Elastic Optical Network with Spectrum Slicing for Fragmented Bandwidth Allocation

Nattapong Kitsuwan[a,*], Praphan Pavarangkoon[a], Avishek Nag[b]

[a]*Department of Computer and Network Engineering,*
The University of Electro-Communications, 1-5-1 Chofugaoka Chofu-shi, Tokyo, Japan
[b]*School of Electrical and Electronic Engineering,*
University College Dublin, Dublin 4, Ireland.

Abstract

Elastic Optical Networks (EONs) allow the channel spacing and the spectral width of an optical signal to be dynamically adjusted and hence have become an important paradigm in managing the heterogeneous bandwidth demands of optical backbone networks. The entire available optical spectrum is divided into some spectrum slots which define the smallest granularity of bandwidth and optical signals with variable bandwidths can occupy different number of such slots. The constraints imposed by the physical layer of an EON require that the slots occupied by an optical signal from source to destination have to be consecutive and contiguous in terms of their relative position in the optical spectrum. Furthermore, the same spectrum slots need to be reserved throughout the entire optical signal's path from its source to destination. The above constraints make the routing and spectrum allocation (RSA) in EONs very challenging because unavailability of enough spectrum slots that together equals the spectral width of the optical signal associated with an end-to-end request, will result in blocking of the request. Recent developments in the physical layer technologies have made all-optical 'slicing' of a request possible and make the request to be 'fit' into multiple non-consecutive spectral slots in an EON. But these all-optical

---

[*]Parts of this paper were partially presented at International Conference on Computing, Networking and Communications (ICNC 2019).
[*]Corresponding author
Email address: `kitsuwan@uec.ac.jp` (Nattapong Kitsuwan)

'slicers' employ complex technologies and can be very costly to employ. In this paper, we propose a spectrum allocation scheme for an EON node architecture with these 'slicers' and we also formulate a modified RSA scheme for EONs employing slicers, both as a mixed-integer linear programming (MILP) model and a heuristic algorithm. Our main aim is to analyze the tradeoff between the number of slicers that can be used per node versus the spectrum utilization and bandwidth blocking rate. The numerical results show that the proposed scheme with slicers can significantly improve bandwidth blocking rate, compared to the conventional scheme without slicer.

1. Introduction

With the advent of new paradigms like 5G, the requirements for the new-generation optical networks have been evolving in terms of the bandwidths that they need to support. Applications such as high-definition video streaming,
5 inter-data-center communication, online multi-player gaming, are constantly challenging the capacities of today's networks. Dense wavelength-division multiplexing (DWDM), which is an optical multiplexing technology, is employed in optical backbone networks to support the growing network traffic for more than two decades. Different wavelengths can be modulated with different information-
10 bearing signal and can be simultaneously transmitted over the same optical fiber. Each modulated wavelength occupies a certain spectral width, called 'channel spacing'. A standard for channel spacing, defined by the international telecommunication union (ITU), is traditionally fixed to 50 GHz [1]. Sometimes, the aggregated bandwidth of an information-carrying signal might be too high and
15 can require more than the capacity of one wavelength. Such 'super-wavelength' requests, which require more than 50 GHz, cannot be transmitted in a single channel with the traditional 50 GHz spacing. If they require more than 50 GHz, they cannot be transmitted within the fixed 50 GHz grid, irrespective of

the number of carriers inside. On the other hand, if the information modulating the wavelength is not enough to occupy the entire channel spacing allocated by the ITU standard, it leads to inefficient spectrum utilization.

This leads to the introduction of the elastic optical network (EON) concept [2] for optical backbone networks. EONs facilitate dynamic adjustment of the channel spacing based on a requested bandwidth. Bandwidth-variable transceivers support both high and low demands depending on the required reach. These transceivers may eventually allow for adaptive use of resources, flexible use of spectrum, and a flexible relationship between client technologies such as Internet Protocol (IP) and the optical layer [3]. In EONs, the entire optical spectrum is divided into small frequency slots, which provide better granularity than DWDM and better spectrum utilization. As a result, a wide variety of high- and low-bandwidth requests can be accommodated in an EON with utmost flexibility. The EON is a promising technology to support multiple data center systems and the big data applications [4]. However, allocation of these spectrum slots to the optical requests is a challenging exercise because all the spectrum slots for a request between a source and its destination must be aligned. Spectrum contiguity and continuity constraints must be guaranteed i.e., the slots allocated to a request have to be continuous and adjacent and the same slots have to be reserved throughout the entire optical path of the request from its source to destination.

A common issue in an EON is the bandwidth fragmentation problem which occurs because of dynamically setting up and tearing down of lightpaths [5]. Because of the termination of a few lightpaths at different links of the network, it is possible that those terminated lightpaths are not on continuous and contiguous spectrum slots. So, these slots, though available, are isolated from each other. Because of the spectrum continuity and contiguity constraint associated with the EONs, a new lightpath request in the network may not fit the available spectrum slots due to either non-alignment along the routing path or non-contiguity in the optical domain. In such a situation, the lightpath request is rejected even though enough slots are available but the slots are non-contiguous.

⁵⁰ The fragmentation problem can be solved by two approaches. The first approach is to use a defragmentation method [6]. This method reallocates the occupied spectrum slots to offer enough available consecutive spectrum slots for the new request, e.g., push-pull scheme [7, 8], hop-tuning scheme [9], and make-before-break scheme [10]. The second approach for solving the fragmentation
⁵⁵ problem is to use a spectrum-slot-allocation algorithm to allocate the requests in such a way that there remains as much available consecutive spectrum slots as possible for future requests [11, 12, 13]. A first-fit scheme allocates requests at the lowest consecutive spectrum slots to leave a large band of consecutive spectrum slots in the middle of the spectrum range [14]. A first-last-fit scheme
⁶⁰ categorizes the request into long and short lightpath requests. The long lightpath requests are allocated at the lowest index as the first-fit scheme. The short lightpath requests are allocated at the highest index [15]. In this paper, we have considered the second approach for our allocation scheme because it does not require re-allocation of the existing spectrum bands.

⁶⁵ Since the primary target of this work is to come up with a routing and spectrum allocation scheme in an EON that exploits the exemplary slicing and stitching of optical spectrum, it is worthwhile to discuss this technology briefly. Slicing and stitching is a new technology that breaks the spectrum-contiguity constraint in the optical domain [16]. A request can be split into two components
⁷⁰ with the slicing process. The process generates a copy of the original data on another optical frequency using coherent optical frequency combs and nonlinear wave mixing [17]. Partial spectra of both the original and the copy are sliced into two smaller channels by optical filters. The optical filters select the spectrum band for both the channels based on the allocation process. It should be noted
⁷⁵ that the number of slots per sliced component can be changed for different requests. Two sliced components can then be allocated into two separated zones of consecutive spectrum slots. At the receiver, the two sliced components are combined to recover the original data by phase-preserving wavelength conversion, called a stitching process. An experiment in [16] showed successful three-channel
⁸⁰ slices of a 28-Gbaud quadrature-phase-shift-keying (QPSK) channel by using an

4

optical frequency comb. Different number of slice components can be created by the above process. However, the system performance can degrade with the following reasons as explained in [16]. First, the power of the optical signal is attenuated by the loss of different equipment. Second, nonlinear wave mixing in both stages of the channel slicing and stitching requires sufficient signal power as provided by a 2W erbium-doped fiber amplifier (EDFA) with a 6 dB noise figure. Third, there are optical components with limited bandwidth. It should be noted that each slicer is not able to slice multiple requests, simultaneously. Only one request can be sliced by a slicer. Furthermore, though the authors in [16] verified successful use of the slicing processes for a request, the splitting position and the required number of slots for each component is unknown.

This paper proposes a scheme to determine the splitting position and required number of slots for each slice component in an EON network with spectrum slicing. Each node in the network is assumed to have slicing devices, called slicers. Since the technology for slicing is quite new and can be expensive, the number of slicers is limited and therefore, all requests cannot be sliced. Some of the preliminary concepts of this paper were presented in [18]. Compared to [18], the allocation problem to minimize the number of slicers and maximize the utilized slots is formulated as a mixed-integer linear programming (MILP) problem. A heuristic algorithm of multiple shortest paths for each request is also introduced to solve the problem for a large network. The dependencies on the number of paths and guardband, for each request are also investigated.

Apparently, this work may seem identical to the multipath routing schemes reported in [19, 20, 21] but there is a subtle difference between these works and our work. Our research is fundamentally based on the physical slicing technology proposed by Y. Cao et al., [16]. Owing to this technology proposed by Cao et al., an end-to-end lightpath can be supported on several combinations of non-contiguous spectrum slots. A given source-to-destination demand can be supported by multiple lightpaths, however, each of those paths corresponding to a particular demand are not split further as in the case of multipath splitting proposed by [19, 20, 21]. The paths of a source-destination demand, instead of
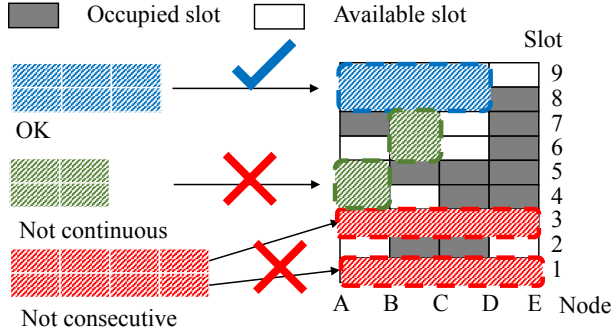
5

Figure 1: Allocation constraints in general EON.

further splitting into subpaths just distribute their demands into non-contiguous slots owing to the slicing and stitching technology.

The rest of the paper is organised as follows: In Section 2, we discuss the conventional routing and spectrum allocation scheme in EONs and the constraints therein. Then in Section 3, we present our scheme of spectrum slicing, with a proposed node architecture for an EON employing spectrum slicers. We also present a mixed-integer linear programming (MILP) model for routing and spectrum allocation in an EON with spectrum slicers, in Section 3. In Section 4, we present the results of our analysis in detail and finally in Section 5, we conclude the paper.

2. Conventional scheme

As discussed earlier, in an EON, there are two constraints for spectrum slot allocation: the spectrum-contiguity constraint and the spectrum-continuity constraint. The spectrum-contiguity constraint suggests that a requested spectrum band cannot be split. All the slots allocated to a request have to be contiguously adjacent in terms of their position on the spectrum. On the other hand, the spectrum-continuity constraint enforces that all the contiguous slots allocated to a request have to remain same in each link throughout the entire optical path of a request, if a spectrum converter is not applied. If any one of the above constraints is not satisfied, the request will be rejected.

6

Figure 1 illustrates the above-mentioned constraints in a general EON. Let us consider we have three requests, marked with red, green, and blue. The blue request needs two slots from node A to node D and in all the three links between A to D the blue request is allocated using slots 8 and 9 as these slots are continuous and contiguous between nodes A to D. Next, the green request needs two slots from node A to node C. For this request, an allocation option can be to use slots 4 and 5 on link AB, and slots 6 and 7 on link BC. This request is rejected since the spectrum band from node A to node C is not continuous. Finally, in case of the red request, it needs two slots from node A to node E. It has the option to occupy slots 1 and 3. This request is also rejected since the used slots on each link are not contiguous. It should be noted that Fig. 1 only focuses on the constraints in the general EON so that guardband is not considered in this figure. However, the guardband must be considered in an allocation process.

The routing and spectrum slot assignment (RSA) scheme is used to select a route from source to destination and to assign spectrum slots on the selected route, for a request [22]. The scheme aims to pack the end-to-end requests by maintaining the spectrum continuity and contiguity constraints as much as possible, so that future requests can be accommodated efficiently.

Based on this principle, several RSA schemes exist. For example, a first-fit scheme tends to solve the slot fragmentation problem. This scheme allocates the new request on the possible consecutive slots with the lowest slot index number [23]. Therefore, it leaves a large number of empty slots near the high-slot-index zone for accommodating future requests.

A first-last fit scheme on the other hand, categorizes the lightpath requests into disjoint and non-disjoint paths. Requests with the disjoint paths are allocated at the lowest slot index [24]. Requests with the non-disjoint paths are allocated at the highest slot index. Another scheme called the first-last-exact fit scheme categorizes the lightpath requests as the first-last fit scheme [25]. But in this case, the requests with the disjoint and non-disjoint paths are allocated at the lowest and highest indexed available slots, respectively. It ensures that
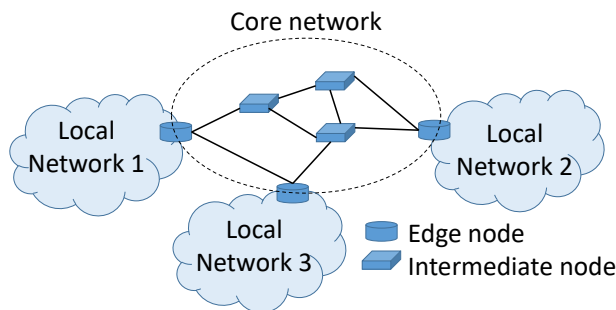
Figure 2: Network architecture using in proposed scheme.

the number of available consecutive slots matches with the number of requested slots.

Although the above schemes provide consecutive spectrum slots as much as possible for the future requests, bandwidth blocking occurs when the available consecutive slots are not enough for the request.

3. Proposed scheme

We propose an RSA scheme in an EON to break the consecutive-slots constraint by adopting a technology to slice and to recover the signal, so that the contiguity constraint can be relaxed and the allocation of requests similar to the red request in Fig. 1 is possible. We focus on the network architecture as shown in Fig. 2. The network consists of several local networks. Each local network connects each other via a core network. The technology to slice and to recover the optical signal are placed at the edge nodes of the core network. A request is sliced at the ingress edge node of the core network using the newly invented slicing technology [16], into several spectrum components if the spectrum band does not fit any available consecutive spectrum slots. It should be noted that the proposed scheme is not able to slice the request at intermediate nodes. The spectrum slicing is performed at only the ingress edge nodes. Each spectrum component consists of several consecutive spectrum slots. The sliced spectrum components are inserted into the available non-contiguous
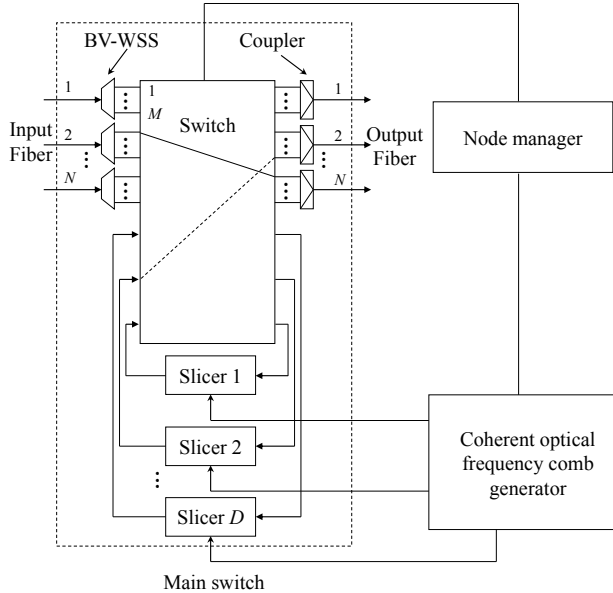
8

Figure 3: Edge node architecture for proposed scheme.

spaces along the requested optical path to a desired egress node in the core network. The components are recovered, using a stitching technology [16] at the

185 egress node. In our proposed allocation scheme, our main focus is to allocate the requested spectrum slots into the available spaces. Therefore, while abstracting the EON and formulating a modified RSA scheme, only the slicing technology will be considered for the allocation. We do not consider stitching in the allocation scheme. Stitching comes with slicing for analyzing the performance of

190 the RSA scheme, there is no need of abstracting the stitching technology. It should be noted that the proposed scheme considers only the requests that pass through the core network.

3.1. Node architecture

The node architecture is shown in Fig. 3. A typical node in an EON em-

195 ploying the slicing technology, contains three modules: the node manager, the coherent optical frequency comb generator, and the main switch. The node

9

manager orchestrates the connections between the input and the output ports. It also selects the frequency combs for each slicer as decided by an EON controller. The function of the coherent optical frequency comb generator is to generate frequency combs to the slicers. The main switch consists of N input and output fibers and D slicers. The optical signals at each input fiber are separated using the bandwidth variable wavelength selective switches (BV-WSS), where each BV-WSS has M output ports. Each output port of the BV-WSS has a one-to-one correspondence with an input port of the switch fabric. Slicers are connected to the switch fabric and are shared by all connections, if required, that originate from a node. A group of sliced spectrum is combined by an optical coupler before being forwarded to the output fiber. It should be noted that a slicer is not able to be shared with other requests if it is being used.

3.2. Problem formulation for spectrum allocation with slicing

The objective of the problem formulation is minimizing two things. The major objective is to minimize the number of utilized slicers, followed by minimizing the maximum number of slots index as the minor objective. In other words, if there are several solutions that result the same number of utilized slicers, the maximum number of slot index in the network is minimized. The parameters used in the problem formulation are defined in Table 1.

Table 1: Parameters used in problem formulation.

Parameter	Meaning
E	Set of links.
F	Set of spectrum slots.
K	Set of source-destination $(s-d)$ pairs.
P	Set of paths.
M	Set of routes for every path, where the 4-tuple $(i,j,k,p) \in M$ indicates that link $(i,j) \in E$ is on the route for pair $k \in K$ on path $p \in P$.
S	Set of slice components.
Q	Set of slicing patterns.
T_k	Number of requested slots for pair $k \in K$.
O_{kq}^s	Number of utilized slots for slicer component $s \in S$ with slicing pattern $q \in Q$ for $k \in K$.
l^{kq}	Number of required slicers used for slicing pattern $q \in Q$ on path $k \in K$.
r_{ij}^{kp}	A given binary parameter, which indicates the utilized link $(i,j) \in E$ for pair $k \in K$ on path $p \in P$.
x_{fij}^{kpqs}	A binary decision variable that is set to 1 if slot index $f \in F$ is the starting spectrum slot index for pair $k \in K$ on path $p \in P$ that is allocated on link $(i,j) \in E$ for s-th optical component with slicing pattern $q \in Q$.
y_{fij}^{kpqs}	A binary decision variable that is set to 1 if slot index $f \in F$ is used for pair $k \in K$ on path $p \in P$ on link $(i,j) \in E$ for s-th optical component with slicing pattern $q \in Q$.
c	A decision variable that indicates the maximum spectrum slot in the network.
a^{kpq}	A binary decision variable that is set to 1 if path $p \in P$ for pair $k \in K$ and slicing pattern $q \in Q$ is selected.
G	Number of guardband slots to separate adjacent spectrum components.
U	Set of occupied slots on links.
g_{fij}	A given parameter, where $(f,i,j) \in U$, indicating that slot index $f \in F$ on link $(i,j) \in E$ is occupied if it is set to 1, and 0 otherwise.

The objective function is presented in Eq. (1a). The first term minimizes the number of utilized slicers. The second term, $c \times \epsilon$, is a secondary objective to minimize the maximum utilized slot index in the network. ϵ is set to a sufficiently small value so that the second term in Eq. (1a) cannot affect the minimization of the first term. The constraints of the problem are formulated as Eqs (1b)-(1p).

$$\min \quad \sum_{k \in K} \sum_{p \in P} \sum_{q \in Q} a^{kpq} l^{kq} + (c \times \epsilon) \tag{1a}$$

subject to:

$$\sum_{f \in F} x_{fij}^{kpqs} = 1, \quad \forall (i,j,k,p) \in M, q \in Q, s \in S \tag{1b}$$

$$x_{fij}^{kpqs} a^{kpq} \leq y_{f'ij}^{kpqs},$$
$$\forall (i,j,k,p) \in M, f \in \{1, \cdots, |F| - O_{kq}^{s} + 1\},$$
$$f' \in \{f, \cdots, f + O_{kq}^{s} - 1\}, s \in S, q \in Q \tag{1c}$$

$$x_{fij}^{kpqs} a^{kpq} = 0,$$
$$\forall (i,j,k,p) \in M, s \in S, q \in Q,$$
$$f \in \{|F| - O_{kq}^{s} + 2, \cdots, |F|\} \tag{1d}$$

$$x_{fij}^{kpqs} a^{kpq} \leq r_{ij}^{kp},$$
$$\forall (i,j,k,p) \in M, s \in S, q \in Q,$$
$$f \in \{1, \cdots, |F| - O_{kq}^{s} + 1\}, O_{kq}^{s} \neq 0 \tag{1e}$$

$$y_{fij}^{kpqs} = y_{fjj'}^{kpqs},$$
$$\forall (i,j,k,p), (j,j',k,p) \in M,$$
$$f \in F, q \in Q, s \in S \tag{1f}$$

$$\sum_{s \in S} \sum_{f \in F} y_{fij}^{kpqs} = (T_k + (l^{kq} + 1)G) a^{kpq},$$
$$\forall (i,j,k,p) \in M, q \in Q \tag{1g}$$

$$y_{fij}^{kpqs} + y_{fij}^{k'p'qs'} \leq 1,$$
$$\forall f \in F, (i,j,k,p), (i,j,k',p') \in M,$$

12

$$s, s' \in S, (k \neq k' \text{ or } p \neq p' \text{ or } s \neq s') \qquad \text{(1h)}$$

$$\sum_{k \in K} \sum_{p \in P} \sum_{q \in Q} \sum_{s \in S} y_{fij}^{kpqs} \leq 1,$$
$$\forall f \in F, (i,j) \in E \qquad \text{(1i)}$$

$$y_{f'ij}^{kpqs} \leq (x_{fij}^{kpqs} a^{kpq}) + (O_{kq}^s \times a^{kpq}),$$
$$\forall q \in Q, s \in S, f \in \{1, \cdots, |F| - O_{kq}^s + 1\},$$
$$f' \in \{f, \cdots, f + O_{kq}^s - 1\}, (i,j,k,p) \in M \qquad \text{(1j)}$$

$$(x_{fij}^{kpqs} a^{kpq} \times f) + O_{kq}^s - 1 \leq c,$$
$$\forall f \in F, (i,j,k,p) \in M, q \in Q, s \in S \qquad \text{(1k)}$$

$$\sum_{p \in P} \sum_{q \in Q} a^{kpq} = 1, \quad \forall k \in K \qquad \text{(1l)}$$

$$\textcolor{red}{y_{fij}^{kpqs} = 0,}$$
$$\textcolor{red}{\forall (f,i,j) \in U, k \in K, p \in P, q \in Q, s \in S,}$$
$$\textcolor{red}{g_{fij} = 1} \qquad \text{(1m)}$$

$$x_{fij}^{kpqs}, y_{fij}^{kpqs} \in \{0,1\},$$
$$\forall f \in F, (i,j,k,p) \in M, q \in Q, s \in S \qquad \text{(1n)}$$

$$a^{kpq} \in \{0,1\}, \quad \forall k \in K, p \in P, q \in Q \qquad \text{(1o)}$$

$$0 \leq O_{kq}^s \leq F, \quad \forall k \in K, s \in S, q \in Q \qquad \text{(1p)}$$

Eq. (1b) guarantees that one starting spectrum slot index exists for each $s-d$ pair and each slice component. Eq. (1c) represents that the starting spectrum slot index must be the lowest index allocated for a $s - d$ pair. Eqs. (1d)-(1e) exclude the impossible-starting-spectrum-slot-index condition. Eq. (1d) eliminates the slot indices that are impossible to be the first slot in case of a^{kpq} is one. The number of elements of $f \in \{|F| - O_{kq}^s + 2 \cdots, |F|\}$ is less than the number of utilized slots O_{kq}^s. Eq. (1e) indicates the first slot indices of the selected path, pair, and pattern. Eq. (1f) is a continuity constraint specifying that the $s - d$ pair $k \in K$ uses the same slot $f \in F$ on every link. Eq. (1g) indicates the utilized spectrum slots on each link of the $s - d$ pair $k \in K$. Eq. (1h) assures that only one utilized spectrum slot is used for only

13

one $s - d$ pair. Eq. (1i) guarantees that only one slot is usable on each link. Eq. (1j) states that the utilized slots must be in a range between x_{fij}^{kpqs} and $x_{fij}^{kpqs} + O_{kq}^s$. Eq. (1k) indicates that the maximum slot index must be equal to or less than c. This equation denotes that c is greater than each combination of the values of the variables in the left-hand side. These combinations of the values of the variables in the left-hand side of this equation is obtained by iterating over all the combinations of the subscripts and superscripts denoted by $\forall f \in F, (i, j, k, p) \in M, q \in Q, s \in S$. The left-hand side of this equation denotes the spectrum utilization for a particular request, occupying a particular slot, on a particular path that is passing through a particular link. Therefore, technically this equation denotes that c is a variable which is the maximum of all possible cases of spectrum utilization and we are minimizing that variable c. Eq. (1l) assures that only one slicing pattern and one path is selected for each pair. Eq. (1m) is a lookup constraint which disables the slot f on link (i, j) if it is occupied. It should be noted that the range of f and f' in every equation are different, which is written in each equation, depending on the condition of each constraint.

The constraints in Eqs. (1c)-(1e), and (1j)-(1k) are nonlinear in nature. We introduce binary variables $\pi_{fij}^{kpqs} = x_{fij}^{kpqs} a^{kpq}$ to linearize them and represent the linearized constraints in Eqs. (2a) to (2i).

$$\pi_{fij}^{kpqs} \leq y_{f'ij}^{kpqs},$$
$$\forall (i, j, k, p) \in M, f \in \{1, \cdots, |F| - O_{kq}^s + 1\},$$
$$f' \in \{f, \cdots, f + O_{kq}^s - 1\}, s \in S, q \in Q \tag{2a}$$
$$\pi_{fij}^{kpqs} = 0,$$
$$\forall (i, j, k, p) \in M, s \in S, q \in Q,$$
$$f \in \{|F| - O_{kq}^s + 2, \cdots, |F|\} \tag{2b}$$
$$\pi_{fij}^{kpqs} \leq r_{ij}^{kp},$$
$$\forall (i, j, k, p) \in M, s \in S, q \in Q,$$

14

$$f \in \{1, \cdots, |F| - O_{kq}^s + 1\}, O_{kq}^s \neq 0 \tag{2c}$$

$$y_{f'ij}^{kpqs} \leq \pi_{fij}^{kpqs} + (O_{kq}^s \times a^{kpq}),$$

$$\forall (i, j, k, p) \in M, f \in \{1, \cdots, |F| - O_{kq}^s + 1\},$$

$$f' \in \{f, \cdots, f + O_{kq}^s - 1\}, q \in Q, s \in S \tag{2d}$$

$$(\pi_{fij}^{kpqs} \times f) + O_{kq}^s - 1 \leq c,$$

$$\forall f \in F, (i, j, k, p) \in M, q \in Q, s \in S \tag{2e}$$

$$\pi_{fij}^{kpqs} \leq x_{fij}^{kpqs},$$

$$\forall f \in F, (i, j, k, p) \in M, q \in Q, s \in S \tag{2f}$$

$$\pi_{fij}^{kpqs} \leq a^{kpq},$$

$$\forall f \in F, (i, j, k, p) \in M, q \in Q, s \in S \tag{2g}$$

$$\pi_{fij}^{kpqs} \geq x_{fij}^{kpqs} + a^{kpq} - 1,$$

$$\forall f \in F, (i, j, k, p) \in M, q \in Q, s \in S \tag{2h}$$

$$\pi_{fij}^{kpqs} \in \{0, 1\},$$

$$\forall (i, j, k, p) \in M, f \in F, q \in Q, s \in S \tag{2i}$$

The completed problem formulation becomes as follows.

$$\min \quad \sum_{k \in K} \sum_{p \in P} \sum_{q \in Q} a^{kpq} l^{kq} + (c \times \epsilon) \tag{3a}$$

subject to:

$$\text{C1:} \quad \sum_{f \in F} x_{fij}^{kpqs} = 1, \quad \forall (i, j, k, p) \in M, q \in Q, s \in S \tag{3b}$$

$$\text{C2:} \quad \pi_{fij}^{kpqs} \leq y_{f'ij}^{kpqs},$$

$$\forall (i, j, k, p) \in M, f \in \{1, \cdots, |F| - O_{kq}^s + 1\},$$

$$f' \in \{f, \cdots, f + O_{kq}^s - 1\}, s \in S, q \in Q \tag{3c}$$

$$\text{C3:} \quad \pi_{fij}^{kpqs} = 0,$$

$$\forall (i, j, k, p) \in M, s \in S, q \in Q,$$

$$f \in \{|F| - O_{kq}^s + 2, \cdots, |F|\} \tag{3d}$$

$$\text{C4:} \quad \pi_{fij}^{kpqs} \leq r_{ij}^{kp},$$

15

$$\forall (i, j, k, p) \in M, s \in S, q \in Q,$$

$$f \in \{1, \cdots, |F| - O_{kq}^s + 1\}, \textcolor{red}{O_{kq}^s \neq 0} \tag{3e}$$

C5: $\quad y_{fij}^{kpqs} = y_{fjj'}^{kpqs},$

$$\forall (i, j, k, p), (j, j', k, p) \in M,$$

$$f \in F, q \in Q, s \in S \tag{3f}$$

C6: $\quad \displaystyle\sum_{s \in S} \sum_{f \in F} y_{fij}^{kpqs} = (T_k + (l^{kq} + 1)G)a^{kpq},$

$$\forall (i, j, k, p) \in M, q \in Q \tag{3g}$$

C7: $\quad y_{fij}^{kpqs} + y_{fij}^{k'p'qs'} \leq 1,$

$$\forall f \in F, (i, j, k, p), (i, j, k', p') \in M,$$

$$s, s' \in S, (k \neq k' \text{ or } p \neq p' \text{ or } s \neq s') \tag{3h}$$

C8: $\quad \displaystyle\sum_{k \in K} \sum_{p \in P} \sum_{q \in Q} \sum_{s \in S} y_{fij}^{kpqs} \leq 1,$

$$\forall f \in F, (i, j) \in E \tag{3i}$$

C9: $\quad y_{f'ij}^{kpqs} \leq \pi_{fij}^{kpqs} + (O_{kq}^s \times a^{kpq}),$

$$\forall (i, j, k, p) \in M, f \in \{1, \cdots, |F| - O_{kq}^s + 1\},$$

$$f' \in \{f, \cdots, f + O_{kq}^s - 1\}, q \in Q, s \in S \tag{3j}$$

C10: $\quad (\pi_{fij}^{kpqs} \times f) + O_{kq}^s - 1 \leq c,$

$$\forall f \in F, (i, j, k, p) \in M, q \in Q, s \in S \tag{3k}$$

C11: $\quad \displaystyle\sum_{p \in P} \sum_{q \in Q} a^{kpq} = 1, \quad \forall k \in K \tag{3l}$

C12: $\quad \textcolor{red}{y_{fij}^{kpqs} = 0,}$

$$\textcolor{red}{\forall (f, i, j) \in U, k \in K, p \in P, q \in Q, s \in S,}$$

$$\textcolor{red}{g_{fij} = 1} \tag{3m}$$

C13: $\quad \pi_{fij}^{kpqs} \leq x_{fij}^{kpqs},$

$$\forall f \in F, (i, j, k, p) \in M, q \in Q, s \in S \tag{3n}$$

C14: $\quad \pi_{fij}^{kpqs} \leq a^{kpq},$

$$\forall f \in F, (i, j, k, p) \in M, q \in Q, s \in S \tag{3o}$$

$$\text{C15:} \quad \pi_{fij}^{kpqs} \geq x_{fij}^{kpqs} + a^{kpq} - 1,$$

$$\forall f \in F, (i,j,k,p) \in M, q \in Q, s \in S \tag{3p}$$

$$\text{C16:} \quad x_{fij}^{kpqs}, y_{fij}^{kpqs}, \pi_{fij}^{kpqs} \in \{0,1\},$$

$$\forall (i,j,k,p) \in M, f \in F, q \in Q, s \in S \tag{3q}$$

$$\text{C17:} \quad a^{kpq} \in \{0,1\}, \quad \forall k \in K, p \in P, q \in Q \tag{3r}$$

$$0 \leq O_{kq}^{s} \leq F, \quad \forall k \in K, s \in S, q \in Q \tag{3s}$$

255 *3.3. Algorithm for spectrum allocation with slicing*

An algorithm for the spectrum allocation with slicing is introduced since the MILP problem formulation is not scalable for a large practical network requiring a large number of spectrum slots.

We consider an EON with V nodes, where each node is equipped with D
260 slicers. Each link contains maximum $|F|$ slots and $|P|$-shortest path fixed routing is considered for each request. The other notations used in the algorithm are as follows:

17

i: Slot index, where $1 \leq i \leq |F|$

p: Path index, where $1 \leq p \leq |P|$

N: The maximum number of logical sliced components

n: Logical sliced components index, where $1 \leq n \leq N$

M: The maximum number of physical sliced components

m: Physical sliced components index, where $1 \leq m \leq M$

a_n: The number of utilized slots for the n-th logical sliced component

b_m: The number of utilized slots for the m-th physical sliced component

x_n: The first slot index of the nth logical sliced component

y_m: The first slot index of the mth physical sliced component

d_v^p: The remaining number of physical slicers at node v on path p, where $1 \leq v \leq V$. Each d_v^p is set to D initially

r^p: The number of available slots that can be used on path p

T: The number of requested slots

G: The number of slots for guardband

u^p: The number of utilized slicer(s) for the request on path p

c: The maximum utilized slot index

The allocation process is separated into three phases. Phase 1 logically assigns the spectrum band of the request into available slots. In phase I, the algorithm compares the number of consecutive available slots and the number of requested slots. If the number of requested slots can fit into the consecutive available slots, the request is assigned to those available slots. Otherwise, the algorithm increases the number of portions, where each portion contains equally consecutive slots. The process repeats until consecutive slots of all portions are assigned. For example, the requested number of slots is six, phase 1 searches for six consecutive slots, which is considered as one portion, to assign. If there are no consecutive slots for six slots, phase 1 split the request by increasing the number of portions from one to two, where each portion contains three consecutive slots, and search for the location to assign. If all of two portions cannot be assign, phase 1 split the request from two to three portions, where each portion contains 2 consecutive slots. The number of portions is increased until all of

Phase 2 considers the number of slicers used for the assigned slots from phase 1. This phase checks the consecutive por-

²⁸⁰ tions. If different portions are consecutively assigned, the algorithm considers those portions as one portion so that slicers are not required for this portion. Otherwise, a slicer is needed for non-consecutive portions. Phase 3 determines the minimum of the maximum utilized slot index. This phase determines the number of required slicers for each path. The algorithm selects the path with

²⁸⁵ the minimum number of required slicers. Initially, p is set to zero and c is set to inf. The procedure of the allocation process for each request is described as in the pseudo code.

Phase 1: Logical assignment

1 $N \leftarrow 1$;

2 while <u>request is not logically assigned</u> **do**

3 $n \leftarrow 1$;

4   **if** <u>$N - 1 \leq d_v^p$ AND $N \leq T$</u> **then**

5   **if** <u>$T + (N \times G) \leq r^p$</u> **then**

6   **if** <u>$n < N - (T \bmod N) + 1$</u> **then**

7 $a_n = \lfloor \frac{T}{N} \rfloor + G$;

8   **else**

9 $a_n = \lfloor \frac{T}{N} \rfloor + G + 1$;

10   **end**

11   **else**

12   Reject request

13   **end**

14   **for** <u>$i = 1$ to $|F| - T + 1$</u> **do**

15   **if** <u>consecutive slots from i to $i + a_n - 1$ are available</u> **then**

16 $x_n \leftarrow i$;

17 $n{+}{+}$;

18   **else**

19   break;

20   **end**

21   **end**

22   **else**

23   Reject request

24   **end**

25 $N{+}{+}$;

26 end

Phase 2: Physical assignment

---

1 $m \leftarrow 1$;

2 **for** $\underline{n = 1 \text{ to } N - 1}$ **do**

3 $b_m \leftarrow 0$;

4 $y_m \leftarrow x_n$;

5   **while** $\underline{x_n = y_m + b_m}$ **do**

6 $b_m = b_m + a_n$;

7 $n{+}{+}$;

8   **end**

9 $y_m \leftarrow x_n$;

10 $b_m \leftarrow a_n$;

11 $m{+}{+}$;

12 **end**

13 $d_v^p = d_v^p + m - 1$;

14 $u^p = m - 1$;

Phase 3: Determine the minimum of the maximum utilized slot index

---

1 **if** $\underline{p < 0 \text{ OR } u^p \geq u^{p-1}}$ **then**

2   Repeat from phase 1

3 **else**

4   **if** $\underline{b_m < c}$ **then**

5 $c \leftarrow b_m$;

6   **else**

7   **if** $\underline{p < |P|}$ **then**

8 $p{+}{+}$;

9   Repeat from phase 1

10   **end**

11   **end**

12 **end**

290

21

After finishing the assignment in phase 3, the request is allocated into spectrum slots as the result from phase 3. The total time complexity of the algorithm is $\mathcal{O}(|P|(T + \lceil \frac{T}{N} \rceil))$. The number of slicing is controlled in the algorithm (lines 3-13 of phase 1). If we have n slices, we try to accommodate all the demand using these slices. Only if the n slices are not enough, we go for $n + 1$ slices. In this way, minimum number of slicers is guaranteed.

Figure 4 illustrates how the allocation process in conjunction with spectrum slicing functions. Let us assume there is a request to be carried over a lightpath established from node A to node D via nodes B and C and requires 5 spectrum slots, as shown in Fig. 4(a). In Fig. 4(b) we illustrate what happens if the number of slicing increases gradually. First, we start with the assumption that there is no slicing. This means that the 5-slots must be contiguous from node A to node D. A quick glance on the available spectrum slots shows that there is no available spectrum chunk that can be formed from 5 contiguous available slots. Therefore, spectrum slicing is certainly required.

So, we start with one slicing. The spectrum band of the request is broken into two portions, and let us say, one portion is a 2-slots slice and the other is 3-slots slice. Although a 4-slots chunk from slots 6 to 9 are available, this space is able to support only either the 2-slots slice or the 3-slots slice and the other slice cannot be supported at all. Therefore, applying one slicing to split a 5-slots request into 3-slots and 2-slots is not feasible for this particular scenario. At this point, we might try slicing the already sliced bands into another slice resulting in three portions of 1 slot, 2 slots, and 2 slots. The 1-slot slice can be assigned to the available slot 1, the first 2-slots slice can be assigned to slots 6 and 7, and the last slice can be assigned to slots 8 and 9. This eventually ends up in having the two 2-slots portions in contiguous slots from slot 6 to 9. Therefore, in the physical slicing, assigning a 1-slot slice at slot 1 and a 4-slots (2-slots + 2-slots) slice at slots 6 to 9 can allocate the total spectrum band of the request successfully. As a result, only one slicing to split the 5-slots request into 4-slots and 1-slot is enough in this example. In the physical assignment phase of the algorithm the slicing will be adjusted to make the number of slices

22

optimum as discussed in the above example.

4. Performance Evaluation

Performances of EONs with slicing are evaluated in this section. Bandwidth blocking ratio (BBR), which is a defined as the ratio of the number of rejected bandwidth to the total requested bandwidth, is used as an indicator to show the effect of the proposed scheme. We firstly compare the performance of the proposed scheme by using the problem formulation in Section 3.2 and the algorithm in Section 3.3 for a 6-node network topology in Fig. 5 when the number of paths for each source and destination pair is one, $|P| = 1$. The following assumptions are used for the simulation. Each link has maximum 50 spectrum slots. The bandwidth of a spectrum slot is set to 12.5 GHz. Two slots are used as a guardband, G. It should be noted that every node in this network is considered as both edge node and intermediate node, depending on a function of request processing. Ingress and egress nodes for a request are considered as the edge nodes. Nodes that the request passes through are considered as intermediate nodes.

Figure 6 shows an example of a result to confirm the correctness of the MILP problem formulation. The paths are determined by k-shortest path. Two slots are set as guardband. Initially, we assume that slots {3,4,5}, {6,8,9}, and {3,5,7,12} are occupied on links (1,2), (2,4), and (3,4), respectively. The maximum number of paths is set to two. There are three requests, req. 1 to 3. Req. 1 requests five slots from node 1 to node 4. Path 1 is $1 \rightarrow 2 \rightarrow 4$. Path 2 is $1 \rightarrow 3 \rightarrow 4$. Req. 2 requests four slots from node 2 to node 5. Path 1 is $2 \rightarrow 4 \rightarrow 5$. Path 2 is $2 \rightarrow 4 \rightarrow 6 \rightarrow 5$. Req. 3 requests four slots from node 3 to node 4. Path 1 is $3 \rightarrow 4$. Path 2 is $3 \rightarrow 1 \rightarrow 2 \rightarrow 4$. MILP outputs a result as follows. Req. 1 selects path 2. One slicer is used for this request. The first component uses slots 8 and 9 with slots 10 and 11 as guardband. The second component uses slots 13 to 15 with slots 16 and 17 as guardband. Req. 2 selects path 2. One slicer is used for this request. The first component uses slots 1 to

23

3 with slots 4 and 5 as guardband. The second component uses slot 10 with slots 11 and 12 as guardband. Req. 3 selects path 2. No slicer is used for this request. Slots 13 to 14 with slots 15 and 16 as guardband are used to allocate the request.

The requests are generated randomly based on a Poisson process with λ arrival rate. The holding time of the requests follows an exponential distribution with an average time of $\mu = 10$ units. The routing is fixed and is determined by using Dijkstras algorithm. The traffic load (ρ) in the network is given in Erlangs, where $\rho = \lambda \times H$. The number of requested spectrum slots for a request is assumed to be uniformly distributed between 1 and 6. We generate 100,000 requests randomly for the simulation. The maximum number of slicers per node, D takes values from zero to three with $D = 0$ referring to the conventional scheme, where no slicing is considered. It should be noted that inter-symbol interference (ISI) occurs because of non-ideal filtering in both spectrum filtering and slice selection. Two sliced spectrum components may have a partially overlapped spectrum. But the effect of ISI is omitted in this simulation and can be taken up as part of a future research study. The performance of the proposed scheme for both the MILP problem formulation and the heuristic is evaluated on a server with Intel(R) Core(TM) i7-2600K CPU@3.40GHz, 32GB memory. We used CPLEX version 12.9.0.0 to solve the MILP problem formulation.

Figure 7, shows the comparison of BBR performance as obtained from the MILP problem formulation and the heuristic algorithm for the 6-node topology in Fig. 5. The number of slicers per node is set to $D = 3$ and the guardband is set to $G = 2$. It is observed that both the MILP and the heuristic performs identically with respect to BBR when $|P| = 1$ and the MILP performs better than the heuristic when $|P| = 2$ with low traffic load.

Next, in Figs. 8 and 9, we present the comparison of the computational times for the MILP and the heuristic for the 6-node topology for different values of $|P|$, i.e., the number of shortest paths as a parameter. The number of slicers per node is set to $D = 3$ and the guardband is set to $G = 2$. From these two graphs, few interesting observations can be noted. Firstly, in case of the total

24

computation times, the MILP runs for much longer times, e.g., the order of a few 100 thousand seconds as compared to the heuristic which runs for a few hundred seconds. Secondly, for both cases, the computation time is higher for higher value of $|P|$. This is because with higher value of $|P|$ the solution space increases for both the MILP and the heuristic and it takes more time to iterate over all possible feasible solutions.

Furthermore, it is also interesting to note that the computation time for the MILP actually shows a decreasing trend with increasing traffic. This can be explained as follows. When the traffic increases, so does the number of requests. The CPLEX solver removes all those requests that are not feasible to be accommodated and prunes the solution space to only those requests that are feasible and then tries to find the global optimum allocation. Therefore, with higher traffic more requests are blocked and the solver has to work with less computationally intensive solution space exhibiting a lower computation time for higher traffic.

However, in case of the heuristic, it works with all the requests in an iterative manner and keeps a track of the requests that are blocked, calculating BBR at the end once it completes looking up all the requests. Hence, in case of the heuristic's computational time in Fig. 9, a more conventional upward trend is exhibited with increase in traffic.

Next, in Fig. 10 we illustrate the effect of the slicing with traffic load. The x-axis of Fig. 10 represents the varying traffic load, the y-axis represents how many times a request has been sliced (note that this number is capped to 3 as $D = 3$ for this plot), and the z-axis represents the percentage of requests that are sliced. Note that there are 100,000 requests generated for all traffic loads. So with higher traffic loads more number of requests out of these 100,000 requests are likely to be unfit for allocation if they are not sliced. So, with increasing traffic the relative percentage of requests that require 3 times slicing as compared to 1 time slicing or 2 times slicing are increasing. The general trend of the percentage of sliced requests is also upwards with traffic, for each of the three scenarios (i.e., 1-time slicing or 2-times slicing or 3-times slicing).

25

So, from Fig. 10, it is evident, as expected, that with increasing traffic, not only more requests are sliced but also each request is sliced more number of times.

In Figs. 11 and 12 we represent the sensitivity of BBR for the 6-node topology with number of slicers per node and the number of shortest paths allowed per connection. It is observed that for both the MILP and heuristic the results almost follow each other. Furthermore, there is a downward trend in BBR with increasing slicers and increasing paths. This is again intuitive as more slicers or more paths always increase the option of successfully setting up a request and hence reducing BBR. Also it is evident that the choice of guardband also has a significant role to play in terms of BBR.

Next, in Figs. 13 and 14 we study the effect of the number of shortest paths in the computation time for both the MILP and the heuristic. As the number of shortest paths increases, both the MILP and the heuristic show an upward trend in the computation time, because more number of paths make the solution space large and hence it takes longer time to span the entire solution space to establish all the requests. However, as observed in Figs. 8 and 9, an interesting observation can be made in Figs. 13 and 14 too in terms of the dependence of the computation time with traffic for both the MILP and the heuristic. Due to the same reasons discussed previously for Figs. 8 and 9, we can see, that the computation times are lower for higher traffic in case of the MILP whereas the trend is just opposite in case of the heuristic from Figs. 13 and 14.

Figure 15 reports BBR for different guardband slots for the 6-node topology and again as intuitive, BBR increases with increased number of guardband slots. This is because more guardband slots mean less available slots for the actual data resulting in more blocking on the average.

Next, we evaluate the performance of the proposed scheme in larger networks by using the algorithm in Section 3.3 for the COST239 and NSFNET networks, as shown in Fig. 16. The simulation environment is the same as the simulation using the problem formulation, except the number of maximum spectrum slots for each link is set to 400 slots and the number of requested spectrum slots for a request is uniformly distributed between 1 and 16. It should be noted that

26

every node in these networks is considered as both edge node and intermediate node, depending on a function of request processing. Ingress and egress nodes for a request are considered as the edge nodes. Nodes that the request passes through are considered as intermediate nodes.

Figure 17 reports BBR with traffic load for the COST239 topology. The different parameters are the number of slicers D and the number of shortest paths $|P|$. The usual trend of increase in BBR with traffic is exhibited. In addition, we can notice that BBR is also sensitive to value of D. There is an order of magnitude improvement in BBR with slicing compared to without slicing for 300 Erl of traffic. BBR is also highly sensitive to the number of shortest paths which is also expected.

In Fig. 18 we report the sensitivity of BBR with increasing number of slicers. It shows while a jump from no slicers to 1, 2, and 3 slicers bring down BBR but after that there is not much difference. This is also an important find because the slicers, as discussed before, employ complex optical technologies and are expensive. So, how many of them are just enough to improve BBR performance is also worth finding out and Fig. 18 provides us with that information.

Figure 19 presents BBR of the COST239 topology with the number of shortest paths and the fall in BBR is quite sharp here as compared to the 6-node topology case reported in Fig. 12. This is because COST239 is a highly meshed topology with more number of alternative routes for a particular request than in the case of the 6-node topology for particular $|P|$ value. Therefore, the number of blocked bandwidth is reduced more sharply with $|P|$ in COST239 topology. The sensitivity of BBR with the number of slicers D is also evident from Fig. 19.

Next, in Fig. 20, we illustrate the dependence of BBR with the number of guardband slots for the COST239 topology and similar to Fig. 15 for the 6-node topology, we see an upward trend in BBR with number of guardband slots and reason behind this is also similar as we discussed for Fig. 15. Here also, the sensitivity of BBR with the number of slicers D is evident.

In Figs. 21 - 24, we present similar analysis for the NSFNET topology. We

27

can observe that similar to the COST239 topology, these graphs also exhibit the similar interplay of BBR with other parameter like traffic, number of slicers (D), guardband (G), and number of shortest paths ($|P|$).

5. Conclusion

In this paper, we proposed routing and spectrum assignment scheme for elastic optical networks (EONs) employing the novel spectrum slicing and stitching technology. By developing an mixed-integer linear programming (MILP) problem formulation and a heuristic algorithm that emulates the performance of the MILP we analyzed the performance of an EON in terms of bandwidth blocking rate in presence of slicing. Other usual parameters like the number of shortest paths and the amount of guardbands were also considered in our analysis. Our results showed that the heuristic algorithm closely follows the MILP performance for a 6-node topology. Furthermore, our results also showed the sensitivity of BBR with the amount of slicing that can be provided. We also captured the subtle interplay of BBR with the amount of slicing and the other EON parameters like the guardband and the number of shortest paths. Furthermore, we also showed that there is a threshold amount of slicers beyond which the blocking performance does not improve much. Last but not the least, we also showed the consistency of our model by applying it to different practical network topologies and discovering similar trends in our results.

References

[1] Recommendation ITU-T G.694.1, Spectral grids for WDM applications: DWDM frequency grid, 2012.

[2] M. Jinno, H. Takara, B. Kozicki, Y. Tsukishima, Y. Sone, S. Matsuoka, Spectrum-efficient and scalable elastic optical path network: architecture, benefits, and enabling technologies, IEEE Commun. Mag. 47 (11) (2009) 66-73.

28

[3] O. Gerstel, M. Jinno, A. Lord, S.J.B. Yoo, Elastic optical networking: a new dawn for the optical layer?, IEEE Commun. Mag. 50 (2) (2012) s12-s20.

[4] P. Lu, L. Zhang, X. Liu, J. Yao, Z. Zhu, Highly efficient data migration and backup for big data applications in elastic optical inter-data-center networks, IEEE Netw. 29 (5) (2015) 36-42.

[5] B.C. Chatterjee, S. Ba, E. Oki, Fragmentation Problems and Management Approaches in Elastic Optical Networks: A Survey, IEEE Commun. Surv. Tutor. 20 (1) (2017) 183-210.

[6] R. Wang, B. Mukherjee, Provisioning in elastic optical networks with non-disruptive defragmentation, J. Light. Technol., 31 (15) (2013) 2491-2500.

[7] F. Cugini, F. Paolucci, G. Meloni, G. Berrettini, M. Secondini, F. Fresi, N. Sambo, L. Poti, P. Castoldi, Push-pull defragmentation without traffic disruption in flexible grid optical networks, IEEE/OSA J. Lightw. Technol. 31 (1) (2013) 125-133.

[8] S. Ba, B.C. Chatterjee, S. Okamoto, N. Yamanaka, A. Fumagalli, E. Oki, Route Partitioning Scheme for Elastic Optical Networks With Hitless Defragmentation, IEEE/OSA J. Opt. Commun. Netw. 8 (6) (2016) 356-370.

[9] R. Proietti, C. Qin, B. Guan, Y. Yin, R.P. Scott, R. Yu, S. Yoo, Rapid and complete hitless defragmentation method using a coherent RX LO with fast wavelength tracking in elastic optical networks, Opt. Exp. 20 (24) (2012) 26958-26968.

[10] T. Takagi, H. Hasegawa, K. Sato, Y. Sone, A. Hirano, M. Jinno, Disruption minimized spectrum defragmentation in elastic optical path networks that adopt distance adaptive modulation, in: Proceedings of the ECOC 2011, September 2011.

[11] Y. Yin, H. Zhang, Mi. Zhang, M. Xia, Z. Zhu, S. Dahlfort, S. J. B. Yoo, Spectral and spatial 2D fragmentation-aware routing and spectrum assign-

ment algorithms in elastic optical networks, J. Optic. Commun. Netw., 5
(10) (2013) A100-A106.

[12] L. Gong, X. Zhou, X. Liu, W. Zhao, W. Lu, Z. Zhu, Efficient resource allocation for all-Optical multicasting over spectrum-sliced elastic optical networks, J. Optic. Commun. Netw., 5 (8) (2013), 836-847.

[13] L. Gong, Z. Zhu, Virtual optical network embedding (VONE) over elastic optical networks, J. Light. Technol., 32 (3) (2014), 450-460.

[14] W. Fadini, E. Oki, A subcarrier-slot partition scheme for wavelength assignment in elastic optical networks, in: Proceedings of the HPSR 2014, July 2014.

[15] B.C. Chatterjee, E. Oki, Lightpath threshold adaptation algorithm for dispersion-adaptive first-last fit spectrum allocation scheme in elastic optical networks, in: Proceedings of the ICTON'16, July 2016.

[16] Y. Cao, A. Almaiman, M. Ziyadi, A. Mohajerin-Ariaei, C. Bao, P. Liao, F. Alishahi, A. Fallahpour, Y. Akasaka, C. Langrock, M.M. Fejer, J.D. Touch, M. Tur, A.E. Willner, Reconfigurable Channel Slicing and Stitching for an Optical Signal to Enable Fragmented Bandwidth Allocation Using Nonlinear Wave Mixing and an Optical Frequency Comb, J. Light. Technol. 36 (2) (2018) 440-446.

[17] Y. Cao, A. Almaiman, M. Ziyadi, A. Mohajerin-Ariaei, C. Bao, P. Liao, F. Alishahi, A. Falahpour, Y. Akasaka, C. Langrock, M. Fejer, J. Touch, M. Tur, A.E. Willner, Experimental demonstration of tunable optical channel slicing and stitching to enable dynamic bandwidth allocation, in: Proceedings of the OFC'17, June 2017.

[18] N. Kitsuwan, R. Matsuura, Performance of Elastic Optical Network with Spectrum Slicing for Fragmented Bandwidth Allocation, in: Proceedings of the ICNC 2019, February, 2019.

30

[19] F. Yousefi, A. G. Rahbar, Novel fragmentation-aware algorithms for multipath routing and spectrum assignment in elastic optical networks-space division multiplexing (EON-SDM), Optic. Fiber Techn., 46 (2018) 287-296.

[20] N. Kadu, S. Shakya, X. Cao, Modulation-aware multipath routing and spectrum allocation in elastic optical networks, in: Proceedings of the ANTS 2014, December 2014.

[21] D. Din, Y. Wu, B. Guo, C. Chen, P. Wu, Spectrum Expansion/Contraction Problem for Multipath Routing with Time-Varying Traffic on Elastic Optical Networks, in: Proceedings of the ICC'17, March 2017.

[22] Z. Zhu, W. Lu, L. Zhang, N. Ansari, Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing, J. Light. Technol. 31 (1) (2013) 15-22.

[23] R. Wang, B. Mukherjee, Spectrum management in heterogeneous bandwidth optical networks, Opt. Switch. Netw. 11 (A) (2014) 83-91.

[24] B. C. Chatterjee, W. Fadini, E. Oki, A spectrum allocation scheme based on first-last-exact fit policy for elastic optical networks, J. Netw. Comput. Appl., 68 (2016) 164-172.

[25] B. C. Chatterjee, N. Kitsuwan, E. Oki, Performance evaluation of first-last-exact fit spectrum allocation policy for elastic optical networks, in: Proceedings of the ICTON 2017, July 2017.

31
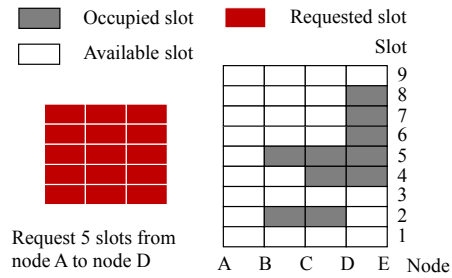
(a) Request and available slots
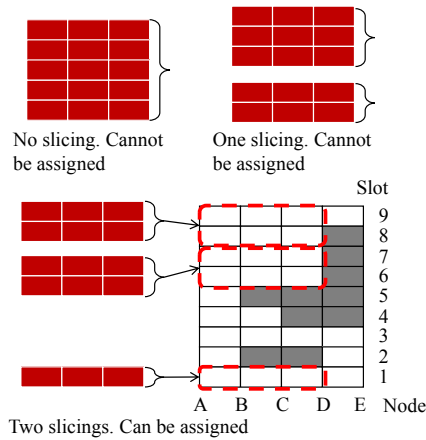
(b) Logical slicing
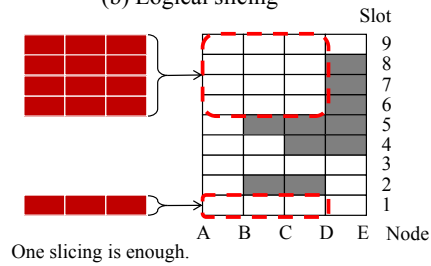
(c) Physical slicing

Figure 4: Example of request allocation with spectrum slicing.
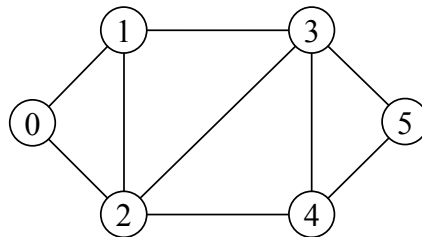
Figure 5: 6-node topology.
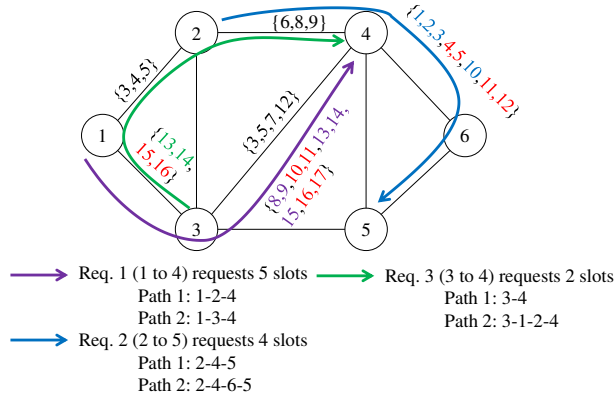
Figure 6: Example of result for MILP.
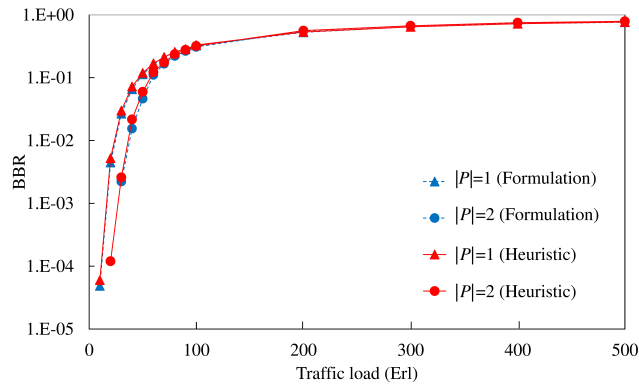
Figure 7: BBR for different traffic in 6-node topology ($G = 2$, $D = 3$).
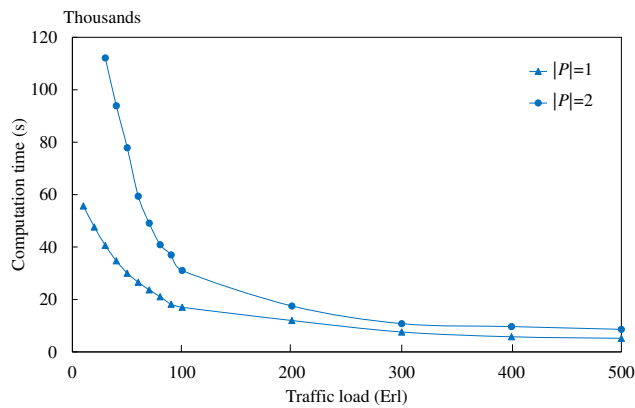
Figure 8: Computational time of the MILP problem formulation for different traffic in 6-node topology ($G = 2$, $D = 3$).
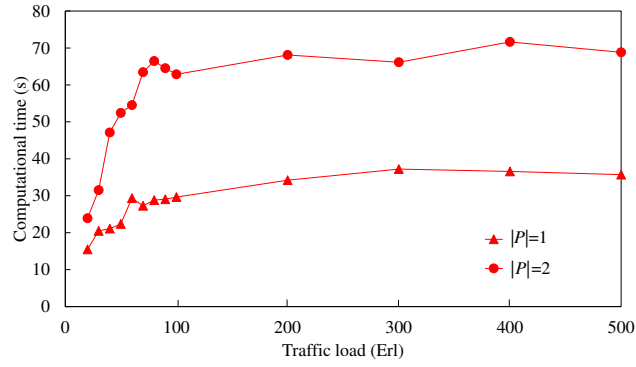
Figure 9: Computational time of heuristic algorithm for different traffic in 6-node topology ($G = 2$, $D = 3$).
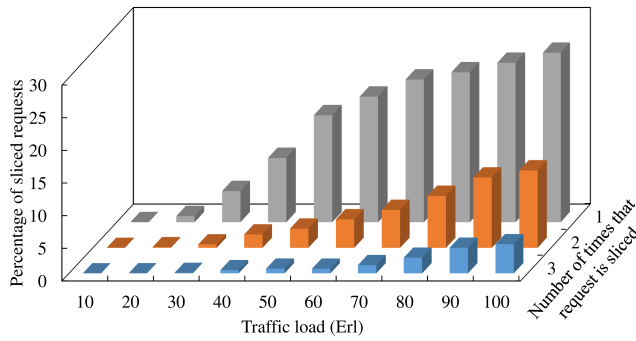
Figure 10: Effect of slicing for different traffic load in 6-node topology ($G = 2$, $D = 3$, $|P| = 1$) using the MILP.
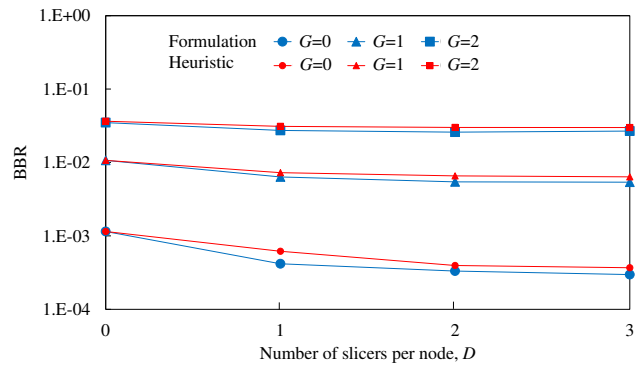
Figure 11: BBR for different number of slicers per node in 6-node topology (traffic load = 30 Erl, $|P| = 1$).
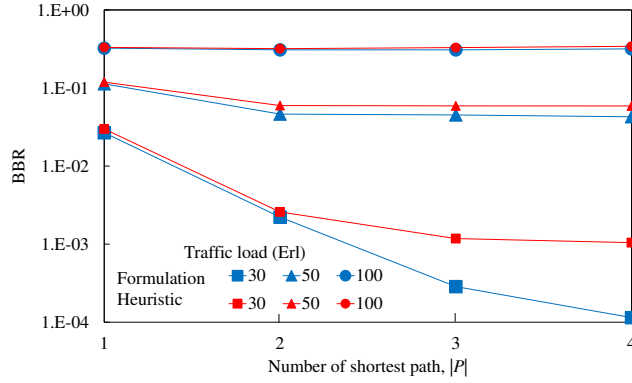
Figure 12: BBR for different number of shortest paths in 6-node topology ($G = 2$, $D = 3$).
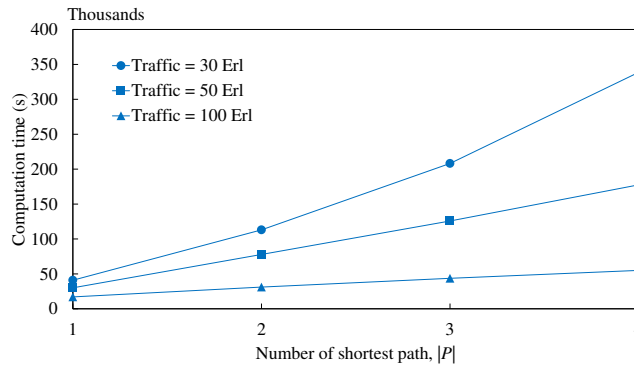
Figure 13: Computational time of problem formulation for different number of shortest paths in 6-node topology ($G = 2$, $D = 3$).
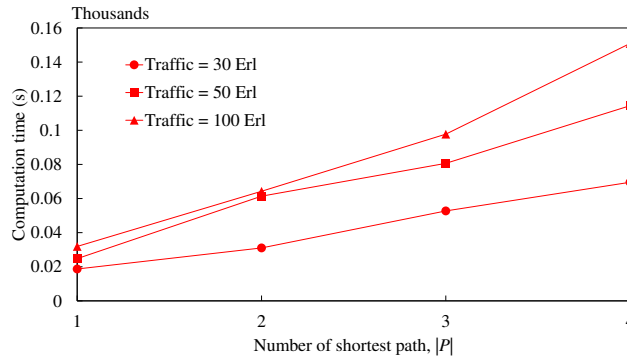
Figure 14: Computational time of heuristic algorithm for different number of shortest paths in 6-node topology ($G = 2$, $D = 3$).
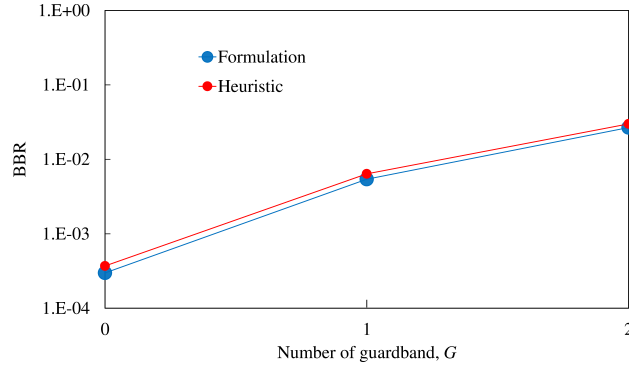
Figure 15: BBR for different number of guardband slots in 6-node topology (traffic load = 30 Erl, $|P| = 1$, $D = 3$).
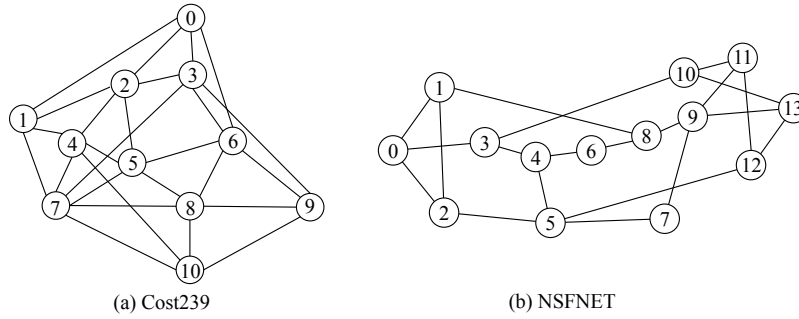
(a) Cost239

(b) NSFNET

Figure 16: COST239 and NSFNET topologies.
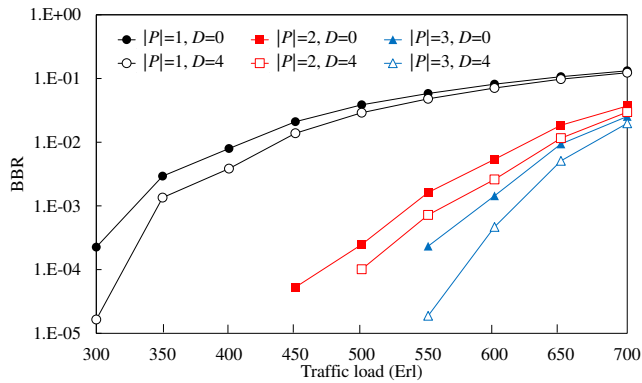
Figure 17: BBR for different traffic in COST239 topology ($G = 2$).
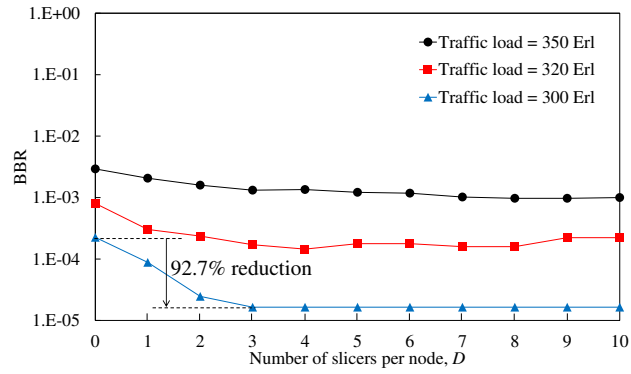
Figure 18: BBR for different number of slicers per node in COST239 topology ($|P| = 1$, $G = 2$).
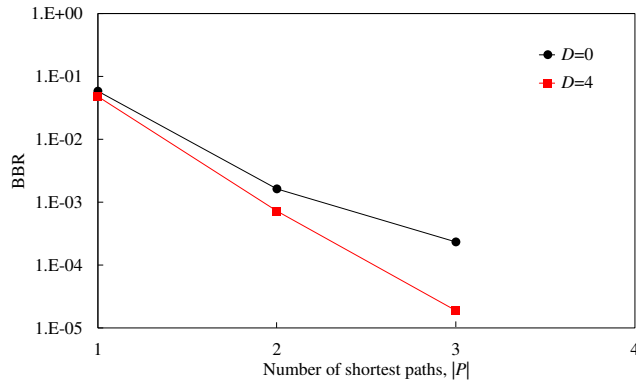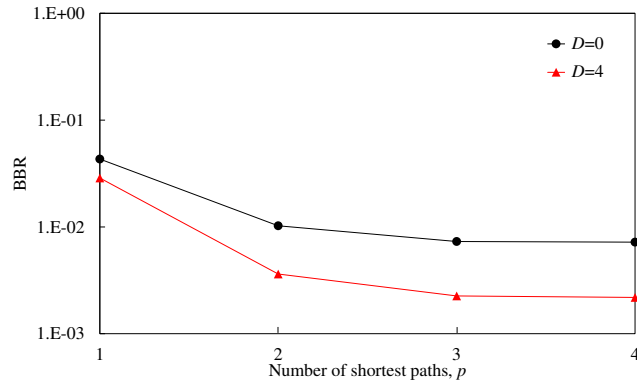

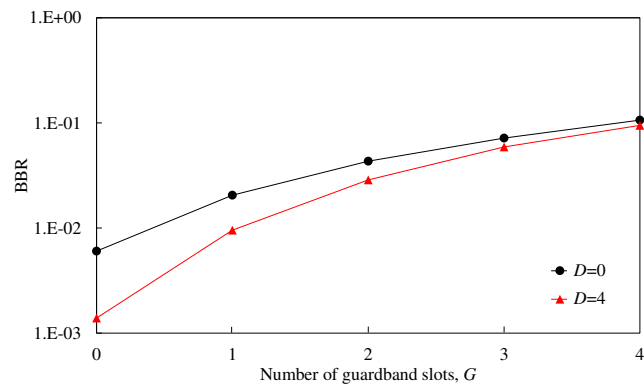Figure 19: BBR for different number of shortest paths in COST239 topology (traffic load = 550 Erl, $G = 2$).
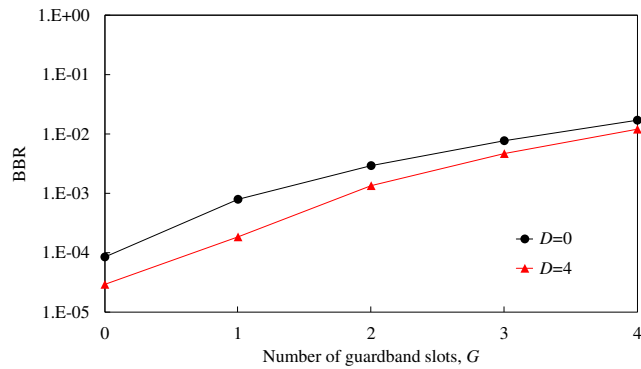
Figure 20: BBR for different number of guardband slots in COST239 topology (traffic load = 350 Erl, $|P| = 1$).
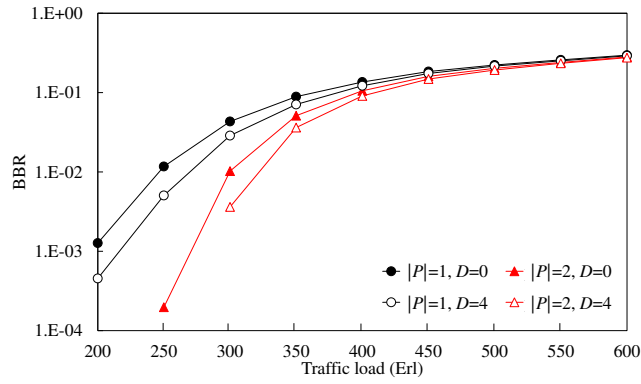
Figure 21: BBR for different traffic in NSFNET topology ($G = 2$).
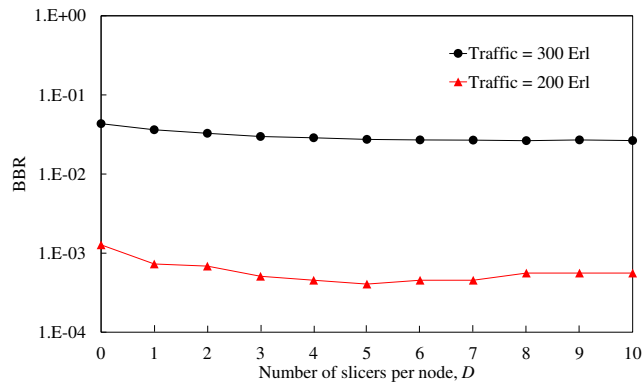
Figure 22: BBR for different number of slicers per node in NSFNET topology ($p = 1$, $G = 2$).

Figure 23: BBR for different number of shortest paths in NSFNET topology (traffic load = 300 Erl, $G = 2$).

Figure 24: BBR for different number of guardband slots in NSFNET topology (traffic load = 300 Erl, $p = 1$).

Conflicts of Interest Statement

Manuscript title: Elastic Optical Network with Spectrum Slicing for Fragmented Bandwidth Allocation

The authors whose names are listed immediately below certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

Author names:

Nattapong Kitsuwan

Praphan Pavarangkoon

Avishek Nag

The authors whose names are listed immediately below report the following details of affiliation or involvement in an organization or entity with a financial or non-financial interest in the subject matter or materials discussed in this manuscript. Please specify the nature of the conflict on a separate sheet of paper if the space below is inadequate.
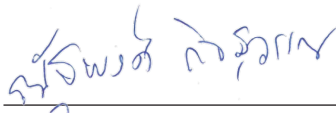
Author names:

This statement is signed by all the authors to indicate agreement that the above information is true and correct (*a photocopy of this form may be used if there are more than 10 authors*):
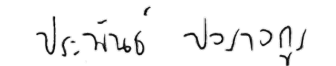
Author's name (typed)	Author's signature	Date
Nattapong Kitsuwan		2020/02/28
Praphan Pavarangkoon		2020/02/28
Avishek Nag		2020/03/02