Abstract—This letter studies how to optimize the availability-aware service provisioning (AaSP) with failure-independent path-protecting pre-configured cycles (FIPP-$p$-cycles) in elastic optical networks (EONs). We first propose a novel AaSP-FIPP scheme by incorporating the bandwidth-squeezed restoration, and develop a mathematical model to analyze the service availability of such scheme theoretically. Then, to improve the scheme’s scalability, we design and analyze a topology partitioning method to assist the design of AaSP with FIPP-$p$-cycle protection. The proposed algorithms are evaluated with simulations that consider both offline planning and online provisioning, and the results confirm that our algorithms can facilitate more efficient spectrum usage and improve the clients’ satisfaction ratios on availability.

Index Terms—Elastic optical networks (EONs), Availability-aware service provisioning (AaSP), Failure-independent path-protecting pre-configured cycle (FIPP-$p$-cycle).

I. INTRODUCTION

It is known that flexible-grid elastic optical networks (EONs) use narrow-band spectrally-contiguous frequency slots (FS’s) to achieve high spectral efficiency and adaptive bandwidth allocation in the optical layer [1–4]. As maintaining network survivability is crucial in optical networks, previous studies have considered both path- and link-based protection schemes to deal with the link failures in EONs [5–9]. However, these schemes suffer from either long recovery latency or low resource efficiency. In this context, the failure-independent path-protecting pre-configured cycle (FIPP-$p$-cycle), which can integrate the advantages of path- and link-based protection schemes (i.e., fast restoration speed and high resource efficiency, respectively), has been put forward in [10] for realizing survivable EONs. Note that, in practical network operations, network survivability is usually quantified with service availability, which is defined as the ratio of service-on time to total provisioning period. Service availability is usually specified explicitly in the service-level agreement (SLA) between a client and its network operator [8]. Hence, a more practical angle to study survivable EONs is to consider availability-aware service provisioning (AaSP), i.e., to satisfy the clients’ availability requirements with the minimum spectrum usage.

Perviously, people have studied how to realize AaSP in fixed-grid wavelength-division multiplexing (WDM) networks in [11], and proposed effective algorithms. Nevertheless, because the spectrum allocation schemes in WDM networks and EONs are fundamentally different in a few aspects, we still need to revisit this problem for EONs. For instance, with the flexible spectrum allocation in EONs, one can leverage bandwidth-squeezed restoration to further improve the efficiency of AaSP [9], which is not feasible in WDM networks.

In this letter, we study how to optimize the scheme of AaSP with FIPP-$p$-cycle protection (AaSP-FIPP) in EONs for enhanced resource efficiency. We first propose a novel AaSP-FIPP scheme by incorporating bandwidth-squeezed restoration [12], and develop a mathematical model to analyze the service availability of such scheme theoretically. Then, to make the AaSP-FIPP scheme more scalable, we design a topology partitioning method that divides an EON into smaller domains and protect lightpaths with FIPP-$p$-cycles in multiple domains. Numerical simulations consider both offline planning and online provisioning with AaSP-FIPP, and the results confirm that our proposed schemes can make the spectrum usage more efficient and provide higher satisfaction ratio on availability.

The rest of the paper is organized as follows. Section II describes the working principle of AaSP-FIPP in EONs. Then, in Section III, we propose a time-efficient topology partitioning algorithm to assist the design of AaSP-FIPP. The algorithms are evaluated with extensive simulations in Section IV. Finally, Section V summarizes this paper.

II. AASP-FIPP in EONs

We model the topology of an EON as $G(V, E)$, where $V$ represents the set of nodes and $E$ is the link set. A lightpath request is denoted as $LR(s, d, B, A, T)$, where $s, d \in V$ are the source and destination nodes, its bandwidth requirement is $B$ Gb/s, $A$ is the availability requirement from SLA, and $T$ is its service duration. Then, with $B$, we can derive the number of FS’s to be allocated based on the quality-of-transmission of $LR$’s working path [9]. Next, to satisfy $A$, AaSP-FIPP configures one or more FIPP-$p$-cycles for $LR$ if necessary.

Fig. 1 shows an intuitive example for AaSP-FIPP in EONs. Basically, a working path can be protected by an FIPP-$p$-cycle, if the $p$-cycle includes both of its end-nodes and can provide a backup path that is link-disjoint with it. Meanwhile,
we incorporate the shared protection scheme in the FIPP-p-cycle design, allowing two lightpaths to be protected by the same backup FS’ allocated on a p-cycle when their working or backup paths are link-disjoint. Therefore, the p-cycle 1→2→3→6→5→4→1 in Fig. 1(a) can protect the working paths of the three requests, i.e., LR_1, LR_2 and LR_3 share the backup FS’ reserved on the p-cycle with the scheme depicted in Fig. 1(b). Furthermore, we can leverage the bandwidth-squeezed restoration technique to make the AaSP-FIPP in EONs more resource efficient. Specifically, for LR, the bandwidth allocated during restoration (i.e., denoted as B’) can be smaller than B [12], while the minimum amount of backup bandwidth that is needed to recover the service of LR (also derived from SLA) is assumed as B_m, i.e., B’ ∈ [B_m, B]. In such a situation, the acquired availability during restoration (i.e., the availability corresponds to this specific failure restoration scenario) is A’ = \frac{B’}{B} [9]. For example, in Fig. 1(c), we can allocate 6 and 5 FS’ (including 1 guard-band FS) to restore the services of LR_1 and LR_2, respectively, when their working paths fail simultaneously. Consequently, the acquired availabilities of the lightpaths are A_1’ = \frac{6}{10} = 0.5 and A_2’ = \frac{5}{10} = 0.5.

Note that, to facilitate the design of AaSP-FIPP, we need to analyze the service availability of each request precisely. Hence, we develop a theoretical model. Firstly, it is easy to obtain the availability of an unprotected LR as \rho^{H_w}, where \rho is the link availability (assumed to be identical for every link in the EON) and H_w is the hop-count. For an LR that is protected by FIPP-p-cycles, we can get its availability by enumerating the situations in which its service is available: 1) its working path is intact, and 2) its working path is broken but its backup path provided by FIPP-p-cycle(s) is available with sufficient bandwidth to ensure a successful recovery (i.e., B’ ∈ [B_m, B]). Specifically, its availability is [13]

\begin{equation}
A_L = \rho^{H_w} \left\{ \left(1 + H_w(1 - \rho)\rho^{H_w-1}\sum_{c \in C} \rho|\mathbb{L}|^{-1}(1 - \rho) \left(\frac{1}{2}A_0' + \frac{1}{2}A_c' \right) + \frac{1}{2}(H_w - 1)(1 - \rho)\rho|\mathbb{L}|^{-1}A_0' \right) \right\}, \tag{1}
\end{equation}

where \mathbb{L} denotes the set of the links on the working paths of other lightpaths, which share backup FS’ with LR, H_w is the hop-count of LR’s backup path, and A_0’ and A_c’ are the acquired availabilities when LR is restored with full or partial working bandwidth, respectively. Note that, the derivation of Eq. (1) ignores the situations in which there are more than two simultaneous link failures, and this is because their probability is so small (e.g., in the magnitude of 10^{-6} if \rho = 0.99) that their contributions to the overall availability are negligible.

Then, we design an AaSP-FIPP algorithm that determines the protection scheme of each request based on the spectral efficiency (SE) of FIPP-p-cycles, namely, AaSP-SE-FIPP, whose procedure is shown in Algorithm 1. Note that, in Line 6, the minimum number of backup FS’ N_c refers to the FS’ that need to be reserved specifically for LR, while those that can be shared with other in-service requests are not included. In Line 9, if no feasible p-cycle can be found, we still provision LR with the working path but mark it as availability unsatisfied.

\begin{algorithm}
1. precalculate C as a set of candidate FIPP-p-cycles in the EON;
2. for each LR(s, d, B, A, T) do
   3. obtain the working path P_w of LR as the shortest available one;
   4. if A cannot be satisfied with P_w, then
      5. for each candidate p-cycle c ∈ C that can protect P_w do
         6. calculate N_c as the minimum number of FS’ to be reserved on c for LR to satisfy A while the availabilities of all the other in-service requests are still satisfactory;
         7. set the spectral efficiency of c as SE(c) = \frac{\rho}{N_c};
      8. end
   9. select p-cycle c’ = \arg\min\{SE(c)|\};
   10. assign N_c’ FS’ on c’ to protect P_w;
12. end
\end{algorithm}

\section{III. AaSP-FIPP with Topology Partitioning}

Although AaSP-SE-FIPP can improve the spectral efficiency of FIPP-p-cycle protection, its time complexity is relatively high. This is because Algorithm 1 needs to check all the available FS’ on all the feasible FIPP-p-cycles to determine LR’s protection scheme. In other words, the complexity of the for-loop that covers Lines 5-8 is \mathcal{O}(F \cdot |C|), where F represents the total number of FS’ that a link can accommodate. However, in a relatively large EON topology, |C| can easily be thousands or more. Hence, we try to leverage the topology partitioning, which is to divide the topology into a few protection domains and apply AaSP-SE-FIPP to each of them, to improve the time-efficiency of AaSP-FIPP.

Fig. 2 shows an example for AaSP-FIPP with topology partitioning. Here, we calculate the availability of an LR that traverses multiple domains by considering both intra- and inter-domain cases, i.e., link failures happen in single or multiple domains. While the availability associated with the intra-domain case can be obtained with Eq. (1), we analyze the availability of the inter-domain case by considering the two scenarios in Figs. 2(a) and 2(b). Here, we still only consider the situations with two or less simultaneous link failures. Fig. 2(a) shows the scenario in which dual failures happen on

\begin{figure}
\centering
\includegraphics[width=\textwidth]{fig2.png}
\caption{Examples on AaSP-FIPP with topology partitioning that can restore dual failures on (a) the working path, and (b) the working path and a common link of two domains.}
\end{figure}

\begin{figure}
\centering
\includegraphics[width=\textwidth]{fig3.png}
\caption{An example of cyclic partition.}
\end{figure}
LR’s working path, which is restored with the p-cycles in two domains independently. The scenario in Fig. 2(b) is more complicated as it involves a failure on the common link of two domains, and thus the domains need to work cooperatively to determine the backup path segments (i.e., \(1 \rightarrow 5 \rightarrow 4\) and \(4 \rightarrow 8 \rightarrow 7\)). Then, the availability of the inter-domain case is

\[
A_I = (1 - \rho)^2 \rho \left( \sum_{v \in V} \frac{1}{|L^w_v|} \right) \left( \sum_{v \in V} \frac{1}{|L^w_v|} \right) \left( \sum_{u \in V} \frac{1}{|L^w_u|} \right) \left( \sum_{u \in V} \frac{1}{|L^w_u|} \right)
\]

where \(D\) is the set of domains in the EON, \(L^w_i\) and \(L^p_i\) are the sets of links in domain \(D_i \in D\), which are on LR’s working and backup paths, respectively, and \(L^p_i\) denotes the set of links on the backup path determined by the scenario shown in Fig. 2(b). Finally, we obtain the overall availability of LR as

\[
A_L = \rho \left( \sum_{v \in V} \frac{1}{|L^w_v|} \right) \left( \sum_{v \in V} \frac{1}{|L^w_v|} \right) \left( \sum_{u \in V} \frac{1}{|L^w_u|} \right) \left( \sum_{u \in V} \frac{1}{|L^w_u|} \right) + A_I,
\]

where \(A_{L,i}\) is the intra-domain availability in domain \(D_i\).

Algorithm 2 shows the procedure of CP. The while-loop that covers Lines 2-21 divides the topology into several cyclic-type domains. Line 3 selects a node \(v \in V\) with the highest node degree as the center of a domain since this can potentially include more nodes in the cyclic-type domain. Then, Lines 4-14 find all the adjacent nodes of \(v\) and connect them sequentially with shortest paths, to form a path \(P\). Next, we use Lines 15-16 to check whether a cyclic-type domain can be formed. Specifically, we try to find a new shorter path \(P'\) to connect the end-nodes of \(P\), and if \(P'\) exists and the number of nodes in \(P' \cup P\) plus 1 (node \(v\)) does not exceed \(\chi\) (restriction on the size of each domain as defined in Line 1), a new cyclic-type domain \(D_i\) can be formed in Line 17. When all the feasible cyclic-type domains have been formed, either we have an empty set \(V\) or all the remaining nodes in \(V\) have been checked. Then, if \(V \neq \emptyset\), Lines 22-25 form the rest of the non-cyclic domains.

Fig. 3 shows an example of CP. We first select \(v\) as Node 3 and obtain its adjacent node set \(V_1 = \{2, 4, 6\}\). Assume Node 2 is selected as the first \(u\), we calculate paths 2-6 and 6-7-4 in sequence to form \(P\) as 2-6-7-4 according to Lines 8-14 of Algorithm 2. Then, as the end-nodes of \(P\) (i.e., Nodes 2 and 4) can be connected with a new shorter path 2-1-4-4, we can obtain a cyclic-type domain as Domain 1 in Fig. 3. Next, we repeat the same procedure with Node 9 to get Domain II. Finally, since no more cyclic-type domains can be formed, we calculate \(C_p\) containing cycles 6-7-11-10 and 10-11-13-12 and merge them to form Domain III.

### IV. PERFORMANCE EVALUATION

The performance of the proposed AaSP-FIPP algorithm (denoted as AaSP-CP-FIPP) is evaluated with simulations using the US Backbone topology in [8]. We assume that each fiber link accommodates \(F = 358\) FS’, each of which occupies a bandwidth of 12.5 GHz [14]. The availability of each link is set as \(\rho = 0.999\) [8, 11]. The lightpath requests are generated with bandwidth requirements evenly distributed within \([25, 250]\) GHz, availability requirements evenly distributed within \([0.970, 0.999]\), and their minimum restoration ratios (i.e., \(\frac{\chi}{B_{ECC}}\)) are randomly selected within \([0.5, 0.9]\). Regarding the baseline algorithms for performance comparisons, we use the PE-FIPP algorithm in [10], the AaSP-TP-FIPP algorithm in [13] and the dedicated path protection based AaSP algorithm (AaSP-DPP, which applies the AaSP principle in Section II but configures protection resources according to DPP).

We first consider the offline planning in which all the requests are known and served simultaneously. For AaSP-CP-FIPP, we investigate the trade-off in the number of partitioned domains by restricting the maximum number of nodes in each domain to be 5, 7 and 12, resulting in the partitioning results containing 13, 7 and 5 domains respectively. Fig. 4(a) shows the results on spectrum utilization, which indicate that AaSP-CP-FIPP can improve the spectral efficiency of the service provisioning effectively compared with the baseline algorithms. Meanwhile, we observe that the performance from AaSP-CP-FIPP improves with the number of partitioned domains. This is because by partitioning the topology into more but smaller domains, we can avoid configuring relatively long backup paths and make the FIPP-p-cycle more flexible, i.e., being able to design the protection structures within each small domain independently based on the actual service availability requirements from requests. Table I summarizes the results on average availability satisfaction ratio when the number of requests is

2We set the maximum availability requirement being 0.999 as with the setup of \(\rho\), the simulations indicate the maximum achievable availability from the algorithms is 0.99987.
200. Consistently with the observations from the results in Fig. 4(a), AaSP-CP-FIPP can significantly improve the percentage of LRs whose availability requirements get satisfied with FIPP-p-cycle protection, especially when more domains are obtained. On the other hand, we should notice that having more partitioned domains also increases the cost of transponder usage as we need to reserve an additional transponder on each FIPP-p-cycle configured for a lightpath. Specifically, simulation results indicate that the average numbers of FIPP-p-cycles configured for each lightpath are 6.9, 3.7 and 2.1 when we obtain 13, 7 and 5 domains respectively. Therefore, network designers should carefully address these trade-offs according to their performance targets and budgets. Fig. 4(b) shows the results on the running time of the algorithms, confirming that the proposed topology partitioning mechanisms reduce the time-complexity effectively. The running time from AaSP-SE-FIPP decreases with the number of requests due to the fact that fewer FS-blocks need to be inspected for each request when the network gets more saturated.

We then simulate the scenario of online provisioning. Specifically, the dynamic lightpath requests are generated according to the Poisson traffic model, and we assume that they can come and leave on-the-fly. Here, we only compare AaSP-DPP, AaSP-TP-FIPP and AaSP-CP-FIPP, since the results of offline planning have already shown that AaSP-SE-FIPP and PE-FIPP perform significantly worse than AaSP-CP-FIPP. Table I presents the results on availability satisfaction ratio from online simulations when the traffic load is 330 Erlangs. It is interesting to notice that the availability satisfaction ratio from AaSP-DPP drops sharply to only 80.5% while the performance of the other algorithms maintain relatively stable. The rationale behind this can be explained by the results on blocking probability in Table II, where we can see that AaSP-

**TABLE I**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>PE-FIPP</td>
<td>93.1</td>
<td>-</td>
<td>AaSP-CP-FIPP-5</td>
<td>98.3</td>
<td>97.9</td>
</tr>
<tr>
<td>AaSP-SE-FIPP</td>
<td>96.1</td>
<td>97.0</td>
<td>AaSP-CP-FIPP-7</td>
<td>98.8</td>
<td>97.6</td>
</tr>
<tr>
<td>AaSP-TP-FIPP</td>
<td>96.3</td>
<td>97.0</td>
<td>AaSP-CP-FIPP-13</td>
<td>99.0</td>
<td>99.1</td>
</tr>
<tr>
<td>AaSP-DPP</td>
<td>96.1</td>
<td>80.5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TABLE II**

<table>
<thead>
<tr>
<th>Traffic Load (Erlangs)</th>
<th>Algorithms</th>
<th>130</th>
<th>170</th>
<th>210</th>
<th>250</th>
<th>290</th>
<th>330</th>
</tr>
</thead>
<tbody>
<tr>
<td>AaSP-DPP</td>
<td>2.47</td>
<td>10.50</td>
<td>13.10</td>
<td>18.80</td>
<td>23.83</td>
<td>25.17</td>
<td></td>
</tr>
<tr>
<td>AaSP-TP-FIPP</td>
<td>0.38</td>
<td>0.70</td>
<td>1.20</td>
<td>1.98</td>
<td>2.67</td>
<td>2.95</td>
<td></td>
</tr>
<tr>
<td>AaSP-CP-FIPP</td>
<td>0.02</td>
<td>0.03</td>
<td>0.14</td>
<td>0.67</td>
<td>0.90</td>
<td>0.98</td>
<td></td>
</tr>
<tr>
<td>AaSP-CP-FIPP-7</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.03</td>
<td>0.12</td>
<td>0.18</td>
<td></td>
</tr>
<tr>
<td>AaSP-CP-FIPP-13</td>
<td>0</td>
<td>0</td>
<td>0.08</td>
<td>0.12</td>
<td>0.33</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

DPP rejects ~25% requests at the highest traffic load. This implies that AaSP-DPP has exhausted the spectra in the EON, making it difficult to find sufficient spectra for satisfying the availability requirements from future requests. Again, AaSP-CP-FIPP-13 performs the best among all the algorithms.

**V. CONCLUSION**

In this letter, we studied how to optimize the scheme of AaSP-FIPP in EONs for enhanced resource efficiency. We first proposed a novel AaSP-FIPP scheme by incorporating the bandwidth-squeezed restoration, and developed a theoretical model to analyze the service availability of such scheme. Then, to make the scheme more scalable, we designed and analyzed a topology partitioning method to assist AaSP-FIPP. Simulation results verified that our algorithm can make the spectrum usage of AaSP-FIPP much more efficient while the clients’ satisfaction ratios on availability are improved as well.

**ACKNOWLEDGMENTS**

This work was supported in part by the NSFC Project 61371117, the SPR Program of the CAS (XDA06011202), the Key Project of the CAS (QYZDY-SSW-JSC003), and the NGBWMCN Key Project (2017ZX03001019-004).

**REFERENCES**


