhem, "Low-cost, delay-bounded point-to-point communication to support multicasting over WDM networks," *Comput. Netw.*, vol. 38, no. 4, pp. 423–445, 2002.

[15] http://en.wikipedia.org/wiki/Power_budget.

[16] R. Libeskind-Hadas, "Efficient collective communication in WDM networks with a power budget," in *Proc. of IEEE ICCCN*, 2000, pp. 612–616.

[17] Y. Xin and G. N. Rouskas, "Multicast routing under optical layer constraints," in *Proc. of IEEE INFOCOM*, 2004, pp. 2731–2742.

[18] R. Libeskind-Hadas and R. Melhem, "Multicast routing and wavelength assignment in multihop optical networks," *IEEE/ACM Trans. Netw.*, vol. 10, no. 5, pp. 621–629, 2002.

[19] S. Gao, X. Jia, X. Hu, and D. Li, "Wavelength requirements and routing for multicast connections in lightpath and light-tree models of WDM networks with limited drops," *IEE Proc.-Commun.*, vol. 148, no. 6, pp. 363–367, 2001.

[20] X. D. Hu, T. P. Shuai, X. Jia, and M. H. Zhang, "Multicast routing and wavelength assignment in WDM networks with limited drops," in *Proc. of IEEE INFOCOM*, 2004, pp. 487–494.

[21] A. M. Hamad and A. E. Kamal, "Routing and wavelength assignment with power aware multicasting in WDM networks," in *Proc. of IEEE BROADNETS*, 2005, pp. 31–40.

[22] S. Pachnicke, T. Paschenda, and P. M. Krummrich, "Physical impairment based regenerator placement and routing in translucent optical networks," in *Proc. of OSA OFC/NFOEC*, 2008, pp. 1–3.

[23] Y. Peng, W. Hu, W. Sun, X. Wang, and Y. Jin, "Impairment constraint multicasting in translucent WDM networks: architecture, network design and multicasting routing," *Photonic Network Commun.*, vol. 13, no. 1, pp. 93–102, 2007.

[24] L. 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, 1999.

[25] Y. Yang, J. Wang, and C. Qiao, "Nonblocking WDM multicast switching networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 11, no. 12, pp. 1274–1287, 2000.

[26] H. Takahashi and A. Matsuyama, "An approximate solution for the steiner problem in graphs," *Math. Japonica*, vol. 24, no. 6, pp. 573–577, 1980.

[27] Edward F. Moore, "The shortest path through a maze," in *Proc. of the Int. Symp. on the Theory of Switching*, 1959, vol. 2, pp. 285–292.

[28] C. Y. Lee, "An algorithm for path connection and its applications," *IRE Trans. Electron. Comput.*, vol. 10, no. 3, pp. 346–365, 1961.

[29] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numer. Math.*, vol. 1, no. 1, pp. 269–271, 1959.