Malleable Reservation Based Bulk-Data Transfer to Recycle Spectrum Fragments in Elastic Optical Networks

Wei Lu, *Student Member, IEEE*, and Zuqing Zhu, *Senior Member, IEEE*

***Abstract*—In this paper, we propose to facilitate efficient bulk-data transfer in elastic optical networks (EONs) with malleable reservation (MR). The MR scheme performs adjustable routing and spectrum assignment (RSA) for the data-oriented requests that each has certain amount of data to transfer in an EON, where there also exist flow-oriented requests that each requires a fixed bandwidth. We enable RSA reconfigurations for each data-oriented request served by MR, to effectively recycle the 2-D spectrum fragments (i.e., fragments existing in the time and spectrum domains with a correlated manner) generated by the flow-oriented requests. We first formulate a mixed integer linear programming (MILP) model for the MR problem to maximize the percentage of transmitted data using a limited number of RSA reconfigurations. Then, in order to reduce the time complexity, we propose a dynamic programming method (DPM) that can provide the exact solution to the MR problem in polynomial time. Simulation results suggest that compared with the MILP, DPM can provide exact MR solutions for the data-oriented requests with significantly reduced time complexity. The results also verify that without affecting the provisioning of flow-oriented requests, DPM can recycle 2-D spectrum fragments and improve the EON's spectrum utilization effectively.**

***Index Terms*—Bulk data-transfer, elastic optical networks (EONs), malleable reservation (MR), 2-D spectrum fragments.**

I. Introduction

DUE to the enhanced network agility brought by flexible grid, elastic optical networks (EONs) have been considered as a promising physical infrastructure of the next-generation backbone networks and future datacenter (DC) networks [1], [2]. Hence, we expect future EONs to have the capability of supporting a wide range of bandwidth-intensive applications that have diverse traffic characteristics and quality-of-service (QoS) requirements. For instance, in addition to the flow-oriented applications that require strict QoS guarantees (i.e., setup delay and transmission bandwidth), there will also be noticeable data-oriented applications that need to accomplish bulk-data transfers timely. These applications, e.g., grid computing in e-science [3] and data backup and migration in inter-DC networks [4], usually need to transfer huge amounts of data across geographically distributed networks using as short time as possible. Different from the flow-oriented ones, they do not have rigid requirements on setup delay and transmission bandwidth.

Previously, people have studied various EON planning and provisioning schemes for flow-oriented requests that each has a fixed bandwidth, including both the immediate reservation (IR) and advance reservation (AR) scenarios. The IR scenario provisions bandwidth immediately upon receiving a request [5]–[9], while AR reserves bandwidth in the future for data transmission [10], [11]. As both IR and AR scenarios reserve a fixed amount of bandwidth for the whole life-time of a request, they are not suitable for data-oriented applications to accomplish bulk-data transfers. For simplicity, in the rest of the paper, we refer to the flow-oriented fixed-bandwidth requests that are in either the IR or AR scenario as "flow-oriented requests," in contrast with the data-oriented ones. On the other hand, dynamic bandwidth allocation methods for time-varying traffic have been proposed in [12], [13], where the authors leveraged spectrum expansion, contraction, and re-allocation to adapt to traffic dynamics. Even though these methods were still flow-oriented, they started to consider adjustable resource allocation in both the time and spectrum domains.

In this paper, inspired by the aforementioned dynamic bandwidth allocation methods, we propose to facilitate efficient bulk-data transfer in EONs with malleable reservation (MR) [14]. More specifically, we investigate how to optimize spectrum allocations for the data-oriented requests that each has certain amount of data to deliver, in an EON environment where there also exist flow-oriented requests. We enable spectrum retuning and transmission pausing for each data-oriented request to effectively recycle the 2-D spectrum fragments (i.e., spectrum fragments existing in the time and spectrum domains with a correlated manner) generated by the flow-oriented ones. It is known that spectrum fragments, which refer to nonaligned and isolated spectrum blocks in the spectrum domain, exist in EONs where there are dynamic IR-type flow-oriented requests [15], [16]. Meanwhile, when AR-type flow-oriented requests look into future for more transmission opportunities, they also generate spectrum fragments in the time domain, i.e., spectrum blocks that are only available for a very short period of time [10]. These 2-D spectrum fragments lead to low spectrum utilization in EONs, while the rationale behind our investigation is to efficiently recycle them for bulk-data transfers.

Manuscript received September 10, 2014; revised December 22, 2014 and February 13, 2015; accepted February 19, 2015. Date of publication February 19, 2015; date of current version March 16, 2015. This work was supported in part by the NCET program under Project NCET-11-0884, the NSFC Project 61371117, the Fundamental Research Funds for the Central Universities (WK2100060010), Natural Science Research Project for Universities in Anhui (KJ2014ZD38), and the Strategic Priority Research Program of the CAS (XDA06030902).

The authors are with the School of Information Science and Technology, University of Science and Technology of China, Hefei, Anhui 230027, China (e-mail: zqzhu@ieee.org).

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.2015.2406251

We assume that each data-oriented request has a fixed number of routing path candidates, while its routing and spectrum assignment (RSA) is changeable in different time intervals, i.e., the network operator can provide it with various RSA configurations in different service provision periods with spectrum retuning. Note that such spectrum retuning is feasible in bandwidth-variable transponders (BV-Ts), as researchers have already experimentally demonstrated a hop-tuning-based technique that can accomplish full-spectrum retuning within micro-seconds [17]. Meanwhile, bandwidth-variable wavelength-selective switches (BV-WSS') can also be reconfigured dynamically to facilitate RSA reconfigurations. Moreover, people have proposed and experimentally demonstrated the combination of software-defined networking (SDN) and EONs for software-defined EONs (SD-EONs), and shown that SD-EONs can make optical networks adapt to highly dynamic traffic better with agile resource management and timely operation reconfiguration [18]–[21].

Based on the network model mentioned above, we solve the MR problem in which the maximum number of RSA reconfigurations is constrained for each bulk-data transfer with two steps, namely, 1) locating 2-D spectrum fragments and 2) performing MR. With the objective to maximize the transmitted data, we formulate a mixed integer linear programming (MILP) model for MR. In order to solve the MR problem within acceptable running time, we propose a dynamic programming method (DPM), which can provide the exact solution to the MR problem in polynomial time. Extensive simulations are then performed to evaluate the performance of the proposed MR algorithms.

The rest of the paper is organized as follows. Section II provides a survey of the related work. In Section III, we describe the problem of MR-based bulk-data transfer that recycles 2-D spectrum fragments in EONs. Section IV formulates the MILP model for the MR problem and analyzes its complexity. The DPM-based MR algorithm is proposed in Section V. We then present the performance evaluation with simulations in Section VI. Finally, Section VII summarizes the paper.

II. Related Work

Previous investigations have studied how to serve data-oriented requests in the Internet with emphases on routing and scheduling [4], [22]–[25]. In [22], Guerin *et al.* proved that the problem of finding the optimum routing path to minimize the bulk-data transfer time of a data-oriented request is $\mathcal{NP}$-hard and proposed an approximation algorithm to solve a special case of the problem where all the links in the network have homogeneous traffic. Gangulay *et al.* [23] investigated how to find the path that has the highest throughput based on a time-varying graph, under the assumption that path switchings are allowed but only within a limited times during bulk-data transfer. The problem of path computation for bulk-data transfer schemes with and without path switchings was discussed in [24], where the authors proved that the problem is $\mathcal{NP}$-complete for both schemes and proposed a heuristic algorithm for each of them. In [4], Laoutaris *et al.* investigated the scheduling problem for multiple concurrent data-oriented requests, reduced it to the maximum concurrent multi-commodity flow problem, and designed a store-and-forward scheme for it. Wang *et al.* [25] extended the store-and-forward scheme by introducing the max-min fairness in request scheduling. Nevertheless, all of the studies mentioned above did not consider the resource allocation in the physical layer and only modeled the links as bandwidth pipes.

Meanwhile, with optical networks as the background, people have studied both bulk-data transfer [26]–[28] and dynamic bandwidth allocation for time-varying traffic [12], [13]. Naiksatam *et al.* proposed an elastic bandwidth reservation algorithm for LambdaGrids, which reconfigures in-service data-oriented requests to improve the bandwidth utilization [26]. In [27], the problem of provisioning data-oriented requests in wavelength-division multiplexing (WDM) networks was studied and the authors formulated an MILP model to optimize the bandwidth allocation for two objectives, 1) finishing the data-transfer before a preset deadline, and 2) maximizing the network's bandwidth utilization. More recently, Patel *et al.* investigated the problem of routing and scheduling for variable-bandwidth AR requests in WDM networks, and proposed three heuristics to minimize the data transfer time in [28].

Although these studies pioneered in the area of provisioning data-oriented requests in optical networks and proposed a few interesting algorithms, they only focused on data-oriented requests and did not consider the network environments where heterogenous requests (i.e., data-oriented and flow-oriented) coexist. As commented in [4], [25], for the networks with heterogeneous requests, the MR scheme discussed in them, which leverages the delay-tolerance nature of data-oriented requests and utilizes instantaneous leftover bandwidth for bulk-data transfers, fits in better. Moreover, the studies only addressed fixed-grid WDM networks, but did not consider flexible-grid EONs, whose bandwidth allocation in the optical layer is more agile but also more complicated.

On the other hand, dynamic bandwidth allocation for time-varying traffic in EONs was independently studied in [12], [13] and both work considered spectrum expansion, contraction, and re-allocation to adapt to traffic dynamics. However, these approaches were still flow-oriented and the authors designed their algorithms to minimize the request blocking probability. Also, the bandwidth fluctuation of each request was predetermined. Therefore, their network models are different from ours too, and the proposed algorithms cannot be used to solve the MR problem. Note that we leverage the interesting idea of spectrum expansion, contraction, and re-allocation in this paper, to recycle the 2-D spectrum fragments effectively.

III. Problem Description

A. Models of Network and Requests

We model the EON's physical topology as a directed graph $G(V, E)$, where V and E denote the node and fiber link sets, respectively. Each link can accommodate B frequency slots (FS$'$) at most. In addition, we assume that the network operates according to discrete time slots (TS$'$). Note that the practical value for the TS$'$ duration actually depends on two factors, 1) the speed of lightpath reconfiguration, and 2) the dynamics of the traffic demands in EONs. First of all, the TS$'$ duration has to be much longer than the time required for lightpath reconfiguration.

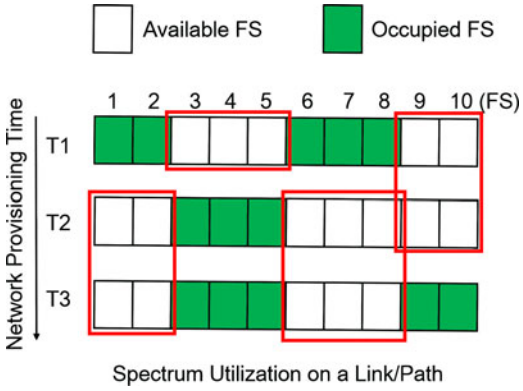

Fig. 1. Examples of 2-D spectrum fragments in an EON.

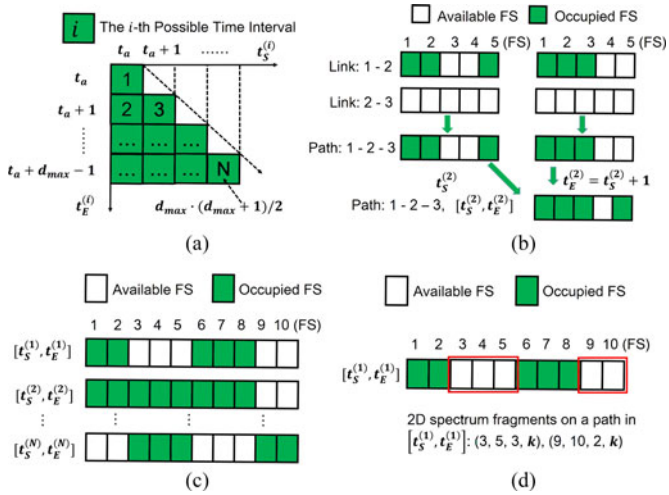

Fig. 2. Examples on the procedure to locate 2-D spectrum fragments.

Otherwise, we may not be able to apply the MR scheme in an EON. On the other hand, the dynamics of the traffic demands in the EON determine how fast we should adjust the network operation to adapt to the traffic load change. Considering the fact that it may take the operator more than a minute to reconfigure a lightpath [29], [30] and certain emerging applications, such as data backup and migration in inter-DC networks, may need bandwidth-on-demand on the scale of tens of minutes [30], we expect TS be on the magnitude of tens of minutes. Meanwhile, considering the fact that in the future, the reconfiguration process can be automated and the demands can occupy much shorter durations in many cases, we anticipate that it would be beneficial if a much shorter TS could be implemented. Note that the algorithm proposed in this paper does not prevent the EON from selecting a much shorter TS, since it can find optimal solutions with relatively low complexity (as we will show in Section V).

Here, T represents the maximum number of TS$'$ that we can look ahead in the future for MR. In order to represent the availability of each FS on a link $e \in E$ during a TS $t \in [1, T]$ ($t \in \mathbb{Z}^+$) [1], we define a matrix $[\mathbf{U}]_{|E|\times T}$, whose element $u_{e,t}$ is a bitmask that contains B bits to record the availabilities of all the FS$'$ on link e during the tth TS. For instance, if the ith FS on link e is available during the tth TS, we set $u_{e,t}[i] = 1$, otherwise $u_{e,t}[i] = 0$. Each data-oriented request is denoted by a tuple $R(s, d, \mathcal{F}, t_a, d_{\max})$, where s and d are the source and destination nodes, $\mathcal{F}$ is the size of the data to be transmitted in terms of the usage of FS$'$ over time (e.g., if $\mathcal{F} = 6$, we can finish the bulk-data transfer using 2 FS$'$ over 3 TS$'$), t_a is the TS at which the request arrives, and $d_{\max}$ is the longest look-ahead period in terms of TS$'$, i.e., the longest period that the network operator can look into in the future to coordinate the request's data transfer.

B. MR to Recycle 2-D Spectrum Fragments

Fig. 1 illustrates the examples of 2-D spectrum fragments in an EON, which are generated by IR- and AR-types of flow-oriented requests. Basically, we mark the availabilities of the FS$'$ on a link/path over network provisioning time, and it can be seen that these fragments can hardly be utilized to set up flow-oriented requests as they are either small in terms of bandwidth, or short in time duration, or both.

In order to realize MR for a data-oriented request $R(s, d, \mathcal{F}, t_a, d_{\max})$, we need to solve two subproblems.

1) *Routing:* We need to calculate a set of K routing paths from s to d, which can be denoted as $\{p_{s,d,k}, k = 1, 2, \ldots, K\}$, where k is the path ID. We assume that RSA reconfigurations with path switching are allowed during each data transfer.
2) *2-D-Fragmentation-Aware Spectrum Allocation:* We need to allocate spectrum resources in a time-varying way to maximize the transmitted data for each data-oriented request. As RSA reconfigurations cause increased operational complexity and cost, we restrict the reconfiguration times below Q for each data transfer.

The objective is to maximize the transmitted data with the fewest RSA reconfigurations. We propose to obtain the solution of the MR problem with two steps, i.e., 1) Locating 2-D spectrum fragments, and 2) Performing MR-based RSA.

1) Locating 2-D Spectrum Fragments: Based on the information provided by $[\mathbf{U}]_{|E|\times T}$, we locate all the 2-D spectrum fragments on $\{p_{s,d,k}, k \in [1, K]\}$. Fig. 2 explains the procedure to do so. First, with t_a and $d_{\max}$, we obtain all the possible time intervals, i.e., $\{[t_S^{(i)}, t_E^{(i)}] \subseteq [t_a, t_a + d_{\max} - 1], i \in [1, N]\}$, where $t_S^{(i)}$ and $t_E^{(i)}$ denote the starting and ending TS$'$ of the ith time interval, and N is the total number of possible time intervals. Here, N can be calculated with $d_{\max}$ using the expression below

$$N = \frac{d_{\max} \cdot (d_{\max} + 1)}{2}. \tag{1}$$

Fig. 2(a) shows how to obtain all the possible time intervals, where the horizontal and vertical axes represent $t_S^{(i)}$ and $t_E^{(i)}$, respectively, i.e., each slot represents a possible time interval. Secondly, we analyze the availabilities of the FS$'$ on the path candidates $\{p_{s,d,k}, k \in [1, K]\}$ during each possible time interval. Figs. 2(b) and (c) show some illustrative examples. Basically,

[1] $\mathbb{Z}^+$ refers to the set of all the positive integers.

in Fig. 2(b), we analyze the spectrum utilization on each link during $[t_S^{(2)}, t_E^{(2)}]$ to obtain the availabilities of the FS′ on *Path* 1−2−3. Fig. 2(c) shows the availabilities of the FS′ on $p_{s,d,k}$ during all the possible time intervals. We use a bitmask $u_{s,d,k}^{(i)}$ to present the availabilities of all the FS′ on $p_{s,d,k}$ during the ith time interval, which is calculated as[2]

$$u_{s,d,k}^{(i)} = \prod_{t=t_S^{(i)}}^{t_E^{(i)}} \prod_{e \in p_{s,d,k}} u_{e,t} \quad \forall i,k. \tag{2}$$

Then, according to $\{u_{s,d,k}^{(i)}\}$, we locate the 2-D spectrum fragments on $\{p_{s,d,k}\}$ during each possible time interval. Note that the 2-D spectrum fragments should be those that are the largest in terms of the product of FS′ and TS′, under the spectrum contiguous and time continuous constraints (i.e., the red rectangles in Fig. 1). We use $\Psi^{(i)}$ to represent the set of 2-D spectrum fragments on $\{p_{s,d,k}\}$ during the ith time interval, where each element in $\Psi^{(i)}$ is a tuple $(f_{S,j}^{(i)}, f_{E,j}^{(i)}, w_j^{(i)}, p_j^{(i)})$, where j is the ID of the element, $f_{S,j}^{(i)}$ and $f_{E,j}^{(i)}$ are the starting and ending FS-indices of the fragment, $w_j^{(i)}$ is the size of the fragment (i.e., the product of FS′ and TS′, or $w_j^{(i)} = (f_{E,j}^{(i)} - f_{S,j}^{(i)} + 1)(t_E^{(i)} - t_S^{(i)} + 1)$), and $p_j^{(i)}$ is the routing path that carries the fragment. Fig. 2(d) provides an example to explain how to obtain $(f_{S,j}^{(1)}, f_{E,j}^{(1)}, w_j^{(1)}, p_j^{(1)})$ to locate the 2-D spectrum fragments, when $p_j^{(1)} = p_{s,d,k}$.

2) Performing MR-Based RSA: Based on $\{\Psi^{(i)} \quad \forall i\}$, we determine the MR-based RSA for each data-oriented request. Basically, during each TS $t \in [t_a, t_a + d_{\max} - 1]$, we should assign a routing path $p^{(t)}$ and a spectrum block $[f_S^{(t)}, f_E^{(t)}]$ on it to transmit data for the request. For any two consecutive TS′ t and $t+1$, if we have $p^{(t+1)} \neq p^{(t)}$ or $f_S^{(t+1)} \neq f_S^{(t)}$ or $f_E^{(t+1)} \neq f_E^{(t)}$, there is a RSA reconfiguration in between. Note that in order to limit the operation complexity, we restrict the RSA reconfiguration times for each data-oriented request below Q[3]. Note that since we limit the look-ahead period for handling each data-oriented request as $d_{\max}$, there is a possibility that only a part of its data gets transferred within $[t_a, t_a + d_{\max} - 1]$. The percentage of the transmitted data, denoted by η, can be calculated as

$$\eta = \min \left(\frac{\sum_{t=t_a}^{t_a + d_{\max} - 1} (f_E^{(t)} - f_S^{(t)} + 1)}{\mathcal{F}}, 1 \right), \tag{3}$$

which is a performance metric to evaluate the MR algorithms.

IV. MILP Model for MR

In this section, we formulate an MILP model to solve the MR problem and analyze its complexity.

[2]Here, $\prod$ executes bitwise dot-product, i.e., $\bigotimes$.

[3]In order to save BV-Ts, we assume that each request only consumes a pair of BV-Ts, i.e., we do not consider spectrum-splitting in this paper.

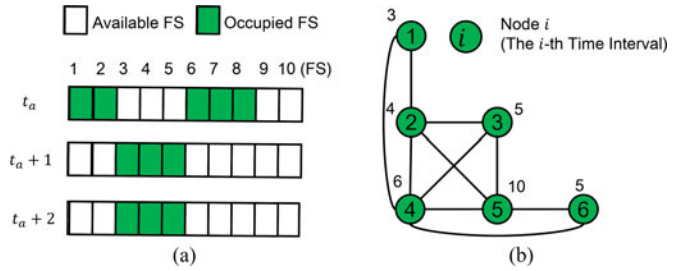

Fig. 3. Example of auxiliary graph $G^a(V^a, E^a)$ for $d_{\max} = 3$ TS′.

A. MILP Formulation

For a data-oriented request $R(s, d, \mathcal{F}, t_a, d_{\max})$, the MR scheme tries to transfer the largest amount of data with the least RSA reconfigurations within $[t_a, t_a + d_{\max} - 1]$. Then, the scheme can be understood as follows:

1) The whole data-transfer time $[t_a, t_a + d_{\max} - 1]$ is divided into several time intervals, which are bound with RSA reconfigurations.
2) Within each time interval, we select the largest feasible 2-D spectrum fragment to transfer data.
3) The accumulated FS′ provided by the selected 2-D spectrum fragments maximize η defined in Eq. (3).

With all the 2-D spectrum fragments on $\{p_{s,d,k}\}$ located, we construct an auxiliary graph $G^a(V^a, E^a)$ with $|V^a| = N$, which is the total number of possible time intervals. Node v_i^a in G^a represents the time interval $[t_S^{(i)}, t_E^{(i)}]$ and its weight $w(v_i^a)$ is the size of the largest 2-D spectrum fragment in it as

$$w(v_i^a) = \max_j (w_j^{(i)}). \tag{4}$$

For any two nodes in $G^a(V^a, E^a)$, if their time intervals overlap, they are directly connected. Fig. 3 shows an example of $G^a(V^a, E^a)$. For simplicity, we only use one feasible routing path $p_{s,d,1}$ in the example, i.e., $K = 1$. Based on the spectrum utilization in Fig. 3(a), we construct $G^a(V^a, E^a)$ in Fig. 3(b) for $d_{\max} = 3$ TS′, where the numbers on the nodes are their IDs and those around them are their weights, i.e., $w(v_i^a)$. With $G^a(V^a, E^a)$, we formulate an MILP model to solve the MR problem for each data-oriented request.

Parameters:

1) V^a: Set of nodes in the auxiliary graph $G^a(V^a, E^a)$.
2) E^a: Set of links in the auxiliary graph $G^a(V^a, E^a)$.
3) $w(v_i^a)$: Weight of node $v_i^a \in V^a$.
4) $\mathcal{F}$: Data size of the data-oriented request.
5) Q: Maximum RSA reconfiguration times.

Variables:

6) x_i: Boolean variable that equals 1 if node v_i^a is selected for MR, otherwise $x_i = 0$.
7) y: Real variable that ranges within $[0, 1]$, indicating the percentage of transmitted data.

Objectives:

$$Maximize \quad \alpha \cdot y - \sum_{\{i:\ v_i^a \in V^a\}} x_i, \tag{5}$$

where α is a big positive integer and we use $\alpha \cdot y$ to make the optimization's major objective as maximizing the percentage of transmitted data. Meanwhile, for the solutions that provide similar percentages of transmitted data, we use the second term ($\sum_{\{i:\ v_i^a \in V^a\}} x_i$) to minimize the RSA reconfiguration times.

Constraints:

$$x_i + x_j \leq 1, \quad \{i, j:\ v_i^a, v_j^a \in V^a, (v_i^a, v_j^a) \in E^a\}. \tag{6}$$

Eq. (6) ensures that the selected time intervals are not overlapping in the time domain

$$\sum_{\{i:\ v_i^a \in V^a\}} x_i \leq Q + 1. \tag{7}$$

Eq. (7) ensures that the RSA reconfiguration times will not exceed the preset limit

$$0 \leq y \leq \frac{\sum_{\{i:\ v_i^a \in V^a\}} x_i \cdot w(v_i^a)}{\mathcal{F}}. \tag{8}$$

Eq. (8) limits the value of y,

$$x_i \in \{0, 1\}, \{i:\ v_i^a \in V^a\}. \tag{9}$$

Eq. (9) defines the ranges of the variables.

B. Complexity Analysis

In the worst case, we can obtain the optimum solution by running an exhaustive search, whose complexity is tightly related to the size of feasible solution space. A feasible solution includes a set of independent node(s), and the set's size Δ ranges within $[1, (Q+1)]$. For a specific Δ, if we suppose that the time interval(s) of the corresponding independent node(s) can be denoted as $\{[t_{S,i}, t_{E,i}], i \in [1, \Delta]\}$, where $t_{S,i}$ and $t_{E,i}$ are the starting and ending TS′ of the ith node and $t_{S,1} < t_{S,2} < ... < t_{S,\Delta}$ (if $\Delta \geq 2$), the size of the feasible solution space can be calculated as

$$\begin{cases} \sum_{t_{S,1}=t_a}^{t_{up}-1} (t_{up} - t_{S,1}), & \Delta = 1, \\ \sum_{t_{S,1}=t_a}^{t_{up}-\Delta} \sum_{t_{E,1}=t_{S,1}}^{t_{up}-\Delta} \sum_{t_{S,2}=t_{E,1}+1}^{t_{up}-\Delta+1} ...(t_{up} - t_{S,\Delta}), & \Delta \geq 2, \end{cases} \tag{10}$$

where t_{up} is defined as $t_{up} = t_a + d_{\max}$. With Eq. (10), we can see that the complexity for performing an exhaustive search in the whole feasible solution space is $O((d_{\max})^{2\Delta})$. Hence, according to the range of Δ, i.e., $[1, (Q+1)]$, we can obtain the lower- and upper-bounds of the complexity as $O((d_{\max})^2)$ and $O((d_{\max})^{2(Q+1)})$, respectively.

V. DPM for MR

Since solving the MILP model with an exhaustive search yields relatively high time complexity, we propose a DPM in this section, which can provide the exact solution to the MR problem in polynomial time.

A. DPM

Before solving MR with DPM, we leverage the weight $w(v_i^a)$ obtained in Section IV-A as the weight of the time interval $[t_S^{(i)}, t_E^{(i)}]$, i.e., $w([t_S^{(i)}, t_E^{(i)}]) = w(v_i^a)$. Basically, $w([t_S^{(i)}, t_E^{(i)}])$ represents the maximum amount of data that can be transmitted in $[t_S^{(i)}, t_E^{(i)}]$ without any RSA reconfiguration, i.e., the size of the largest 2-D spectrum fragment in $[t_S^{(i)}, t_E^{(i)}]$. Then, DPM schedules the MR-based data transmission based on $w([t_S^{(i)}, t_E^{(i)}])$ to achieve the largest amount of transmitted data for each data-oriented request.

Algorithm 1 shows the detailed procedure of the proposed DPM, which selects proper time intervals and 2D spectrum fragments to carry the data-oriented request $R(s, d, \mathcal{F}, t_a, d_{\max})$. Here, we define several variables as

1) $M_{t,r}$: State variable that represents the maximum amount of data that can be transmitted within the time interval $[t_a, t]$ with r RSA reconfigurations. Here, we have $t \in [t_a, t_a + d_{\max} - 1]$ and $r \in [0, Q]$.
2) $\Gamma_{t,r}$: Variable that records the set of used time intervals for achieving the MR state described in $M_{t,r}$.
3) Ω: Variable that records the set of time intervals that are finally selected by DPM for the data-oriented request.

Lines 1–4 are for the initialization. Note that $M_{t_a-1,r}$ and $\Gamma_{t_a-1,r}$ do not have physical meanings since the MR scheme has to be scheduled starting from TS t_a, but we need to initialize them to make sure that the subsequent lines can be executed correctly. The for-loop that covers *Lines* 5–28 illustrates the procedure for state transition. Specifically, *Line* 7 makes sure that $M_{t,r}$ equals $M_{t-1,r}$ when there is no additional RSA reconfiguration, and *Lines* 8–26 try to maximize $M_{t,r}$ at the cost of one additional RSA reconfiguration. When we have obtained all $M_{t,r}$ and $\Gamma_{t,r}$, *Lines* 29–31 select the optimal ones and use them to determine the MR scheme for the data-oriented request. Note that, in *Line* 30, in order to obtain the optimal MR scheme (i.e., represented by M_{t^*,r^*} and Γ_{t^*,r^*}), we first find out all the MR schemes that can provide the largest η, and then select the one that can achieve this with the least RSA reconfigurations as the optimal one.

Once the optimal time interval set Ω has been determined, we perform RSA to implement MR, the detailed procedure of which is listed in *Algorithm*. Note that when the largest $\eta = 1$, it is possible that a small 2-D spectrum fragment would be left in the last time interval in Ω, considering the situation in which the remaining data to be transmitted is smaller than the size of the last 2-D spectrum fragment. However, since not all the RSA schemes for the data-oriented requests would end up with this situation in MR and at most one 2-D small spectrum fragment per request would be left in the worst case, the adverse effect from it is actually very limited, which will be verified with the simulation results in Section VI-B. *Lines* 4-12 show the detailed RSA procedure for the case that the largest $\eta = 1$, where Υ represents the remaining data to be transmitted. When the time interval is not the last one, *Lines* 5-7 find a proper 2-D spectrum fragment in it and fill it with data. In *Lines* 9-12, we handle the situation in which a small 2-D spectrum fragment would be left

Algorithm 1: Dynamic Programming Method (DPM) for MR

Input: t_a, d_{max}, $\mathcal{F}$, $\{w([t_S,t_E])\}$, and Q;
Output: Ω, η;

$M_{t_a-1,0}=0$, $\Gamma_{t_a-1,0}=\emptyset$;
for $r=1$ *to* Q **do**
  $M_{t_a-1,r}=-\infty$, $\Gamma_{t_a-1,r}=\emptyset$;
end
for $r=0$ *to* Q **do**
  for $t=t_a$ *to* $t_a+d_{max}-1$ **do**
    $M_{t,r}=M_{t-1,r}$, $\Gamma_{t,r}=\Gamma_{t-1,r}$;
    if $r=0$ **then**
      for $tt=(t_a-1)$ *to* $(t-1)$ **do**
        if $M_{t,r}<w([tt+1,t])$ **then**
          $M_{t,r}=w([tt+1,t])$;
          $\Gamma_{t,r}=\{[tt+1,t]\}$;
        else
          continue;
        end
      end
    else
      for $tt=(t_a-1)$ *to* $(t-1)$ **do**
        if $M_{t,r}<M_{tt,r-1}+w([tt+1,t])$ **then**
          $M_{t,r}=M_{tt,r-1}+w([tt+1,t])$;
          $\Gamma_{t,r}=\Gamma_{tt,r-1}\cup\{[tt+1,t]\}$;
        else
          continue;
        end
      end
    end
  end
end
calculate η with Eq. (3) for all $M_{t,r}$;
select the optimal M_{t^*,r^*} and Γ_{t^*,r^*} with the largest η as well as the least RSA reconfigurations;
$\Omega=\Gamma_{t^*,r^*}$;
return (Ω,η);

Algorithm 2: RSA for Implementing MR

Input: $R(s,d,\mathcal{F},t_a,d_{max})$, $\{p_{s,d,k}:k=1,...,K\}$, Ω, $\{w([t_S,t_E]):t_a\le t_S\le t_E\le t_a+d_{max}-1\}$, η;
Output: $\{(p^{(t)},[f_S^{(t)},f_E^{(t)}]):t\in[t_a,t_a+d_{max}-1]\}$;

$\Upsilon=\mathcal{F}$;
for *all* $[t_S,t_E]\in\Omega$ *in descending order of* $w([t_S,t_E])$ **do**
  if $\eta=1$ **then**
    if $[t_S,t_E]$ *is not the last one in* Ω **then**
      $\Upsilon=\Upsilon-w([t_S,t_E])$;
      locate 2D fragment whose size is $w([t_S,t_E])$;
      set $\{(p^{(t)},[f_S^{(t)},f_E^{(t)}]):t\in[t_S,t_E]\}$ according to the RSA of the located 2D fragment;
    else
      locate 2D fragment whose size is $w([t_S,t_E])$;
      calculate the number of FS' that need to be assigned with $\lceil\frac{\Upsilon}{t_S-t_E+1}\rceil$;
      get the actual RSA based on the located 2D fragment and required FS number;
      set $\{(p^{(t)},[f_S^{(t)},f_E^{(t)}]):t\in[t_S,t_E]\}$ according to the actual RSA;
    end
  else
    locate 2D fragment whose size is $w([t_S,t_E])$;
    set $\{(p^{(t)},[f_S^{(t)},f_E^{(t)}]):t\in[t_S,t_E]\}$ according to the RSA of the located 2D fragment;
  end
end
return $\{(p^{(t)},[f_S^{(t)},f_E^{(t)}]):t\in[t_a,t_a+d_{max}-1]\}$;

in the last time interval. On the other hand, when the largest $\eta<1$, the 2-D spectrum fragments in all the time intervals are fully filled and no small 2-D spectrum fragment would be left, as in *Lines* 15-16.

B. Complexity Analysis

When performing DPM in *Algorithm*, we need to calculate $d_{\max}\cdot(Q+1)$ state variables, and the time complexity of computing each one is $O(d_{\max})$. According to the principle of MR, we have $Q<d_{\max}$, i.e., the maximum RSA reconfiguration times must be less than the longest look-ahead period in terms of TS′, because a RSA reconfiguration can only happen in between two consecutive TS′. Hence, the time complexity of DPM is $O((d_{\max})^3)$, which is in polynomial time.

VI. Performance Evaluation

In this section, we present the simulation results for performance evaluation. The simulations use MATLAB R2013a and run on a personal computer with 2.93 GHz Intel Core i3 CPU and 6 GB RAM. The MILP model is implemented and solved with the GNU linear programming kit (GLPK) [31].

TABLE I
Simulation Parameters for the Quasi-Static Scenario

T, Time period for MR	150 TS′
B, Number of FS′ on each fiber link	358
$\frac{\lambda_f}{\mu_f}$, Load of background flow-oriented traffic	800 Erlangs
Book-ahead time of flow-oriented requests	$[0,10]$ TS′
Bandwidth requirements of flow-oriented requests	$[1,16]$ FS′
Number of data-oriented requests	100
$\mathcal{F}$, Data amount of data-oriented requests	$[10,160]$ FS′·TS′
$d_{\max}$, Longest look-ahead period	$15,20,25$ TS′
Q, Maximum RSA reconfiguration times	$1,3,5,7$
K, Number of alternative routing paths	5

A. Simulations With Quasi-Static Network Scenario

We first compare the proposed DPM with the MILP model and conduct simulations with a quasi-static network scenario. Basically, we first perform dynamic network provisioning for flow-oriented requests to generate time-varying network spectrum utilization. Then, we introduce data-oriented requests and serve them based on the generated network status.

1) Simulation Parameters: The simulations use the NSFNET topology in [9]. In order to generate the time-varying network spectrum utilization, we create flow-oriented requests

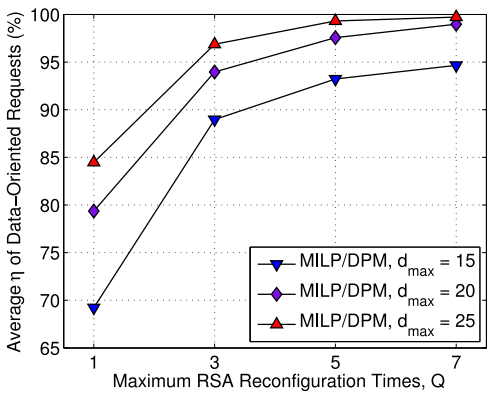

Fig. 4. Results on average percentage of transmitted data achieved by MILP and DPM.

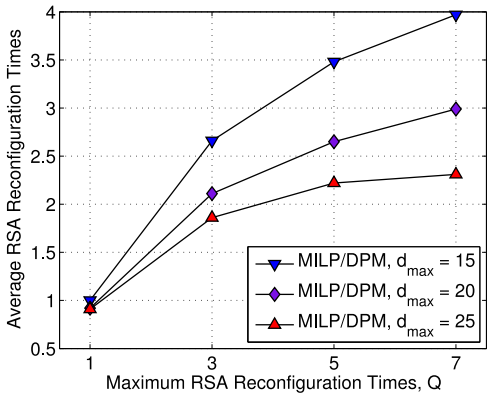

Fig. 5. Results on average RSA reconfiguration times provided by MILP and DPM.

TABLE II
RESULTS ON COMPUTATION TIME

Network Parameters		Average Running Time (s)	
$d_{\max}$ (TS′)	Q	MILP	DPM
15	1	0.5913	0.0049
	3	3.0012	0.0095
	5	8.2978	0.0137
	7	11.3763	0.0187
20	1	1.7856	0.0078
	3	19.3251	0.0155
	5	83.5315	0.0227
	7	273.7078	0.0313
25	1	4.9130	0.0120
	3	56.1916	0.0232
	5	363.8915	0.0347
	7	1482.5391	0.0466

TABLE III
SIMULATION PARAMETERS FOR THE DYNAMIC SCENARIO

$\frac{\lambda_f}{\mu_f}$, Flow-oriented traffic load	$[300, 1200]$ Erlangs
$\frac{\lambda_d}{\mu_d}$, Data-oriented traffic load	$\{0, 180, 240\}$ Erlangs
Book-ahead time of flow-oriented requests	$[0, 10]$ TS′
$d_{\max}$, Longest look-ahead period	$[6, 10]$ TS′
Q, Maximum RSA reconfiguration times	5

with the Poisson process, i.e., they come in with an average rate of λ_f requests per TS and their durations follow the negative exponential distribution with an average of $\frac{1}{\mu_f}$ TS′. Then, the background flow-oriented traffic can be quantified with $\frac{\lambda_f}{\mu_f}$ in Erlangs. Note that, to generate 2-D spectrum fragments, we apply the IR-and-AR-mixed traffic model to the flow-oriented requests and use the specific starting-time and specific duration (STSD) scheme. The book-ahead time of the flow-oriented requests varies within $[0, 10]$ TS′. When its book-ahead time is 0, the request is IR, otherwise, it is an AR request. We serve the flow-oriented requests with shortest-path routing and first-fit spectrum assignment (SPR-FFSA) [10] and generate time-varying spectrum utilization over the period of $[1, T]$ TS′.

Then, each data-oriented request $R(s, d, \mathcal{F}, t_a, d_{\max})$ is generated as follows. The s-d pair is randomly selected, and the arrival time t_a is uniformly distributed within $[0.3T, 0.6T]$ TS′. We change the longest look-ahead period $d_{\max}$ to study its effect on algorithms' time complexity. In each simulation, we make sure that all the data-oriented requests have the same $d_{\max}$. Then, by comparing the average running time of simulations with different $d_{\max}$, we investigate the effect of $d_{\max}$ on the time complexities of the MILP model and DPM. Table I summarizes the key simulation parameters.

2) Simulation Results: Fig. 4 compares the results on average η of the data-orient requests when the simulations use $d_{\max} = 15$ TS′, $d_{\max} = 20$ TS′ and $d_{\max} = 25$ TS′, respectively. Since DPM can also provide the exact solution to the MR problem, it achieves the same results as the MILP. Hence, we only plot one curve in each scenario. In the figure, we observe that the average η (i.e., the percentage of the transmitted data achieved by MR) increases with Q (i.e., the number of the maximum RSA reconfiguration times). Basically, when we increase Q from 1 to 7, the average η increases from 69% $\sim$ 84% to 95% $\sim$ 99%. This is because when Q is larger, the number of 2-D spectrum fragments that MR can use to serve a data-oriented request also becomes larger, and hence more data can be transmitted for the request. Also, it can be seen that for a larger $d_{\max}$, the average η is higher when Q is the same. This is because a larger $d_{\max}$ leads to a larger solution space for DPM to search for proper 2-D spectrum fragments to serve each data-oriented request, and hence it can achieve more data transmission on average.

Fig. 5 shows the results on average RSA reconfiguration times per data-oriented request, for the simulations using $d_{\max} = 15$ TS′, $d_{\max} = 20$ TS′ and $d_{\max} = 25$ TS′, respectively. The MILP model and DPM still provide the same results. It can be seen that for most of the cases, the results on average RSA reconfiguration times are smaller than Q. This observation verifies that the MR algorithms try to use as few RSA reconfigurations as possible to achieve the highest percentage of data transmitted for each data-oriented request. We can also see that when $d_{\max}$ increases, the average RSA reconfiguration times become less when Q is the same. Again, this is because a larger $d_{\max}$ leads to a larger solution space for DPM to find the proper MR schemes, and thus it can serve each data-oriented request with less RSA reconfigurations.

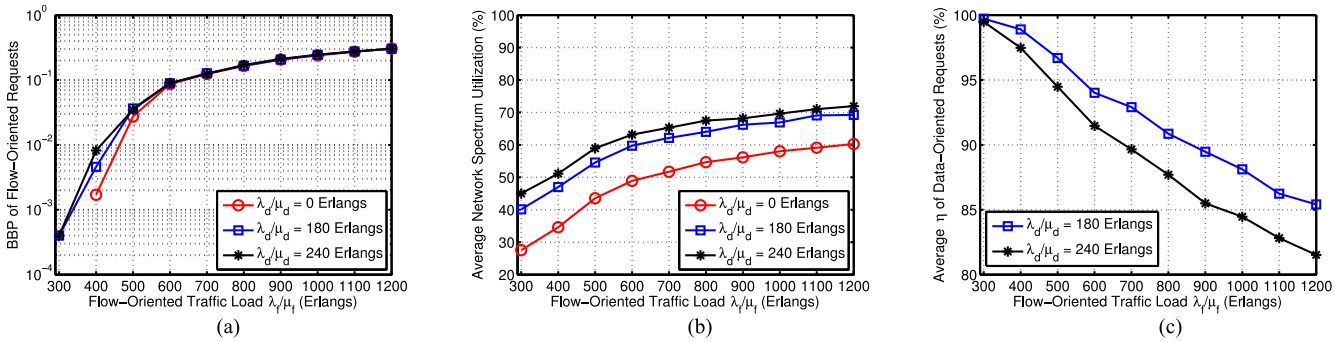

Fig. 6. Results from simulations with dynamic network scenario. (a) BBP of flow-oriented requests. (b) Average network spectrum utilization. (c) Average percentage of transmitted data.

TABLE IV
DISTRIBUTION OF DATA-ORIENTED REQUESTS BASED ON THEIR ACHIEVED η

$\frac{\lambda_f}{\mu_f}$	$\frac{\lambda_d}{\mu_d}$	Distribution of Data-Oriented Requests				
(Erlangs)	(Erlangs)	$\eta \in [0, 25\%)$	$\eta \in [25\%, 50\%)$	$\eta \in [50\%, 75\%)$	$\eta \in [75\%, 100\%)$	$\eta = 100\%$
300	180	0.04%	0.15%	0.21%	0.17%	99.43%
	240	0.04%	0.26%	0.50%	0.44%	98.76%
800	180	1.90%	5.93%	7.10%	6.00%	79.07%
	240	4.40%	7.76%	8.55%	6.82%	72.47%
1200	180	4.84%	9.24%	9.64%	7.93%	68.35%
	240	7.53%	11.62%	10.89%	7.24%	62.72%

3) Computation Time: Table II shows the average running time of the MILP and DPM for serving 100 data-oriented requests. It can be seen that DPM is much more time-efficient than the MILP. We also observe that the running time increases with $d_{\max}$ and Q, and the MILP has the running time increased much more dramatically. For example, the running time of the MILP becomes around 2500 times longer when the combination of $\{d_{\max},\ Q\}$ changes from $\{15\ TS',\ 1\}$ to $\{25\ TS',\ 7\}$, while for the same parameter change, the running time of DPM only increases for about 10 times.

B. Simulations With Dynamic Network Scenario

In order to further evaluate the performance of the proposed DPM, we also design simulations with a dynamic scenario for MR-based bulk-data transfer to recycle 2-D spectrum fragments in EONs. Specifically, we make flow-oriented and data-oriented requests coexist in the network, i.e., they both come and leave on-the-fly and are served simultaneously.

1) Simulation Parameters: Most of the simulation parameters are the same as those in Section VI-A. Here, we quantify the flow-oriented and data-oriented requests separately. The traffic load of flow-oriented requests is still quantified as $\frac{\lambda_f}{\mu_f}$ in Erlangs, and they have the book-ahead time evenly distributed within $[0, 10]$ TS′. Meanwhile, the data-oriented requests are also generated according to the Poisson process, and their traffic load is quantified as $\frac{\lambda_d}{\mu_d}$ in Erlangs, where λ_d and μ_d are the corresponding statistical parameters. Table III lists the new/different key simulation parameters.

2) Simulation Results: When provisioning the heterogenous requests in the simulations, we always serve the flow-oriented ones first to provide them with better QoS guarantee. The flow-oriented requests are still served with the SPR-FFSA scheme, and the data-oriented ones are served with DPM with $Q = 5$.

At each flow-oriented traffic load, we insert different amounts of data-oriented requests in the EON and simulate the network operation. Fig. 6(a) compares the results on bandwidth blocking probability (BBP) of the flow-oriented requests, when the data-oriented traffic load is 0, 180 and 240 Erlangs. It can be seen that the BBP results are almost unchanged when the data-oriented traffic load increases. This observation verifies that our proposed algorithm can recycle 2-D spectrum fragments in EONs while not affecting the provisioning of flow-oriented requests. The results also confirm that the adverse effect from the fact that MR can leave small 2-D spectrum fragments in the network is very limited.

Fig. 6(b) plots the results on average network spectrum utilization. We notice that when the data-oriented traffic load increases from 0 to 240 Erlangs, the network spectrum utilization also increases significantly. This can be viewed as another proof of the proposed algorithm's effectiveness on recycling 2-D spectrum fragments in EONs. Fig. 6(c) shows the results on average percentage of transmitted data per data-oriented request (i.e., η), which indicate that DPM can still achieve over 80% data-transfer on average, when the EON is as crowded as that the traffic loads of the flow- and data-oriented requests are 1200 and 240 Erlangs, respectively.

In the simulations, we calculate η under the constraints of $d_{\max}$ and Q and treat the data-oriented requests as incomplete if they are served with $\eta < 1$. Table IV shows the distribution of data-oriented requests based on their achieved η at different traffic loads. As expected, the percentage of data-oriented requests

that are transmitted partially (i.e., with $\eta < 1$) increases with the traffic loads. Nevertheless, we also observe that more than 60% data-oriented requests can be transmitted completely, even when the traffic loads of flow- and data-oriented requests are as high as 1200 and 240 Erlangs, respectively. We will study how to handle the "left-over" demands in future work. For instance, we can assume that data-oriented requests are served with the best-effort mechanism. Therefore, each "left-over" demand can be treated as a new data-oriented request to schedule, and we will try to finish the data transfer with multiple MR rounds (i.e., providing degraded QoS on latency).

VII. Conclusion

In this paper, we investigated how to optimize malleable reservation (MR) for data-oriented requests in EONs where there are also flow-oriented requests. RSA reconfigurations were used for each data-oriented request to effectively recycle the 2-D spectrum fragments generated by the flow-oriented ones. We first formulated an MILP model for the MR problem. The MILP model tried to maximize the percentage of transmitted data for each data-oriented request under the constraint that we can only use a limited number of RSA reconfigurations. Then, we proposed a dynamic programming method (DPM), which can solve the MR problem exactly in polynomial time. Simulation results suggested that DPM can provide the same exact solutions as the MILP, while its time complexity is significantly lower. The results also verified that DPM could recycle 2-D spectrum fragments in EONs successfully while not affecting the provisioning of flow-oriented requests.

References

[1] O. Gerstel, M. Jinno, A. Lord, and S. Yoo, "Elastic optical networking: A new dawn for the optical layer?," *IEEE Commun. Mag.*, vol. 50, pp. s12–s20, Feb. 2012.

[2] M. Chen, H. Jin, Y. Wen, and V. Leung, "Enabling technologies for future data center networking: A primer," *IEEE Netw.*, vol. 27, no. 4, pp. 8–15, Aug. 2013.

[3] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," in *Proc. Grid Comput. Workshop*, Nov. 2008, pp. 12–16.

[4] N. Laoutaris, M. Sirivianos, X. Yang, and P. Rodriguez, "Inter-datacenter bulk transfers with netstitcher," in *Proc. ACM SIGCOMM*, Aug. 2011, pp. 74–85.

[5] 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.

[6] Y. Wang, X. Cao, and Y. Pan, "A study of the routing and spectrum allocation in spectrum-sliced elastic optical path networks," in *Proc. INFOCOM*, Apr. 2011, pp. 1503–1511.

[7] W. Lu, X. Zhou, L. Gong, and Z. Zhu, "Scalable network planning for elastic optical orthogonal frequency division multiplexing (OFDM) networks," in *Proc. 8th Int. Conf. Commun. Syst. Netw. Digital Signal Process.*, Jul. 2012, pp. 1–4.

[8] L. Gong, X. Zhou, W. Lu, and Z. Zhu, "A two-population based evolutionary approach for optimizing routing, modulation and spectrum assignments (RMSA) in O-OFDM networks," *IEEE Commun. Lett.*, vol. 16, no. 9, pp. 1520–1523, Sep. 2012.

[9] Z. Zhu, W. Lu, L. Zhang, and N. Ansari, "Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing," *J. Lightw. Technol.*, vol. 31, no. 1, pp. 15–22, Jan. 2013.

[10] W. Lu and Z. Zhu, "Dynamic service provisioning of advance reservation requests in elastic optical networks," *J. Lightw. Technol.*, vol. 31, no. 10, pp. 1621–1627, May 2013.

[11] S. Shen, W. Lu, X. Liu, L. Gong, and Z. Zhu, "Dynamic advance reservation multicast in data center networks over elastic optical infrastructure," in *Proc. 39th Eur. Conf. Opt. Commun.*, Sep. 2013, pp. 1–3.

[12] K. Christodoulopoulos, I. Tomkos, and E. Varvarigos, "Time-varying spectrum allocation policies and blocking analysis in flexible optical networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 1, pp. 13–25, Jan. 2013.

[13] M. Klinkowski, M. Ruiz, L. Velasco, D. Careglio, V. Lopez, and J. Comellas, "Elastic spectrum allocation for time-varying traffic in flexgrid optical networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 1, pp. 26–38, Jan. 2013.

[14] L. Burchard, H. Heiss, and C. De Rose, "Performance issues of bandwidth reservations for grid computing," in *Proc. 15th Symp. Comput. Archit. High Perform. Comput.*, Nov. 2003, pp. 82–90.

[15] M. Zhang, Y. Yin, R. Proietti, Z. Zhu, and S. Yoo, "Spectrum defragmentation algorithms for elastic optical networks using hitless spectrum retuning techniques," in *Proc. Opt. Fiber Commun. Conf.*, Mar. 2013, pp. 1–3.

[16] W. Shi, Z. Zhu, M. Zhang, and N. Ansari, "On the effect of bandwidth fragmentation on blocking probability in elastic optical networks," *IEEE Trans. Commun.*, vol. 61, no. 7, pp. 2970–2978, Jul. 2013.

[17] R. Proietti, C. Qin, B. Guan, Y. Yin, R. P. Scott, R. Yu, and S. Yoo, "Rapid and complete hitless defragmentation method using a coherent RX LO with fast wavelength tracking in elastic optical networks," *Opt. Exp.*, vol. 20, pp. 26-958–26-968, Nov. 2012.

[18] J. Zhang, Y. Zhao, H. Yang, Y. Ji, H. Li, Y. Lin, G. Li, J. Han, Y. Lee, and T. Ma, "First demonstration of enhanced software defined networking (eSDN) over elastic grid (eGrid) optical networks for data center service migration," in *Proc. Opt. Fiber Commun. Conf.*, Mar. 2013, pp. 1–3.

[19] R. Casellas, R. Martinez, R. Munoz, R. Vilalta, L. Liu, T. Tsuritani, and I. Morita, "Control and management of flexi-grid optical networks with an integrated stateful path computation element and OpenFlow controller," *J. Opt. Commun. Netw.*, vol. 5, pp. A57–A65, Oct. 2013.

[20] R. Casellas, R. Munoz, J. Fabrega, M. Moreolo, R. Martinez, L. Liu, T. Tsuritani, and I. Morita, "Design and experimental validation of a GMPLS/PCE control plane for elastic CO-OFDM optical networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 1, pp. 49–61, Jan. 2013.

[21] S. Ma, C. Chen, S. Li, M. Zhang, S. Li, Y. Shao, Z. Zhu, L. Liu, and B. Yoo, "Demonstration of online spectrum defragmentation enabled by OpenFlow in software-defined elastic optical networks," in *Proc. Opt. Fiber Commun. Conf.*, Mar. 2014, pp. 1–3.

[22] R. Guerin and A. Orda, "Networks with advance reservations: The routing perspective," in *Proc. INFOCOM*, Mar. 2000, pp. 118–127.

[23] S. Gangulay, A. Sen, G. Xue, B. Hao, and B. Shen, "Optimal routing for fast transfer of bulk data files in time-varying networks," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2004, pp. 1182–1186.

[24] Y. Lin and Q. Wu, "Path computation with variable bandwidth for bulk data transfer in high-performance networks," in *Proc. INFOCOM Workshops*, Apr. 2009, pp. 1–6.

[25] Y. Wang, S. Su, S. Jiang, Z. Zhang, and K. Shuang, "Optimal routing and bandwidth allocation for multiple inter-datacenter bulk data transfers," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2012, pp. 5538–5542.

[26] S. Naiksatam and S. Figueira, "Elastic reservations for efficient bandwidth utilization in LambdaGrids," *Future Gener. Comput. Syst.*, vol. 23, pp. 1–22, Jan. 2007.

[27] D. Andrei, M. Tornatore, M. Batayneh, C. Martel, and B. Mukherjee, "Provisioning of deadline-driven requests with flexible transmission rates in WDM mesh networks," *IEEE/ACM Trans. Netw.*, vol. 18, no. 2, pp. 353–366, Apr. 2010.

[28] A. Patel and J. Jue, "Routing and scheduling for variable bandwidth advance reservation," *J. Opt. Commun. Netw.*, vol. 3, pp. 912–923, Dec. 2011.

[29] Y. Katsuyama, M. Hashimoto, K. Nishikawa, A. Ueno, M. Nooruzzaman, and O. Koyama, "Lightpath reconfiguration in regional IP-over-WDM Networks by a centralized control system," in *Proc. IEEE 32nd Local Comput. Netw.*, Oct. 2007, pp. 63–72.

[30] A. Mahimkar, A. Chiu, R. Doverspike, M. Feuer, P. Magill, E. Mavrogiorgis, J. Pastor, S. Woodward, and J. Yates, "Bandwidth on demand for inter-data center communication," in *Proc. ACM HotNets*, Nov. 2011, pp. 24–29.

[31] GNU linear programming kit (GLPK). [Online]. Available: http://www.gnu.org/software/glpk/

Authors' biographies not available at the time of publication.