Multicast Routing and Distance-Adaptive Spectrum Allocation in Elastic Optical Networks With Shared Protection

Anliang Cai, Jun Guo, *Member, IEEE*, Rongping Lin, Gangxiang Shen, *Senior Member, IEEE*, and Moshe Zukerman, *Fellow, IEEE*

***Abstract*—We consider an elastic all-optical network, where each node is multicast-capable and does not support spectrum conversion. In such a network, for a given set of static multicast demands, we consider distance-adaptive spectrum resource allocation, and aim to optimize multicast routing, modulation, and spectrum assignment with shared protection in a way that minimizes the required spectrum resources for accommodating all multicast sessions. In our design, we provision each multicast demand by a light-tree where spectrum resources are allocated in all links included in the tree. We protect each light-tree from any single link failure in both directions by having a backup path that is link-disjoint to the path from the source to each destination on the primary tree. We reserve spectrum resources in the links that are not in the primary tree but in the backup paths between all source-destination pairs. The reserved spectrum resources can be shared to protect multiple light-trees as long as they do not fail simultaneously. For such a problem, we provide a mixed integer linear programming formulation. We also develop a scalable heuristic algorithm with an attribute that enables it to improve the quality of the results at the cost of longer running times. Numerical results for small problems show that the heuristic algorithm performs close to the optimum. In addition, we use a Markov chain simulation of the network to evaluate the performance of our proposed algorithm in terms of blocking probability in a dynamic environment, which demonstrates a significant improvement over straightforward approaches.**

***Index Terms*—Distance-adaptive transmission, elastic optical network (EON), multicast routing, shared protection, spectrum assignment.**

I. Introduction

The elastic optical network (EON) [2]–[6] based on the *coherent optical orthogonal frequency-division multiplexing* technique [7] has recently attracted attention as a viable means for further increasing efficiency and data rate of optical networks. It allows partial spectrum overlapping and flexible bandwidth allocation in a finer granularity, where multiple low-data-rate subcarriers, or frequency slots (FSs), are adequately allocated to a connection. This provides better flexibility in spectrum assignment and higher spectrum utilization than traditional wavelength-division multiplexing (WDM) optical networks [8] that follow the ITU-T fixed frequency-grid standard. A considerable effort has been made to tackle the problem of routing and spectrum assignment (RSA) in EONs [9], [10]. Recently, an important attribute in EONs, known as *distance-adaptive spectrum resource allocation*, has been proposed [11]. Distance-adaptive transmission entails the choice of a modulation scheme (MS) for a signal that is adaptive to the transmission distance of the optical path. Thus, the RSA problem is extended into a more flexible and complex problem involving routing, modulation and spectrum assignment (RMSA) [12]–[14].

A. Background

Multicast services, including synchronization of databases among geographically distributed datacenters and ultra-high-definition TV delivery, are gaining popularity. Such point-to-multi-point applications are typically bandwidth-intensive. For example, uncompressed ultra-high-definition TV with 8K resolution requires 72 Gb/s, and 3-D high-definition TV would require even higher bit rate [15]. Compared to IP multicasting, multicasting in the optical layer based on the WDM technology is more spectrum- and power-efficient [16]. Extensive studies on multicast routing and wavelength assignment have been conducted for WDM optical networks [17]–[19]. As a promising candidate of the next-generation optical transport network, the EON demonstrates better performance in multicast service provisioning than the traditional WDM optical network [20].

As an extended concept of lightpath, a light-tree is an optical channel from a source to multiple destinations [16]. Recent studies focus on provisioning multicast connections using the light-tree technology [21]–[25], and experiments have been demonstrated for EONs in [26]. Similar to lightpath-based RMSA in EONs, the light-tree-based *multicast-capable* RMSA (MC-RMSA) problem does not consider *spectrum conversion* and is required to satisfy three constraints, namely, *spectrum continuity*, *spectrum contiguity*, and *spectrum non-overlapping*. The spectrum continuity constraint ensures that the same FSs should be utilized in all fiber links that are included in a light-tree. The spectrum contiguity constraint ensures that the FSs in

Manuscript received February 22, 2016; revised June 02, 2016; accepted June 21, 2016. Date of publication July 18, 2016; date of current version August 12, 2016. This work was supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China [CityU 11216214] and a grant from the National Natural Science Foundation of China [NSFC 61401070]. Part of this paper was presented in ACP 2015 [1].

A. Cai, J. Guo, and M. Zukerman are with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong (e-mail: anliancai2-c@my.cityu.edu.hk; j.guo@cityu.edu.hk; m.zu@cityu.edu.hk).

R. Lin is with the School of Communication & Information Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: linrp@uestc.edu.cn).

G. Shen is with the School of Electronic and Information Engineering, Soochow University, Suzhou 215006, China (e-mail: shengx@suda.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JLT.2016.2592999

each fiber link assigned to a light-tree are contiguous. Then, the spectrum non-overlapping constraint prevents any FS in a fiber link from being allocated to two or more connections.

Protection has been considered an important attribute in optical networks. Fiber cuts often occur to an optical network and even a single fiber cut could result in severe service disruption. Therefore, adequate protection should be preplanned for optical networks so that they can continue to operate under failures. This is especially true when a trunk of the multicast tree fails, in which case multiple destinations are affected and could not receive data from the source. This highlights the importance of incorporating multicast survivability in the design of optical networks. Protection in the optical layer is beneficial in many aspects, e.g., faster recovery speed and simpler operation than that in higher layers [27]. However, existing research rarely focuses on multicast protection or survivability in EONs [28], [29]. In [28], the authors used multiple lightpaths to support a multicast connection and proposed a dedicated path protection scheme. Our approach in this paper is based on shared protection that is more spectrally efficient than dedicated protection [30]. The authors of [29] considered shared protection for multicast connections. However, they did not consider distance-adaptive spectrum resource allocation, which is our focus in the present paper. A brief (3-page) and preliminary version of this paper was presented in [1].

B. Key Contributions

To the best of our knowledge, there is no published work on light-tree-based MC-RMSA with shared protection in EONs considering distance-adaptive spectrum resource allocation. In this paper, we address this important problem. Specifically, we focus on the problem for static traffic where a set of multicast demands are given *a priori*. Each multicast session is supported by a light-tree with the assumption that all nodes in the network are MC by adopting splitter-and-delivery switches [31]. In addition, we consider a shared protection scheme to protect the light-trees from any single link failure. Our objective is to minimize the required spectrum resources subject to the condition that all the given multicast demands are accommodated. We present a mathematical formulation in the form of mixed integer linear programming (MILP) for the problem. Since MILP is known to be computationally prohibitive for large networks, we also develop an efficient heuristic algorithm with an attribute that enables it to improve the quality of the results at the cost of longer running times. The effectiveness of the proposed algorithm is demonstrated by extensive numerical results and a Markov chain simulation of the network in a dynamic environment.

C. Organization

The remainder of this paper is organized as follows. In Section II, we review the related work. In Section III, we provide a detailed description of the problem considered in this paper. The MILP formulation is presented in Section IV, and the heuristic algorithm is presented in Section V. Numerical results and discussions are provided in Section VI. Section VII concludes the paper.

II. Related Work

Extensive studies on multicasting in optical networks have been reported in the literature. When all or part of the network nodes are based on the multicast-incapable tap-and-continue switch [32], solutions based on the light-trail technology [33], [34] have been proposed for provision of multicast demands [35]–[39]. In WDM networks where there is a mix of MC and multicast-incapable nodes, it was demonstrated in [40] that the light-trail-based solution, namely, light-hierarchy, achieves a lower blocking probability than the one based on the light-tree technology. Also, the light-trail-based solutions have an advantage in grooming sub-wavelength connections in WDM networks [37].

In this paper, we focus on using the light-tree technology to provision multicast demands under the assumption that all network nodes are MC. We note that, for any light-trail-based solution associated with a multicast connection in a network that contains multicast-incapable nodes, there is an equivalent light-tree-based solution in the corresponding MC network. In our context, since we consider flexi-grid networks where just-enough FSs are allocated to each demand, and since the FS capacity can be small relative to the high bit rate applications that we consider, the weakness of light-tree in grooming may not be a major issue. As a result, a light-tree-based solution has an advantage over a light-trail-based solution to provision high bit rate demands in MC EONs as we consider in the present paper.

For the MC-RMSA problem in EONs, Gong *et al.* [21] and Walkowiak *et al.* [22] proposed integer linear programming (ILP) formulations and heuristic algorithms based on the light-tree technology. Ruiz and Velasco [23] evaluated three provisioning schemes for multicast connections, namely, path, tree, and subtree schemes. Walkowiak *et al.* [24] investigated the impact of the degree limitation of a multicast tree node and the transmission distance implications of optical signal splitting. Yu *et al.* [25] considered a network with modulation-enabled nodes, where the MSs of the input and output signals can be different. While the above studies mainly aimed at the optical layer, Ruiz and Velasco [41] also considered serving multicast demands in multiple layers.

Numerous studies have been conducted on multicast protection in WDM optical networks. Singhal *et al.* [30] proposed self-sharing schemes for protecting a multicast session from any single link failure. Singhal *et al.* [42] further proposed the more spectrally efficient cross-sharing scheme for protecting multiple multicast sessions from any link failure. Constantinou *et al.* [43] proposed a directed-graph multicast protection scheme based on the minimum-cost path heuristic (MPH) [44]. Network coding has been considered in [45] for provisioning multicast connections with 1+1 dedicated protection. In this paper, we address the important problem of MC-RMSA in EONs considering shared protection for multiple multicast sessions.

TABLE I
TRANSPARENT REACH AND CAPACITY PER FS FOR EACH MS [14]

MS	Transparent Reach [km]	Capacity per FS [Gb/s]
BPSK	4000	12.5
QPSK	2000	25
8QAM	1000	37.5

III. MC-RMSA WITH SHARED PROTECTION

In this section, we begin by providing the model of the EON considered in this paper. Then, we describe the concept of distance-adaptive transmission and discuss its implications in spectrum resource allocation in EONs. Next, we present the shared protection scheme used in our context for protecting multiple multicast sessions in EONs. Finally, we provide the statement of the MC-RMSA problem with shared protection.

A. Network Model

The EON is represented by a directed graph $G = (\boldsymbol{V}, \boldsymbol{L})$, where $\boldsymbol{V}$ is the set of nodes and $\boldsymbol{L}$ is the set of directed links. Every two adjacent nodes i and j are connected by two directed links in opposite directions, each of which corresponds to a unidirectional fiber link, denoted by (i, j) for the one from node i to node j and (j, i) for the one from node j to node i. Let ℓ_{ij} denote the weight of fiber link (i, j) representing its physical length. We assume $\ell_{ij} = \ell_{ji}$ for all $(i, j) \in \boldsymbol{L}$ without loss of generality. Let $\mathcal{G}$ denote the bandwidth of an FS in units of GHz. The set of MSs is $\boldsymbol{M}$. For $m \in \boldsymbol{M}$, we denote by τ_m the transparent reach of a signal modulated by MS m, and by $\mathcal{C}_m$ the capacity per FS modulated by MS m.

B. Distance-Adaptive Transmission

To efficiently utilize the spectrum resources, we consider distance-adaptive spectrum resource allocation in EONs, where minimum spectrum resources are adaptively allocated to an all-optical channel according to its physical condition [46]–[48]. As in [14], [46], the channel condition is measured by its physical transmission distance. In this context, a more spectrally efficient (or higher-order) MS implies fewer spectrum resources for serving a demand but is associated with a shorter transparent reach. As an example, Table I provides the transparent reach and the capacity per FS of three MSs, namely, BPSK, QPSK and 8QAM. The values are used in [14] based on [46].

If the transmission distance of the connection is longer than the transparent reach of the chosen MS, regenerators are required to regenerate the signal. This indicates a fundamental tradeoff between the cost of the transparent reach and the spectrum usage in choosing an MS. Accordingly, in distance-adaptive spectrum resource allocation, the highest-order available MS is chosen so that the transmission distance of the connection does not exceed the transparent reach of the MS. In this way of distance-adaptive transmission [5], the number of regenerators required in the network can be minimized, while the spectrum usage is kept at a reasonable level.

C. Shared Protection

A light-tree for a multicast session is a unidirectional connection where an optical signal is transmitted from a given source to multiple destinations. Under normal operations where no failure occurs, such a tree is called a primary tree, which can be viewed as a set of primary paths between the source and each destination. A primary link is a link in a primary tree. One protection scheme to protect a light-tree is by having each of its primary paths protected via a link-disjoint backup path. The backup paths for the corresponding source-destination (SD) pairs can share spectrum resources in their common link(s). The advantages of such path-based protection have been recognized in [30].

To protect a light-tree, there are two mutually exclusive cases in the protection scheme. One case is that the backup path of any SD pair shares no common link with the primary tree. The other case is that at least one of the backup paths shares common link(s) with the primary tree. However, any common link must be a link in the primary paths of other SD pairs since, for each SD pair, the backup path and the primary path are link-disjoint. This is known as self-sharing protection [42]. Thus, a light-tree can be protected as long as for each SD pair of the multicast session, a backup path is link-disjoint to its primary path, and we reserve spectrum resources in the backup-only links that are not in the primary tree but in the backup paths of all SD pairs. These reserved spectrum resources could be utilized when the primary tree fails.

For the case of protecting multiple light-trees, the reserved spectrum resources in the backup-only links can be shared as long as these light-trees do not fail simultaneously. Such shared protection among multiple light-trees is known as cross-sharing [42].

In this paper, we consider four types of links that are included for serving a multicast connection, namely, primary-only link, backup-only link, self-sharing link, and cross-sharing link. For a multicast connection, a primary-only link is a link included only in the primary tree. Similarly, a backup-only link is a link included only as backup, while a self-sharing link is a link that is included in both primary and backup paths of different SD pairs. Hence, primary-only, backup-only, and self-sharing links are mutually exclusive. A cross-sharing link is a link that is included in multiple multicast connections only as backup. Thus, a cross-sharing link for multiple multicast connections is a backup-only link in each of these multicast connections, while a backup-only link may not be a cross-sharing link.

Fig. 1 illustrates the shared protection scheme that we consider in this paper based on the above basic concepts. For the example network of Fig. 1(a), Figs. 1(b) and (c) show the routing case of two multicast sessions, i.e., $M1 = \langle A; \{B, C\} \rangle$ and $M2 = \langle B; \{C, D\} \rangle$, respectively. Here, a multicast session is denoted as $\langle s; \{d_1, d_2, \ldots, d_n\} \rangle$ where s is the source node, and $\{d_1, d_2, \ldots, d_n\}$ are the set of destination nodes. As shown in Fig. 1(b), the primary tree of $M1$ contains links (A, B) and (A, C). The backup path to destination node B and that to

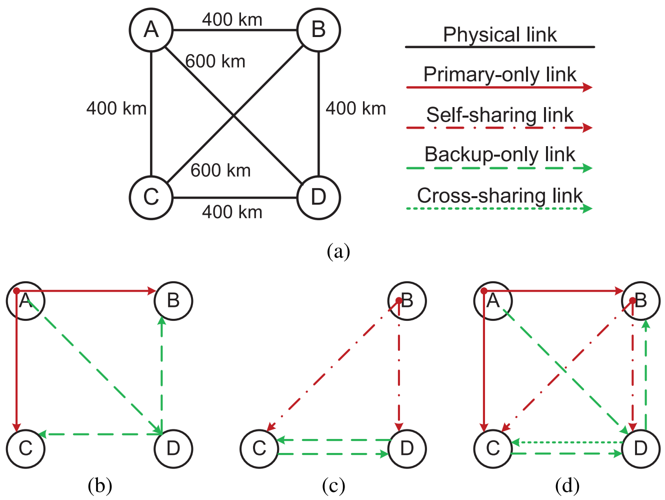

Fig. 1. An example of the shared protection scheme: (a) a four-node fully-mesh network; (b) routing for $M1 = \langle A; \{B, C\}\rangle$; (c) routing for $M2 = \langle B; \{C, D\}\rangle$; and (d) sharing between $M1$ and $M2$.

destination node C are $A \to D \to B$ and $A \to D \to C$, respectively. These two backup paths share a common link (A, D), but share no common link with the primary tree. In this case, a link included is either a primary-only link or a backup-only link, and there are no self-sharing links. For $M2$, as shown in Fig. 1(c), the primary tree contains links (B, C) and (B, D). To protect the primary path $B \to C$ from a single link failure, a link-disjoint backup path $B \to D \to C$ is found, which uses the link (B, D) in the primary path $B \to D$. Similarly, a link-disjoint backup path $B \to C \to D$ is found for the primary path $B \to D$, which uses the link (B, C) in the primary path $B \to C$. In this case, both (B, C) and (B, D) are self-sharing links. Links (C, D) and (D, C) are backup-only links, where the reserved spectrum resources could be used to protect other multicast connections by cross-sharing. In Fig. 1(d), link (D, C) is a backup-only link for both $M1$ and $M2$, so the reserved spectrum resources in the link can be used to protect both multicast connections. This is because their primary connections contain no common links and will not fail simultaneously in case of a single link failure, and therefore there is no competition on utilizing the reserved resources in the cross-sharing link for service restoration.

D. Problem Statement

In this paper, we focus on the MC-RMSA problem with shared protection for static traffic. Specifically, we consider a set of multicast demands $\boldsymbol{R}$. For $r \in \boldsymbol{R}$, the multicast demand r is denoted by $\langle s_r; \boldsymbol{F}_r; t_r\rangle$, requesting a data transmission from the source s_r to the set of destinations $\boldsymbol{F}_r$ at bit rate t_r. Let ω_m^r denote the number of FSs to be assigned to serve the multicast demand r for the required bit rate given that MS m is utilized. By definition, we have $\omega_m^r = \lceil t_r/\mathcal{C}_m \rceil$ as presented in [14], where $\lceil x \rceil$ is the smallest integer that is greater than or equal to x. For instance, based on the values of MSs in Table I, an FS modulated by QPSK has a capacity $\mathcal{C}_{\text{QPSK}} = 25$ Gb/s and a transparent reach $\tau_{\text{QPSK}} = 2000$ km. We can observe from Table I that, for an example multicast connection requesting $t_r = 45$ Gb/s and a transmission distance of 1,800 km, the number of required FSs given that QPSK is utilized is two, i.e., $\omega_{\text{QPSK}}^r = 2$. Here, the highest-order MS that can be assigned to this particular connection is QPSK since $\tau_{\text{QPSK}} > 1800$ km but $\tau_{8\text{QAM}} < 1800$ km.

Based on the network model described above, we further make the following assumptions. We assume that the EON does not support spectrum conversion. Thus, the spectrum continuity constraint must be satisfied. To support distance-adaptive spectrum resource allocation, all transponders are central frequency tunable and also MS tunable. Each node is MC based on the splitter-and-delivery switch [31]. An input signal arriving at such an MC node can be dropped locally and/or switched to one, many, or all of its output ports. We assume that the power loss of the signal due to the splitting is compensated by amplifiers in the node. For simplicity, in this paper we do not consider the use of regenerators. We assume that every SD pair has at least one pair of disjoint paths, of which the transmission distance is within the transparent reach of the lowest-order MS considered, e.g., 4000 km for BPSK as in Table I. This assumption is made for simplicity and extension to larger networks that involve regenerators is left for future work.

In this case, to implement the shared protection scheme where spectrum resources used in a link of the primary path between an SD pair can be utilized to protect another SD pair within the same multicast session, we consider that the same spectrum resources (modulated by the same MS) are allocated in each link of the primary tree and reserved in the backup-only links of the multicast connection. Such consideration also implies faster recovery and transponder savings. Thus, for MS assignment, all primary and backup paths should be taken into account in the transparent reach constraint. For example, for the multicast session $M2$ in Fig. 1(c), the backup paths of the primary paths $B \to C$ and $B \to D$ are $B \to D \to C$ and $B \to C \to D$, respectively. The longest transmission distance among all four paths is 1000 km, thus the MS assigned to the multicast connection of $M2$ should have a transparent reach of no less than 1000 km.

The objective of the MC-RMSA problem with shared protection is to minimize the maximum spectrum among the required spectrum in all links subject to the condition that all the given multicast demands are accommodated with the consideration of distance-adaptive spectrum resource allocation. The light-tree of each multicast session is protected on an SD pair basis, where for each SD pair of the multicast connection the primary path is protected by a link-disjoint backup path from any single link failure in both directions. This means that the network can continue to operate when only one link fails at a time.

IV. MILP Formulation

To solve the MC-RMSA problem with shared protection, we provide here the MILP formulation. Let Δ be a large number. We use a set, $\boldsymbol{O}$, of operation indicator as presented in the following. To find a primary and a backup paths for each SD pair for protection concern, we set $\boldsymbol{O} = \{1, 2\}$, where the variables with a superscript of elements 1 and 2 are for the primary and backup connection, respectively. We use a general term, i.e., *routing subgraph*, to denote the primary tree and backup paths for all SD pairs used to serve a protected multicast demand. Here for each SD pair, the primary path and the backup path does not share any common link, not even a link in opposite directions.

A. Variables

- $P_{d,ij}^{r,o}$ Binary; equals one if a path to destination d, $d \in \boldsymbol{F}_r$, of multicast connection r, $r \in \boldsymbol{R}$, traverses fiber link (i,j), $(i,j) \in \boldsymbol{L}$, for operation (primary or backup) indicator o, $o \in \boldsymbol{O}$; zero, otherwise.
- X_{ij}^r Binary; equals one if fiber link (i,j), $(i,j) \in \boldsymbol{L}$, is included to serve multicast connection r, $r \in \boldsymbol{R}$.
- Y_{ij}^r Binary; equals one if fiber link (i,j), $(i,j) \in \boldsymbol{L}$, is a primary link of multicast connection r, $r \in \boldsymbol{R}$.
- D_r Real; denotes a distance that is longer than or equal to the longest distance among all the paths included in multicast connection r, $r \in \boldsymbol{R}$.
- N_r Integer; denotes the number of FSs in primary links allocated to multicast connection r, $r \in \boldsymbol{R}$; $N_r \geq 0$.
- K_m^r Binary; equals one if MS $m, m \in \boldsymbol{M}$, is assigned to multicast connection r, $r \in \boldsymbol{R}$; zero, otherwise.
- S_r Integer; denotes the starting FS index of multicast connection r, $r \in \boldsymbol{R}$; $S_r \geq 1$.
- E_r Integer; denotes the ending FS index of multicast connection r, $r \in \boldsymbol{R}$; $E_r \geq 1$.
- T_{ij}^r Integer; denotes a number that is no smaller than the number of FSs in fiber link (i,j), $(i,j) \in \boldsymbol{L}$, assigned to multicast connection r, $r \in \boldsymbol{R}$; $T_{ij}^r \geq 0$.
- H_{r2}^{r1} Binary; equals one if a fiber link included to serve multicast connection $r1$, $r1 \in \boldsymbol{R}$, is a primary link of another multicast connection $r2$, $r2 \in \boldsymbol{R}$, where $r1 \neq r2$.
- Z_{r2}^{r1} Binary; equals zero if the ending FS index E_{r2} of multicast connection $r2$, $r2 \in \boldsymbol{R}$, is smaller than the starting FS index S_{r1} of another multicast connection $r1$, $r1 \in \boldsymbol{R}$, where $r1 \neq r2$, i.e., $S_{r1} \geq E_{r2} + 1$.
- C Integer; maximum number among the numbers of required FSs in all fiber links.

B. Objective

$$\textbf{Minimize } \mathcal{G} \cdot C \tag{1}$$

The objective is to minimize the maximum spectrum among the required spectrum in all fiber links in the entire network.

C. Constraints

The constraints can be classified into five groups. The first group of constraints is called *searching and constructing a routing subgraph*, where the constraints ensure that a pair of link-disjoint paths is found for each SD pair of a multicast connection and that a routing subgraph for the multicast connection is constructed by these paths for all SD pairs. The second group is called *modulation determination*. It consists of constraints ensuring that an MS is assigned to each multicast connection, and the transparent reach constraint is met. The third is the *spectrum assignment* group, where the constraints guarantee that a sufficient number of FSs are assigned to each multicast connection, and that the three constraints, namely, spectrum continuity, spectrum contiguity, and spectrum non-overlapping, are all met. The fourth is a group of *redundancy constraints* for faster solutions. The last group provides a *lower bound* on the number of FSs required in all links.

1) Searching and Constructing a Routing Subgraph

$$\sum_{(i,j)\in \boldsymbol{L}} P_{d,ij}^{r,o} = \begin{cases} 1, & i = s_r \text{ or } j = d, \\ 0, & j = s_r \text{ or } i = d, \end{cases} \forall o \in \boldsymbol{O}, r \in \boldsymbol{R}, d \in \boldsymbol{F}_r \tag{2}$$

$$\sum_{(i,x)\in \boldsymbol{L}} P_{d,ix}^{r,o} = \sum_{(x,j)\in \boldsymbol{L}} P_{d,xj}^{r,o}, \forall o \in \boldsymbol{O}, r \in \boldsymbol{R}, d \in \boldsymbol{F}_r, \quad x \in \boldsymbol{V} \setminus \{s_r, d\} \tag{3}$$

$$\sum_{o\in \boldsymbol{O}} (P_{d,ij}^{r,o} + P_{d,ji}^{r,o}) \leq 1, \quad \forall r \in \boldsymbol{R}, d \in \boldsymbol{F}_r, (i,j) \in \boldsymbol{L} \tag{4}$$

$$\Delta \cdot Y_{ij}^r \geq \sum_{d\in \boldsymbol{F}_r} P_{d,ij}^{r,1}, \quad \forall r \in \boldsymbol{R}, (i,j) \in \boldsymbol{L} \tag{5}$$

$$\sum_{(i,j)\in \boldsymbol{L}} Y_{ij}^r \leq 1, \quad \forall r \in \boldsymbol{R}, j \in \boldsymbol{V} \tag{6}$$

$$\Delta \cdot X_{ij}^r \geq \sum_{d\in \boldsymbol{F}_r, o\in \boldsymbol{O}} P_{d,ij}^{r,o}, \quad \forall r \in \boldsymbol{R}, (i,j) \in \boldsymbol{L}. \tag{7}$$

Constraints (2) and (3) guarantee that the flow conservation requirement is met. They are used to search routing paths for all SD pairs of a multicast connection. Constraint (4) ensures that the primary and the backup paths for each SD pair of a multicast connection do not share common link(s), not even a link in opposite directions, i.e., they are link-disjoint. However, paths from different SD pairs could share common link(s). Constraint (5) ensures that a fiber link included in a primary path is a primary link included for the multicast connection. Constraint (6) guarantees that a primary tree is constructed by the primary paths by ensuring that each node in a tree has only one ingress fiber link. Constraint (7) ensures that a fiber link included in any path is used to serve the multicast connection.

2) Modulation Determination

$$\sum_{m\in \boldsymbol{M}} K_m^r = 1, \quad \forall r \in \boldsymbol{R} \tag{8}$$

$$D_r \geq \sum_{(i,j)\in \boldsymbol{L}} \ell_{ij} \cdot P_{d,ij}^{r,o}, \quad \forall o \in \boldsymbol{O}, r \in \boldsymbol{R}, d \in \boldsymbol{F}_r \tag{9}$$

$$\tau_m - D_r \geq \Delta \cdot (K_m^r - 1), \quad \forall r \in \boldsymbol{R}, m \in \boldsymbol{M}. \tag{10}$$

Constraint (8) ensures that one of the MSs is selected for each multicast connection. Constraints (9) and (10) guarantee that the transparent reach constraint is met. Constraint (9) ensures that a distance used to determine the MS assigned to each multicast connection is no shorter than the longest distance among all paths included in the multicast connection. Constraint (10) guarantees that the transparent reach of the selected MS for the multicast connection is no shorter than the longest distance among all paths.

3) Spectrum Assignment

As the MS is determined from the above constraints, the corresponding number of FSs for the multicast connection can be obtained. Further, spectrum assignment subject to the three constraints is presented as follows.

$$N_r = \sum_{m \in \boldsymbol{M}} \omega_m^r \cdot K_m^r, \forall r \in \boldsymbol{R} \quad (11)$$

$$E_r = S_r + N_r - 1, \forall r \in \boldsymbol{R} \quad (12)$$

$$H_{r2}^{r1} \geq X_{ij}^{r1} + Y_{ij}^{r2} - 1, \forall (i,j) \in \boldsymbol{L}, r1, r2 \in \boldsymbol{R}, r1 \neq r2 \quad (13)$$

$$Z_{r2}^{r1} + Z_{r1}^{r2} = 1, \quad \forall r1, r2 \in \boldsymbol{R}, r1 \neq r2 \quad (14)$$

$$E_{r2} - S_{r1} \leq \Delta \cdot (Z_{r2}^{r1} + 1 - H_{r2}^{r1}) - 1, \forall r1, r2 \in \boldsymbol{R}, r1 \neq r2 \quad (15)$$

$$E_{r1} - S_{r2} \leq \Delta \cdot (Z_{r1}^{r2} + 1 - H_{r2}^{r1}) - 1, \forall r1, r2 \in \boldsymbol{R}, r1 \neq r2. \quad (16)$$

Constraint (11) enforces that a number of FSs in accordance to the selected MS are assigned to the multicast connection. Constraint (12) guarantees that spectrum contiguity constraint is satisfied by assigning a number, calculated by constraint (11), of contiguous FSs, from the starting FS index to the ending FS index, to the multicast connection. Constraint (13) models the situation when a fiber link in the primary tree of one multicast connection is also included to serve another multicast connection. Constraints (14)–(16) ensure spectrum continuity and spectrum non-overlapping between two multicast connections. For the former, the same FSs, from the starting FS index to the ending FS index, in its primary links and backup-only links are allocated to and reserved for the multicast connection, respectively. For the latter, if a primary link of a multicast connection is also used (as either primary or backup link) to serve another multicast connection, the indexes of the FSs of one multicast connection should be smaller than those of the FSs of the other multicast connection. This is because in shared protection, only the FSs for the primary connection are exclusive, FSs on a backup-only link are not exclusive for the multicast connection and can be cross-shared by other connections also only as backup.

4) Lower Bound

$$C \geq E_r, \quad \forall r \in \boldsymbol{R}. \quad (17)$$

Constraint (17) ensures that the maximum number among the numbers of required FSs in all fiber links is greater than or equal to the maximum number among the ending FS indexes of all multicast connections.

5) Redundancy Constraint

$$T_{ij}^r \geq \Delta \cdot (Y_{ij}^r - 1) + N_r, \quad \forall r \in \boldsymbol{R}, (i,j) \in \boldsymbol{L} \quad (18)$$

$$C \geq \sum_{r \in \boldsymbol{R}} T_{ij}^r, \quad \forall (i,j) \in \boldsymbol{L}. \quad (19)$$

Constraints (18) and (19) are used as redundancy to reduce the search region for faster solutions. Constraint (18) ensures that the number of FSs required in each fiber link is greater than or equal to that of the FSs assigned to the primary connection of each multicast connection. Constraint (19) ensures that each fiber link, included as a primary link of the multicast connections, should have at least the sum of the number of FSs assigned to those multicast connections.

V. Heuristic Algorithm

The mathematical formulation of the MC-RMSA problem with shared protection presented in Section IV can be used by a MILP solver to find optimal solutions to the problem in principle. However, MILP is computationally prohibitive for realistically sized networks. In this section, we develop an efficient heuristic algorithm to obtain near optimal solutions.

We first present a routing scheme for the multicast session of a multicast demand. As in our MILP formulation, we also consider distance-adaptive spectrum resource allocation in the heuristic approach. To find a routing subgraph, we assume that the transparent reach is given, which implies that the distance of a path from the source to each destination should not exceed the transparent reach. The main idea of our routing scheme for the multicast session of a multicast demand is to find a routing subgraph that includes a minimum number of links.

Then, we present a heuristic algorithm to provision a multicast demand by calling the routing scheme. The main idea of our heuristic algorithm is to allocate the fewest FSs to a multicast demand, which implies the highest-order possible MS. The reason is that the higher-order MS is assigned to a connection, the fewer FSs are required in each traversed link. Also, having a higher-order MS corresponds to having a shorter transparent reach, which generally in return limits the number of links in the routing subgraph. In this way, the number of links in the routing subgraph is minimized, and the number of FSs required in each of these links is minimized, thus each multicast demand is accommodated with minimum resources.

Finally, to provision multiple demands, we present a greedy algorithm. It firstly orders the given multicast demands in a sequence for accommodation. Then, for each demand, it repeatedly calls the aforementioned heuristic algorithm and adds one FS in each link when currently available FSs are not sufficient until it is accommodated.

A. Multicast Routing Scheme

Our aim is to find a protected tree that will provide a disjoint path pair for each SD pair of a multicast session subject to the transparent reach constraint. This is achieved in part by an algorithm that we call *distance-constrained minimum-cost anycast path* that finds a minimum-cost path from a source to one of the destinations within a given distance. In this way, we can implement MPH [44] as shown in the following to obtain a minimum-cost tree by repeatedly resetting to zero the cost of the links traversed by paths that have already been found and finding a minimum-cost path among the remaining destinations until a path is obtained for each SD pair. Moreover, such an algorithm can be applied to route anycast (one-to-one-of-many) traffic in EONs with a bound on the path distance.

The distance-constrained minimum-cost anycast path algorithm is based on the breadth-first search algorithm, and its pseudocode is provided by Algorithm 1. Since in our case, the link cost is either zero or one, the cost of a path P_i from a given source, s_r, to node i is the sum of the costs of the links traversed by P_i. Thus, the path cost is a non-negative integer and is smaller than or equal to $|\boldsymbol{V}| - 1$, where $|\boldsymbol{V}|$ is the number of the network nodes. For a given path cost, it is possible that no path of that given cost can be found from s_r to node i. It is also possible that there is at least one shortest-distance path of that given cost, where we randomly select one of these shortest paths for possible consideration in the following. Thus, for a given cost value, out of $|\boldsymbol{V}|$ possible cost values, each node has at most one selected shortest-distance path from s_r. Consequently, in total, each node has up to $|\boldsymbol{V}|$ selected shortest-distance paths. Each of these paths is considered at most once (see Line 4 in Algorithm 1), and therefore we need to consider at most $|\boldsymbol{V}|^2$ paths until a minimum-cost path from s_r to one of the destinations within a given distance is found. In this case, the condition/expression of the *while* loop in Line 3 is checked at most $|\boldsymbol{V}|^2$ times. Within the *while* loop, once a shortest-distance path from s_r to node n of a certain cost is considered, we scan the nodes adjacent to n for the *for* loop in Line 8, so the number of the adjacent nodes is at most $|\boldsymbol{V}| - 1$, which implies at most $|\boldsymbol{V}| - 1$ potential paths. Then, within the *for* loop, for each node v of the adjacent nodes, we compare its distance of the newly found path, P'_v, to the distance of the path, P_v (if there is one recorded at node v), with the same cost, and delete P_v from a list of (up to $|\boldsymbol{V}|$) paths with the same cost if the list contains P_v and insert the path with the shorter distance into the list in an increasing order of the distance. To sum up, the *while* loop exits within $|\boldsymbol{V}|^2$ times. Within the *while* loop, the *for* loop exits within $|\boldsymbol{V}|$ times. Within the *for* loop, the insertion of a path has a complexity of $|\boldsymbol{V}|$. Thus, Algorithm 1 has a complexity of $O(|\boldsymbol{V}|^4)$.

For MC-RMSA problem with shared protection, based on Algorithm 1, we introduce a routing scheme for survivable routing for the multicast session of a multicast demand. The routing scheme is called APPF that finds All Primary Paths First for all the SD pairs and then the corresponding link-disjoint ones as their backups. The spectrum used in a primary link of a connection cannot be shared with other connections. However, in shared protection, for better spectrum efficiency, the spectrum used in a backup-only link can be shared among multiple multicast connections also only as backup. Thus the graphs for finding the primary tree and the backup paths that are link-disjoint from their corresponding primary paths could be different, and we denoted them by $G_p = (\boldsymbol{V}_p, \boldsymbol{L}_p)$ and $G_b = (\boldsymbol{V}_b, \boldsymbol{L}_b)$, respectively.

Algorithm 1: Distance-Constrained Minimum-Cost Anycast Path.

Input: A graph $G = (\boldsymbol{V}, \boldsymbol{L})$, a multicast session $r = \langle s_r; \boldsymbol{F}_r; t_r \rangle$, and a distance τ;
Output: null or a minimum-cost path from the source to one of the destinations, where its distance does not exceed τ.
1: Create $|\boldsymbol{V}|$ sets of subgraphs, i.e., $\mathbb{G}_c$, $c = 0, 1, \ldots, |\boldsymbol{V}| - 1$, where the subgraphs in $\mathbb{G}_c$ have a cost of c;
2: Create a subgraph g with a node s_r, reset its cost $g.c$ to 0, and distance $g.t$ to 0, set its end node $g.d$ to s_r, and its set of traversed links $g.\Gamma$ to $\emptyset$, and add it in $\mathbb{G}_0$;
3: **while** $\boldsymbol{F}_r \neq \emptyset$ and $\mathbb{G} \neq \emptyset$, where $\mathbb{G} \leftarrow \bigcup_{c=0}^{|\boldsymbol{V}|-1} \mathbb{G}_c$, **do**
4: Remove the first element g_1 which has firstly a minimum cost and secondly a minimum distance from $\mathbb{G}$;
5: **if** the end node of g_1, i.e., $g_1.d \in \boldsymbol{F}_r$, **then**
6: **return** g_1;
7: **end if**
8: **for all** v in the neighboring node set of node $g_1.d$, **do**
9: **if** v is not in g_1, and link $(g_1.d, v)$, denoted by $\bar{l}$, $\bar{l} \in \boldsymbol{L}$, **then**
10: Create a subgraph g' and set $g'.\Gamma \leftarrow g_1.\Gamma \cup \{\bar{l}\}$;
11: Set $g'.c \leftarrow g_1.c + \bar{l}.c$, $g'.t \leftarrow g_1.t + \bar{l}.t$, where $l.c$ and $l.t$ are the cost and distance of link l, respectively;
12: Set $g'.d \leftarrow v$, $c' \leftarrow g'.c$;
13: **if** the distance of g', $g'.t \leq \tau$, **then**
14: **if** there is already a subgraph g_y terminated at v, $g_y.c = c'$, and $g'.t < g_y.t$, **then**
15: Replace g_y with g';
16: Delete g_y from $\mathbb{G}_{c'}$, and insert g' into $\mathbb{G}_{c'}$ in an increasing order of the distance;
17: **else if** there is no subgraph of cost c', **then**
18: Record g' for v of cost c' and insert g' into $\mathbb{G}_{c'}$ in an increasing order of the distance;
19: **end if**
20: **end if**
21: **end if**
22: **end for**
23: **end while**
24: **return null**;

In APPF, firstly, we initialize the cost of the links in $\boldsymbol{L}_p$ to one. Secondly, we implement MPH by repeatedly calling Algorithm 1 for G_p to find a primary path P_d from the source to destination d, and resetting to zero the cost of the links traversed by P_d, until a path is found for each SD pair. Then, to guarantee that a primary tree is obtained by the found primary paths, we use Dijkstra's algorithm to find a shortest-path tree in a subgraph consisting of all links in the primary paths. In this case, the longest path among the paths to all destinations along the primary tree should have a distance of no longer than the longest path among the primary paths for all SD pairs. Thirdly, we arrange the destinations in an increasing order of the number of links in their primary paths of the obtained primary tree. After that, we reset the cost of each link in $\boldsymbol{L}_b \setminus \boldsymbol{L}_p$ to zero, and the cost of each link in the obtained primary tree to zero, and the cost of the remaining links in $\boldsymbol{L}_b$ to one. Thirdly, to find a backup path P'_d that is link-disjoint from P_d for each destination d as ordered, we call Algorithm 1 for $G'_b = (\boldsymbol{V}_b, \boldsymbol{L}'_b)$, where $\boldsymbol{L}'_b$ is obtained by removing the links traversed by P_d from $\boldsymbol{L}_b$ and also the links in the reversed direction of P_d from $\boldsymbol{L}_b$ if they are in $\boldsymbol{L}_b$, and reset the cost of the links traversed by P'_d to zero. Finally, we obtain a primary tree with a minimum number of links and the backup paths that are link-disjoint from their corresponding primary paths for all SD pairs.

APPF one by one calls Dijkstra's algorithm once and Algorithm 1 for a total of $2|\boldsymbol{F}_r|$ times, where $|\boldsymbol{F}_r|$ is the number of destinations of the multicast session. The former can be achieved at a complexity of $O(|\boldsymbol{V}|^2)$, and the latter has a complexity of $O(|\boldsymbol{V}|^4)$. Thus, APPF has a complexity of $O(|\boldsymbol{V}|^4|\boldsymbol{F}_r|)$.

B. Heuristic Algorithm for Provisioning a Single Demand

Before we present the heuristic algorithm for provisioning a multicast demand, we introduce the concepts of spectrum window (SW) and spectrum-window plane (SWP) proposed in [10]. An SW in a fiber link is a window of spectrum containing a certain number of contiguous FSs. The availability of an SW is subject to the availability of the FSs it contains. If any FS within an SW cannot be used by other connections because of the limitation of the spectrum non-overlapping constraint, the SW is unavailable; otherwise, it is available. Then, an SWP is a plane of virtual graph corresponding to an SW. The virtual graph contains all the nodes of the original one and the links, in each of which, the SW is available.

Fig. 2 illustrates the concepts of SW and SWP. The original graph and the usage of FSs in each fiber link are shown in Figs. 2(a) and (b), respectively. Assume that a new demand, namely, D3, requests three FSs. Then, to accommodate it, an SW should contain three FSs. In each fiber link, there are several possible such SWs, e.g., the first SW, second SW, and third SW occupying FSs 1 to 3, 2 to 4, and 3 to 5, respectively. The first SW corresponds to the first SWP. If the first SW in a link is available, the link should be in the first SWP, and vice versa. A primary tree provisioned for a multicast connection cannot reuse the spectrum resources in fiber links that are either allocated along the primary tree or reserved as backup of other connections, while the backup paths, which are link-disjoint to their corresponding primary paths, can share backup-only spectrum resources with other multicast connections. In Figs. 2(c) and (d), we provide a graph of the first SWP used for finding a primary tree and a graph of the first SWP for finding the backup paths, respectively. The latter graph contains two more backup-only links, namely, links (X, Z) and (Z, Y), than the former graph.

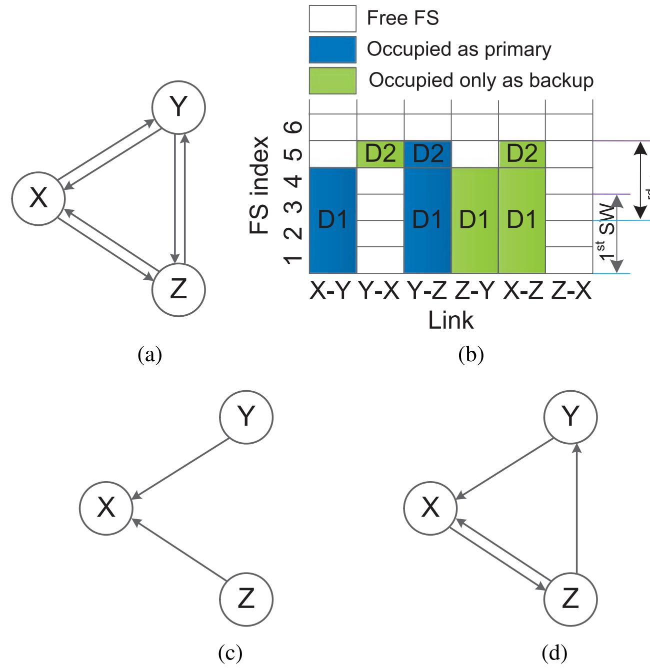

Fig. 2. Illustration for the concepts of SW and SWP: (a) an example network graph; (b) FS usage; (c) a graph G_p on the first SWP for finding a primary tree for a demand requesting for 3 FSs; (d) a graph G_b of the first SWP for finding the backup paths for a demand requesting for 3 FSs.

Such an SWP scheme provides a simple, but efficient, way to satisfy the three constraints in spectrum assignment. Spectrum continuity is guaranteed since the graph of an SWP contains links with the same available FSs. The requirement of spectrum contiguity is met since each SW of the links of the SWP corresponds to a number of contiguous FSs. We also ensure spectrum non-overlapping by removing links from the graph of the SWP. The FSs in these links are utilized by other connections and cannot be reused.

To reduce the number of SWPs to be considered for accommodating a demand, we introduce the concept of an *SWP starting-FS list*, which is a list of FS indexes that includes the lowest numbered FS in every SWP considered for allocating any future demand. Each time after a multicast demand is served, the SWP starting at FS $n + 1$ following the end FS n of the latest served multicast connection is usually different from other SWPs. We store the former FS indexes and also FS 1 in an increasing order in the SWP starting-FS list. Any SWP starting from an FS between the FSs of i-th and $(i + 1)$-th elements of the SWP starting-FS list will have a subset of links that are available in the SWP starting at the FS of i-th element. For the example of Fig. 2(b), the SWP starting FS list includes only FSs 1, 5, and 6. If we consider, for example possible demands requesting a number of contiguous FSs, a possible SWP starting at FS 2, 3, or 4 does not need to be considered, either because it is equivalent to the one starting at FS 1, or because it contains a subset of links that are available in the SWP starting at FS 1.

Assuming that a set of feasible MSs is given, for each multicast demand, we try to assign the MSs from the highest to the lowest modulation order. For a given MS, we calculate the number of required FSs. Then, for given FSs in each fiber link, we scan the SWPs to obtain two graphs, denoted by G_p and G_b, which are used to find the primary tree and the backup paths, respectively. After that, we call a routing scheme to find a pair of primary and backup paths for each SD pair of the multicast connection. The details are presented in Algorithm 2. Algorithm 2 serves a single demand, thus can be easily applied for the case of dynamic traffic where demands arrive sequentially and randomly, then hold network resources for the required service duration, and finally depart. To reduce running time, we use Line 7–9 in Algorithm 2 to pre-determine if the SWP has the possibility for finding such a routing subgraph. Note that to meet the backup requirement, there should be at least two ingress links at each destination and two egress links from the source in G_b, and similarly, to meet the primary requirement, at least one ingress link at each destination and one egress links from the source in G_p.

Since APPF has a complexity, i.e., $O(|\boldsymbol{V}|^4|\boldsymbol{F}_r|)$, Algorithm 2 calls it at most $|\boldsymbol{M}|\Omega$ times, where $|\boldsymbol{M}|$ and Ω are the numbers of considered MSs and FSs in each fiber link. Thus, Algorithm 2 has a complexity of $O(|\boldsymbol{V}|^4|\boldsymbol{F}_r||\boldsymbol{M}|\Omega)$.

Algorithm 2: Provisioning a demand with APPF.

Input: A network graph $G = (\boldsymbol{V}, \boldsymbol{L})$, a multicast demand $r = \langle s_r; \boldsymbol{F}_r; t_r \rangle$, a set $\boldsymbol{M}_r$ of feasible MSs and their corresponding transparent reaches for r, the maximum number Ω among the FS indexes in a link, and an SWP starting-FS list;
Output: MC-RMSA for accommodating r.
1: **while** r has not been accommodated, **do**
2:   **for all** MS m in $\boldsymbol{M}_r$ from the highest to the lowest modulation order, **do**
3:     Obtain the number, i.e., ω_m^r, of required FSs for r and the transparent reach τ_m assuming that MS m is utilized;
4:     **for all** FS index α, $\alpha + \omega_m^r - 1 \leq \Omega$, from lowest to highest in the SWP starting-FS list, **do**
5:       Obtain an SWP whose SW starts from this FS index α and ends at FS index ε, $\varepsilon = \alpha + \omega_m^r - 1$;
6:       Obtain two graphs, namely, $G_p = (\boldsymbol{V}_p, \boldsymbol{L}_p)$ and $G_b = (\boldsymbol{V}_b, \boldsymbol{L}_b)$;
7:       **if** G_b does not have two ingress links at each destination and two egress links from the source, or G_p does not contain one ingress link at each destination and one egress links from the source, **then**
8:         **continue**;
9:       **end if**
10:       Call APPF with the inputs of the two graphs, a multicast session $\langle s_r; \boldsymbol{F}_r; t_r \rangle$, and a distance τ_m, to find a routing subgraph for r;
11:       **if** such a routing subgraph is found, **then**
12:         Accommodate r by allocating and reserving the FSs of the present SW in its primary links and backup-only links, respectively;
13:         Insert FS index $\varepsilon + 1$ into the SWP starting-FS list in an increasing order;
14:       **end if**
15:     **end for**
16:   **end for**
17: **end while**

C. Provisioning of Multiple Demands

We assume that we can find a routing subgraph for every multicast session by running Dijkstra's algorithm to first find a shortest path tree in the original graph and then running Dijkstra's algorithm again for each SD pair to find a shortest path in a modified graph by removing the links traversed by the path in the tree of both directions. The distances of the paths to all destinations are within the transparent reach of the lowest-order MS considered, e.g., 4000 km for BPSK.

Based on the above assumption and the heuristic algorithm for provisioning a multicast demand, we present a greedy algorithm that increases the spectrum resources required in each fiber link in a greedy way when currently available resources in the network are not sufficient to accommodate it. The greedy algorithm includes three steps, namely, *initialization*, *order-operation of the multicast demands*, and *connection setup one by one*. We also introduce a *multi-iteration process* [10] to improve performance.

1) Initialization

To serve a multicast demand, given a set of MSs, we find a set of feasible MSs that it can utilize and a routing subgraph that includes the fewest links for the feasible highest-order MS. Firstly, we obtain a routing subgraph $\Re_1$ by running Dijkstra's algorithm to find a shortest path tree for the multicast, and for each SD pair a shortest path that is link-disjoint from the path in the tree. Then, we can obtain the longest distance among all paths, and therefore obtain the highest-order MS m_1 that $\Re_1$ can utilize. We also try the given MSs from the highest to the lowest order until a routing subgraph $\Re_2$ can be found by APPF. Similarly, we can obtain the highest-order MS m_2 that $\Re_2$ can utilize. After that, we select the higher-order MS, denoted by m, between m_1 and m_2, and record the corresponding routing subgraph as a *candidate routing subgraph*. If m_1 and m_2 are the same, $m = m_1$ and we record as a candidate routing subgraph the routing subgraph that has fewer primary links (first consideration) and fewer total links (secondary consideration). Accordingly, we obtain a set of feasible MSs that are not of higher-order than m.

2) Order Operation of the Demands

The order of serving the demands also affects the result. In this case, we consider two ordering methods. Similar to the traffic-volume-decreasing order in [10], one method is to arrange the requested multicast connections in a decreasing order of the number of required FSs assuming that the highest-order feasible

MS is assigned to each demand. The other is to randomly shuffle the demands to obtain a random sequence of demands. Then sequentially serve the requested multicast connections one by one as follows.

3) Connection Setup One by One

We first reset the number of FSs in each fiber link of the network to zero. If the demand cannot be accommodated for the current network resources, we add one FS in each fiber link, and run Algorithm 2 on the newly available SWP. This procedure is repeated until the demand is accommodated, or the number of added FSs reaches the number n of FSs required by this multicast demand assuming that the highest-order feasible MS is used. If the procedure stops because of the latter, we do not use Algorithm 2, instead we accommodate the demand by allocating and reserving the n newly added FSs in the primary and backup-only links of the candidate routing subgraph obtained in the initialization step, respectively.

After all the demands are served, we can obtain the number of required FSs in each link and thus the required spectrum in units of GHz.

4) Multi-Iteration Process

Since the performance of the proposed greedy algorithm is dependent on the order that the demands are served, we adopt a multi-iteration process to further improve the performance. A result can be obtained for each sequence of demands served via the greedy algorithm. Thus, in multi-iteration process, we randomly shuffle a set of the demands multiple times, then obtain multiple demand sequences and thus multiple results, and select the best one as the final result. In this way, the quality of the result for the multi-iteration process is dependent on the number of distinct sequences of the demands. The larger the number of demand sequences, a better result can be achieved, however, at the cost of a longer computational time. Note that, to reduce computation time, we stop the single run for any demand sequence for which the number of required FSs exceeds the current minimal result based on the previous runs, and continue a new run for another demand sequence.

VI. Numerical Results

In this section, we present numerical results for the MC-RMSA problem with shared protection. We compare the performance of the proposed heuristic algorithm with the optimum obtained by solving the MILP formulation. We investigate the impact of the number of considered demand sequences on the performance of the heuristic algorithms. Moreover, we simulate a more straightforward approach where arriving demands are served one by one.

A. Optimization for Static Multicast Traffic

We consider two ordering methods as mentioned in Section V-C. One is to arrange the demands in a decreasing order of the number of FSs required by each demand assuming that the feasible highest-order MS is utilized. The other is to randomly shuffle the demands to obtain a randomly ordered demand sequence. For the case of random demand sequence, to further improve the solution quality, we consider a multi-iteration process. For the multi-iteration process, it is important to know the right number of demand sequences to be used. If this number is too large it adversely affects the running time. If it is too small the accuracy is compromised. It is therefore important to investigate the impact of this number on the performance of the algorithm.

1) Test Conditions

We consider the following three test networks: 1) a six-node nine-link (n6s9) network, 2) the 11-node 26-link COST239 network, and 3) the 24-node 43-link USNET network. In particular, the n6s9 network we consider is obtained by adding a link with a distance of 700 km between nodes 1 and 4 in the six-node network in [14]. The COST239 network and the USNET network can be found in [49] and [50], respectively. The bandwidth of an FS in each fiber is 12.5 GHz. We consider three MSs, namely, BPSK, QPSK, and 8QAM. The MSs and the corresponding transparent reaches are set as shown in Table I. We consider ten sets of multicast demands. Because of the known computational limitations of the MILP formulation, we only consider that each set contains ten multicast demands for the n6s9 network, and compare the performance of the algorithm against the optimal MILP solution. We use a commercial optimization software, i.e., AMPL/Gurobi 6.5.1 [51], to solve the MILP problem. For the other three larger networks, we consider that each set contains 50 multicast demands, and compare the performance of the proposed heuristic algorithm with different ordering methods. The multicast sessions of the considered demands are obtained by randomly shuffling the set of network nodes. The bit rate values of the demands follow a uniform distribution of range (100, 200) Gb/s. We use up to 10 000 random sequences for each set of demands to investigate the number of random sequences on the performance of the algorithm. We also look into the relationship between the required spectrum and the number of destinations.

Henceforth, the following notations, names and abbreviations will be used. We use "MILP" to stand for the MILP approach. For the heuristic algorithm, we use a prefix, i.e., "APPF_G," to denote the algorithm that uses Algorithm 2 with APPF as the routing scheme in the Greedy algorithm. We also use several suffixes to denote the type of ordering of the demands used in the algorithm. For instance, suffix "_DO" denotes that the set of demands is arranged in a Decreasing Order mentioned above. The remaining suffixes in this paper are used to denote a multi-iteration process with a certain number of considered random sequences, e.g., "_1000" for 1000 randomly shuffled sequences. In this way, we can obtain the short name of a greedy algorithm by combining the prefix and a suffix. For example, "APPF_G_100" is the short name of a greedy algorithm that employs Algorithm 2 with APPF considering 100 random sequences for each set of demands.

2) Numerical Results and Performance Comparison

For the n6s9 network, we compare the performance of the proposed heuristic algorithm to the optimal MILP algorithm. As shown in Fig. 3, with the increase of the number of destinations, the amount of required spectrum increases steadily. We observe that the heuristic algorithm, APPF_G_DO, which is based on ordering the demands in a decreasing order of their FS requirements, requires on average 11.8% more spectrum than the

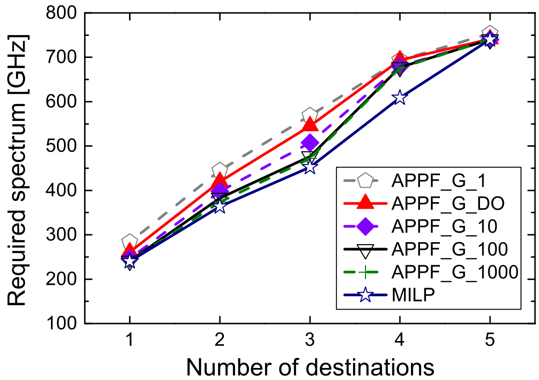

Fig. 3. Performance comparison for the n6s9 network (10 demands).

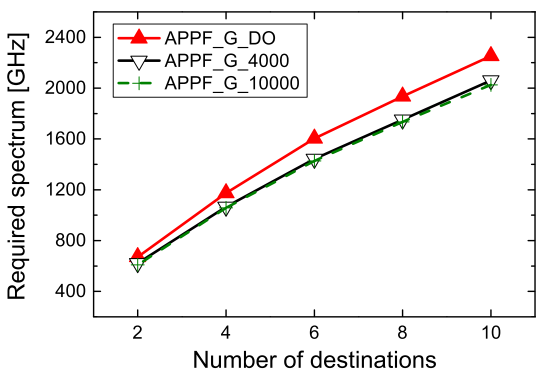

Fig. 4. Performance comparison for the COST239 network (50 demands).

optimum. For the multi-iteration methods, the one with one random sequence performs the worst among all approaches. The one with 100 random sequences, i.e., APPF_G_100, has a significant improvement over APPF_G_1, and outperforms APPF_G_DO. Also, APPF_G_100 achieves close performance to MILP, and consumes on average 4.4% more spectrum than the optimum. This demonstrates the benefit of the multi-iteration process. Further (but not so significant) improvement is achieved by increasing the number of the random sequences to 1000. Thus, 100 random sequences are considered sufficient to achieve near optimum for the n6s9 network with 10 demands. For the broadcast case, the multi-iteration process does not help much and the heuristic approaches achieve optimum and near optimum since the average nodal degree, i.e., 3, is low, and thus the probability of accommodating two or more multicast connections that cross-share backup spectrum resources is low.

For the COST239 network, the performance comparison is shown in Fig. 4. Similarly, the required spectrum increases steadily as the number of destinations increases. Since we have more demands than in the previous case, for multi-iteration processes, we increase the number of considered random sequences to look into the impact of the number of demand sequences on the performance. The heuristic approach with 4000 random sequences, i.e., APPF_G_4000, saves on average around 9% spectrum compared to the other one with a single decreasingly ordered sequence, i.e., APPF_G_DO. We also observe that by comparing APPF_G_10000 to APPF_G_4000, only marginal

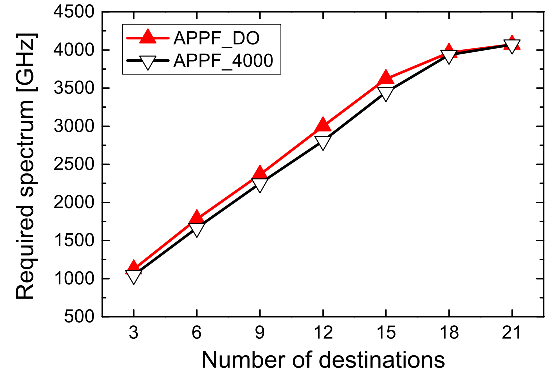

Fig. 5. Performance comparison for the USNET network (50 demands).

improvement is obtained for two-and-one-half times the number of random sequences considered. Thus, 4000 random sequences are considered sufficient for 50 demands. For the remaining USNET network, we also consider 4000 random sequences for 50 demands. Moreover, for the case of broadcast, the COST239 network achieves more benefit of such multi-iteration process than the n6s9 network. The reason is that COST239 has a relatively high nodal degree, i.e., 4.7, than n6s9, and thus there is a higher possibility of finding a solution with spectrum resource sharing.

We also consider 50 demands for the USNET network as shown in Fig. 5. With the increase in the number of destinations, the amount of required spectrum climbs. Also, for the ordering methods of the demands, an approach considering 4000 random sequences, i.e., APPF_G_4000, saves on average 4.3% spectrum compared to APPF_G_DO. Similar to the case in the n6s9 network, such a multi-iteration process does not improve much when there are many destinations in a multicast session, e.g., 18, as the average nodal degree of USNET is also low, i.e., 3.6.

B. *Markov-Chain Simulation for Dynamic Multicast Traffic*

So far we have considered a static model aiming to minimize the maximum spectrum among the required spectrum in all links for accommodating a given set of unordered demands. Here, we model the network as a dynamic system where multicast demands are admitted and complete their service stochastically over time. In particular, we consider a finite set of multicast demands, each of which can be either *active* or *inactive* at any point in time. We use the term, *arrival*, to designate a multicast demand attempting to obtain service in the network. An arrival can be either admitted or blocked. If an arrival is admitted, the state of the multicast demand associated with that arrival will change its state from inactive to active, and it will stay active for a period of time (holding time) which is exponentially distributed with mean $1/\mu$. Then, it completes its service and becomes inactive again. A multicast demand will stay inactive for an exponential amount of time with mean $1/\lambda$ until it will attempt to enter the system to generate an arrival. Our network has a capacity limitation defined by a finite number of FSs. If a multicast demand arrives and there are no sufficient FSs to accommodate the demand, this new arrival will be blocked.

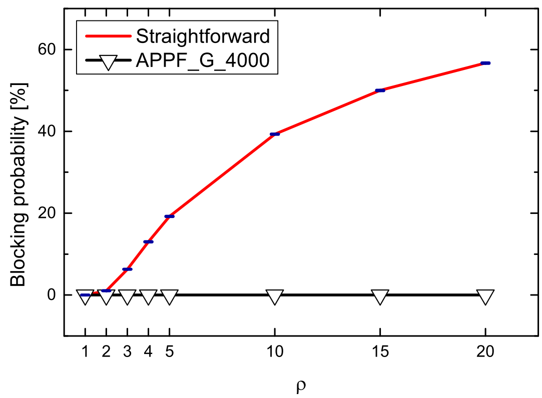

Fig. 6. Blocking probability comparison between the straightforward and our solutions versus ρ for the USNET network (50 demands).

A common performance measure for such a dynamic system is the blocking probability, defined as the ratio of the number of blocked arrivals of multicast demands to the total number of arrivals. Let $\rho = \lambda/\mu$ be a measure of traffic load in our dynamic system.

1) Simulation Conditions

We use the USNET network as a test network, and consider a set of 50 randomly generated demands. The numbers of destinations of the demands follow a uniform distribution from one (unicast) to 23 (broadcast). The bit rates also follow a uniform distribution with a range (100, 200) Gb/s. For the above settings, a static optimization problem considering that the demands are given unordered, is to minimize the maximum number among the numbers of required FSs in all links under the condition that all multicast demands are accommodated. For such a static problem, we employ APPF_G_4000 to obtain the nearly optimal number $\mathcal{F}$ of FSs required in each link. Then, in the dynamic case, simulated by a Markov chain simulation, we assume that there are $\mathcal{F}$ FSs in each link for the considered network. In such a network, the demands can always be accommodated by the solution of APPF_G_4000, and multiple arrivals of the same demand is accommodated exactly the same. Thus, we have no blocking for the approach based on APPF_G_4000 with $\mathcal{F}$ FSs in each link. However, for a more straightforward method in dynamic systems, where demands are served one by one, the blocking probability can be significant. Admitting connections one by one does not have a wider horizon view and will provide different solutions for different returns of the same demand since the accommodation of the demand is attempted based on the available network resources at the moment of the arrival. Under this straightforward approach, we use Algorithm 2 with the APPF routing scheme to attempt to admit each arrival of the demands. We consider a range of scenarios from light to heavy traffic load. For each traffic load scenarios, we conduct 11 simulation experiments, each considering one million arrivals for the given demands, and take the average over the 11 results with a confidence interval of 95% as the final result.

2) Simulation Results

The blocking performance is shown in Fig. 6. As we can see, the straightforward approach for dynamic systems, denoted as "Straightforward" in the figure, shows to have losses. The blocking probability increases with the increase of ρ. Specifically, when traffic load is light $\rho = 1$, the blocking probability is low, 1.154×10^{-5}. However, it rises dramatically to about 39% when $\rho = 10$, and a further increase of blocking probability can be observed for a larger ρ, i.e., heavier traffic load. In contrast, our approach, denoted as "APPF_G_4000" in the figure, does not have service blocking at all for any ρ. This is because that we use the APPF_G_4000 approach to minimize the maximum spectrum among the required spectrum in all links subject to the condition that all the given demands are accommodated. For a network where each fiber link is equipped with the minimized maximum required spectrum, each time a call arrives, the demand can always be served using the solution based on APPF_G_4000, and thus there will be no blocking.

VII. Conclusion

We have considered the MC-RMSA problem for an elastic all-optical network with shared protection. For such a network, we have provided a MILP formulation, and developed a new polynomial time heuristic algorithm for a range of cases. Because the serving order of the demands affects the result, we have considered two cases, one is where the demands are arranged in a sequence in decreasing order of their FS requirements, and the other is to randomly shuffle the demands and obtain a randomly ordered demand sequence. Numerical results showed that the heuristic algorithm achieves close performance to the optimal MILP solution. The heuristic algorithm based on shuffling the demands outperforms the one with specifically ordered demands. We provided complexity analysis to prove that the heuristic algorithm has polynomial time complexity. We considered various cases to demonstrate the scalability of the heuristic algorithm and the improvement in the quality of its result that can be achieved by considering more demand sequences at the cost of longer running time. In this way, we now have a solution where a tradeoff exists between the performance and the running time. We also compared our approach to a more straightforward method in a dynamic setting using a Markov-chain simulation, where multicast demands are admitted and complete their service stochastically over time. Simulation results show that the straightforward method leads to significant losses under heavy traffic load while our heuristic solution has no losses. In the future, we will consider provisioning multicast demands using the light-trail technology for sparse-splitting networks.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments that contributed to the improved quality of this paper.

References

[1] A. Cai, M. Zukerman, R. Lin, and G. Shen, "Survivable multicast routing and spectrum assignment in light-tree-based elastic optical networks," in *Proc. Asia Commun. Photon. Conf. Exhib.*, 2015, pp. 1–3.

[2] I. Tomkos, S. Azodolmolky, J. Solé-Pareta, D. Careglio, and E. Palkopoulou, "A tutorial on the flexible optical networking paradigm: state of the art, trends, and research challenges," *in Proc. IEEE*, vol. 102, no. 9, pp. 1317–1337, Sep. 2014.

[3] O. Gerstel, M. Jinno, A. Lord, and S. J. B. Yoo, "Elastic optical networking: A new dawn for the optical layer?" *IEEE Commun. Mag.*, vol. 50, no. 2, pp. s12–s20, Feb. 2012.
[4] G. Shen and M. Zukerman, "Spectrum-efficient and agile CO-OFDM optical transport networks: Architecture, design, and operation," *IEEE Commun. Mag.*, vol. 50, no. 5, pp. 82–89, May 2012.
[5] K. Walkowiak, "Elastic Optical Networks," in *Modeling and Optimization of Cloud-Ready and Content-Oriented Networks*. Cham, Switzerland: Springer, 2016, pp. 101–193.
[6] M. Jinno, H. Takara, B. Kozicki, Y. Tsukishima, Y. Sone, and S. Matsuoka, "Spectrum-efficient and scalable elastic optical path network: Architecture, benefits, and enabling technologies," *IEEE Commun. Mag.*, vol. 47, pp. 66–73, Nov. 2009.
[7] W. Shieh and I. Djordjevic, *OFDM for Optical Communications*. New York, NY, USA: Academic, 2010.
[8] B. Mukherjee, *Optical Communication Networks*. New York, NY, USA: McGraw-Hill, 1997.
[9] M. Klinkowski and K. Walkowiak, "Routing and spectrum assignment in spectrum sliced elastic optical path network," *IEEE Commun. Lett.*, vol. 15, no. 8, pp. 884–886, Aug. 2011.
[10] A. Cai, G. Shen, L. Peng, and M. Zukerman, "Novel node-arc model and multiiteration heuristics for static routing and spectrum assignment in elastic optical networks," *J. Lightw. Technol.*, vol. 31, no. 21, pp. 3402–3413, Nov. 2013.
[11] M. Jinno *et al.*, "Distance-adaptive spectrum resource allocation in spectrum-sliced elastic optical path network," *IEEE Commun. Mag.*, vol. 48, no. 8, pp. 138–145, Aug. 2010.
[12] K. Christodoulopoulos, I. Tomkos, and E. Varvarigos, "Elastic bandwidth allocation in flexible OFDM-based optical networks," *J. Lightw. Technol.*, vol. 29, no. 9, pp. 1354–1366, May 2011.
[13] P. S. Khodashenas, J. M. Rivas-Moscoso, D. Klonidis, G. Thouénon, C. Betoule, and I. Tomkos, "Impairment-aware resource allocation over flexi-grid network with all-optical add/drop capability," in *Proc. Euro. Conf. Opt. Commun.*, 2015, pp. 1–3.
[14] C. Wang, G. Shen, and S. K. Bose, "Distance adaptive dynamic routing and spectrum allocation in elastic optical networks with shared backup path protection," *J. Lightw. Technol.*, vol. 33, no. 14, pp. 2955–2964, Jul. 2015.
[15] S. Namiki *et al.*, "Ultrahigh-definition video transmission and extremely green optical networks for future," *IEEE J. Sel. Top. Quantum Electron.*, vol. 17, no. 2, pp. 446–457, 2011.
[16] L. H. Sahasrabuddhe and B. Mukherjee, "Light-trees: Optical multicasting for improved performance in wavelength routed networks," *IEEE Commun. Mag.*, vol. 37, no. 2, pp. 67–73, Feb. 1999.
[17] G. N. Rouskas, "Optical layer multicast: Rationale, building blocks, and challenges," *IEEE Netw.*, vol. 17, no. 1, pp. 60–65, Jan. 2003.
[18] Y. Zhou and G. S. Poo, "Optical multicast over wavelength-routed WDM networks: A survey," *Opt. Switch. Netw.*, vol. 2, no. 3, pp. 176–197, Nov. 2005.
[19] R. Lin, M. Zukerman, G. Shen, and W. D. Zhong, "Design of light-tree based optical inter-datacenter networks," *J. Opt. Commun. Netw.*, vol. 5, no. 12, pp. 1443–1455, Dec. 2013.
[20] Q. Wang and L. Chen, "Performance analysis of multicast traffic over spectrum elastic optical networks," in *Proc. Opt. Fiber Commun. Conf. Expo./Nat. Fiber Optic Eng. Conf*, 2012, pp. 1–3.
[21] L. Gong, X. Zhou, X. Liu, W. Zhao, W. Lu, and Z. Zhu, "Efficient resource allocation for all-optical multicasting over spectrum-sliced elastic optical networks," *J. Opt. Commun. Netw.*, vol. 5, no. 8, pp. 836–847, Aug. 2013.
[22] K. Walkowiak, R. Goscien, M. Klinkowski, and M. Wozniak, "Optimization of multicast traffic in elastic optical networks with distance-adaptive transmission," *IEEE Commun. Lett.*, vol. 18, no. 12, pp. 2117–2120, Dec. 2014.
[23] M. Ruiz and L. Velasco, "Performance evaluation of light-tree schemes in flexgrid optical networks," *IEEE Commun. Lett.*, vol. 18, pp. 1731–1734, Oct. 2014.
[24] K. Walkowiak, R. Goscien, M. Tornatore, and M. Wozniak, "Impact of fanout and transmission reach on performance of multicasting in elastic optical networks," in *Proc. Opt. Fiber Commun. Conf. Exhib.*, 2015, pp. 1–3.
[25] Z. Yu, Y. Zhao, J. Zhang, X. Yu, B. Chen, and X. Lin, "Multicast routing and spectrum assignment in elastic optical networks," in *Proc. Asia Commun. Photon. Conf. Exhib.*, 2012, pp. 1–3.
[26] N. Sambo *et al.*, "Demonstration of data and control plane for optical multicast at 100 and 200 Gb/s with and without frequency conversion," *J. Opt. Commun. Netw.*, vol. 5, no. 7, pp. 667–676, Jul. 2013.
[27] D. Zhou and S. Subramaniam, "Survivability in optical networks," *IEEE Netw.*, vol. 14, no. 6, pp. 16–23, Nov. 2000.
[28] W. Kmiecik, R. Goścień, K. Walkowiak, and M. Klinkowski, "Two-layer optimization of survivable overlay multicasting in elastic optical networks," *Opt. Switch. Netw.*, vol. 14, pp. 164–178, Aug. 2014.
[29] D. R. Din and I. R. Lai, "Multicast protection problem on elastic optical networks using segment-base protection," in *Proc. Int. Conf. Informat., Electron. Vis.*, Jun. 2015, pp. 1–6.
[30] N. K. Singhal, L. H. Sahasrabuddhe, and B. Mukherjee, "Provisioning of survivable multicast sessions against single link failures in optical WDM mesh networks," *J. Lightw. Technol.*, vol. 21, no. 11, pp. 2587–2594, Nov. 2003.
[31] W. S. Hu and Q. J. Zeng, "Multicast optical cross connects employing splitter-and-delivery switch," *IEEE Photon. Technol. Lett.*, vol. 10, no. 7, pp. 970–972, Jul. 1998.
[32] M. Ali and J. S. Deogun, "Power-efficient design of multicast wavelength-routed networks," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 10, pp. 1852–1862, Oct. 2000.
[33] A. Gumaste and I. Chlamtac, "Light-trails: A novel conceptual framework for conducting optical communications," in *Proc. IEEE Int. Conf. High Perform. Switching Routing*, 2003, pp. 251–256.
[34] N. A. VanderHorn, S. Balasubramanian, M. Mina, and A. K. Somani, "Light-trail testbed for ip-centric applications," *IEEE Commun. Mag.*, vol. 43, no. 8, pp. S5–10, Aug. 2005.
[35] F. Zhou, M. Molnár, and B. Cousin, "Light-hierarchy: The optimal structure for multicast routing in WDM mesh networks," in *Proc. IEEE Symp. Comput. Commun.*, 2010, pp. 611–616.
[36] Y. Li, J. Wang, A. Gumaste, Y. Xu, and Y. Xu, "Multicast routing in light-trail WDM networks," in *Proc. IEEE Glob. Telecommun. Conf.*, 2008, pp. 1–5.
[37] R. Lin, W. D. Zhong, S. K. Bose, and M. Zukerman, "Multicast traffic grooming in tap-and-continue WDM mesh networks," *J. Opt. Commun. Netw.*, vol. 4, no. 11, pp. 918–935, Nov. 2012.
[38] D. D. Le, M. Molnár, and J. Palaysi, "Multicast routing in WDM networks without splitters," *IEEE Commun. Mag.*, vol. 52, no. 7, pp. 158–167, Jul. 2014.
[39] D. D. Le and M. Merabet, "Light-trail based hierarchy: The optimal multicast route in WDM networks without splitters and converters," in *Proc. Int. Conf. Photon. Optics Laser Technol.*, 2014, Art no. 8.
[40] D. D. Le, F. Zhou, and M. Molnár, "Minimizing blocking probability for the multicast routing and wavelength assignment problem in WDM networks: Exact solutions and heuristic algorithms," *J. Opt. Commun. Netw.*, vol. 7, no. 1, pp. 36–48, Jan. 2015.
[41] M. Ruiz and L. Velasco, "Serving multicast requests on single-layer and multilayer flexgrid networks," *J. Opt. Commun. Netw.*, vol. 7, no. 3, pp. 146–155, Mar. 2015.
[42] N. K. Singhal, C. Ou, and B. Mukherjee, "Cross-sharing vs. self-sharing trees for protecting multicast sessions in mesh networks," *Comput. Netw.*, vol. 50, no. 2, pp. 200–206, Feb. 2006.
[43] C. K. Constantinou, G. Ellinas, and K. Manousakis, "Survivability of multicast requests in mesh optical networks," in *Proc. Int. Conf. Opt. Netw. Des. Model.*, 2014, pp. 7–12.
[44] H. Takahashi and A. Matsuyama, "An approximate solution for the Steiner problem in graphs," *Math. Japonica*, vol. 24, no. 6, pp. 573–577, Feb. 1980.
[45] M. Mohandespour and A. E. Kamal, "Multicast 1+1 protection: The case for simple network coding," in *Proc. IEEE Int. Conf. Comput. Netw. Communi.*, 2015, pp. 853–857.
[46] A. Bocoi, M. Schuster, F. Rambach, M. Kiese, C.-A. Bunge, and B. Spinnler, "Reach-dependent capacity in optical networks enabled by OFDM," in *Proc. Opt. Fiber Commun. Conf. Expo./Nat. Fiber Optic Eng. Conf.*, 2009, pp. 1–3.
[47] A. Klekamp, R. Dischler, and F. Buchali, "Limits of spectral efficiency and transmission reach of optical-OFDM superchannels for adaptive networks," *IEEE Photon. Technol. Lett.*, vol. 23, no. 20, pp. 1526–1528, Oct. 2011.
[48] C. T. Politi *et al.*, "Dynamic operation of flexi-grid OFDM-based networks," in *Proc. Opt. Fiber Commun. Conf. Expo./Nat. Fiber Optic Eng. Conf.*, 2012, pp. 1–3.
[49] S. Peng, R. Nejabati, and D. Simeonidou, "Role of optical network virtualization in cloud computing [invited]," *J. Opt. Commun. Netw.*, vol. 5, no. 10, pp. A162–A170, Oct. 2013.
[50] Y. Lui, G. Shen, and W. Shao, "Design for energy-efficient IP over WDM networks with joint lightpath bypass and router-card sleeping strategies," *J. Opt. Commun. Netw.*, vol. 5, no. 11, pp. 1122–1138, Nov. 2013.
[51] Gurobi Optimization, Inc., "Gurobi optimizer reference manual," 2015. [Online]. Available: http://www.gurobi.com

Authors' photographs and biographies not available at the time of publication.