Traffic Grooming and Spectrum Overlap in FIPP p-cycle for Protection of Elastic Optical networks

Helder M. N. S. Oliveira and Nelson L. S. da Fonseca
Institute of Computing - University of Campinas, Brazil
Email: helder@lrc.ic.unicamp.br, nfonseca@ic.unicamp.br

***Abstract*—Elastic optical networks has emerged as a solution for dealing with the diversity of bandwidth demands of network applications. However, networks are not sufficiently resilient, survivable, highly available and dependable, therefore, protection techniques have been developed to cope with failures. Among these techniques, p-cycle is a very attractive once since it provides ring-like speed of restoration in mesh topologies. This paper presents a new algorithms to provide path protection using p-cycle path, traffic grooming and overlap spectrum in elastic optical networks. The proposed algorithm is compared to others existing in the literature. Results indicate that the our algorithm can provide up to 40% less blocking when compared to existing other algorithms.**

***Keywords*—p-Cycle, Survivability, Multi-core Fiber, Elastic Optical Network with Space Division Multiplexing.**

I. INTRODUCTION

In Wavelength Division Multiplexing (WDM), the fixed capacity of a wavelength accommodates demands of different sizes. This leads to under utilization of the spectrum since demands rarely match the exact capacity of a wavelength. Sub-wavelength demands are usually groomed to decrease the waste of capacity. Such rigidness has recently led to the emergence of spectrum-sliced elastic optical path networking. In this technology, (Optical) Orthogonal Frequency Division Multiplexing (OFDM) is employed. OFDM is a multi-carrier transmission technology that slits high data rate channels into a number of orthogonal channels, called subcarriers, each with (sub-wavelength) low data rates. Elastic optical networks have gained great momentum and have attracted attention from industry and academia due to the technology maturity that enables their development and deployment.

Differently than WDM networks in which the optical spectrum is divided into frequency slots of fixed width 50GHz or 100GHz, allowing up to 40 and 80 wavelengths respectively, in elastic optical networks the spectrum is divided into slots with finer granularity, e.g. 12.5GHz or even 6.25GHz; and slots can be combined and assigned to a connection according to the requested bandwidth and modulation technique applied to convert the electrical signal into the optical signal.

Due to the wide capacity of optical fibers, any interruption implies massive loss of data. This vulnerability has motivated the development of different protection and restoration schemes. p-Cycle is one these protection techniques which has been intensively investigated in recent years [1] [2] due to their advantages. The p-cycle technique combines the speed recovery in ring networks with efficiency in restorable network. One kind of p-cycle of particular interest is the p-cycle FIPP that provides protection to paths fully.

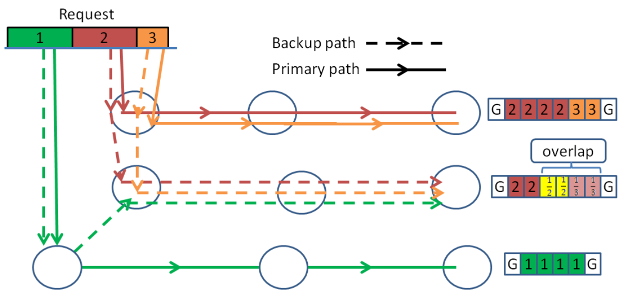


Figure 1: Spectrum overlap

Spectrum overlap is a technique in which two backup lightpaths can use the same links and the same spectrum, since the working paths of the two connections are physically disjoint links [4]. Figure 1 show an example of spectrum overlap. The primary path of request 1 is disjoint from the primary path of the request 2 and the primary path of request 1 is disjoint from the primary path of the request 3. In case of failure on a link only one of the connections will make use of spectrum overlap, making possible the spectrum overlap between the backup paths of these requests. Figure 1 shows two spectrum overlap in backup paths of requests 1 and 2, and two spectrum overlap in backup paths of requests 1 and 3.

In elastic optical networks, traffic grooming is a technique that combines multiple connections in an optical path without needing guard bands between them [3]. The combination of traffic grooming and spectrum overlap allows a significant gain in spectrum utilization, which decreases the blocking of connections.

This paper introduces an algorithm called FIPPSh-Flex for providing FIPP *p*-cycle protection in elastic optical networks. The algorithm creates protection paths, using the p-cycle FIPP technique, spectrum overlap and traffic grooming. The algorithm FIPPSh-Flex extends the algorithm FIPP-Flex, [5] to use the spectrum overlay technique in optical paths (p-cycles). The combination of spectrum overlap, traffic grooming and p-cycle schemes produces results better than existing algorithms [4], [5] e [6].

This paper is organized as follows. Section II reviews related work and Section III introduces the concepts of *p*-cycle and FIPP. Section IV introduces the RSA-FLEX algorithm and Section V the FIPPSh-FLEX algorithm. Section VI evaluates the performance of the proposed algorithm and Section VII concludes the paper.

II. RELATED WORK

The emergence of flexgrid networks has motivated several investigations, mainly on RSA algorithms but only recently investigations have addressed protection issues [4]–[7].

A novel elastic shared path protection (ESPP), was proposed in [4], it does not only provide the traditional shared path protection (SPP), but explores a new opportunity for sharing enabled by the tunability of transponders: In fact, the backup spectrum can be shared between two adjacent lightpaths on a link, if their corresponding working paths are link-disjoint. Oliveira *et al.* [5] proposed a novel algorithm to provide Failure-independent path protecting p-cycle for path protection in elastic optical networks. Shao *et al.* [6] proposed and evaluated conservative and aggressive backup sharing policies in OFDM-based optical networks with elastic bandwidth allocation but *p*-cycles were not investigated. They introduced a policy in which backup lightpaths with different allocated capacity can protect primary lightpaths with disjoint paths, leading to better use of resources to provide path protection. In [7], the multicast protection problem on elastic optical networks (EONs) for the single link-failure case is studied. Two segment-based protection algorithms are proposed to solve this problem.

None of these studies used the p-cycle protection technique combined with spectrum overlap and traffic grooming in optical elastic networks as shown in this paper.

III. P-CYCLE

p-Cycle is a protection scheme in which the spare capacity is pre-connected to form ring-like structures [1]. *p*-Cycles provide Bi-directional Line Switching Ring (BLSR) protection which is considered a generalization of the 1:1 protection scheme [8]. The main difference to conventional ring protection is that *p*-cycles provide two protection paths for each link that straddles the cycle. The straddling links can have working capacity but no spare capacity [2]. Moreover, working paths can be freely routed over a mesh structure and it is not necessary to follow ring-constrained routing topology. In networks protected by cycles, in an event of failure, only two switching actions at the end nodes of the failed span are necessary to switch the traffic to a protection path, as in conventional ring. *p*-Cycles provide fast restoration not because they are rings but because they are fully pre-connected before failure [2].

A special case of *p*-cycle for path protection is the so called Failure-Independent Path Protecting *p*-cycles (FIPP) [9]. FIPP *p*-cycles furnish protection to end-to-end working (primary) path with end nodes on the *p*-cycle. FIPP is an extension of the *p*-cycle concept in which the failure is not limited to be in a link or path segment immediately adjacent to the end nodes. FIPP is based on disjointness of working and backup paths, and provides the advantage that fault detection is independent of the fault location which is called failure independence. Failure independence is quite advantageous when location of fault is slow or difficult such as in transparent or translucent networks.

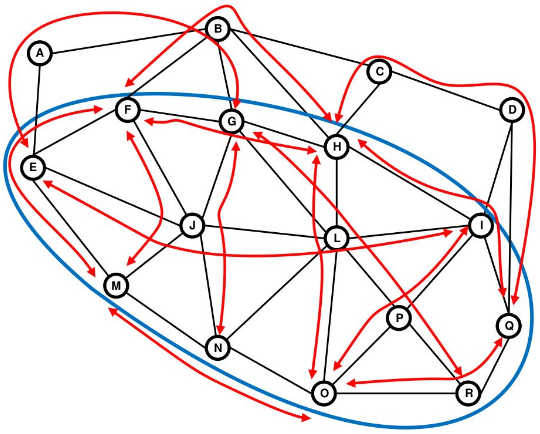


Figure 2: FIPP p-Cycle

This is an advantage over traditional path protection schemes and over the so called flow *p*-cycles [9].

Figure 2 illustrates the concept of the p-cycle FIPP. The cycle EFGHIQRONM is a p-cycle, and the arrows show the various paths which it protects. In this example, a single p-cycle protects the fourteen different paths which have their endpoints on this p-cycle path.

IV. RSA-FLEX ALGORITHM

In this paper, we used the RSA-Flex algorithm [5], for routing and spectrum allocation, the Routing and Spectrum Allocation problem is an NP-hard problem and heuristics are needed to solve the problem. In an RSA problem, besides the spectrum continuity constraint that imposes the allocation of the same spectrum in each fiber along the route of a lightpath, slots (carrier) must be contiguously allocated in the spectrum (the spectrum contiguity constraint). As in [5], the proposed algorithm models the spectrum availability in the network as labeled multigraph. A multigraph is a graph which can have multiple edges (also called "parallel edges"), that is, edges that have the same end vertice. In this auxiliary graph, vertices represent OXCs and edges the slots in the link connecting OXCs. The multigraphs presented in this paper, all vertices are connected by N edges which is the number of slots in the spectrum of each network link. The label on an edge represent the slot availability. An ∞ value means that the slot is already allocated whereas the value 1 means that the slot is available for allocation. These values were defined to facilitate the employment of traditional shortest path algorithms.

The multigraph is transformed into $N-b+1$ graphs where b is the bandwidth demand in slot of the requested channel. These graphs are generated by fixing an edge of the multigraph and considering the b consecutive edges to the fixed edge. This set of b edges of the multigraph are mapped onto a single edge of the generated graph. Its weight is given by applying a specific weight function that considers the b edges. Figure 2 illustrates the multigraph representing the spectrum and one of the generated graph. For each of the generated graphs, a shortest path algorithm is executed and the chosen path is the one that has the lowest weight among all shortest paths found.

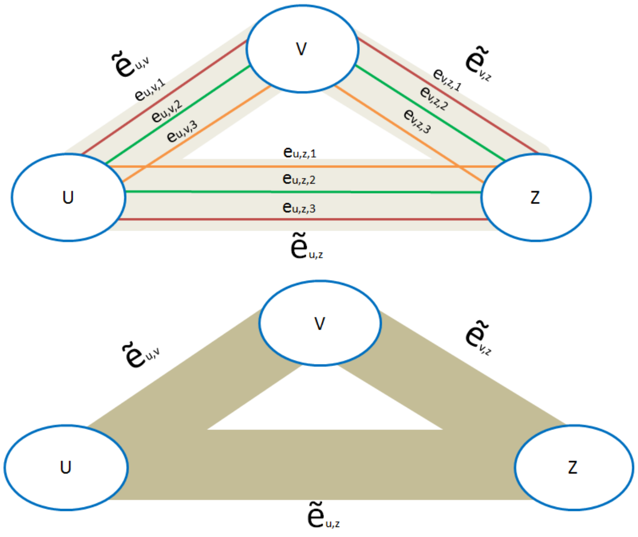


Figure 3: Multigraph

For a demand of b slots, $N-b+1$ graphs of type $\widetilde{G}_{n,b}$ will be generated, each edge of the $\widetilde{G}_{n,b}$ graph corresponds to the mapping of b edges of G starting on the n^{th} edge of G. Since the same ordered edges connecting any two nodes in G are mapped onto edges of $\widetilde{G}_{n,b}$, the spectrum continuity is assured.

We introduce the notation used in this paper:

s: source node;

d: destination node;

b: bandwidth demand in slots, $b = 1 \dots N$;

$r(s,d,b)$: request from the node s to the node d with bandwidth demand b in slots;

N: number of slots between two nodes;

$G = (V,E,W)$: labeled multigraph composed by a set of nodes V, a set of edges E and a set of edge weight W, $|E| = N \cdot |V|$. The edges connecting two vertices of G represent the N slots in the link connecting two network nodes;

$G' = (V,E,W)$: labeled multigraph considers that an edge that is being used by a protection path as available as long as the path protected by the protection path is disjoint the new path to be created. The multi-graph consists of a set of nodes V, a set of edges E and a set of weights of edges W, $|E| = N \cdot |V|$.

$\widetilde{V} = V$: set of nodes;

$E = \{e_{u,v,n}\}$: set of n edges;

$e_{u,v,n}$: the n^{th} edges connecting u and v;

$w(e_{u,v,n})$: weight of the edge $e_{u,v,n}$;

$w(e_{u,v,n}) = 1$ if the n^{th} slot in the link connecting OXC u and v is free and $w(e_{u,v,n}) = \infty$ if the slot is already allocated;

$W = \{w(e_{u,v,n})\}$:set of edge weights

$\widetilde{G}_{n,b} = (\widetilde{V}, \widetilde{E}, \widetilde{W})$: the n^{th} labeled graph such that $\widetilde{E}$ is the set of edges connecting $\{\widetilde{u}, \widetilde{v}\} \in \widetilde{V}$ and $\widetilde{C}$ is the set of costs associated to $\widetilde{E}$. The edges in $\widetilde{E}$ correspond to the mapping of b edges in G starting at the n^{th}*edge*;

$\widetilde{e}_{u,v} \in \widetilde{E}$: edge connecting $\widetilde{u}$ and $\widetilde{v}$; $\widetilde{e}_{\widetilde{u},\widetilde{v}} = \{e_{u,v,n}\} \in E$ is a chain such that $e_{u,v,n}$ is the least ordered edge, $e_{u,v,n+b}$ is the greatest ordered edge and $|\widetilde{e}_{u,v}| = b$;

$\widetilde{w}_n(\widetilde{e}_{\widetilde{u},\widetilde{v}})$: weight of the edge $\widetilde{e}_{\widetilde{u},\widetilde{v}}$;

$\widetilde{W}_n = \{\widetilde{c}_n(\widetilde{e}_{\widetilde{u},\widetilde{v}})\}$: set of edge weights;

P_n: chain of $\widetilde{G}_n$ such that the source node s is the least ordered node and d is the greatest ordered node;

$W(\widetilde{P}_n)$: $\sum_{\widetilde{e}_{\widetilde{u},\widetilde{v}} \in \{\widetilde{P}_n\}} \widetilde{e}_{\widetilde{u},\widetilde{v}}$: the weight of the path $\widetilde{P}_n$ (the sum of the weights of all the edges in the chain;

$W_{s,d}$ = weight of the shortest path between s and d;

$\widetilde{c}_{u,v,b}$: p-cycle containing vertices u and v and edges corresponding to the mapping of b edges of the multigraph G;

$\widetilde{C}_{u,v,b} = \widetilde{c}_{u,v,b}$: set of all p-cycles containing vertices u and v and edges corresponding to the mapping of b edges of the multigraph G;

$\widetilde{C}$: set of all established p-cycles;

$P_1 \oplus P_2$: concatenation of disjointness paths P_1 and P_2

Algorithm 1 RSA-Flex

```
1: $\forall n = 1 \dots N-b$
2:     $(W(P_n), P_n) = ShortestPath(\widetilde{G}_{n,b}, r(s,d,b))$
3: $W_{s,d} = W(P_n) |\ \forall i\ W(P_n) \le W(P_i)$
4: **if** $W_{s,d} = \infty$ **then**
5:     $block\ r(s,d,b)$
6: **else**
7:     $W(e_{u,v,i}) = \infty \quad \forall \{u,v\} \in \widetilde{P}_i \quad n = n \dots i+b-1$
8: **end if**
```

Algorithm 1 [5] details the RSA-Flex Algorithm. In this algorithm, Line 1 establishes all the set of edges that will be mapped onto $\widetilde{G}_{n,b}$ edges. Line 2 solves a shortest path algorithm for the graph $\widetilde{G}_{n,b}$ and provides the path and its weight. If the weight of the shortest path is ∞, it means that was not possible to find a path under the contiguity constraint for the demand b with allocation starting at the n^{th} slot. Line 3 selects the path among the $N-b+1$ shortest paths that has the lowest weight value. In case the weight of all shortest path is ∞ (Line 4), there is no path in the network that satisfies the request of b slots under the contiguity constraint. Therefore, the request has to be blocked (Line 5). Otherwise, the shortest path with the lowest value is chosen (Line 7) and the corresponding edges in the multigraph G have their weight changed to ∞ (Line 8) meaning that the slots were allocated to the newly established lightpath. Since the RSA-Flex Algorithm executes a shortest path algorithm $N-b$ times and considering the use of the Djkstra Shortest Path algorithm, the computational complexity of the proposed algorithm is $N \cdot (|V| + |E|) \cdot \log(|V|)$.

V. FIPPSh-Flex Algorithm

The FIPP-Flex algorithm [5] decides on the establishment of lightpaths in an FIPP p-cycle protected network. The FIPPSh-Flex algorithm extends FIPP-Flex algorithm to allow traffic grooming on their optical paths and spectrum overlap between disjoint backup paths.

Spectral overlap is performed on the spectrum of backup paths to protect primary paths adjacent. As the FIPP-Flex algorithm generates better use of network resources than other existing algorithms, the use of traffic grooming further reduces the use of network resources. The best use of resources in FIPPSh-Flex algorithm is possible also due to the fact that the backup path allows the establishment of other paths using slots already occupied by other backup paths.

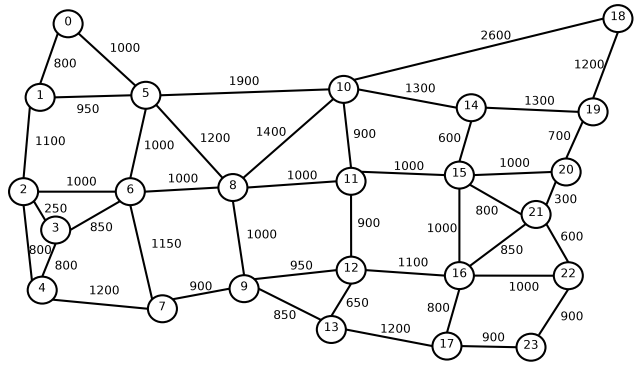
Algorithm 2 FIPPSh-Flex

```
1:  $(W(P_n), P_n) = RSA - Flex(G, s, d, b)$
2:  if $W_{s,d} = \infty$ then
3:      $block\ r(s, d, b)$
4:  else
5:      if $C_{u,v,i} \neq \emptyset \forall i \geq b$ then
6:          establish $r(s, d, b)$ as $P_n$
7:      else
8:          $(W(P_1), P_1)$ = RSA-Flex $(G', r(s, d, b))$
9:          $(W(P_2), P_2)$ = RSA-Flex $(G', r(s, d, b))$
10:         if $W(P_1) = \infty$ or $W(P_2) = \infty$ then
11:             block $r(s, d, b)$
12:         else
13:             establish $r(s, d, b)$ as $P_n$
14:             establish $P_1$ and $P_2$
15:             $\widetilde{c}_{u,v,b}$ = $P_1 \oplus P_2$
16:         end if
17:     end if
18: end if
```

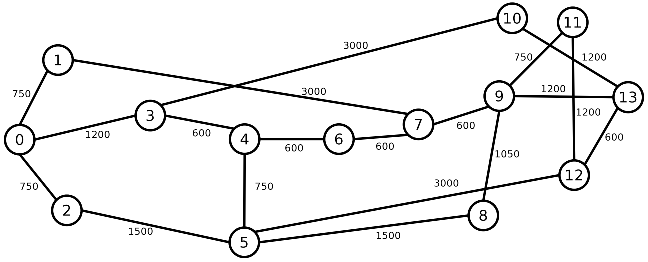
In FIPPSh-Flex, a lightpath is established if and only if it can be protected by an FIPP *p*-cycle which can have both on-cycle and straddling links. The FIPP-Flex algorithm assures a protection path for each established lightpath and the protection is guaranteed for single failures. Line 1 tries to find a path to establish the request $r(s, d, b)$. If there is no path available (Line 2) then the request is blocked (Line 3). Otherwise, a *p*-cycle to protect the lightpath to be established is searched (Line 5). If a *p*-cycle exists, the lightpath is established. Otherwise, a *p*-cycle to protect the lightpath to be established should be created (Lines 8 and 9). The major difference between FIPP-Flex and FIPPSh-Flex algorithms are lines 8 and 9, because in FIPPSh-Flex algorithm these lines consider the traffic grooming and spectrum overlap to create the p-cycle, providing better efficiency of backup feature used. When no *p*-cycle can be created to protect the lightpath then the request is blocked (Line 11), otherwise the lightpath (Line 13) as well as the *p*-cycle (Lines 14 and 15) are established to satisfy the request.

VI. PERFOMANCE EVALUATION

To assess the performance of the FIPPSh-Flex algorithm, simulation experiments were employed and results compared to those of networks without any protection scheme as well to those produced by FIPP-Flex, SBPP, ESPP and BPP algorithms. The FlexGridSim [10] simulator was used. In each simulation, 100,000 requests were generated and simulations for each algorithms used the same set of seeds. Confidence intervals with 95% confidence level were generated. The NSF (Figure 4b) and the USA (Figure 4a) topologies were used. The NSF topology has 16 nodes and 25 links whereas the USA topology has 24 nodes and 43 links. In the simulated network, the spectrum was divided in 240 slots of 12,5 GHz each.



(a) USA Topology

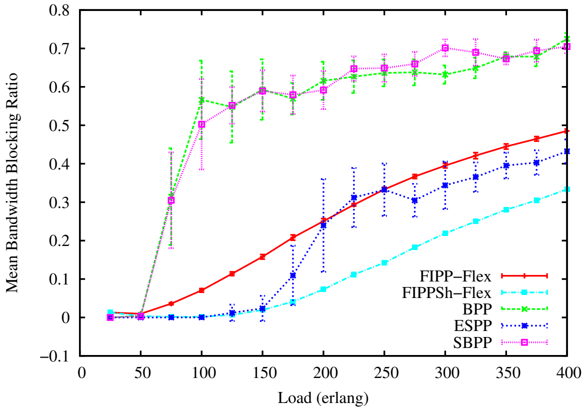


(b) NSF Topology

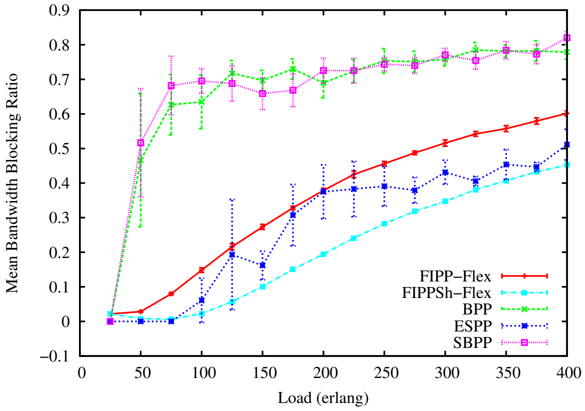
Figure 4: Topologies

In the figures, curves labeled BPP show the results for networks protected by protection scheme 1+1, curves labeled SBPP show the results for networks protected by algorithm proposed in [6], curves labeled FIPP-Flex show the results for networks protected by algorithm FIPP-Flex proposed in [5], curves labeled ESPP show the results for networks protected by algorithm proposed in [4] which uses traffic grooming and spectrum overlap and curves labeled FIPPSh-Flex show the results for networks protected by the algorithm FIPPSh-Flex.

Figures 5a and 5b show the bandwidth blocking ratio (BBR) as a function of the load for the USA and NSF topologies, respectively. The SBPP and BPP algorithms produce similar BBR behavior in both topologies. For the USA topology (Figure 5a), BPP and SBPP algorithms saturate the network under load of 100 erlangs. Due to the high connectivity of the USA topology there is no blocking up to 50 erlangs. The FIPP-Flex produces 50% less blocking than do the BPP and SBBP algorithms, this is due to the sharing of the FIPP p-cycles. Meanwhile ESPP produces about 33% less blocking than does FIPP-Flex between loads 50 and 200 erlangs and between loads 250 and 400 erlangs. Despite the FIPP-Flex algorithm adopts no traffic grooming and spectrum overlap, the sharing of p-cycle produces low blocking, which is very close to the blocking generated by the ESPP algorithm. Until load of 150 erlangs, the FIPPSh-Flex and ESPP algorithms hardly produce blocking, as a consequence of the adoption spectrum overlap and traffic grooming. FIPPSh-Flex combines the advantages of sharing, traffic grooming and spectrum overlap, producing less blocking. FIPPSh-Flex algorithm produces 30% less blocking that does ESPP under 150 erlangs, showing the advantage of

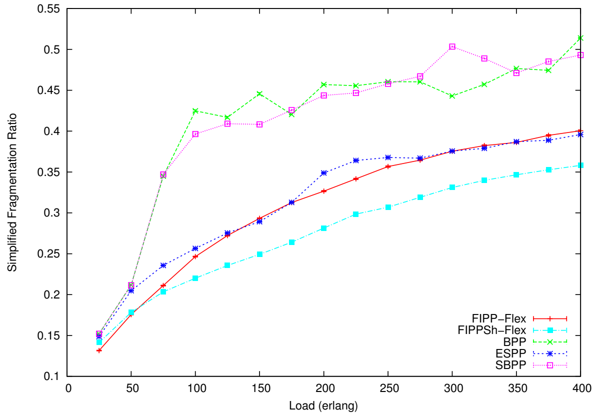


(a) USA

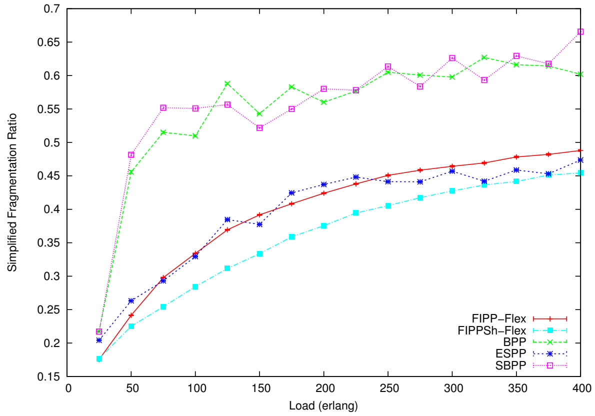


(b) NSF

Figure 5: Bandwidth Blocking Ratio



(a) USA



(b) NSF

Figure 6: Fragmentation Ratio (%)

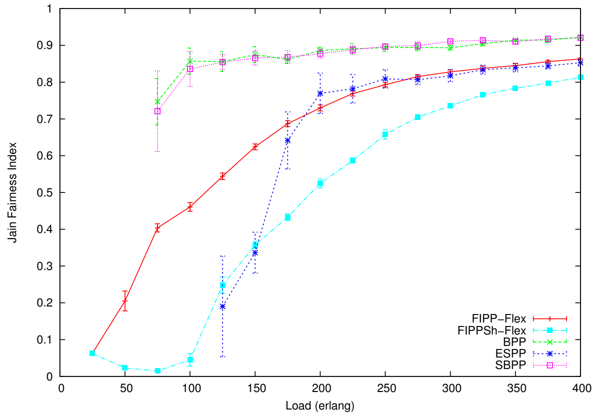
using p-cycle since both algorithms use traffic grooming and spectral overlap.

For the NSF topology (Figure 5b), the SBPP and BPP algorithms saturate the network under loads of 75 erlangs. The FIPP-Flex produces 42% less blocking than do the BPP and SBBP algorithms. This is due to the sharing of FIPP p-cycle. Meanwhile ESPP produces about 50% less blocking than does FIPP-Flex between loads 50 and 175 erlangs and between loads 250 and 400 erlangs. The FIPPSh-Flex produces low blocking, very close to that produced by ESPP. Until load 75 erlangs, the FIPPSh-Flex and ESPP algorithms produce almost no blocking. The FIPPSh-Flex algorithm produces 15% less blocking than ESPP algorithm under 75 erlangs. The combination of spectrum overlap and traffic grooming in the ESPP and FIPPSh-Flex algorithms evinces advantages in both topologies, when compared with the others algorithms presented in the paper.

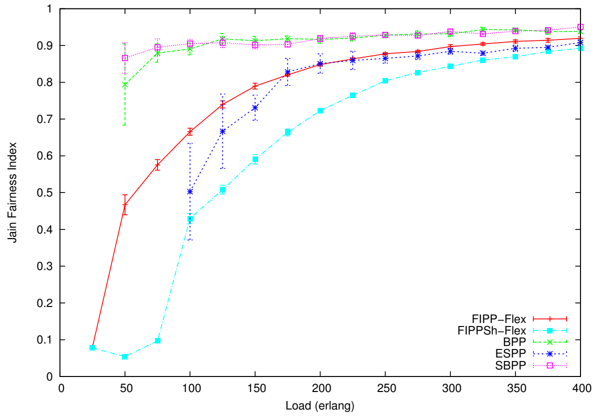
Figures 6a and 6b depict the Fragmentation Ratio as function of the load for USA and NSF topologies, respectively. In flexgrid networks, the establishment and tear down of lightpaths leads to the fragmentation of the spectrum which is a state in which there are available slots, that cannot be gathered in a way to be used to accept new requests. The fragmentation ratio is defined as the average ratio between the number of types of demands that cannot be accepted to the total number of types of demands.

For the USA topology (Figure 6a), the SBPP algorithm produces fragmentation ratio 5% lower than that given by BPP algorithm, as a consequence of the sharing of paths. The FIPP-Flex algorithm produces fragmentation ratio 23% lower than that given by SBPP, since p-cycles promote greater sharing. FIPP-Flex produce fragmentation ratio 6% lower than that given by ESPP algorithm despite the ESPP algorithm producing less blocking that does the FIPP-Flex algorithm. This happens since p-cycle promote greater sharing, generating a smaller amount of disconnected backup paths, and reducing the number of available spectrum that can not be used. The FIPPSh-Flex algorithm produces fragmentation ratio 17% lower than that given by ESPP algorithm.

For the NSF topology (Figure 6b), the SBPP algorithm produces fragmentation ratio 3% lower than that given by BPP algorithm. FIPP-Flex algorithm produces fragmentation ratio 18% less than that by the SBPP algorithm, because the p-cycle can be shared between all node in the p-cycle. The FIPP-Flex algorithm produces fragmentation rate 3% lower than that given by the ESPP algorithm. The ESPP algorithm produces lower BBR values than does the FIPP-Flex algorithm, since p-cycle promote greater sharing between nodes and the number of required paths of backup is smaller, which influences the fragmentation. The FIPPSh-Flex algorithm produces fragmentation ration 20% lower than that given by the ESPP algorithm. BPP and SBPP algorithms have high fragmentation ratio as a result of blocking and the low number of alternative paths.



(a) USA



(b) NSF

Figure 7: Jain fairness index

Figures 7a and 7b display the Jain Fairness Index (JFI) of the BBR for different source destination pairs, for the USA and NSF topologies, respectively.

The BPP and SBPP algorithms have high Jain index values, distributing the blocking requests more evenly between the source and destination pairs, because these algorithms produce greater BBR. The FIPP-Flex, FIPPSh-Flex and ESPP algorithms have low Jain index values due to the low BBR values that they produce, especially under low loads.

For the USA topology (Figure 7a), the FIPP-Flex algorithm produces Jain fairness index 13% less than SBPP algorithm, due to the higher BBR values produced by the SBPP algorithm. The ESPP algorithm produces Jain index 60% lower than that given by the FIPP-Flex under load 175 erlangs, as a result of the use of spectrum overlap and traffic grooming use in ESPP. Under 150 erlangs, the FIPPSh-Flex algorithm produces index Jain 16% lower than given by the ESPP algorithm, due to the large BBR produced by ESPP algorithm.

For the NSF topology (Figure 7b), the FIPP-Flex algorithm produces Jain fairness index 14% less than SBPP algorithm, due to the large blocking produced by the SBPP algorithm. The ESPP algorithm produces Jain fairness index 30% less than the FIPP-Flex algorithm, due to spectrum overlap and traffic grooming used in the ESPP algorithm. The FIPPSh-Flex algorithm generates Jain fairness index 24% less than the ESPP algorithm, due to the larger BBR produced by ESPP algorithm and sharing of p-cycle FIPP.

VII. CONCLUSION

This paper introduced an algorithm to support the establishment of lightpaths in elastic optical networks jointly using FIPP *p*-cycles, traffic grooming and spectrum overlap. The *p*-cycle method benefits from the fast restoration of ring-like protection and high capacity efficiency of mesh protection. The algorithm was evaluated for different topologies and loads. Algorithms using overlapping spectrum present more attractive results for both topologies in comparison with other algorithms. FIPPSh-Flex algorithm produces up to 20% less blocking than ESPP algorithm, which also uses spectrum overlap and traffic grooming. The spectrum overlap capacity is noticeably effective, especially when combined with p-cycle FIPP.

ACKNOWLEDGMENTS

The authors would like to thank the grant 165446/2015-3, CNPq, and FOTONICOM for the financial support.

REFERENCES

[1] R. Asthana, Y. Singh, and W. Grover, “p-cycles: An overview,” *IEEE, Communications Surveys Tutorials*, vol. 12, no. 1, pp. 97 –111, quarter 2010.

[2] Schupke, “Analysis of p-cycle capacity in WDM networks,” *Photonic Network Communications*, pp. p. 41–51, 2006.

[3] G. Zhang, M. De Leenheer, and B. Mukherjee, “Optical traffic grooming in ofdm-based elastic optical networks [invited],” *Optical Communications and Networking, IEEE/OSA Journal of*, vol. 4, no. 11, pp. B17–B25, Nov 2012.

[4] M. Liu, M. Tornatore, and B. Mukherjee, “Survivable traffic grooming in elastic optical networks -shared protection,” *Lightwave Technology, Journal of*, vol. 31, no. 6, pp. 6230–6234, March 2013.

[5] H. M. N. S. Oliveira and N. L. S. da Fonseca, “Algorithm for fipp p-cycle path protection in flexgrid networks,” in *2014 IEEE Global Communications Conference*, Dec 2014, pp. 1278–1283.

[6] X. Shao, Y.-K. Yeo, Z. Xu, X. Cheng, and L. Zhou, “Shared-path protection in OFDM-based optical networks with elastic bandwidth allocation,” in *Optical Fiber Communication Conference and Exposition (OFC/NFOEC), 2012 and the National Fiber Optic Engineers Conference*, 2012, pp. 1–3.

[7] D.-R. Din and I.-R. Lai, “Multicast protection problem on elastic optical networks using segment-base protection,” in *Informatics, Electronics Vision (ICIEV), 2015 International Conference on*, June 2015, pp. 1–6.

[8] M. Kiaei, C. Assi, and B. Jaumard, “A survey on the p-cycle protection method,” *IEEE, Communications Surveys Tutorials*, vol. 11, no. 3, pp. 53 –70, quarter 2009.

[9] A. Kodian and W. Grover, “Failure-independent path-protecting pcycles: efficient and simple fully preconnected optimal-path protection,” *IEEE, J. Lightwave Technol.*, 2005.

[10] P. M. Moura and A. C. Drummond, “FlexGridSim: Flexible Grid Optical Network Simulator,” http://www.lrc.ic.unicamp.br/FlexGridSim/.