This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2021.3064643, IEEE Journal on Selected Areas in Communications

Disruption Minimized Bandwidth Scaling in EON-enabled Transport Network Slices

Nashid Shahriar, Mubeen Zulfiqar, Shihabur Rahman Chowdhury, *Member, IEEE*, Sepehr Taeb, Raouf Boutaba, *Fellow, IEEE*, Jeebak Mitra, and Mahdi Hemmati *Member, IEEE*

***Abstract*—Elastic Optical Networks (EONs) enable finer-grained resource allocation and tuning of transmission configurations for right-sized resource allocation. These features make EONs excellent choice for 5G transport networks supporting highly dynamic traffic with diverse Quality-of-Service (QoS) requirements. 5G network slices are expected to host applications with a dynamic nature (*e.g.*, augmented/virtual reality broadcasting), which will result in slice resource requirement changing over time. The initial resource allocation to network slices has to be adapted to accommodate such changes without causing significant disruption to existing traffic and using minimal additional resources. In this paper, we address the problem of scaling bandwidth demand of network slices on an EON-enabled 5G transport network. In contrast to the state-of-the-art, we do not assume any specific technologies for minimizing disruption when accommodating the scaling request. Rather, we propose an Integer Linear Program (ILP) and a heuristic algorithm for accommodating scaling requests by choosing from a comprehensive set of reconfiguration actions. We carefully design a novel cost model for capturing traffic disruptions and additional resource usage by these different actions. Our extensive simulations using realistic network topologies shed light on the trade-off between additional resource usage and disruption while accommodating slice scaling requests by employing a comprehensive set of reconfiguration actions. Simulation results also show that our heuristic algorithm can find solutions that remain within 10% of ILP-based solutions, while executing several orders of magnitude faster than ILP.**

***Index Terms*—Elastic Optical Network, Network Virtualization, Slice Scaling, Transport Network**

I. Introduction

Elastic Optical Network (EON) virtualization is gaining significant traction in recent times, especially because of its importance in the upcoming 5G transport network slicing [2], [3]. A transport network slice is often represented as a virtual network (VN) that connects virtual nodes through virtual links [4]. These VNs are typically requested by service providers (SPs) or business verticals with Quality-of-Service (QoS) requirements such as bandwidth, latency, and reliability of virtual links or virtual paths [5], [6]. EONs allow fine-grained spectrum slice allocation and tuning of transmission configurations such as modulation format, forward error correction (FEC) overhead, and baudrate to rightsize resource allocation to these VNs or network slices [3], [7]. For instance, 5G transport networks enabled by EONs can leverage flexible resource allocation for making best use of network resources while establishing network slices for services such as enhanced mobile broadband and ultra-reliable and low latency communications [8].

Applications hosted on a VN or network slice[1] can evolve over time due to changes in number of users, traffic volume, and communication pattern [9]–[11]. For instance, an augmented/virtual reality broadcast application [12] will require more bandwidth with increasing number of viewers over time. Therefore, a network slice, initially provisioned to satisfy an SP's QoS requirements, may become inadequate to accommodate the dynamics of an application. As a result, the infrastructure provider (InP) will need to adapt the resources for accommodating the SP's requests for scaling their slices. When accommodating such requests for scaling network slices, the InP must ensure that the other SPs' traffic is not disrupted. In addition, the InP should strive to minimize (i) disruption to the requesting SP's existing traffic and (ii) use of any additional resources such as transponders or spectrum slices.

Network slice scaling involves scaling both the virtual links' bandwidth and the virtual nodes' computation capacity [13]. Virtual nodes of a network slice typically have location constraints and host the network functions of a slice. These nodes are usually mapped to an EON node equipped with compute resources located at a metro data center (MDC) or a Point-of-Presence (PoP) site, which in turn are connected by the EON-enabled transport network. EONs in this way form the backbone of telecommunications infrastructure and are primarily concerned with bandwidth resources. In this paper, our focus is on scaling the bandwidth resources of an EON-based transport network slice under the assumption that EON nodes have sufficient resources to accommodate the scaling request. Network slice scaling with joint consideration for virtual node resources at the MDCs/PoPs and virtual link bandwidth on the transport network merits a separate investigation that we plan to pursue in the future. We also

This work was supported in part by Huawei Canada and in part by an NSERC Collaborative Research & Development (CRD) Grant. This work benefited from the use of CrySP RIPPLE Facility at University of Waterloo. Part of this work was presented at the Optical Fiber Communication Conference (OFC) 2020 [1]. (Corresponding author: Raouf Boutaba)

Nashid Shahriar (email: nashid.shahriar@uregina.ca)) is with Department of Computer Science, University of Regina, SK S4S 0A2, Canada.

Mubeen Zulfiqar (email: mubeen.zulfiqar@uwaterloo.ca), Shihabur Rahman Chowdhury (email: sr2chowdhury@uwaterloo.ca), Sepehr Taeb (email: staeb@uwaterloo.ca), and Raouf Boutaba (email: rboutaba@uwaterloo.ca) are with David R. Cheriton School of Computer Science, University of Waterloo, ON N2L 3G1, Canada.

Jeebak Mitra (email: jeebak.mitra@huawei.com) and Mahdi Hemmati (email: mahdi.hemmati@huawei.com) are with with Huawei Technologies Canada Research Center, Ottawa ON K2K 3J1, Canada.

[1]We use the terms *network slice* and *VN* interchangeably in the remainder of this paper

consider an EON that is capable of splitting a virtual links' bandwidth over multiple lightpaths called *splits*[2] hereafter [8], [14]–[16]. Such splitting may become unavoidable for deploying high-bandwidth virtual links in two scenarios [17]: i) when there are not enough contiguous spectrum slots to serve the demand due to the spectrum fragmentation; ii) when a particular transmission configuration needed to transmit the desired bit-rate of the virtual link becomes infeasible due to physical layer impairments that limit the transmission reach. To circumvent these two scenarios, a virtual link demand can be split into sub-demands and mapped over multiple lightpaths to take advantage of the spectral gaps [8], [17].

A request for scaling up virtual link bandwidth of an already embedded VN on an EON can be accommodated by employing different actions, *e.g.*, modifying the transmission configuration (*e.g.*, modulation format) of the current embedding without changing spectrum allocation, expanding the current spectrum allocation in one or both directions, among others. In the presence of virtual link demand splitting, several other reconfiguration actions can become possible to apply. For instance, new splits can be created with their own transmission configuration and embedding paths. These techniques exhibit different levels of disruption to existing traffic as well as require different amounts of additional resources [18]. For instance, expanding a split's spectrum allocation may require no additional transponder, however, it has a higher degree of disruption on existing traffic. In contrast, creating a new split to accommodate a demand increase causes a lower degree of disruption to existing traffic thanks to the *make-before-break* (MBB) approach [19]. However, creating a new split requires a pair of additional transponders as well as additional spectrum. Therefore, a key challenge here is to strike a balance between these aspects, *i.e.*, disruption to existing traffic versus additional resource footprint incurred by the InP.

In this paper, we address the scaling of bandwidth resources of an EON-based transport network slice while minimizing the resource consumption and disruption to existing traffic. We distinctly differ from the state-of-the-art in terms of how we address disruption to existing traffic while accommodating a scaling request. The current literature on lightpath reconfiguration for accommodating time-varying traffic assumes the existence of specific technologies that cause minimal traffic disruption, *e.g.*, push-pull retuning [20], [21] or others mentioned in [10], [22], [23]. However, many of these techniques have not yet made it to production transponders (*e.g.*, push-pull retuning [21] and hop-retuning [23]) or require a completely new transponder design [22]. In contrast, we do not assume the existence of any specialized technologies. Rather, we capture the disruption caused by different actions by carefully designing a cost function that gives adaptive weights to spectrum slots[3] based on their initial allocation and the actions being applied to them.

To the best of our knowledge, our work is the first that captures different levels of disruptions caused by different types of re-configuration actions and assigns appropriate costs to each of the actions. We use the proposed cost function to devise a multi-objective Integer Linear Program (ILP) formulation for solving the problem. Our multi-objective ILP formulation provides a simple way to compute the overall disruptions of different possible solutions to the virtual link scaling problem and tune the trade-off between disruption and additional resource footprint. The ILP formulation exploits k shortest paths between pairs of EON nodes to approximate the optimal solution of the bandwidth scaling problem. Although the ILP formulation with a finite value of k cannot give us the guarantee of optimality, k shortest paths based approaches have been extensively used in the literature that were shown to achieve near-optimal solutions for routing and spectrum allocation problems in EONs [16]. We also propose a heuristic solution for solving larger problem instances within a reasonable time. We empirically demonstrate that prioritizing disruption minimization can save more than 30% spectrum slots from being affected by disruptive actions compared to prioritizing the minimization of additional resource footprint.

This paper extends our preliminary work presented in [1] on several aspects. First, we present a more elaborate discussion on the trade-off between disruption and additional resource usage of different actions to accommodate requests for scaling slice bandwidth demand in Section III. We also significantly expand the discussion on disruption cost modeling and present a full-fledged ILP formulation for solving the problem. Furthermore, we introduce a scalable heuristic algorithm in addition to the ILP-based solution. We present additional evaluation results that provide further insights into our ILP solution. We also present new results that analyze the solution quality and scalability of the heuristic. Finally, we expand our discussion of the related works and contrast our solution with the state-of-the-art.

The rest of the paper is organized as follows. We first discuss the related works and contrast our approach to them in Section II. Then, we mathematically define the problem, and present the possible actions for accommodating the scaling request and the trade-off between disruption and additional resource usage Section III. We present our proposed disruption cost model and the ILP formulation in Section IV. Then, we present our heuristic algorithm in Section V. We discuss our evaluation results in Section VI. Finally, we conclude with some future research directions in Section VII.

II. Related Works

Traditionally, lightpaths in optical networks are provisioned for peak traffic, as overprovisioning avoids tedious and disruptive lightpath reconfiguration procedure. Such static provisioning can be very inefficient from resource utilization perspective, especially considering the higher envisioned dynamicity of traffic in 5G and beyond 5G networks [9], [24]. Recent advances of key optical networking technologies, such as bandwidth-variable elastic transponders, sliceable transponders, and Reconfigurable Add Drop Multiplexers (ROADM) are now paving the way for lighpath adaptation to support more dynamic resource provisioning. As such, there has been a growing interest in adaptive spectrum allocation for lightpaths in dynamic traffic scenarios [9], [10], [18], [25], [26].

[2]A *split* is a contiguous spectrum allocation on a physical lightpath, provisioning partial or full bandwidth of a virtual link

[3]The term *slot* represents a spectrum/frequency slot unit in this paper

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2021.3064643, IEEE Journal on Selected Areas in Communications

As one the earliest works in this field, [27] studies lightpath adaptation under variable traffic demands in EON. The work proposes three spectrum reconfiguration schemes to accommodate changes in traffic demand. The schemes allow spectrum expansion/contraction with or without change in the central frequency or re-allocation of spectrum with a new central frequency and spectrum width. This work recognizes that spectrum expansion/contraction can create traffic disruption assuming that the variant modifying the central frequency causes the highest degree of disruption, and it suggests to use MBB strategy to avoid interruptions due to re-allocation of spectrum with a new central frequency and spectrum width. For spectrum expansion/contraction without change in the central frequency, the work of [27] proposes to activate/de-activate exterior sub-carriers without breaking the connectivity of the interior ones. However, such partial activation is only feasible for Optical-Orthogonal Frequency Division Multiplexing (O-OFDM) technologies. In addition, expansion/contraction requires optical filters on intermediate nodes of lightpath to be adjusted, and this adjustment also causes traffic disruption.

Similarly, [20] proposes three spectrum expansion/contraction (SEC) policies for modifying the spectrum allocated to each connection to adapt to time-varying transmission rate. The policies include constant spectrum allocation, dynamic high-expansion low-contraction, and dynamic alternate direction. This work allows to route the excess traffic over a different spectrum-path, or reroute the entire connection over a different spectrum-path. It assumes that transponders are capable of dynamically adapting the spectrum they utilize using techniques such as the push-pull to avoid traffic interruptions [21]. However, the problem with push-pull is that it can only re-tune the frequency if there is no other occupied spectrum allocation between the previous and new frequency positions. A technique that can alleviate this assumption is hop re-tuning [23], where the receiver can detect a frequency-position change with the help of thermal arrayed waveguide grating (AWG) and photo-detectors, and tune the transponder to the new position. This process requires 500ns, but the impact of this delay can be eliminated by the use of a buffer at the source. Also, hop re-tuning requires a completely new transponder design that may not be feasible in practice. Moreover, note that, with both push-pull and hop re-tuning techniques, the signaling protocol should be extended to propagate the spectrum policies to intermediate nodes, and this causes additional delays and possible disruptions.

Recent research studies have looked into lightpath reconfiguration for accommodating time-varying traffic from different perspectives. For example, [9] represents a bandwidth demand of a connection request as a 3-tuple consisting of minimum, average, and maximum required transmission rates and applies a stochastic optimization technique to develop a dynamic spectrum allocation approach using the most optimized modulation format. This work assumes to use hit-less transponders to ensure zero loss of data during transponder reconfiguration. However, such hit-less transponders are still at the experimental level and have not yet made it to production. Similarly, [25] proposes a heuristic scheme for routing and spectrum allocation for incremental traffic scenarios. This work creates a new lightpath to handle incremental traffic that is one of the reconfiguration actions in our solution. Some studies have proposed to improve network bandwidth and availability by adaptively tuning the modulation format without changing the spectrum allocation [28], [29]. Among them [29] proposes dynamic adjustment of physical link capacities in centrally controlled wide area networks, where reconfiguration is triggered locally by transponders [28]. In addition, [29] benchmarks modulation reconfiguration latency using both commercial transponders and evaluation boards, and proposes ways of reducing the latency.

Very recently, [26] proposes to completely relocate connections at the beginning of a reconfiguration interval by predicting traffic-demand matrices for each reconfiguration interval. In contrast, [10] addresses short-term traffic fluctuations by dividing a lightpath into multiple segments. This allows the shorter lightpath segments to use advanced modulation format that supports the traffic surge with or without changing spectrum allocation. However, such division of a lightpath into multiple segments introduces additional intermediate devices that may increase power consumption and end-to-end latency. In addition, [10] acknowledges that service interruption can occur due to dividing an operational lightpath into segments. Although [10] suggests to perform lightpath division in advance to avoid service interruption, such scheduling may not always be feasible. In the same line, [30] proposes to use spectrum expansion/reduction with multi-path routing to accommodate time-varying traffic. However, [30] does not take into account the traffic disruption introduced by the spectrum expansion/reduction operation.

The only work that considers revenue loss due to traffic disruption occurring during the reconfiguration of optical circuits for dynamic traffic as part of the optimization objective is [18]. However, [18] considers only the worst case lightpath reconfiguration time multiplied by the revenue loss as the cost. We significantly differ from [18] by modeling several alternative types of reconfigurations and by building a mechanism for selecting the reconfiguration that is based on assigning appropriate costs to the various reconfiguration types. To the best of our knowledge, our work is the first that captures different levels of disruptions caused by different types of reconfiguration actions and assigns appropriate costs to each of the actions.

III. Problem Definition

A. Problem Statement

We are given a substrate EON $G = (V, E)$ consisting of a set of EON nodes and links represented by V and E, respectively. We are also given a set of EON slices $\mathcal{G}$ embedded on G. Each EON slice $\bar{G} \in \mathcal{G}$ consists of a set of virtual nodes (VNodes) $\bar{V}_{\bar{G}}$ with location constraints and virtual links (VLinks) $\bar{E}_{\bar{G}}$. We assume that the set of all VLinks currently embedded on the EON is $\bar{E}$ where each VLink $\bar{e} \in \bar{E}$ has a bandwidth demand $b_{\bar{e}}$ (see VLink qr with initial demand $400G$ in Fig. 1). Each VNode is assigned to an EON node present in V as per location constraint and each VLink is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2021.3064643, IEEE Journal on Selected Areas in Communications

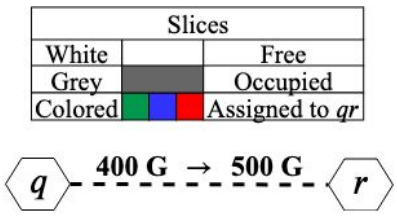


Fig. 1: Slice scaling request

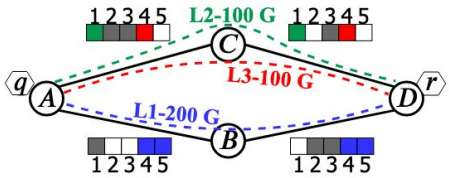


Fig. 2: Embedding before scaling

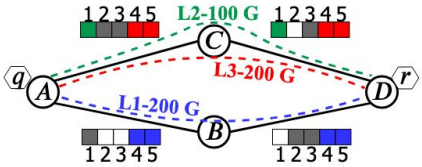


Fig. 3: Embedding after scaling

embedded to a set of paths (called as splits) in the EON where each path p is configured with a transmission configuration or tuple $t = (d, b, m, f) \in \mathbb{T} = (\mathbb{D} \times \mathbb{B} \times \mathbb{M} \times \mathbb{F})$ to provide a data-rate so that sum of data-rates is at least $b_{\bar{e}}$. Here, d, b, m, and f represent *data-rate*, *baud-rate*, *modulation format*, and *FEC* selected from the set of possible values $\mathbb{D}$, $\mathbb{B}$, $\mathbb{M}$, and $\mathbb{F}$, respectively. Each tuple t has a spectrum requirement and a maximum optical reach within which t can be used with satisfactory signal quality. For illustration, Fig. 2 shows the embedding of VLink qr on EON $ABCD$ with three splits of 200G, 100G, and 100G. These three splits are realized by three lightpaths established on EON, where blue, green, and red lightpaths are assigned spectrum slots 4 and 5, 1, and 4 on paths ABD, ACD, and ACD, respectively. Fig. 2 shows that slot assignments to lightpaths satisfy spectrum continuity and contiguity constraint [3], [13].

The slice scaling request comes as an increase of the bandwidth demand of a VLink $\bar{e} \in \bar{E}$ belonging to one of the EON slices $\bar{G} \in \mathcal{G}$ from $b_{\bar{e}}$ to $b'_{\bar{e}}$. The scaling request comes from the SP as a result of user demand changes or generated through traffic demand predictions [26], [31]. The objective of our problem is to accommodate a VLink demand increase while minimizing costs in terms of (i) number of transponders (Tx), (ii) spectrum occupation (Sp), and (iii) disruption to existing traffic (Ds). We assume that the spectrum on a lightpath p is divided into equal-width spectrum slots represented by the set S and enumerated as $1, 2 \ldots |S|$. We also assume that VNode assignment to SNode are not changed and the VLink's increased demand ($b'_{\bar{e}}$) can be spread up to a maximum number splits q, while accommodating the scaling request [8], [32].

B. Pre-computations

In this work, we focus on bandwidth scaling in EON network slices that provide transport connectivity to a wide geographic area. A VNode of a transport network slice is mapped to an EON node equipped with computing capacity located in an MDC or a POP site. Migrating a VNode from one MDC/PoP to another MDC/PoP would violate the location constraints provided with the network slice request. Hence, the only option for changing node mapping is to migrate a VNode from one EON node to another node within the same MDC/PoP. For this case, we assume that an EON node has enough capacity to accommodate a scaling request and hence, the VNode mapping remains unchanged. In addition, for each VLink $\bar{e} \in \bar{E}$, we pre-compute $\mathcal{P}^k_{\bar{e}}$, a set of k shortest paths between the pair of EON nodes on which VLink $\bar{e}$'s endpoints have been mapped. For each substrate path $p \in \mathcal{P}^k_{\bar{e}}$ on the EON, $|p|$ and $length(p)$ represent the number of EON links present in the path p and the physical length of the path p in terms of kilometers, respectively. Additionally, we denote the association between a path $p \in \mathcal{P}^k_{\bar{e}}$ and any EON link $e \in E$ by the binary input value $\delta_{p,e}$, which is set to 1 when e is on p, 0 otherwise. For each path $p \in \mathcal{P}^k_{\bar{e}}$, we pre-compute the set of admissible transmission configurations, $\mathcal{T}_{\bar{e},p} \subseteq \mathcal{T}$, such that each configuration $t \in \mathcal{T}_{\bar{e},p}$ results in a reach $r_t \geq length(p)$ and has a data-rate $t^{(d)}$.

C. Reconfiguration Actions and Disruption model

To accommodate a scaling request for $\bar{e} \in \bar{E}$, each split (*i.e.*, lightpath) in $\bar{e}$'s current embedding adopts one of the reconfiguration actions presented in Table I. These actions cause different levels of disruptions (Ds) to existing traffic and require different amounts of transponders (Tx) and spectrum slots (Sp). Each action can be represented by the slot s, tuple t, and path p combinations of a current split of $\bar{e}$ and a split in the re-embedding of $\bar{e}$ as specified in Table I. To quantify disruptions caused by each action, we assign a disruption cost $c(\bar{e}, p, t, s)$ to each s, t, and p combination of $\bar{e}$'s re-embedding by comparing them with $\bar{e}$'s current splits. Note that costs of the actions in Table I can be arbitrarily set based on the technology and desired objective.

Among the actions, R_1 does not change anything of a current split of $\bar{e}$. R_1 is the most preferable one as it requires no additional resource and does not disrupt traffic, hence has zero cost. However, R_1 alone may not support the increased demand, necessitating further actions. On the contrary, R_2 uses an advanced transmission configuration (*e.g.*, modulation format with a higher bits-per-symbol) for an existing split that does not change the spectrum allocation of the split but can provide a higher data-rate than the original data-rate of the split. R_2 needs no extra resources but has non-negligible disruption caused by a transponder reconfiguration (*e.g.*, modulation change takes $\sim$ 70 seconds [29]). Although R_3 creates a new lightpath to re-allocate a split's spectrum or to add a new lightpath similar to [25], it incurs moderate disruptions as a new lightpath whose spectrum does not overlap with any existing lightpath can be created with MBB [19]. In contrast, R_4 and R_5 change the spectrum width of an existing split either by expanding (R_4) or contracting (R_5) the width. Similarly, R_6 creates a new split whose spectrum allocation overlaps with the spectrum allocation of an existing split on any EON link. Hence, MBB cannot be applied to R_4, R_5, or R_6 as they require a change in filters on intermediate nodes of an existing lightpath, thus disrupting the lightpath's traffic [27]. Note that R_4 and R_5 cannot be represented by one s, t, and p combination and need to examine more than one slots.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2021.3064643, IEEE Journal on Selected Areas in Communications

TABLE I: Different reconfiguration actions and corresponding cost

ID	Reconfiguration action	Ds level	Extra Tx?	Extra Sp?	Combination of slot s, tuple t, and path p for the re-embedding of a VLink	c($\bar{e}$, p,t,s)
R_1	Reuse spectrum and transmission configuration of an existing split	Zero	No	No	s is used in current splits of $\bar{e}$ with same tuple t on same path p	0
R_2	Modify only transmission configuration of an existing split	Low	No	No	s is used in current splits of $\bar{e}$ with a different tuple $t' \neq t$, but on same path p	$c_1 >> 0$
R_3	Re-allocate a split or create a new split whose spectrum does not overlap with an existing split's spectrum	Moderate	Yes/ No	Yes/ No	s is not used in current splits of $\bar{e}$ and is not occupied by other VLinks	$c_2 >> c_1$
R_4	Expand spectrum allocation of an existing split	High	No	Yes	s is used in a current split of $\bar{e}$ on path p and another slot s' not used in same split of s, and both s and s' are used in a new split of $\bar{e}$'s on path p	$c_3 >> c_2$
R_5	Contract spectrum allocation of an existing split	High	No	No	s is used in a current and new split of $\bar{e}$ on path p and not all the slots of same current split are used in the new split of $\bar{e}$'s	$c_3 >> c_2$
R_6	Re-allocate a split or create a new split whose spectrum overlaps with an existing split's spectrum	Very High	Yes/ No	Yes/ No	s is used in current splits of $\bar{e}$ on path p' where p' and p have a common link	$c_3 >> c_2$

We now explain these actions and their costs with the help of the embedding in Fig. 2. In Fig. 2, slots assigned to $\bar{e}$'s current splits (L_1, L_2, and L_3) have 0, low, and high cost for R_1, R_2, and R_4-R_6, respectively. Free (white) slots in a path get appropriate cost for R_3-R_6, and remaining (grey) slots are unavailable to use. Fig. 3 shows the re-embedding of VLink qr after its new demand 500G is served by increasing L_3's data-rate to 200G. To do so, L_3's spectrum allocation is expanded to include slot 5 to already allocated slot 4 using R_4, disrupting L_3's traffic. A less-disruptive solution is to create a new lightpath using R_3 on path ACD that provides 100G through slot 5. This less-disruptive solution adds a new split that requires a pair of additional transponders at node A and D. A least-disruptive and resource efficient solution is to change L_3's current modulation format using R_2 to provide 200G without changing its current spectrum allocation, as long as the reach of new modulation format is respected on L_3.

IV. Problem Formulation

We present a path-based ILP formulation for solving the problem of scaling a VLink's ($\bar{e}$) embedding to accommodate a change in $\bar{e}$'s demand. Before applying the ILP formulation, we first free up the slots from the paths used in $\bar{e}$'s current lightpaths and mark the slots on the paths taken by the embedding of other VLinks of the same or different VNs as occupied. Then, we re-embed $\bar{e}$ with the new demand $b'_{\bar{e}}$ using the proposed ILP formulation such that each existing split of $\bar{e}$ adopts one of the reconfiguration actions presented in Table I. Table II summarizes key notations used in our formulation.

A. Inputs

The current embeddings of all VLink $\bar{u} \in \bar{E}$ are considered as input to the ILP formulation for scaling the demand for $\bar{e}$. Following matrix contains the slot mapping information for all VLink $\bar{u} \in \bar{E}$:

$$y^{cur}_{\bar{u},p,i,s} = \begin{cases} 1 & \text{if } \bar{u} \in \bar{E} \text{ currently uses slot } s \in S \text{ on path} \\ & p \in \mathcal{P}^k_{\bar{u}} \text{ for the } i\text{-th split} \\ 0 & \text{otherwise} \end{cases}$$

The disruption cost matrix $c(\bar{e}, p, t, s)$ for each possible s, t, and p combination of $\bar{e}$'s re-embedding as per Table I is also pre-computed and given as input to the ILP formulation as follows:

$$c(\bar{e},p,t,s) = \begin{cases} c_0(=0) & \text{if } s \text{ is used in current embedding of } \bar{e} \text{ with same tuple } t \text{ on same path } p \text{ using } R1 \\ c_1(>>0) & \text{if } s \text{ is used in current embedding of } \bar{e} \text{ with a different tuple } t_1 \neq t \text{ on same path } p \text{ using } R2 \\ c_2(>>c_1) & \text{if } s \text{ is not used in current embedding of occupied } \bar{e} \text{ and is not by other VLinks using } R3 \\ c_3(>>c_2) & \text{if } s \text{ is used in current embedding of } \bar{e} \text{ on path } p_1, \text{ and } p_1 \text{ and } p \text{ have a common EON link using } R6 \\ \infty & \text{if } s \text{ is currently occupied by other VLinks} \end{cases}$$

B. Decision Variables

We use VLink splitting for re-embedding a VLink, *i.e.*, satisfy the new demand of a VLink by allocating spectrum slots over multiple paths up to a maximum of q paths as q is the maximum number of allowable splits [8], [14]. Spectrum slot allocation over multiple paths can be done using a combination of the following two cases: (i) allocate contiguous sets of slots on distinct paths; (ii) allocate different contiguous sets of slots on the same path (with identical or different transmission configuration). The latter represents a scenario when there are sufficient slots available on a path, however, the path length or the spectrum fragmentation do not allow provisioning the total required data-rate using a single contiguous set of slots. To represent the second case, we assume each transmission

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2021.3064643, IEEE Journal on Selected Areas in Communications

TABLE II: Notation Table

Inputs & Pre-computations	
$G = (V, E)$	Substrate EON
$G = (V, E)$	Virtual Network (VN) or Network Slice
S	The set of equal-width frequency slots
q	The maximum number of allowable splits for a VLink
$\bar{e}$, $b_{\bar{e}}$, $b'_{\bar{e}}$	VLink $\bar{e} \in \bar{E}$ whose demand needs to be increased from $b_{\bar{e}}$ to $b'_{\bar{e}}$
$\mathcal{D}, \mathcal{B}, \mathcal{M}, \mathcal{F}$	Set of all possible data-rates, baud-rates, modulation formats, & FEC overhead levels
$\mathcal{T} = (\mathcal{D} \times \mathcal{B} \times \mathcal{M} \times \mathcal{F})$	Set of all possible transmission configurations
$t = (d, b, m, f) \in \mathcal{T}$	A transmission configuration. *i.e.*, a baud-rate b, modulation format m, and FEC overhead f, yielding data-rate d
$t^{(d)} \in \mathcal{D}, t^{(b)} \in \mathcal{B}, t^{(m)} \in \mathcal{M}, t^{(f)} \in \mathcal{F}$	Data-rate, baud-rate, modulation format, and FEC overhead of transmission configuration t, respectively
$\mathcal{R}$	Reach table
$r_t \in \mathcal{R}$	Achievable reach for configuration t
n_t	Number of spectrum slots required for configuration t
$\mathcal{P}^k_{\bar{e}}$	k candidate shortest paths for VLink $\bar{e} \in \bar{E}$
$\mathcal{T}_{\bar{e},p} \subseteq \mathcal{T}$	Admissible transmission configurations on path p for embedding VLink $\bar{e} \in \bar{E}$
$y^{cur}_{\bar{u},p,i,s}$	Indicates if $\bar{u} \in E$ currently uses slot $s \in S$ on path $p \in \mathcal{P}^k_{\bar{e}}$ for the i-th split
$c(\bar{e}, p, t, s)$	Disruption cost for VLink $\bar{e} \in E$, using path p, transmission configuration t, and slot $s \in S$
$\delta_{p,e}$	Binary input for a path $p \in \mathcal{P}^k_{\bar{e}}$ and an EON link $e \in E$ that is set to 1 when e is on p
Decision Variables	
$w_{\bar{e},p,t,i}$	Indicates if $\bar{e} \in E$ uses i-th instance of transmission configuration $t \in \mathcal{T}_{\bar{e},p}$ on $p \in \mathcal{P}^k_{\bar{e}}$
$y_{\bar{e},p,t,i,s}$	Indicates if $\bar{e} \in \bar{E}$ uses slot $s \in S$ on $p \in \mathcal{P}^k_{\bar{e}}$ with the i-th instance of transmission configuration $t \in \mathcal{T}_{\bar{e},p}$

configuration for a path can be instantiated multiple times (up to a maximum of q times as total number of splits cannot exceed q regardless of the case). The following variable relates a VLink to a path and an instance of a transmission configuration on that path:

$$w_{\bar{e},p,t,i} = \begin{cases} 1 & \text{if } \bar{e} \in \bar{E} \text{ uses } i\text{-th instance of } t \in \mathcal{T}_{\bar{e},p} \\ & \text{on path } p \in \mathcal{P}^k_{\bar{e}} \\ 0 & \text{otherwise} \end{cases}$$

The following decision variable creates relationship between a mapped path and the spectrum slots present in its links:

$$y_{\bar{e},p,t,i,s} = \begin{cases} 1 & \text{if } \bar{e} \in \bar{E} \text{ uses slot } s \in S \text{ on path } p \in \mathcal{P}^k_{\bar{e}} \\ & \text{with the } i\text{-th instance of } t \in \mathcal{T}_{\bar{e},p} \\ 0 & \text{otherwise} \end{cases}$$

C. Constraints

a) VLink scaling demand satisfaction: We re-embed a VLink by splitting it's increased demand $b'_{\bar{e}}$ across multiple (up to q) paths. Constraint (1) ensures that for a VLink $\bar{e} \in \bar{E}$, the sum of data-rates resulting from applying the selected transmission configurations on the selected splits is equal or larger than $b'_{\bar{e}}$. Then, (2) enforces an upper limit on the number of splits to be used by the re-embedding of the VLink $\bar{e} \in \bar{E}$.

$$b'_{\bar{e}} \leq \sum_{\forall p \in \mathcal{P}^k_{\bar{e}}} \sum_{\forall t \in \mathcal{T}_{\bar{e},p}} \sum_{i=1}^{q} (w_{\bar{e},p,t,i} \times t^{(d)}) \tag{1}$$

$$\sum_{\forall p \in \mathcal{P}^k_{\bar{e}}} \sum_{\forall t \in \mathcal{T}_{\bar{e},p}} \sum_{i=1}^{q} w_{\bar{e},p,t,i} \leq q \tag{2}$$

b) Slot assignment and spectral contiguity constraints: We ensure by (3) that if a path p is selected with a specific transmission configuration t, then the required number of slots n_t for configuration t to support the data-rate $t^{(d)}$ is allocated on p. (4) ensures that each slot on an EON link is allocated to at most one split during re-embedding. We ensure that a split's spectrum slots allocated on each link of a path are contiguous in the frequency spectrum by (5). Finally, (6) enforces that slots currently used by VLinks except $\bar{e}$ cannot be used by the re-embedding of $\bar{e}$.

$$\forall p \in \mathcal{P}^k_{\bar{e}}, \forall t \in \mathcal{T}_{\bar{e},p}, 1 \leq i \leq q : \sum_{\forall s \in S} y_{\bar{e},p,t,i,s} = n_t \times w_{\bar{e},p,t,i} \tag{3}$$

$$\forall e \in E, \forall s \in S : \sum_{\forall \bar{e} \in \bar{E}} \sum_{\forall p \in \mathcal{P}^k_{\bar{e}}} \sum_{\forall t \in \mathcal{T}_{\bar{e},p}} \sum_{i=1}^{q} y_{\bar{e},p,t,i,s} \times \delta_{p,e} \leq 1 \tag{4}$$

$$\forall p \in \mathcal{P}^k_{\bar{e}}, \forall t \in \mathcal{T}_{\bar{e},p}, 1 \leq i \leq q, 1 \leq s \leq |S| - 1 : \sum_{s'=s+2}^{|S|} y_{\bar{e},p,t,i,s'} \leq |S| \times (1 - y_{\bar{e},p,t,i,s} + y_{\bar{e},p,t,i,(s+1)}) \tag{5}$$

$$\forall p \in \mathcal{P}^k_{\bar{e}}, \forall t \in \mathcal{T}_{\bar{e},p}, 1 \leq i \leq q, \forall s \in S \text{ s. t. } c(\bar{e}, p, t, s) = \infty : y_{\bar{e},p,t,i,s} = 0 \tag{6}$$

D. Objective Function

Our objective is to minimize the cost of transponders, spectrum slots, and the number of slots disrupted for a VLink $\bar{e}$ that needs to be scaled. The following multi-objective function captures all three costs where θ, ω, and λ can be used to tune the priorities of different objectives:

$$\text{minimize}(\theta \times Cost^{Tx}_{\bar{e}} + \omega \times Cost^{Sp}_{\bar{e}} + \lambda \times Cost^{Ds}_{\bar{e}}) \tag{7}$$

In (7), $Cost^{Tx}_{\bar{e}}$ is the number of transponders required to accommodate $\bar{e}$'s new demand that is defined as follows:

$$Cost^{Tx}_{\bar{e}} = \sum_{\forall p \in \mathcal{P}^k_{\bar{e}}} \sum_{\forall t \in \mathcal{T}_{\bar{e},p}} \sum_{i=1}^{q} w_{\bar{e},p,t,i} \tag{8}$$

Similarly, $Cost^{Sp}_{\bar{e}}$ in (7) is the number of spectrum slots required to accommodate $\bar{e}$'s new demand over all the paths that is defined as follows:

$$Cost^{Sp}_{\bar{e}} = \sum_{\forall p \in \mathcal{P}^k_{\bar{e}}} \sum_{\forall t \in \mathcal{T}_{\bar{e},p}} \sum_{i=1}^{q} \sum_{\forall s \in S} y_{\bar{e},p,t,i,s} \times |p| \tag{9}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2021.3064643, IEEE Journal on Selected Areas in Communications

Finally, $Cost_{\bar{e}}^{Ds}$ in (7) represents disruption cost incurred by reconfiguring $\bar{e}$'s lightpaths to accommodate $\bar{e}$'s new demand. To compute $Cost_{\bar{e}}^{Ds}$, we use $c(\bar{e}, p, t, s)$, which captures the disruption caused by reconfiguration actions $R_1 - R_3$ and R_6 on each s, t, and p combination of $\bar{e}$'s re-embedding. However, actions R_4 and R_5 are complicated as they cannot be represented by the s, t, and p combination of $\bar{e}$'s re-embedding, and require to examine more than one slot of $\bar{e}$'s re-embedding. To capture the disruption triggered by expansion (R_4) or reduction (R_5) of spectrum width of a split, we introduce following two additional variables:

$$\mu_{\bar{e}} = \begin{cases} 1 & \text{if re-embedding of } \bar{e} \in \bar{E} \text{ has a split that uses at least a slot of a split } i \text{ in } \bar{e}\text{'s current embedding and at least a slot that is not part of the split } i \\ 0 & \text{otherwise} \end{cases}$$

$$\sigma_{\bar{e}} = \begin{cases} 1 & \text{if re-embedding of } \bar{e} \in \bar{E} \text{ has a split that uses at least a slot of a split } i \text{ in } \bar{e}\text{'s current embedding and does not use all the slots of the split } i \\ 0 & \text{otherwise} \end{cases}$$

Since spectrum slots of a split are assigned contiguously, $\mu_{\bar{e}}$ and $\sigma_{\bar{e}}$ represent the expansion and reduction of spectrum width of a split, respectively. To derive these two variable, we introduce following auxiliary variables:

$$o_{\bar{e},p,t,i,j} = \begin{cases} 1 & \text{if re-embedding of } \bar{e} \in \bar{E} \text{ has a split } i \in N_q \text{ on path } p \in \mathcal{P}_{\bar{e}}^k \text{ with tuple } t \in \mathcal{T}_{\bar{e},p} \text{ that uses at least a slot used by current split } j \leq q_{\bar{e}}^{cur} \\ 0 & \text{otherwise} \end{cases}$$

$$g_{\bar{e},p,t,i,j} = \begin{cases} 1 & \text{if re-embedding of } \bar{e} \in \bar{E} \text{ has a split } i \in N_q \text{ on path } p \in \mathcal{P}_{\bar{e}}^k \text{ with tuple } t \in \mathcal{T}_{\bar{e},p} \text{ that uses at least a slot not used by current split } j \leq q_{\bar{e}}^{cur} \\ 0 & \text{otherwise} \end{cases}$$

$$z_{\bar{e},p,t,i,j} = \begin{cases} 1 & \text{if re-embedding of } \bar{e} \in \bar{E} \text{ has a split } i \in N_q \text{ on path } p \in \mathcal{P}_{\bar{e}}^k \text{ with tuple } t \in \mathcal{T}_{\bar{e},p} \text{ that uses all the slots of a current split } j \leq q_{\bar{e}}^{cur} \\ 0 & \text{otherwise} \end{cases}$$

We can define these auxiliary variable using the following three constraints. For example, constraint (10) states that $o_{\bar{e},p,t,i,j}$ will be one when there is a slot s that is used in the split i on path p of the re-embedding of $\bar{e}$ ($y_{\bar{e},p,t,i,s} = 1$) and the same slot s was used in split j of path p of the current embedding of $\bar{e}$ ($y_{\bar{e},p,j,s}^{cur} = 1$). Similarly, (11) enforces that $g_{\bar{e},p,t,i,j}$ is one when slot s is used in the split i on path p of the re-embedding of $\bar{e}$ ($y_{\bar{e},p,t,i,s} = 1$) and slot s was not used by split j of path p of the current embedding of $\bar{e}$ ($y_{\bar{e},p,j,s}^{cur} = 0$). Finally, (12) ensures that $z_{\bar{e},p,t,i,j}$ is one when split i on path p of the re-embedding of $\bar{e}$ uses all the slots of split j of path p of the current embedding of $\bar{e}$.

$$\forall p \in \mathcal{P}_{\bar{e}}^k, \forall t \in \mathcal{T}_{\bar{e},p}, \forall i \in N_q, \forall j \leq q_{\bar{e}}^{cur}, \forall s \in S : \quad o_{\bar{e},p,t,i,j} \geq (y_{\bar{e},p,t,i,s} \times y_{\bar{e},p,j,s}^{cur}) \tag{10}$$

$$\forall p \in \mathcal{P}_{\bar{e}}^k, \forall t \in \mathcal{T}_{\bar{e},p}, \forall i \in N_q, \forall j \leq q_{\bar{e}}^{cur}, \forall s \in S : \quad g_{\bar{e},p,t,i,j} \geq (y_{\bar{e},p,t,i,s} \times (1 - y_{\bar{e},p,j,s}^{cur})) \tag{11}$$

$$\forall p \in \mathcal{P}_{\bar{e}}^k, \forall t \in \mathcal{T}_{\bar{e},p}, \forall i \in N_q, \forall j \leq q_{\bar{e}}^{cur}, \forall s \in S : \quad z_{\bar{e},p,t,i,j} \leq (y_{\bar{e},p,t,i,s} \times y_{\bar{e},p,j,s}^{cur}) \tag{12}$$

Spectrum width expansion is performed ($\mu_{\bar{e}}$ is one) if both the variable $o_{\bar{e},p,t,i,j}$ and $g_{\bar{e},p,t,i,j}$ are simultaneously one for the same pair of splits i and j. This is achieved by the following constraint:

$$\forall p \in \mathcal{P}_{\bar{e}}^k, \forall t \in \mathcal{T}_{\bar{e},p}, \forall i \in N_q, \forall j \leq q_{\bar{e}}^{cur} : \quad \mu_{\bar{e}} \geq (o_{\bar{e},p,t,i,j} \times g_{\bar{e},p,t,i,j}) \tag{13}$$

Since (13) is a non-linear constraint and $\mu_{\bar{e}}$ is a binary variable, we can easily linearize it as:

$$\forall p \in \mathcal{P}_{\bar{e}}^k, \forall t \in \mathcal{T}_{\bar{e},p}, \forall i \in N_q, \forall j \leq q_{\bar{e}}^{cur} : \quad \mu_{\bar{e}} \geq (o_{\bar{e},p,t,i,j} + g_{\bar{e},p,t,i,j} - 1) \tag{14}$$

Similarly, spectrum width reduction is performed ($\sigma_{\bar{e}}$ is one) if the variable $o_{\bar{e},p,t,i,j}$ is one and the variable $z_{\bar{e},p,t,i,j}$ is zero. This is achieved by the following constraint:

$$\forall p \in \mathcal{P}_{\bar{e}}^k, \forall t \in \mathcal{T}_{\bar{e},p}, \forall i \in N_q, \forall j \leq q_{\bar{e}}^{cur} : \quad \sigma_{\bar{e}} \geq (o_{\bar{e},p,t,i,j} - z_{\bar{e},p,t,i,j}) \tag{15}$$

Using the disruption cost model and the derived variables $\mu_{\bar{e}}$ and $\sigma_{\bar{e}}$, we can define disruption cost as follows:

$$Cost_{\bar{e}}^{Ds} = \Big(\sum_{\forall p \in \mathcal{P}_{\bar{e}}^k} \sum_{\forall t \in \mathcal{T}_{\bar{e},p}} \sum_{i=1}^{q} \sum_{\forall s \in S} c(\bar{e}, p, t, s) \times y_{\bar{e},p,t,i,s} \Big) + (\mu_{\bar{e}} + \sigma_{\bar{e}}) \times c_3 \tag{16}$$

The first part in (16) represents the cost of reconfiguration actions $R_1 - R_3$ and R_6 on each s, t, and p combination of $\bar{e}$'s re-embedding. The second part in (16) imposes a one-time large cost (c_3 is sufficiently large) if any spectrum width expansion/reduction is performed by R_4/R_5 during $\bar{e}$'s re-embedding. Since our objective in (7) is a minimization function where $\mu_{\bar{e}}$ and $\sigma_{\bar{e}}$ are used in additive manner, the ILP formulation will try to use the zero values for both $\mu_{\bar{e}}$ and $\sigma_{\bar{e}}$ to minimize (7). However, nonzero values of $\mu_{\bar{e}}$ or $\sigma_{\bar{e}}$ can also appear in a solution of the ILP formulation if they are the only feasible way to accommodate the scaling request with the minimal cost. Note that the proposed ILP formulation can only be used to solve scaling problem for one VLink as the formulation requires the slots allocated to other VLinks known in advance. An brute-force way of solving $\bar{\mathcal{E}} \subseteq \bar{E}$ VLink re-scaling requests is to run the ILP formulation for all possible ($\bar{\mathcal{E}}!$) orders of VLinks and select the order and corresponding re-embeddings yielding the minimum value of the objective function. Such an approach can easily be parallelized for the possible orders to keep the running time reasonable. An heuristic way is to run the ILP formulation one-by-one for a number of VLinks using any particular order.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2021.3064643, IEEE Journal on Selected Areas in Communications

V. Heuristic Algorithm

The ILP formulation presented in Section IV needs a prolonged period of time to find the near-optimal solution for the network slice scaling problem. To find solutions within a reasonable time, we propose a heuristic algorithm in this section. The algorithm re-embeds the VLink with increased demand without impacting the embedding of other VLinks of the same or different VNs. Note that re-embedding of a VLink involves applying a reconfiguration action from Table I for each of the splits in the current embedding of the VLink, while achieving a certain objective as presented in (7). In this regard, we present a two stage heuristic algorithm (Algorithm 1) and describe it in a top-down manner. We first give an overview of the heuristic in Section V-A, followed by detailed description of the stages in Section V-B and Section V-C, respectively.

Algorithm 1: Adapt with Change in VLink Demand

1 **function** *EmbedDemandChange(*G, $\mathcal{I}_{\bar{e}}, b_{\bar{e}}, b'_{\bar{e}}$*)*
2 $SolutionPool \leftarrow \emptyset$
3 $A \leftarrow$ $PopulateConfigurationChangeVector(\mathcal{I}_{\bar{e}})$
4 $SolutionPool \leftarrow \{\forall H \in A$ such that $TotalDataRate(H) \geq b'_{\bar{e}}\}$
5 **if** $SolutionPool \neq \emptyset$ **then**
6 **return** BestSolution(SolutionPool)
7 $SolutionPool \leftarrow$ $PopulateSpectrumChangeVector(G, \mathcal{I}_{\bar{e}}, b_{\bar{e}}, b'_{\bar{e}}, A)$
8 **if** $SolutionPool \neq \emptyset$ **then**
9 **return** BestSolution(SolutionPool)
10 **return** $\mathcal{I}_{\bar{e}}$

A. Overview of Heuristic

Algorithm 1 takes as input a scaling request in the form of a VLink $\bar{e}$'s current demand $b_{\bar{e}}$, its new demand $b'_{\bar{e}}$ and the current embedding $\mathcal{I}_{\bar{e}}$ of $\bar{e}$, and the present state of G. In the first stage, Algorithm 1 employs Algorithm 2 (details in Section V-B) for populating a set of offsprings of $\mathcal{I}_{\bar{e}}$ as A by changing only the Transmission Configurations (TCs) of different subsets of splits in $\mathcal{I}_{\bar{e}}$ where each offspring of $\mathcal{I}_{\bar{e}}$ has a larger aggregate data-rate than $\mathcal{I}_{\bar{e}}$'s data-rate. Therefore, A contains the current embedding of the VLink ($\mathcal{I}_{\bar{e}}$) and all the offsprings of $\mathcal{I}_{\bar{e}}$ which can be obtained by applying action R_2 on the splits in $\mathcal{I}_{\bar{e}}$. Then, Algorithm 1 identifies the offsprings in A that result in a data-rate equal to or higher than the new demand $b'_{\bar{e}}$, and assigns them to $SolutionPool$. Note that the solutions currently in $SolutionPool$ (if any) are the preferred ones since they neither add new split(s) nor change existing spectrum allocation while supporting the new demand $b'_{\bar{e}}$ using the less disruptive action R_2. Next, Algorithm 1 returns the best solution from $SolutionPool$ evaluated based on the objective function (7).

If the first stage of Algorithm 1 fails to accommodate the new demand $b'_{\bar{e}}$ by only applying action R_2, then the second stage of Algorithm 1 tries to use R_3 – R_6 for re-embedding $\bar{e}$. These actions cause moderate to high disruption to existing traffic since they involve re-allocating spectrum of existing splits or setting up new lightpath. This second stage, *i.e.*, Algorithm 3 (details in Section V-C), computes all possible solutions that can be obtained by applying actions R_3 – R_6 on each offspring in A computed by Algorithm 2. If Algorithm 3 returns a non-empty set of solutions in $SolutionPool$, Algorithm 1 will return the best solution in $SolutionPool$ according to the objective function (7). Otherwise, it returns the current embedding $\mathcal{I}_{\bar{e}}$ indicating that it is not possible to accommodate the new demand $b'_{\bar{e}}$.

B. Stage-1: Populate solutions by changing TC

Algorithm 2 returns a set of embeddings A containing $\mathcal{I}_{\bar{e}}$, the current embedding of the VLink $\bar{e}$, and all the offsprings that can be generated from $\mathcal{I}_{\bar{e}}$ by changing the TC of the splits in $\mathcal{I}_{\bar{e}}$ (applying action R_2), resulting in a data-rate higher than that of $\mathcal{I}_{\bar{e}}$. To obtain such offsprings from $\mathcal{I}_{\bar{e}}$, Algorithm 2 enumerates all subsets of splits in $\mathcal{I}_{\bar{e}}$. For each subset $\delta\mathcal{I}_{\bar{e}}$, Algorithm 2 first computes the complementary set $H = \mathcal{I}_{\bar{e}} \setminus \delta\mathcal{I}_{\bar{e}}$ and computes d^{prev} as the total data-rate of the splits in $\delta\mathcal{I}_{\bar{e}}$ with their existing TCs (lines 4 – 5). d^{prev} will later be used by Algorithm 2 to determine if an increased total data-rate can be achieved by changing the TCs of the splits in $\delta\mathcal{I}_{\bar{e}}$. Note that TCs of the splits in H remain unchanged, while those of splits in $\delta\mathcal{I}_{\bar{e}}$ will be included to H after possible change in their configurations. Hence, Algorithm 2 attempts to change the TC of each split $split(p, t, s)$ in $\delta\mathcal{I}_{\bar{e}}$ for maximizing the resulting aggregate data-rate using the current spectrum allocation of $split(p, t, s)$ on their current embedding paths. To accomplish this, Algorithm 2 selects the TC t' from the set of TCs that has a longer reach than the current embedding path of $split(p, t, s)$ and requires no additional spectrum than what is currently allocated to $split(p, t, s)$ (line 8). In the worst case, the TCs of all the splits in $\delta\mathcal{I}_{\bar{e}}$ can remain unchanged, making the resulting offspring same as $\mathcal{I}_{\bar{e}}$. To avoid such a scenario, Algorithm 2 adds a subset $\delta\mathcal{I}_{\bar{e}}$ to the offspring H only if the changes in TCs of the splits in $\delta\mathcal{I}_{\bar{e}}$ yield an increased data-rate than d^{prev} (line 9 – 11). After computing such an offspring H, Algorithm 2 appends H to the set of offsprings A. This process is repeated for other subsets of splits in $\mathcal{I}_{\bar{e}}$ resulting in more offsprings. Algorithm 2 includes each of the qualified offsprings to A and returns A at the end.

C. Stage-2: Populate solutions by changing spectrum

Algorithm 3 enumerates each offspring H in A and generates a set of solutions from H that requires some change in the spectrum allocation of the splits in H. Algorithm 3 divides the changes in the spectrum allocation in two categories based on the level of disruption they might cause. Consequently, Algorithm 3 works in two steps. In the first step, Algorithm 3 generates solutions that do not overlap with the spectrum allocation of any split (*i.e.*, applies action R_3) in the offspring, while the solutions generated in the second step overlap with the spectrum allocation of any existing split (*i.e.*, applies actions $R_4 - R_6$). Regardless of the step, the basic principle to generate a solution from an offspring in A remains the same.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2021.3064643, IEEE Journal on Selected Areas in Communications

Algorithm 2: Populate Solutions by TC Change

1 **function** *PopulateConfigurationChangeVector($\mathcal{I}_{\bar{e}}$)*
2 $A \leftarrow \{\mathcal{I}_{\bar{e}}\}$
3 **foreach** $\delta\mathcal{I}_{\bar{e}} \subseteq splits(\mathcal{I}_{\bar{e}})$ **do**
4 $H \leftarrow \mathcal{I}_{\bar{e}} \setminus \delta\mathcal{I}_{\bar{e}}$
5 $d^{prev} \leftarrow TotalDataRate(\delta\mathcal{I}_{\bar{e}})$
6 **foreach** $split(p,t,s) \in \delta\mathcal{I}_{\bar{e}}$ **do**
7 $t' \leftarrow$ Best tuple $t' \in T$ considering the length of path p and spectrum allocation s
8 $split(p,t,s) \leftarrow split(p,t',s)$
9 **if** $TotalDataRate(\delta\mathcal{I}_{\bar{e}}) > d^{prev}$ **then**
10 $H \leftarrow H \cup \delta\mathcal{I}_{\bar{e}}$
11 $A \leftarrow A \cup \{H\}$
12 **return** A

Algorithm 3: Populate Solutions by Spectrum Change

1 **function** *PopulateSpectrumChangeVector(G, $\mathcal{I}_{\bar{e}}, b_{\bar{e}}, b'_{\bar{e}}, A$)*
2 $d \leftarrow b'_{\bar{e}} - b_{\bar{e}}$
3 $SolutionPool \leftarrow \emptyset$
4 **foreach** $a \in A$ **do**
5 **foreach** $\delta a \subseteq splits(a)$ **do**
6 $H \leftarrow a \setminus \delta a$
7 $\bar{d} \leftarrow TotalBitRate(\delta a) + d$
8 $newSplits \leftarrow FindSolution(G, \bar{d})$
9 **if** $newSplits \neq \emptyset$ *and* $(|newSplits| + |H|) \leq q$ **then**
10 $H \leftarrow H \cup newSplits$
11 $SolutionPool \leftarrow SolutionPool \cup \{H\}$
12 Temporarily de-allocate the splits of δa from G
13 $H \leftarrow a \setminus \delta a$
14 $newSplits \leftarrow FindSolution(G, \bar{d})$
15 **if** $newSplits \neq \emptyset$ *and* $(|newSplits| + |H|) \leq q$ **then**
16 $H \leftarrow H \cup newSplits$
17 $SolutionPool \leftarrow SolutionPool \cup \{H\}$
18 Embed splits of δa back on G
19 **return** SolutionPool

To generate a solution from an offspring $a \in A$, Algorithm 3 explores all subsets of the set of splits in a. For a particular subset δa, Algorithm 3 removes the splits in δa, keeping only the splits in $a \setminus \delta a$ (Line - 6). Despite removing the splits in δa, the spectrum allocation of the splits in δa are kept unmodified in G to ensure that the generated solutions do not overlap with the originally allocated spectrum of the removed splits. Then, the algorithm computes the data-rate $\bar{d}$ that has to be served by the resulting solution H as the sum of the data-rates of removed splits and the increase in data-rate from the scaling request (Line - 7). After computing the data-rate $\bar{d}$, Algorithm 3 finds the embedding solution ($newSplits$) for the demand $\bar{d}$ using a VLink embedding algorithm similar to the one proposed in [8] while considering the current state of the EON G (Line - 8). If an embedding for the demand $\bar{d}$ can be found and the constraint on maximum number of splits is not violated, we augment H with $newSplits$ and the resulting offspring H is added to the solution pool.

The second step to generate solutions works in the same way except that the spectrum allocation of the removed splits in δa are made available to be used by the re-embedding. This is done by temporarily de-allocating the splits of δa from the links in G and embedding them back after computing a solution (Line - 18). Doing so may result in an overlap between the spectrum allocation of removed splits in Line - 12 and the new splits to be created. Such overlap impedes the use of MBB during reconfiguration, thus causing high level of disruptions. However, these solutions may act as the last resort in extremely resource constrained scenarios where solutions without overlap with existing splits are not feasible.

D. Running time analysis

Algorithm 2 iterates over all the subsets of $splits(\mathcal{I}_{\bar{e}})$ ($|splits(\mathcal{I}_{\bar{e}})| \leq q$), resulting in $O(2^q)$ iterations (line 3). Then, for each such subset, Algorithm 2 iterates over the splits of that subset (line 6), requiring $O(q)$ time per subset. Identifying the best tuple on line 7 of Algorithm 2 can be done in constant time by leveraging a pre-computed hash table (hashed by the pre-computed set of usable paths in the SN) TCs. Therefore, the worst case running time of Algorithm 2 is $O(q2^q)$. The size of A, *i.e.*, the offsprings generated by Algorithm 2, which is also the input to Algorithm 3 is less than or equal to $2^{|splits(\mathcal{I}_{\bar{e}})|} \leq 2^q$. Therefore, Algorithm 3 invokes the $FindSolution$ procedure at most $2 \times 2^q \times 2^q = O(2^q)$ times. The running time of $FindSolution$ procedure as given by [8] is $\left(\sum_{i=1}^{q}\binom{k+i-1}{i}\right) \times \frac{(|\mathcal{D}_{\mathbb{P}^k_{\bar{e}}}|+q-1)!}{(|\mathcal{D}_{\mathbb{P}^k_{\bar{e}}}|-1)! \times \Pi_{d_j \in \mathcal{D}_{\mathbb{P}^k_{\bar{e}}}} m(d_j)!} \times \frac{q!}{\Pi_{p_j \in \mathcal{P}^k_{\bar{e}}} m(p_j)!}$, where $\mathcal{D}_{\mathbb{P}^k_{\bar{e}}}$ is the set of admissible data-rates for embedding a VLink, and $m(d_j)$ and $m(p_j)$ represent the multiplicity of a data-rate $d_j \in \mathcal{D}_{\mathbb{P}^k_{\bar{e}}}$ and a path $p_j \in \mathcal{P}^k_{\bar{e}}$, respectively. Therefore, the total running time of our heuristic, Algorithm 1, is equal to the sum of running times of the Algorithm 2 and the Algorithm 3 in the worst case, *i.e.*, $O\left(q2^q + 2^q\left(\sum_{i=1}^{q}\binom{k+i-1}{i} \times \frac{(|\mathcal{D}_{\mathbb{P}^k_{\bar{e}}}|+q-1)!}{(|\mathcal{D}_{\mathbb{P}^k_{\bar{e}}}|-1)! \times \Pi_{d_j \in \mathcal{D}_{\mathbb{P}^k_{\bar{e}}}} m(d_j)!} \times \frac{q!}{\Pi_{p_j \in \mathcal{P}^k_{\bar{e}}} m(p_j)!}\right)\right)$. However, typical values of q and k are small and hence running time depends on the set of admissible data rates in $\mathcal{D}_{\mathbb{P}^k_{\bar{e}}}$. Since the size of the set of admissible data rates is constrained by the maximum value of the bandwidth demand of a virtual link and the coarse-grained data rates (*e.g.*, 50G, 100G, and so on) of commercial transponders, the running time of Algorithm 1 remains within a reasonable limit (see Fig. 7).

VI. Evaluation

A. Simulation Setup

We implement the ILP formulation using IBM ILOG CPLEX and compare it with a C++ implementation of the

heuristic presented in Section V. We consider a fully-flexible EON using Nobel Germany[4] (17 nodes and 26 links) topology. Each EON link has 4THz spectrum bandwidth divided into 160 slots of 25GHz. To emulate a live EON, we develop a discrete event simulator similar to the simulators used in [33], [34]. The simulator loads the EON with traffic from VNs by simulating VN arrival and departure events and allocating and releasing spectrum resources accordingly. When a VN arrives, the simulator embeds the VN using the algorithm proposed in [8]. This simulator generates snapshots of the EON at different time instances in which varying number of VNs are embedded yielding different EON utilizations. We then select a set of snapshots of the EON for a given value of EON utilization and a number of VNs embedded on the EON. To evaluate the ILP formulation and benchmark the heuristic against ILP, we capture different snapshots of the EON at 20% and 50% utilizations and select about 25 VLinks from those snapshots that have initial bandwidth of 500G to increase their demand by 100G to 500G. We consider scaling of these 25 VLinks as separate problem instances for the same network state and solve them independently to report average performance metrics with good statistical confidence.

To analyse the performance of the heuristic, we take different snapshots of the EON at 50% and 60% utilizations and select VLinks that have initial bandwidth from 300G to 700G, and increase their demand by 50G to 300G. This allows us to run heuristic algorithm for higher number of problem instances to get more statistically confident results. To evaluate the effectiveness of reconfiguration actions presented in Table I under different initial settings, we use two different strategies for the initial VN embedding in our simulator. The first strategy uses the optimal transmission configuration for embedding VLinks of a VN, while the second approach uses a random transmission configuration. For both strategies, we consider scaling of the VLinks as separate problem instances for the same network state and solve them independently using the heuristic to report average performance metrics with good statistical confidence.

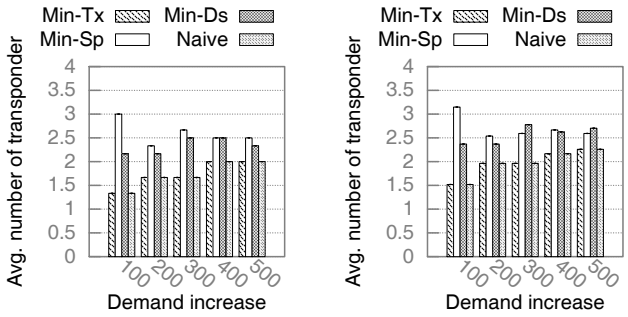
B. Compared Variants

We solve the scaling problem instances using three variants of the objective presented in (7). Among them, *Min-Tx* prioritizes transponder usage over spectrum requirement and disruption by considering $Cost_{\bar{e}}^{Tx}$, $Cost_{\bar{e}}^{Sp}$ and $Cost_{\bar{e}}^{Ds}$ as the primary, secondary, and tertiary objectives using the weights $\theta = 1000$, $\omega = 10$, and $\lambda = 0.0001$. Similarly, *Min-Ds* swaps the roles of $Cost_{\bar{e}}^{Tx}$ and $Cost_{\bar{e}}^{Ds}$ to give the disruption minimization the highest preference using the weights $\theta = 0.01$, $\omega = 1$, and $\lambda = 1000$. In contrast, *Min-Sp* swaps the roles of $Cost_{\bar{e}}^{Tx}$ and $Cost_{\bar{e}}^{Sp}$ to prioritize spectrum minimization using the weights $\theta = 10$, $\omega = 1000$, and $\lambda = 0.0001$. We also compare these variants with a baseline ILP approach, called *Naive*, that uses $Cost_{\bar{e}}^{Tx}$ and $Cost_{\bar{e}}^{Sp}$ as the primary and secondary objectives with $\theta = 1000$ and $\omega = 10$ and sets $\lambda = 0$ to ignore disruption minimization. In addition, we used the cost values $c_0 = 0$, $c_1 = 1$, $c_2 = 10$, and $c_3 = 1000$ for the disruption cost matrix in accordance with Table I.

[4]SNDlib Repository available at http://sndlib.zib.de

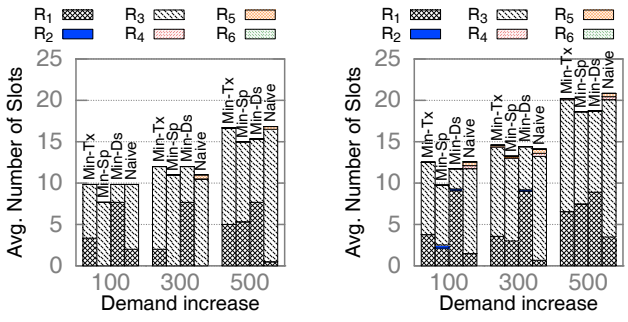
C. Performance metrics

- **Avg. number of transponder usage.** Average number of transponders ($Cost_{\bar{e}}^{Tx}$ in (7)) needed for scaling VLink demands across all problem instances.
- **Avg. number of slots.** Average number of spectrum slots ($Cost_{\bar{e}}^{Sp}$ in (7)) needed to accommodate the new demands for a VLink across all problem instances.
- **Distribution of different actions in Table I.** Average number of slots that are involved in different actions in Table I to scale VLink demands across all problem instances.
- **Avg. number of slots involved in disruption.** Average number of spectrum slots that are involved in disruptive actions *i.e.*, $R_2 - R_6$ in Table I.



(a) 20% EON utilization (b) 50% EON utilization

Fig. 4: Transponder usage



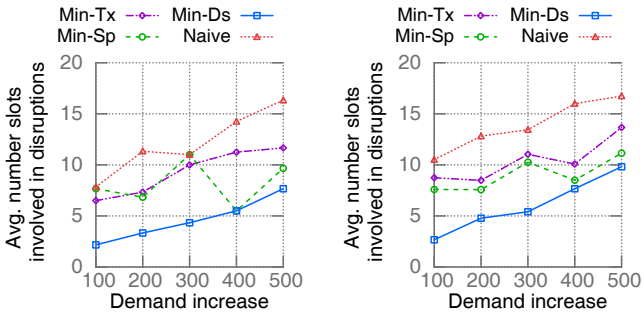
(a) 20% EON utilization (b) 50% EON utilization

Fig. 5: Slots using actions in Table 1

D. Discussion

1) Performance of ILP formulation: Fig. 4 presents transponder usage by all compared variants for two different values of EON utilization. Similarly, Fig. 5 shows the breakdown of slots used by different actions of Table I along with slot usages for the compared variants, and Fig. 6 reports number of slots involved in disruption for the variants. As expected, *Min-Tx* and *Min-Sp* incur the lowest number of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2021.3064643, IEEE Journal on Selected Areas in Communications



(a) 20% EON utilization (b) 50% EON utilization

Fig. 6: Disruptions

transponders and slots in Fig. 4 and Fig. 5, respectively, due to their prioritization of objective. In contrast, *Min-Ds* tries to reuse existing lightpaths as much as possible (see R_1's dominance for *Min-Ds* in Fig. 5) to minimize disruption. Doing so urges *Min-Ds* to create extra lightpaths for satisfying the new demand, forcing *Min-Ds* to use 21% more transponders and 5% more slots than *Min-Tx* and *Min-Sp*, respectively. As adopting R_1 contradicts the goal of *Min-Tx* and *Naive*, *Min-Tx*, and *Naive* prefers re-allocation with R_3 (see R_3's dominance for *Min-Tx* and *Naive* in Fig. 5 with low transponder usages in Fig. 4). Conversely, *Min-Sp* prefers new split creation with R_3 (see R_3's dominance for *Min-Sp* in Fig. 5 with its high transponder usage in Fig. 4).

Comparing the two figures with 20% and 50% EON utilization in Fig. 4 reveals that a higher number of transponders are needed to satisfy the same demand increase in a highly loaded EON than in a low utilized one. This is intuitive since spectrum fragmentation in a highly loaded EON inhibits the variants to find contiguous free spectrum slots in the candidate paths resulting in more splits (*i.e.*, more transponders) to meet new demand during re-embedding. Similar behavior can be observed in Fig. 5 where more spectrum slots are being used by the compared variants in a highly loaded EON since fragmentation prevents them from using the most optimized transmission configuration and path. Note in Fig. 5 that all variants, except *Naive*, adopt other reconfiguration actions than R_1 and R_3 with zero to low probability due to two reasons. First, length of an existing lightpath may not support a transmission configuration with a higher data-rate on the same spectrum allocation inhibiting the use of R_2. Second, R_4-R_6 cause higher level of disruption, and are used only when no other options are feasible.

Fig. 6 reports the average number of spectrum slots that are involved in disruptive actions (R_2-R_6) to accommodate the bandwidth scaling request for all the compared approaches. Fig. 6 shows that *Min-Ds*, on average, disrupts 40%, 31%, and 54% less slots carrying live traffic compared to *Min-Tx*, *Min-Sp*, and *Naive*, respectively. These disruptions come mostly by applying actions R_3 as depicted in Fig. 5. Note that fluctuating behavior is observed for the curves of *Min-Tx*, *Min-Sp*, and *Naive* approaches in Fig. 6. Among them *Min-Tx* and *Min-Sp* give disruption minimization the lowest priority after minimizing transponder and spectrum slot usages. On the other hand, *Naive* does not consider disruption minimization at all while accommodating the bandwidth scaling request. Since none of *Min-Tx*, *Min-Sp*, and *Naive* consider disruption minimization as their primary objective, an arbitrary number of slots is affected by disruptive actions (R_2-R_6) resulting in fluctuations in their curves in Fig. 6. In contrast, *Min-Ds* considers disruption minimization as its primary objective, thus minimizing the number of slots affected by disruptive actions (R_2-R_6). Hence, the curve for *Min-Ds* shows a monotonic increase in the average number of spectrum slots that are involved in disruptive actions with the increase in bandwidth demands. The monotonic increase for *Min-Ds* is due to the fact that a higher bandwidth increase requires a higher number of spectrum slots (see Fig. 4) of which a fraction of slots is involved in disruptive actions.

2) Benchmarking the heuristic algorithm: Fig. 7 shows the mean cost ratio between ILP and heuristic and their mean execution times averaged over different snapshots for the three main compared variants. Our heuristic achieves solution that are close to results produced by the ILP formulation for all the three objectives, as we can observe that cost ratios are very close to one. As for execution times, our heuristic takes 3 to 4 orders of magnitude less computational time. In Fig. 7, note also that the heuristic execution time increases for increasing demand values due to a higher number of combinatorial possibilities to explore. In contrast, in ILP the number of decision variables and constraints remains the same regardless of the demand increase and, hence, there is no noticeable impact on the execution time as demands increase.

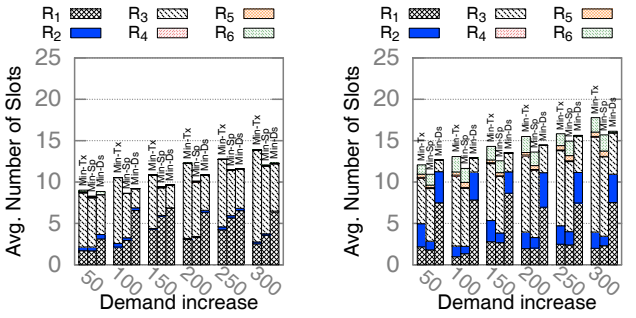
3) Distribution of actions taken by heuristic algorithm: Fig. 9 shows the distribution of actions adopted by the heuristic for two different initial VN embedding strategies. In particular, Fig. 8(a) shows the distribution of actions when initial embedding was done using the most optimized transmission configuration. As seen in Fig. 8(a), *Min-Ds* uses either R_1 or R_3 actions to keep the disruption at the lowest level. In contrast, the other objectives such as *Min-Tx* and *Min-Sp* use higher percentages of R_3 actions to achieve their corresponding objectives. To investigate the reason for lower amount of R_2 actions being taken, we have generated another set of snapshots in our simulator where the initial embedding was done using random transmission configurations. Fig. 8(b) shows that the percentages of R_2 actions taken by all the approaches significantly increase, with the largest increase for *Min-Ds*. This justifies our assumption that if the initial embedding was done with the best transmission configuration, reconfiguration is less likely to be done using R_2 as the virtual link is already using the best configuration considering the lightpath length. However, in situations when initial embedding was not done using the best configuration for some reason (e.g., to keep larger operational margin), our heuristic algorithm can leverage R_2 actions to accommodate new demand with low disruption as shown in Fig. 8(b). Since transponder usage and number of slots being disrupted for these snapshots have patterns similar to Fig. 4 and Fig. 6, respectively, we exclude discussing them for the sake of brevity.

4) Performance analysis by varying network load: To demonstrate how the proposed heuristic algorithm performs

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSAC.2021.3064643, IEEE Journal on Selected Areas in Communications

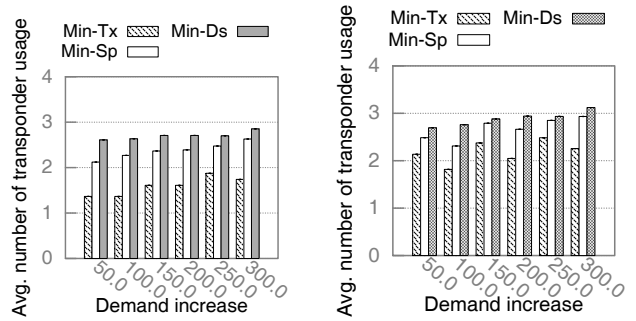
	Cost ratio between ILP and heuristic			ILP Execution Time in seconds			Heuristic Execution Time in seconds		
Demand Increase	Min-Tx	Min-Sp	Min-Ds	Min-Tx	Min-Sp	Min-Ds	Min-Tx	Min-Sp	Min-Ds
100	1.04	1.00	1.00	2552.03	2539.12	2554.60	0.50	0.53	0.50
200	1.04	1.00	1.00	2552.78	2527.52	2593.81	0.36	0.37	0.36
300	1.02	1.00	1.00	2607.62	2548.46	2565.40	0.63	0.62	0.62
400	1.00	1.02	1.08	2576.45	2538.42	2605.07	0.99	1.01	0.98
500	1.01	1.02	1.06	2577.54	2565.72	2588.04	1.37	1.36	1.37

Fig. 7: Benchmarking of the Heuristic Algorithm



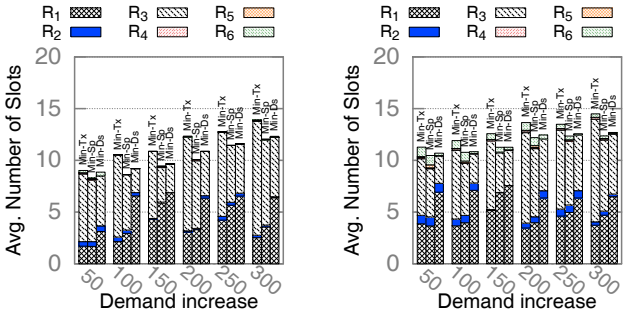
(a) Optimal Initial Embedding (b) Random Initial Embedding

Fig. 8: Slots using actions in Table 1



(a) 50% EON utilization (b) 60% EON utilization

Fig. 9: Transponder usage by varying network load



(a) 50% EON utilization (b) 60% EON utilization

Fig. 10: Slots using actions in Table I by varying network load

under different network loads (*e.g.*, 50% and 60% utilizations), we compare transponder usage and distribution of actions for snapshots with optimal initial embedding in Fig. 9 and Fig. 10, respectively. Fig. 9 shows that higher number of transponders are required to accommodate the same demand increase in an EON with 60% utilization than the one with 50% utilization. This is intuitive since a heavily loaded EON has more spectrum fragmentation in its links that urges to use more splits (*i.e.*, more transponders) to satisfy the same total bandwidth demand. Similarly, Fig. 9 shows that more spectrum slots are required to accommodate the same demand increase in a heavily loaded EON than in a less loaded EON. Again, this is a side-effect of spectrum fragmentation in a highly loaded EON that induces the heuristic algorithm to take longer paths to satisfy spectrum contiguity while serving the same demand increase. Not surprisingly, the heuristic algorithm uses a higher percentage of reconfiguration actions (*e.g.*, R_2 and R_6) that interfere with the existing splits of the virtual link. This is again due to lack of options for reconfiguration in a highly utilized EON that is constrained by additional load and more spectrum fragmentation.

VII. Conclusion

In this paper, we present a novel approach for scaling bandwidth demand of network slices on an EON-enabled 5G transport network. The bandwidth increase is accommodated by re-embedding a virtual link of the network slice while taking into account the corresponding disruption to the existing traffic. In contrast to the state-of-the-art, we do not assume the availability of advanced technologies for minimizing traffic disruption while accommodating the scaling request. Instead, we identify a comprehensive set of reconfiguration actions and propose a cost model that assigns varying costs to different reconfiguration actions according to the level of disruption they cause to the traffic. Our approach can incorporate different objectives such as minimizing the use of transponders, minimizing spectrum slice usage, minimizing disruption, or a combination thereof. We use our proposed cost model to develop an ILP formulation that can find near-optimal solution to the bandwidth scaling problem. We also present a heuristic algorithm that can solve scaling problem instances in three orders of magnitude less time than the ILP formulation.

Our extensive simulation using realistic network topologies and a discrete event simulator shed light on the trade-off

between minimizing disruption and minimizing resource usage in terms of transponder and spectrum slots while accommodating network slice scaling requests on an EON. Simulation results also show that solutions generated by our heuristic algorithm remain with 10% of the ILP-based solution, while executing several orders of magnitude faster than ILP. In the future, we plan to extend the work by incorporating scaling of multiple virtual links or change in the virtual network topology itself. Another interesting direction is to investigate other QoS parameters during scaling such as latency and reliability.

REFERENCES

[1] N. Shahriar, M. Zulfiqar, S. R. Chowdhury, S. Taeb, M. Tornatore, R. Boutaba, J. Mitra, and M. Hemmati, "Disruption-minimized re-adaptation of virtual links in elastic optical networks," in *Proceedings of OFC*, 2020, p. Th2A.30.

[2] S. Aleksic, "Towards fifth-generation (5g) optical transport networks," in *Proceedings of IEEE ICTON*, 2015, pp. 1–4.

[3] R. Boutaba, N. Shahriar, and S. Fathi, "Elastic optical networking for 5G transport," *Journal of Network and Systems Management*, vol. 25, no. 4, pp. 819–847, 2017.

[4] N. M. K. Chowdhury and R. Boutaba, "A survey of network virtualization," *Computer Networks*, vol. 54, no. 5, pp. 862–876, 2010.

[5] "An introduction to network slicing," Whitepaper, GSM Association, 2017, accessed: June 24, 2020. [Online]. Available: https://www.gsma.com/futurenetworks/wp-content/uploads/2017/11/GSMA-An-Introduction-to-Network-Slicing.pdf

[6] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, "Network slicing in 5G: Survey and challenges," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 94–100, 2017.

[7] O. Gerstel *et al.*, "Elastic optical networking: A new dawn for the optical layer?" *IEEE Communications Magazine*, vol. 50, no. 2, 2012.

[8] N. Shahriar, S. Taeb, S. R. Chowdhury, M. Tornatore, R. Boutaba, J. Mitra, and M. Hemmati, "Achieving a fully-flexible virtual network embedding in elastic optical networks," in *Proceedings of IEEE INFOCOM*, 2019, pp. 1756–1764.

[9] M. Hadi, M. R. Pakravan, and E. Agrell, "Dynamic resource allocation in metro elastic optical networks using Lyapunov drift optimization," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 11, no. 6, pp. 250–259, 2019.

[10] Z. Zhong, N. Hua, M. Tornatore, J. Li, Y. Li, X. Zheng, and B. Mukherjee, "Provisioning short-term traffic fluctuations in elastic optical networks," *IEEE/ACM Transactions on Networking*, vol. 27, no. 4, pp. 1460–1473, 2019.

[11] B. Yan, Y. Zhao, X. Yu, W. Wang, Y. Wu, Y. Wang, and J. Zhang, "Tidal-traffic-aware routing and spectrum allocation in elastic optical networks," *Journal of Optical Communications and Networking*, vol. 10, no. 11, pp. 832–842, 2018.

[12] "Network slicing use case requirements," Whitepaper, GSM Association, April 2018, accessed: June 24, 2020. [Online]. Available: https://www.gsma.com/futurenetworks/wp-content/uploads/2020/01/2.0_Network-Slicing-Use-Case-Requirements-1.pdf

[13] P. Soto, P. Maya, and J. F. Botero, "Resource allocation over eon-based infrastructures in a network virtualization environment," *IEEE Transactions on Network and Service Management*, vol. 16, no. 1, pp. 13–26, 2018.

[14] A. Pagès, J. Perelló, S. Spadaro, and J. Comellas, "Optimal route, spectrum, and modulation level assignment in split-spectrum-enabled dynamic elastic optical networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 6, no. 2, pp. 114–126, 2014.

[15] X. Chen, Y. Zhong, and A. Jukan, "Multipath routing in elastic optical networks with distance-adaptive modulation formats," in *2013 IEEE International Conference on Communications (ICC)*. IEEE, 2013, pp. 3915–3920.

[16] B. C. Chatterjee *et al.*, "Routing and spectrum allocation in elastic optical networks: A tutorial," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1776–1800, 2015.

[17] S. Spadaro, J. Perelló Muntan *et al.*, "Virtual network provisioning over flexible optical transport infrastructure," 2014.

[18] V. Eramo and F. G. Lavacca, "Proposal and Investigation of a Reconfiguration Cost Aware Policy for Resource Allocation in Multi-Provider NFV Infrastructures Interconnected by Elastic Optical Networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 37, no. 16, pp. 4098–4114, 2019.

[19] T. Takagi, H. Hasegawa, K.-i. Sato, Y. Sone, A. Hirano, and M. Jinno, "Disruption minimized spectrum defragmentation in elastic optical path networks that adopt distance adaptive modulation," in *Proceedings of ECOC*, 2011, pp. Mo–2.

[20] K. Christodoulopoulos, I. Tomkos, and E. Varvarigos, "Time-varying spectrum allocation policies and blocking analysis in flexible optical networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 1, pp. 13–25, 2012.

[21] F. Cugini, F. Paolucci, G. Meloni, G. Berrettini, M. Secondini, F. Fresi, N. Sambo, L. Poti, and P. Castoldi, "Push-pull defragmentation without traffic disruption in flexible grid optical networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 31, no. 1, pp. 125–133, 2012.

[22] R. Maher, D. S. Millar, S. J. Savory, and B. C. Thomsen, "Widely tunable burst mode digital coherent receiver with fast reconfiguration time for 112 gb/s dp-qpsk wdm networks," *IEEE/OSA Journal of lightwave technology*, vol. 30, no. 24, pp. 3924–3930, 2012.

[23] 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," *OSA Optics express*, vol. 20, no. 24, pp. 26 958–26 968, 2012.

[24] G. Shen, Y. Zhang, X. Zhou, Y. Sheng, N. Deng, Y. Ma, and A. Lord, "Ultra-dense wavelength switched network: A special eon paradigm for metro optical networks," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 189–195, 2018.

[25] L. Mesquita, K. Assis, A. Santos, M. Alencar, and R. Almeida, "A routing and spectrum assignment heuristic for elastic optical networks under incremental traffic," in *2018 SBFoton International Optics and Photonics Conference (SBFoton IOPC)*. IEEE, 2018, pp. 1–5.

[26] T. Panayiotou, K. Manousakis, S. P. Chatzis, and G. Ellinas, "A data-driven bandwidth allocation framework with qos considerations for eons," *Journal of Lightwave Technology*, vol. 37, no. 9, pp. 1853–1864, 2019.

[27] 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 Journal on Selected Areas in Communications*, vol. 31, no. 1, pp. 26–38, 2012.

[28] N. Sambo, K. Christodoulopoulos, N. Argyris, P. Giardina, C. Delezoide, D. Roccato, A. Percelsi, R. Morro, A. Sgambelluri, A. Kretsis, G. Kanakis, G. Bernini, E. Varvarigos, and P. Castoldi, "Field trial: Demonstrating automatic reconfiguration of optical networks based on finite state machine," *IEEE/OSA Journal of Lightwave Technology*, 2019.

[29] R. Singh, M. Ghobadi, K.-T. Foerster, M. Filer, and P. Gill, "Radwan: rate adaptive wide area network," in *Proceedings of ACM SIGCOMM Conference*, 2018, pp. 547–560.

[30] D.-R. Din, Y.-F. Wu, B.-J. Guo, C. Chen, and P.-J. Wu, "Spectrum expansion/contraction problem for multipath routing with time-varying traffic on elastic optical networks," in *Proceedings of the Second International Conference on Internet of things, Data and Cloud Computing*, 2017, pp. 1–8.

[31] S. Ayoubi, N. Limam, M. A. Salahuddin, N. Shahriar, R. Boutaba, F. Estrada-Solano, and O. M. Caicedo, "Machine learning for cognitive network management," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 158–165, 2018.

[32] N. Shahriar, S. Taeb, S. R. Chowdhury, M. Zulfiqar, M. Tornatore, R. Boutaba, J. Mitra, and M. Hemmati, "Reliable slicing of 5G transport networks with bandwidth squeezing and multi-path provisioning," *IEEE Transactions on Network and Service Management*, vol. 17, no. 3, pp. 1418–1431, 2020.

[33] S. Taeb, N. Shahriar, S. R. Chowdhury, M. Tornatore, R. Boutaba, J. Mitra, and M. Hemmati, "Virtual network embedding with path-based latency guarantees in elastic optical networks," in *27th International Conference on Network Protocols (ICNP)*. IEEE, 2019, pp. 1–12.

[34] N. Shahriar, S. Taeb, S. R. Chowdhury, M. Zulfiqar, M. Tornatore, R. Boutaba, J. Mitra, and M. Hemmati, "Reliable slicing of 5g transport networks with dedicated protection," in *2019 15th International Conference on Network and Service Management (CNSM)*. IEEE, 2019, pp. 1–9.

Nashid Shahriar is an assistant professor in the Department of Computer Science at the University of Regina. He received his Ph.D. degree from the School of Computer Science, University of Waterloo in 2020. He received his M.Sc. and B.Sc. degrees in Computer Science and Engineering from Bangladesh University of Engineering and Technology (BUET) in 2011 and 2009, respectively. He was a recipient of Ontario Graduate Scholarship, President's Graduate Scholarship, and David R. Cheriton Graduate Scholarship with the University of Waterloo. He received several recognitions, including the 2020 PhD Alumni Gold Medal from the University of Waterloo, the IEEE/ACM/IFIP CNSM 2019 Best Paper Award, IEEE NetSoft 2019 Best Student Paper Award, and the IEEE/ACM/IFIP CNSM 2017 Best Paper Award. His research interests include network virtualization, 5G network slicing, and network reliability.

Mubeen Zulfiqar is pursuing his doctoral degree at the David R. Cheriton School of Computer Science, University of Waterloo. His research interest include network virtualization and network security.

Shihabur Rahman Chowdhury (S'13) received is a PhD candidate at the David R. Cheriton School of Computer Science, University of Waterloo. He received his B.Sc. degree in computer science and engineering from BUET in 2009. His research interests include virtualization and softwarization of computer networks. He is co-recipient of the Best Paper Award at the IEEE/ACM/IFIP CNSM 2019, IEEE NetSoft 2019, and the IEEE/ACM/IFIP CNSM 2017 conference.

Sepehr Taeb is a research assistant at the David R. Cheriton School of Computer Science, University of Waterloo. He received his B.Sc. in computer engineering from Sharif University of Technology in 2016, and the M.Math. degree in computer science from the University of Waterloo in 2019. His research interest includes network virtualization, optical networks, and Internet of drones.

Raouf Boutaba (F'12) M.Sc. and Ph.D. degrees in computer science from Sorbonne University in 1990 and 1994, respectively. He is currently a University Chair Professor and the Director of the David R. Cheriton School of Computer science at the University of Waterloo (Canada). He also holds an INRIA International Chair in France. He is the founding Editor-in-Chief of the IEEE Transactions on Network and Service Management (2007- 2010) and the current Editor-in-Chief of the IEEE Journal on Selected Areas in Communications. He is a fellow of the IEEE, the Engineering Institute of Canada, the Canadian Academy of Engineering, and the Royal Society of Canada. His research interests include resource and service management in networks and distributed systems.

Jeebak Mitra received the M.A.Sc. and Ph.D. degrees in electrical engineering from The University of British Columbia in 2005 and 2010, respectively. From 2010 to 2011, he was a Senior System Engineer with Riot Micro, leading the system level design for a local thermal equilibrium baseband. From 2011 to 2012, he was a Team Leader for physical layer DSP design with BLINQ Networks, Ottawa, focusing on small cell backhaul products. Since 2013, he has been a Senior Staff Engineer with the Huawei Technologies Canada Research Center, Ottawa, in the areas of algorithm design and implementation for coherent high-speed optical transceivers and flexible optical networks. His research interests lie in the area of high-performance communication systems design focusing on optical and wireless networks. He received the Best Student Paper Award at the IEEE Canadian Conference in Electrical and Computer Engineering 2009. He was a co-recipient of the Best Paper Award at IEEE/ACM/IFIP CNSM 2017 and 2019.

Mahdi Hemmati (M) is a Senior Research Engineer at Huawei Technologies Canada Research Center, Ottawa. He received the PhD degree in Electrical and Computer Engineering from the University of Ottawa in 2017. He received the NSERC Postgraduate Scholarship during his doctoral studies. Hemmati received his MSc degree in Electrical Engineering - Systems & Control from Sharif University of Technology, Iran. His research interests include distributed control and optimization, fair resource allocation, and intelligent network automation. He was a co-recipient of the Best Paper Award at IEEE/ACM/IFIP CNSM 2019.