Dynamic p-Cycle Configuration in Spectrum-Sliced Elastic Optical Networks

Fan Ji[1], Xiaoliang Chen[1], Wei Lu[1], Joel J. P. C. Rodrigues[2], Zuqing Zhu[1,†]
[1]School of Information Science and Technology, University of Science and Technology of China, Hefei, China
[2]Instituto de Telecommunicações, University of Beira Interior, Portugal
[†]Email: {zqzhu}@ieee.org

***Abstract*—We propose three algorithms for dynamic pre-configured-cycle (*p*-cycle) configuration in the spectrum-sliced elastic optical networks (EONs) based on the optical orthogonal frequency-division multiplexing (O-OFDM) technology, aiming to provide 100% restoration against single-link failures. The first one (PE-*p*-cycle) configures the working path and *p*-cycles of a request together according to the protection efficiencies of the *p*-cycles. In order to reduce request blocking probability, we then propose to use the spectrum planning technique to regulate the spectra of working and protection resources and design two algorithms based on the protected working capacity envelope (PWCE) cycles (PWCE-*p*-cycle-SP) and Hamiltonian cycles (Ham-*p*-cycle-SP). The three algorithms are then evaluated in two mesh network topologies, using dynamic Poisson traffic. The simulation results indicate that with the spectrum planning, PWCE-*p*-cycle-SP and Ham-*p*-cycle-SP reduce the blocking probability effectively, when comparing with the one without it, i.e., PE-*p*-cycle. To the best of our knowledge, this is the first attempt to address dynamic *p*-cycle configuration in EONs.**

***Index Terms*—Optical orthogonal frequency-division multiplexing (O-OFDM), Elastic optical networks (EONs), Pre-configured cycle (*p*-cycle), Hamiltonian cycle.**

I. Introduction

Recently, the optical orthogonal frequency-division multiplexing (O-OFDM) technology [1] has attracted intensive research interests because it can achieve high bandwidth efficiency and flexible resource allocation in the optical layer. Different from their wavelength-division multiplexing (WDM) counterparts that operate on discrete wavelength channels, O-OFDM transponders groom the capacities of several narrow-band subcarrier channels (frequency slots) that are contiguous in the spectrum domain and achieve high-speed data transmission over them. Therefore, a network operator can tailor bandwidth allocation with a fine granularity ($\sim$ 12.5 GHz) by adjusting the number of assigned frequency slots (FS's) [2]. With these advantages in mind, we refer the flexible-grid optical networks based on the O-OFDM technology as spectrum-sliced elastic optical networks (EONs) [3].

In order to realize efficient planning and provisioning of EONs, previous works have proposed numerous routing and spectrum assignment (RSA) algorithms [4–9]. However, most of them only addressed how to setup lightpath connections without protection. In EONs, a link failure may lead to severe service disruptions due to the high data-rate. Therefore, it is desired to implement protection schemes in EONs to improve their network survivability. Previously, Sone *et al.* proposed a path protection scheme for EONs based on the bandwidth squeezed restoration technique [10]. A shared backup path protection (SBPP) scheme was investigated in [11] to improve the protection efficiency in EONs. These path-based protection schemes allocate two disjoint lightpaths to each connection as its working and protection paths [12]. Even though path-based protection schemes can achieve high protection efficiency, it suffers from long restoration time as the backup resources can only be reserved but not pre-configured [12]. For speeding up lightpath restoration, researchers have investigated link-based protection schemes and proposed to use pre-configured-cycles (p-cycles) for network resilience in the WDM networks [13–18]. In these schemes, a p-cycle is pre-configured to protect a working link, and when the link fails, only its two end-nodes engage in the restoration, resulting in a short recovery time. The p-cycle configurations in EONs are still under-explored, especially for dynamic network operations where we need to handle setting-up and tearing-down of p-cycles dynamically.

In this paper, we aim to provide 100% restoration against single-link failures, and propose three algorithms for dynamic p-cycle configuration in EONs. The first one configures the working path and p-cycles of a request together according to the protection efficiencies of the p-cycles. In order to reduce request blocking probability, we then propose to use the spectrum planning technique to regulate the spectra of working and protection resources and design two dynamic p-cycle configuration algorithms based on the protected working capacity envelope (PWCE) cycles and Hamiltonian cycles. To the best of our knowledge, this is the first attempt to address dynamic p-cycle configuration in EONs.

The rest of the paper is organized as follows. Section II describes the working principle of dynamic p-cycle configuration in EONs. In Section III, we provide the detailed procedures of the three proposed algorithms. The simulation results are shown in Section IV. We discuss the proposed algorithms and forecast the next-step research directions in Section V. Finally, Section VI summarizes the paper.

II. Dynamic p-Cycle Configuration in EONs

In this section, we explain the working principle of dynamic p-cycle configuration in EONs. We denote the physical topology of an EON as $G(V, E)$, which is a directed graph with V and E as the sets of nodes and fiber links, respectively. On each link $e \in E$, we have F frequency slots (FS's) to allocate.

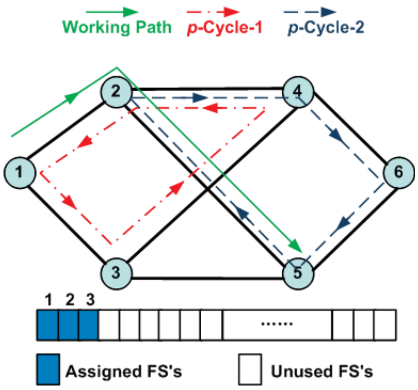


Fig. 1. An example of same-spectrum *p*-cycle protection in EONs.

We define a dynamic connection request as $LR(t, \tau, s, d, n)$, where t is the arrival time, τ is the holding period, $s, d \in V$ are the source and destination nodes, and n is the number of FS's it requires (including the guard-band). In order to serve $LR(t, \tau, s, d, n)$, we calculate a working path from s to d (i.e., $\mathcal{R}_{s,d}$) and assign n contiguous FS's along it. Since we aim to provide 100% restoration against single-link failures, a *p*-cycle $\mathcal{C}_e$ is also configured for each link $e \in \mathcal{R}_{s,d}$ and n FS's are reserved on each $\mathcal{C}_e$ as the protection resources. Note that we assume there are no spectrum converters in the network. Therefore, in order to achieve seamless restoration, we have to make sure that the FS's assigned to the working path are the same as those reserved on all of its *p*-cycles. We refer this scheme as the "same-spectrum" *p*-cycle protection. The start and end indices of the assigned FS's are defined as f_s and f_e,

$$f_e - f_s = n - 1, \quad f_e, f_s \in [1, F] \tag{1}$$

If there are insufficient bandwidth resources to set up both the working path and the associated *p*-cycles of a request at time t, the request is considered as blocked.

Fig. 1 shows an example of the same-spectrum *p*-cycle protection scheme. For the request $LR(t, \tau, 1, 5, 3)$, we determine its working path as 1→2→5 and assign 3 contiguous FS's on the path (e.g., $f_s = 1$ and $f_e = 3$). For link 1→2, we configure the *p*-cycle as 1→3→4→2→1, while the *p*-cycle for link 2→5 is 2→4→6→5→2. In order to achieve the same-spectrum protection, we reserve the same FS's on these two *p*-cycles with $f_s = 1$ and $f_e = 3$. When link 1→2 fails, the request is routed as 1→3→4→2→5 after restoration. Similarly, the working resources on link 2→5 are protected by the 3 FS's on *p*-cycle 2→4→6→5→2.

III. *p*-Cycle Configuration Algorithms in EONs

In this section, we propose three algorithms for dynamic *p*-cycle configuration in EONs. The first algorithm configures the *p*-cycles according to their protection efficiencies, and we call it as protection efficiency based *p*-cycle (PE-*p*-cycle). The other two algorithms incorporate a spectrum planning procedure to regulate the spectra of the working and protection resources. We name them as protected working capacity envelope *p*-cycle with spectrum planning (PWCE-*p*-cycle-SP) and Hamiltonian *p*-cycle with spectrum planning (Ham-*p*-cycle-SP).

A. *PE-*p*-cycle Algorithm*

In PE-*p*-cycle, we calculate a large set of cycles in the network topology $G(V, E)$ and store them in $\mathbb{C} = \{\mathcal{C}_k\}$, where k is the unique index of a cycle. In order to model link spectrum utilization, we define a bit-mask $b_e[1 \cdots F]$ that contains F bits for each link $e \in E$. We set $b_e[j] = 0$, if the j-th FS on link e is unavailable (i.e., already used for working path or reserved as protection resource), otherwise, $b_e[j] = 1$. Then, by multiplying the bit-masks of all its links, we can get the spectrum utilization of a path, e.g.,

$$b_{\mathcal{R}_{s,d}} = \prod_{e \in \mathcal{R}_{s,d}} b_e \tag{2}$$

We also define a bit-mask array $b_{\mathcal{C}_k}[1 \cdots |E|][1 \cdots F]$ for each cycle $\mathcal{C}_k$ to record its protection capability. Specifically, we set $b_{\mathcal{C}_k}[e][j] = 1$, if cycle $\mathcal{C}_k$ can protected the j-th FS on link e, otherwise, $b_{\mathcal{C}_k}[e][j] = 0$. If a cycle $\mathcal{C}_k$ can protect M links in a working path $\mathcal{R}_{s,d}$, we define the protection efficiency (PE) $\eta(\mathcal{R}_{s,d}, \mathcal{C}_k)$ as,

$$\eta(\mathcal{R}_{s,d}, \mathcal{C}_k) = \frac{M}{|\mathcal{C}_k|}, \tag{3}$$

where $|\mathcal{C}_k|$ returns the number of links in $\mathcal{C}_k$.

Algorithm 1 shows the operations of PE-*p*-cycle at each service provision time. *Lines* 1-3 are for determining the working path, and if there are insufficient spectrum resources on the working path, the request is blocked right away. *Line* 4 sorts the requests that are still not blocked (i.e., valid requests) according to the hop-counts of their working paths (i.e., $|\mathcal{R}_{s,d}|$). If the hop-counts are the same, we serve the request whose n is larger first. *Lines* 6-8 show the initialization procedures we perform for configuring the *p*-cycles for a request. To achieve high protection efficiency, *Lines* 9-26 try to use the cycle candidates in the descending order of $\eta(\mathcal{R}_{s,d}, \mathcal{C}_k)$ and configure *p*-cycles until all links in $\mathcal{R}_{s,d}$ are protected.

For each $\mathcal{C}_k$, the availability of the FS's are determined upon a few bit-masks, the working path bit-mask $b_{\mathcal{R}_{s,d}}$ calculated with Eq. (2), the cycle bit-mask array $b_{\mathcal{C}_k}$ and the bit-masks of the links in $\mathcal{C}_k$, i.e., $\prod_{e \in \mathcal{C}_k} b_e$. Note that in *Lines* 17 and 19, the "$\bigcup$" is for the bit-OR operation between two bit-masks. Due to the restriction from the same-spectrum protection scheme, a cycle can only be used if there are n common contiguous 1's between it and the working path. Multiple working paths can share the same *p*-cycle as long as they are link-disjoint. Finally, as in *Lines* 27-32, if all *p*-cycles are found, we assign FS's according to the spectrum utilizations of the working path and the *p*-cycles, otherwise, the request is considered as blocked.

B. *PWCE-*p*-cycle-SP Algorithm*

The PWCE-*p*-cycle-SP algorithm incorporates a spectrum planning procedure to regulate the spectra of the working and protection resources. Specifically, we reserve certain FS's on the links as protection resources and leave the rest for working paths. With the pre-reserved FS's, we obtain a set of cycles

Algorithm 1: PE-p-cycle Algorithm

1 **for** *each pending request* $LR(t, \tau, s, d, n)$ **do**
2 assign its working path $\mathcal{R}_{s,d}$ as the shortest routing path from s to d;
3 **end**
4 sort all valid requests in a descending order based on $|\mathcal{R}_{s,d}|$ firstly and then n;
5 **for** *each valid request* $LR(t, \tau, s, d, n)$ **do**
6 select cycles that can protect link(s) in $\mathcal{R}_{s,d}$ from $\mathbb{C}$;
7 get $\eta(\mathcal{R}_{s,d}, \mathcal{C}_k)$ with Eq. (3) for each selected $\mathcal{C}_k$;
8 $b_0 = b_{\mathcal{R}_{s,d}}$, $flag = 0$;
9 **for** *all selected* $\mathcal{C}_k$ *in the descending order of* $\eta(\mathcal{R}_{s,d}, \mathcal{C}_k)$ **do**
10 $b_1 = b_{\mathcal{C}_k}$;
11 **for** $j = 1$ *to* F, $e = 1$ *to* $|E|$ **do**
12 **if** *the* j*-th FS on* $\mathcal{C}_k$ *protects* e **then**
13 $b_1[e][j] = 1$;
14 **end**
15 **end**
16 **for** *each unprotected link* e **do**
17 **if** $b_0 \cdot ((\prod_{e\in\mathcal{C}_k} b_e) \bigcup b_1[e])$ *has* n *contiguous 1's* **then**
18 mark e as protected by cycle $\mathcal{C}_k$;
19 $b_0 = b_0 \cdot ((\prod_{e\in\mathcal{C}_k} b_e) \bigcup b_1[e])$;
20 **end**
21 **end**
22 **if** *all links in* $\mathcal{R}_{s,d}$ *are protected* **then**
23 $flag = 1$;
24 **break**;
25 **end**
26 **end**
27 **if** $flag = 1$ **then**
28 determine f_s and f_e according to b_0 with first-fit; provision $LR(t, \tau, s, d, n)$;
29 **else**
30 mark $LR(t, \tau, s, d, n)$ as blocked;
31 **end**
32 **end**

that can protect all links in $G(V, E)$, similar to the PWCE-p-cycle design in WDM networks [17]. We formulate a simple integer linear programming (ILP) model to obtain the cycles from $\mathbb{C}$, which have the minimum total number of links.

Notations:

- $\mathbb{C}$: a large set of cycles in $G(V, E)$.
- $\mathcal{C}_k$: the k-th cycle in $\mathbb{C}$.
- $y_{e,k}$: the flag that equals 1, if link e can be protected by cycle $\mathcal{C}_k$, and 0 otherwise.

Variables:

- x_k: Boolean variable that equals 1 if $\mathcal{C}_k$ is selected, and 0 otherwise.

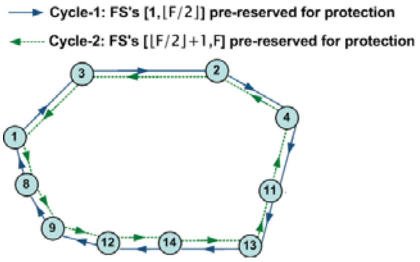


Fig. 2. An example of spectrum planning in PWCE-p-cycle-SP.

Objective:

$$Minimize \qquad z = \sum_k x_k \cdot |\mathcal{C}_k|. \tag{4}$$

Constraint:

$$\sum_k x_k \cdot y_{e,k} > 0, \quad \forall e \in E. \tag{5}$$

Eq. (5) ensures that all links in the network topology are protected by the selected cycles.

Then, we pre-reserve FS's on the selected cycles as protection resources. Fig. 2 illustrates an example of spectrum planning. Assume we find two cycles to cover all links in $G(V, E)$ and assign FS's $[1, \lfloor \frac{F}{2} \rfloor]$ and $[\lfloor \frac{F}{2} \rfloor + 1, F]$ to them as protection resources. Hence, for all links that can be protected by Cycle-1, if a request can use FS's $[1, \lfloor \frac{F}{2} \rfloor]$ on them as the working resources, it is protected. Same thing applies to Cycle-2 for FS's $[\lfloor \frac{F}{2} \rfloor + 1, F]$.

However, the p-cycle configuration scheme mentioned above has an intrinsic drawback when works under the same-spectrum protection constraint. As shown in Fig. 3, we have three cycles to cover all links in the topology. In this case, we cannot configure same-spectrum p-cycles for working path 3→6→14→12. This is because we need at least two cycles to protect the working path, while there are no common protection FS's on any two of the cycles. For example, if we use Cycle-1 to protect link 3→6, the links 6→14 and 14→12 can only be protected by Cycle-2 whose protection FS's do not overlap with those of Cycle-1. In order to solve this problem, we pre-reserve certain FS's on all links as the working and protection resources for this type of connections.

Algorithm 2 shows the detailed procedures of the PWCE-p-cycle-SP algorithm. *Lines* 8-20 explain the procedures of working path and p-cycle configurations for each request. Note that in PWCE protection, the working paths are always protected as long as all its links can be protected by the cycle(s) whose FS assignment(s) have common region(s) in the protection layer. Therefore, in *Lines* 10-16, provided that we can perform routing and spectrum assignment (RSA) for this type of requests, they are provisioned successfully. For the requests that have to be protected by the cycles whose FS assignments do not have common regions, we apply *Algorithm* 1 to use the FS's reserved on all links as in *Lines* 17-19.

Algorithm 2: PWCE-p-cycle-SP Algorithm

```
get a large set of cycles in G(V, E) and store them in ℂ;
select proper protection cycles by solving the ILP;
reserve FS's on all links based on traffic;
reserve FS's on selected cycles and configure p-cycles;
while the network is operational do
    collect pending requests;
    release the resources of the expired requests;
    for each pending request LR(t, τ, s, d, n) in descending order of n do
        calculate K shortest paths from s to d;
        select a shortest path candidate R_{s,d} that all its links can be protected with proper p-cycle(s);
        if there are enough working resources on R_{s,d} then
            determine f_s and f_e with first-fit; provision LR(t, τ, s, d, n);
        else
            mark LR(t, τ, s, d, n) as blocked;
        end
        if all K paths cannot be protected with any existing p-cycle combination then
            apply Algorithm 1 with the FS's reserved on all links to serve the request;
        end
    end
end
```

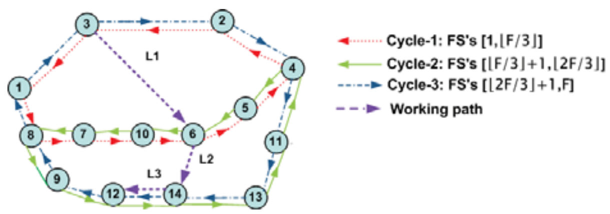


Fig. 3. An example of PWCE-p-cycle-SP algorithm's drawback.

C. *Ham-*p*-cycle-SP*

The PWCE-p-cycle-SP algorithm overcomes the drawback from the limited coverage of the cycles by pre-reserving certain FS's on all links. Nevertheless, this causes additional computation complexity and spectrum fragmentation. Meanwhile, note that in a mesh topology, when the connectivity permits, one can calculate Hamiltonian cycles that traverse all nodes in the topology only once [15]. Therefore, we can solve the issue of PWCE-p-cycle-SP by replacing the cycles found by the ILP with Hamiltonian cycles, and realize the Ham-p-cycle-SP algorithm. Specifically, we place two Hamiltonian cycles whose connections are the same but in opposite directions in the network to establish the protection layer. *Algorithm* 3 shows the detailed procedures of Ham-p-cycle-SP. *Line* 1 indicates that for a given network topology $G(V, E)$, we find the hamiltonian cycles with the ant colony optimization (ACO) algorithm. The rest of Ham-p-cycle-SP is very similar to that of PWCE-p-cycle-SP, except that we do not need to apply *Algorithm* 1 for the special cases.

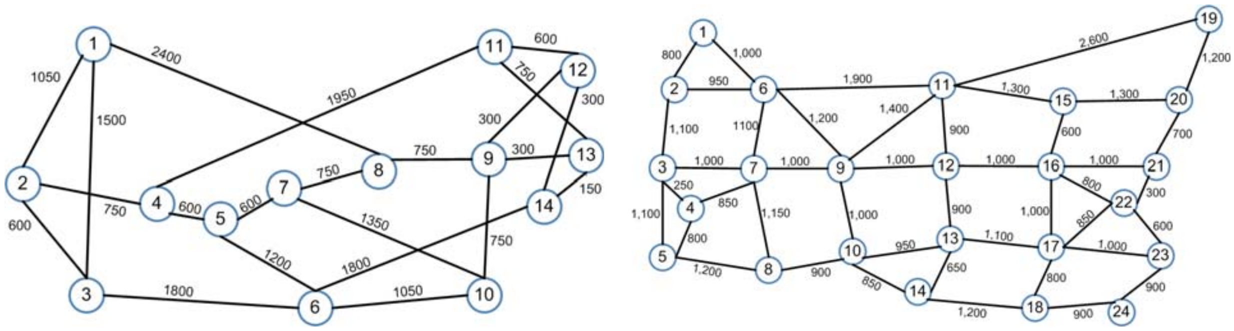
Algorithm 3: Ham-p-cycle-SP Algorithm

```
1  calculate Hamiltonian cycles with the ACO algorithm;
2  place two Hamiltonian cycles (in opposite directions) to establish the protection layer;
3  pre-reserve FS's on the two Hamiltonian p-cycles;
4  while the network is operational do
5      collect pending requests;
6      release the resources of the expired requests;
7      for each pending request LR(t, τ, s, d, n) in descending order of n do
8          calculate the shortest paths R_{s,d} from s to d;
9          if there are enough working resources on R_{s,d} then
10             determine f_s and f_e with first-fit; provision LR(t, τ, s, d, n);
11         else
12             mark LR(t, τ, s, d, n) as blocked;
13         end
14     end
15 end
```

IV. Performance Evaluation

The simulations are carried out with two topologies, the 14-node NSFNET topology and the 24-node US Backbone topology as shown in Fig. 4 [3]. We assume that in the EONs, each fiber can accommodate $F = 358$ FS's. The capacity of a single FS is 12.5 Gb/s and the number of FS's required by each request, n, is uniformly distributed within $[1, 20]$. The dynamic traffic requests are generated according to a Poisson process. For each request, we select the source and destination pair s-d randomly. Fig. 5 illustrates the simulation results on request blocking probabilities. We observe that in both topologies, PE-p-cycle provides the highest blocking probability among three algorithms. This is because it does not incorporate spectrum planning and frequent setting-up and tearing-down working and protection resources can result in significant spectrum fragmentation, which eventually limits the spectrum utilization of the EON.

For PWCE-p-cycle-SP, we set up three p-cycles in the protection layer for the NSFNET topology, assign FS's $[1, 90]$, $[91, 180]$ and $[181, 270]$ to them, and reserve FS's $[271, 358]$ on all links to handle the requests whose working paths cannot be protected by any existing p-cycle combination. Similarly, we pre-configure five p-cycles for the US Backbone topology. The corresponding FS assignments are FS's $[1, 90]$, $[91, 180]$,



(a) NSFNET topology.

(b) US Backbone topology.

Fig. 4. The network topologies for the simulation (fiber lengths in kilometers).

$[181, 270]$, $[1, 90]$ and $[91, 180]$, and we reserve FS's $[271, 358]$ on all links for the same purpose. It can be seen that the spectrum planning scheme reduces the blocking probability in both topologies. The benefit from the spectrum planning is enhanced with the Ham-p-cycle-SP algorithm, as its blocking probabilities are much lower than those of PWCE-p-cycle-SP. For Ham-p-cycle-SP, we only configure two p-cycles, and the corresponding FS assignment are FS's $[1, 179]$ and $[180, 358]$.

Fig. 6 compares the simulation results on the working-to-backup capacity ratios. The working-to-backup capacity ratio (WtB-CR) is defined as the ratio of total working resources to total protection resources. We calculate the average value of this ratio for each traffic load and plot the results in Fig. 6. Since PE-p-cycle sets up both the working and protection resources on-demand in a dynamical way, its WtB-CR stays at around 0.4 for all traffic loads. For PWCE-p-cycle-SP and Ham-p-cycle-SP, the resources in the protection layer is pre-reserved. Therefore, WtB-CR can increase with the traffic load, when more working resources are accommodated in the network. For the high traffic loads ($\geq$ 100 Erlangs), the WtB-CRs from PWCE-p-cycle-SP and Ham-p-cycle-SP are much larger than those from PE-p-cycle, indicating efficient utilization of the protection resources. We also notice that the WtB-CRs from Ham-p-cycle-SP are larger than those from PWCE-p-cycle-SP in both topologies. This is because Ham-p-cycle-SP has a better spectrum planning.

V. Discussion and Future Works

From the simulation results in Figs. 5 and 6, we observe that PWCE-p-cycle-SP and Ham-p-cycle-SP achieve better performance than PE-p-cycle, in terms of both blocking probability and the WtB-CR (for high traffic loads). When comparing their complexities, we can also see that Ham-p-cycle-SP has lower complexity than PE-p-cycle, while PWCE-p-cycle-SP is simpler than PE-p-cycle for most of the cases. However, PWCE-p-cycle-SP and Ham-p-cycle-SP also have their drawbacks. First of all, their WtB-CRs are small for low traffic loads. In our future work, we will investigate whether changing the spectrum planning adaptively with the traffic load can solve this problem. Secondly, the spectrum planning in PWCE-p-cycle-SP can result in some "dangling" spectrum regions on the links, which cannot be used as either working or protection resources. We expect these regions can be used as the working resources of low priority traffics that do not require 100% protection against single-link failures, e.g., the best-effort traffics. Finally, the cycle-length of the Hamiltonian cycles in Ham-p-cycle-SP can be an issue for large networks, as a broken link will be restored with a relatively large number of links. Therefore, after restoration, the transmission delay can increase dramatically. The Hamiltonian cycle cover (HCC) technique discussed in [15] is a potential solution to this. We will incorporate HCC in EONs to partition a large network into a set of protection domains, and design Hamiltonian cycles and spectrum planning to protect each domain.

VI. Conclusion

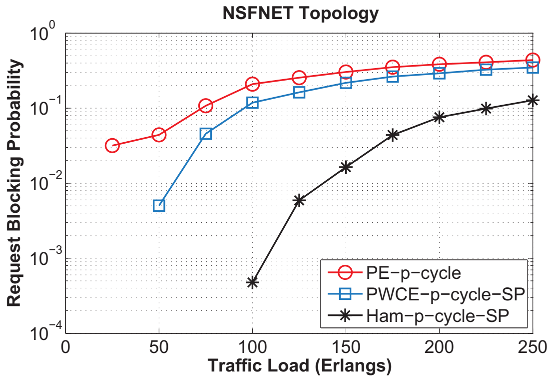
In this paper, we proposed three algorithms for dynamic p-cycle configuration in EONs. We also proposed to use the spectrum planning technique to regulate the spectra of working and protection resources for reducing spectrum fragmentation and request blocking probability. Simulation results indicated that with the spectrum planning, the PWCE-p-cycle-SP and Ham-p-cycle-SP algorithms reduced the blocking probability effectively, when comparing with the one without it, i.e., PE-p-cycle. Between the two algorithms that have spectrum planning, Ham-p-cycle-SP achieved lower blocking probability and higher WtB-CRs for the high traffic loads ($\geq$ 125 Erlangs) than PWCE-p-cycle-SP.

Acknowledgments

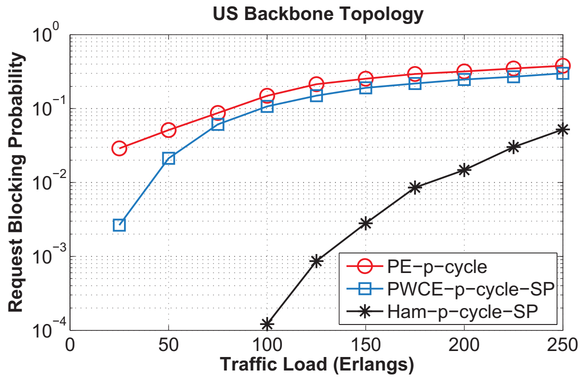
This work was supported in part by the Program for New Century Excellent Talents in University (NCET) under Project NCET-11-0884, and the Natural Science Foundation of Anhui Province under Project 1208085MF88.

References

[1] J. Armstrong, "OFDM for optical communications," *J. Lightw. Technol.*, vol. 27, no. 3, pp. 189–204, Mar. 2009.
[2] M. Jinno *et al.*, "Distance-adaptive spectrum resource allocation in spectrum-sliced elastic optical path network," *IEEE Commun. Mag.*, vol. 48, no. 8, pp. 138–145, Aug. 2010.
[3] Z. Zhu *et al.*, "Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing," *J. Lightw. Technol.*, vol. 31, no. 1, pp. 15–22, Jan. 2013.

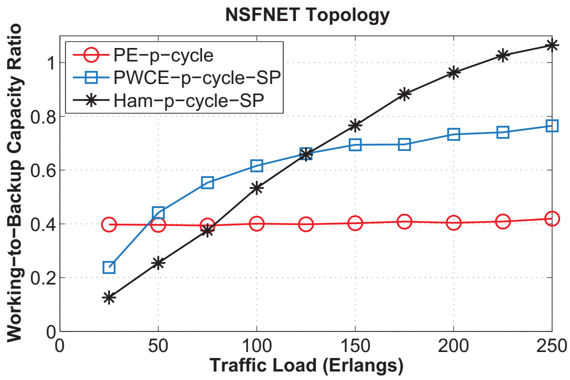


(a) NSFNET topology.

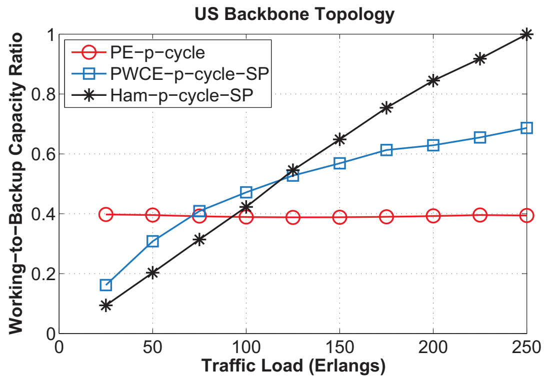


(b) US Backbone topology.

Fig. 5. Simulation results on request blocking probabilities.



(a) NSFNET topology.



(b) US Backbone topology.

Fig. 6. Simulation results on working-to-protection capacity ratio.

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

[5] K. Christodoulopoulos, I. Tomkos, and E. Varvarigos, "Elastic bandwidth allocation in flexible OFDM-based optical networks," *J. Lightw. Technol.*, vol. 29, no. 9, pp. 1354–1366, May 2011.

[6] Y. Sone *et al.*, "Routing and spectrum assignment algorithm maximizes spectrum utilization in optical networks," in *Proc. of ECOC 2011*, pp. 1–3, Sept. 2011.

[7] X. Wan, N. Hua, and X. Zheng, "Dynamic routing and spectrum assignment in spectrum-flexible transparent optical networks," *J. Opt. Commun. Netw.*, vol. 4, no. 8, pp. 603–613, Aug. 2012.

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

[9] W. Lu *et al.*, "Dynamic multi-path service provisioning under differential delay constraint in elastic optical networks," *IEEE Commun. Lett.*, vol. 17, no. 1, pp. 158–161, Jan. 2013.

[10] Y. Sone *et al.*, "Bandwidth squeezed restoration in spectrum-sliced elastic optical path networks (SLICE)," *J. Opt. Commun. Netw.*, vol. 3, no. 3, pp. 223–233, Mar. 2012.

[11] S. Xu *et al.*, "Shared-path protection in OFDM-based optical networks with elastic bandwidth allocation," in *Proc. of OFC 2012*, pp. 1–3, Mar. 2012.

[12] S. Ramamurthy, L. Sahasrabuddhe, and B. Mukherjee, "Survivable WDM mesh networks," *J. Lightw. Technol.*, vol. 21, pp. 870–883, Apr. 2003.

[13] W. Grover and D. Stamatelakis, "Cycle-oriented distributed preconfiguration: ring-like speed with mesh-like capacity for self-planning network restoration," in *Proc. of ICC 1998*, pp. 537–543, Jun. 1998.

[14] D. Schupke, C. Gruber, and A. Autenrieth, "Optimal configuration of p-cycles in WDM networks," in *Proc. of ICC 2002*, pp. 2761–2765, Jun. 2002.

[15] H. Huang and J. Copeland, "A series of Hamiltonian cycle-based solutions to provide simple and scalable mesh optical network resilience," *IEEE Commun. Mag.*, Nov. 2002.

[16] Z. Zhang, W. Zhong, and B. Mukherjee, "A heuristic method for design of survivable WDM networks with p-cycles," *IEEE Commun. Lett.*, vol. 8, pp. 467–469, Jul. 2004.

[17] Z. Zhang, W. Zhong, and S. Bose, "Dynamically survivable WDM network design with p-cycle-based PWCE," *IEEE Commun. Lett.*, vol. 9, no. 8, pp. 756–758, Aug. 2005.

[18] A. Eshoul and H. Mouftah, "Survivability approaches using p-cycles in WDM mesh networks under static traffic," *IEEE/ACM Trans. Netw.*, vol. 17, pp. 671–683, Apr. 2009.