The Minimum Interference p-Cycle Algorithm for Protection of Space Division Multiplexing Elastic Optical Networks

Helder M. N. S. Oliveira   *Nelson L. S. da Fonseca*

Technical Report - IC-17-04 - Relatório Técnico
April - 2017 - Abril

UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE COMPUTAÇÃO

The Minimum Interference p-Cycle Algorithm for Protection of Space Division Multiplexing Elastic Optical Networks

Helder M. N. S. Oliveira * Nelson L. S. da Fonseca †

Abstract

In optical networks, failures can imply in great loss of data due to high transmission rates, leading to the need of employment of protection mechanisms. This paper introduces a novel algorithm to provide Failure-independent path protecting p-cycle with minimum interference for path protection in elastic optical networks using space division multiplexing. The proposed protection algorithm reduces rejections of future requests and make no assumption about specific patterns of arrival of requests. The algorithm is compared to FIPPMC algorithm and a algorithm based on methods of [2]. Results indicate that the 100% protection for single failures can be provided by the proposed algorithm.

1 Introdução

The increasing demand of bandwidth and the rapid approaching capacity limitation of single-core optical fibers has led to the exploitation of the only unused dimension to increase the network capacity. Space division multiplexing (SDM) introduces the concept of using multiple fibers in parallel, providing an n-fold increase in the usable spectral resources and the introduction of a new "space" dimension [3]. Space division multiplexing can be realized using multimode fiber (MMF), multicore Fiber (MCF) and few-mode multicore fiber. In MMF, the number of modes supported by a fiber depends on the core size and the refraction index of the fiber cladding. In MCF, each core acts as a single mode fiber. Moreover, new techniques need to be developed to realize SDM.

The routing and spectrum assignment (RSA) problem is a fundamental problem in elastic optical networks (EON). In RSA, there are constraints assuring contiguous and continuous allocation of the spectrum on all links of the selected route [4]. However, in SDM, it is possible to allocate one or more cores for the establishment of a connection. The inclusion of the space degree of freedom adds another dimension to the RSA problem becoming the routing, spectrum and core allocation (RSCA) problem. Moreover, in RCSA additional issues such as inter-core crosstalk should be taken into account. Inter-core crosstalk happens when the same spectrum propagates through adjacent cores in MCF.

Most of the protection techniques reserve but do not pre-configure backup resources, which may result in long signaling procedure during restoration [5]. p-Cycle is a protection technique with pre-configured backup resources. The spare capacity is used to provide protection to the working paths. p-Cycle can protect all the on-cycle spans as well as straddling spans. p-Cycle combines the advantage of mesh networks with the restoration speed of ring networks [6]. A special case of p-cycle for path protection is the so called Failure-Independent Path Protecting p-cycles (FIPP) [7].

*Institute of Computing - University of Campinas, Brazil, 13081-970 Campinas, SP. This research was partially sponsored by CNPq, process number 165446/2015-3

†Institute of Computing - University of Campinas, Brazil.

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 failure is not limited to a link or path segment immediately adjacent to the end nodes. FIPP p-cycle has been studied for protecting EONs. However, no study has shown FIPP p-cycles for protection of elastic optical network with SDM (SDM-EONs) [7, 8, 9, 10, 5].

The traditional protecting algorithms leads to a rapid saturation of network links, propelling new algorithms, specially those employing minimum interference to promote a balanced utilization of resources. Minimum interference algorithms generate connections along paths that least interfere with incomming requests for connection establishment [11, 12]. p-Cycle protecting a request can overload links, since p-cycle can use the same links of primary paths. The idea is to generate straddling p-cycles preventing p-cycles and paths to use the same links, therefore, minimizing the future request rejection ratio.

In this paper, we propose an algorithm called Minimum Interference and Failure-independent path protecting for MultiCore networks (MIFMC) for providing FIPP p-cycle protection in SDM-EONs. The MIFMC algorithm prioritizes the use of straddling p-cycles in order to generate minimum interference to reduce rejections of future requests. Results show that the proposed algorithm promotes protection effectively without compromising networking blocking.

This paper is organized as follows. Section 2 reviews related work. Section 3 introduces the proposed algorithm. Section 4 evaluates the performance of the proposed algorithm and Section 5 concludes the paper.

2 Related Work

The emergence of elastic optical networks has motivated several investigations, mainly on RSA algorithms but only recently RSCA solutions have been proposed. The authors in [2] divided the RSCA problem into the routing and SCA problems, and introduced a K-shortest path based pre-computation method as the routing solution. They proposed SCA methods with crosstalk awareness. In [13], it is investigated the spectrum fragmentation issue, which undermines the bandwidth efficiency in elastic optical networks. Fujii *et al.*[4] proposed an "on-demand" spectrum and core allocation method to reduce both crosstalk and fragmentation in elastic optical networks with MCFs. Proietti *et al.* [14] extends 2D-EON to include elasticity in all three domains: time, frequency, and space. They investigated algorithms for routing, spectrum, spatial mode, and modulation format assignment. The authors in [15] investigated the routing, spectrum and core allocation (RSCA) problem for flexgrid optical networks. They formulated the RSCA network planning problem using integer linear programming (ILP) formulation as well a heuristic. In [16], it is proposed an architecture for optical cross-connect (OXC) called architecture on demand (AOD) and it is shown that AOD provides much higher flexibility than do other architectures. The optimization problem for a cost-efficient architecture for SDM networks with AoD OXCs is formalized as an integer linear programming (ILP) problem. In [17], it was proposed an FIPP-p-cycle for protection for elastic SDM-EONs. However the minimum interference criteria for p-cycle creation was not considered.

The implementation of FIPP-p-cycles in EONs was studied in [8, 9, 18, 10]. Only in [17], has been proposed protection for SDM-EONs however minimum interference routing was not considered.

3 The Algorithm

Similar to the routing and spectrum assignment (RSA) problem in elastic optical networks, solutions for the Routing, Spectrum and Core Assignment (RSCA) problem in elastic optical networks are needed to efficiently accommodate diverse traffic demands. In an RSCA formulation, routing can switch cores in different links. Moreover, the problem formulation needs to derive the spectrum continuity constraint that imposes that the allocation of the same spectrum in each fiber along the route of a lightpath, and the spectrum contiguity constraint that imposes that the slots must be contiguously allocated in the spectrum.

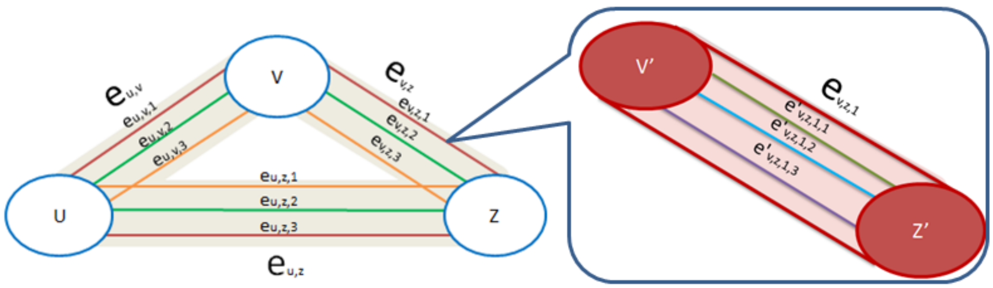


Figure 1: Multigraph

The proposed algorithm models the spectrum availability in the network as labeled multigraph (Figure 1). 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 set of same slots (different cores) in the link connecting the OXCs. All the vertices are connected by N edges which is the number of slots in the spectrum of each network link, and each edges represents the availability of at least one slot, regardless of the core. Labels on an edge represent the availability of a set of slots. An ∞ value means that the all slots are already allocated whereas the value 1 means that at least one slot is available for allocation. These values were defined to facilitate the employment of traditional shortest path algorithms. Notation used in this paper is summarized in Table 1.

3.1 MIFMC Algorithm

The algorithm introduced in this subsection, called Minimum Interference and Failure-independent path protecting for MultiCore networks (MIFMC) decides on the establishment of lightpaths in an FIPP p-cycle protected network. 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 paths.

An FIPP p-cycle protects disjoint primary paths. Requests to lightpath establishment arrive dynamically and for each request an existing p-cycle is searched to protect the potential lightpath. In case no existing p-cycle can protect the potential lightpath then a path is searched to create a new p-cycle for the request. If no path can protect the lightpath then it is not established.

The MIFMC algorithm assures a protection path for each established lightpath and the protection is guaranteed for single failures. The reservation of resources to create the FIPP p-cycle protecting a request can overload links in the network, since p-cycle protecting on-cycle paths can use the same links of the primary path. On the other hand, p-cycle protecting straddling paths tend to reserve more resources, since it has a greater number of hops. p-Cycles protecting on-cycle paths use less network resources than p-cycle protecting straddling paths, and they can overload the links along a path. Therefore, it is necessary to adopt criteria to avoid the formation of bottlenecks,

Table 1: Notation

s: source node;
d: destination node;
b: bandwidth demand in slots, $b = 1 \ldots N$;
$r(s, d, b)$: request from the node s to the node d with bandwidth demand b in slots;
N: number of slots set between two nodes;
C: number of cores set 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;
$E = \{e_{u,v,n}\}$: set of n edges;
$e_{u,v,n}$: the n^{th} edges connecting u and v;
$e'_{u,v,n,j}$: where j is a core chosen to be used.
$w(e_{u,v,n})$: 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 $\widetilde{W}$ 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 $n^{th}edge$;
$\widetilde{V} = V$: set of nodes;
$\widetilde{e}_{u,v} \in \widetilde{E}$: 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 $\widetilde{e}_{\widetilde{u},\widetilde{v}}$;
$\widetilde{W}_n = \{\widetilde{w}_n(\widetilde{e}_{\widetilde{u},\widetilde{v}})\}$: 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;
$W_{P_{s,d}}$ = weight of the shortest path between s and d;
$\widetilde{t}_{u,v,b}$: p-cycle containing vertices u and v and edges corresponding to the mapping of b edges of the multigraph G;
$\widetilde{T}_{u,v,b} = \widetilde{t}_{u,v,b}$: set of all p-cycles containing vertices u and v and edges corresponding to the mapping of b edges of the multigraph G;
$\widetilde{T}$: set of all established p-cycles and active;
T_n: chain of $\widetilde{G}_n$ such that the source node s is the least ordered node and d is the greatest ordered node;
$W(\widetilde{T}_n)$: $\sum_{\widetilde{e}_{\widetilde{u},\widetilde{v}} \in \{\widetilde{T}_n\}} \widetilde{e}_{\widetilde{u},\widetilde{v}}$: the weight of the p-cycle $\widetilde{T}_n$ (the sum of the weights of all the edges in the chain);
$W_{T_{s,d}}$ = weight of the p-cycle will protect the path between s and d;

balancing the load among potential paths, i.e., it is necessary to adopt a minimum interference approach to avoid blocking of incoming connections.

Algorithm 1 MIFMC

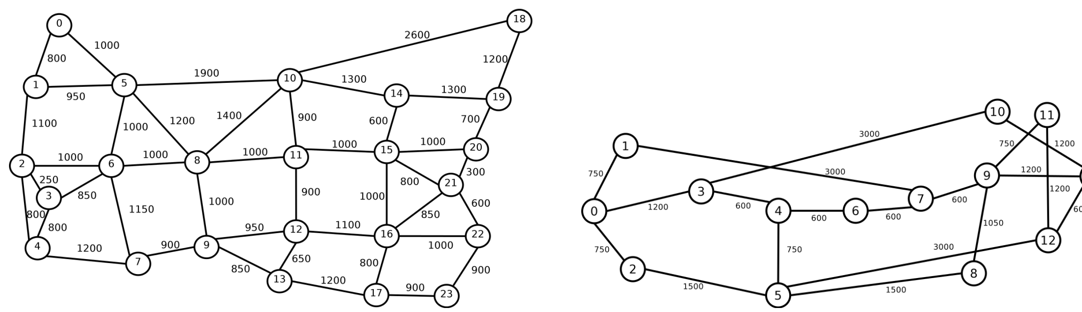
1: $\forall n = 1 ... N{-}b$
2: $\quad (W(P_n), P_n) = ShortestPath(\widetilde{G}_{n,b}, r(s,d,b))$
3: $W_{P_{s,d}} = W(P_n) |\ \forall i\ W(P_n) \leq W(P_i)$
4: **if** $W_{P_{s,d}} = \infty$ **then**
5: $\quad$ *block* $r(s,d,b)$
6: **else**
7: $\quad$ **if** $T_n \neq \emptyset \quad \forall\ T_n \in \widetilde{T}$ **then**
8: $\quad\quad$ establish $r(s,d,b)$ as P_n and T_n
9: $\quad\quad W(e'_{u,v,i}) = \infty \quad \forall \{u,v\} \in \widetilde{P}_i \quad n = n ... i{+}b-1$
10: $\quad$ **else**
11: $\quad\quad (W(T_n), T_n) = StraddlingCycle\ (\widetilde{G}_{n,b}, r(s,d,b))$
12: $\quad\quad W_{T_{s,d}} = W(T_n) |\ \forall i\ W(T_n) \leq W(T_i)$
13: $\quad\quad$ **if** $W_{T_{s,d}} = \infty$ **then**
14: $\quad\quad\quad (W(T_n), T_n) = ShortestCycle\ (\widetilde{G}_{n,b}, r(s,d,b))$
15: $\quad\quad\quad W_{T_{s,d}} = W(T_n) |\ \forall i\ W(T_n) \leq W(T_i)$
16: $\quad\quad\quad$ **if** $W_{T_{s,d}} = \infty$ **then**
17: $\quad\quad\quad\quad$ block $r(s,d,b)$
18: $\quad\quad\quad$ **end if**
19: $\quad\quad$ **end if**
20: $\quad\quad$ **if** $W_{T_{s,d}} \neq \infty$ **then**
21: $\quad\quad\quad$ establish $r(s,d,b)$ as $\widetilde{P}_n$ and $\widetilde{T}_n$
22: $\quad\quad\quad W(e'_{u,v,i}) = \infty \quad \forall \{u,v\} \in \widetilde{P}_i \quad n = n ... i{+}b-1$
23: $\quad\quad\quad W(e'_{u,v,i}) = \infty \quad \forall \{u,v\} \in \widetilde{T}_i \quad n = n ... i{+}b-1$
24: $\quad\quad$ **end if**
25: $\quad$ **end if**
26: **end if**

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 was not possible to find a path under the contiguity constraint for the demand b with allocation starting with 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, a p-cycle to protect the lightpath to be established is searched (Line 7). In case, there exists a p-cycle, the lightpath is established (Line 8) and the corresponding edges in the multigraph G have their weight changed to ∞ (Line 9) meaning that the slots were allocated to the newly established lightpath. Otherwise, a p-cycle with minimum interference is created. To avoid the creation of bottlenecks, a straddling p-cycle to protect the lightpath to be established should be created (Lines 11). In case, no straddling p-cycle can be created to protect the lightpath then the shortest p-cycle is created (Lines 13 and 14). In case, no p-cycle can be created to protect the lightpath then the request is blocked (Line 17), otherwise the lightpath as well as the p-cycle (Line 21) are established to satisfy the request and the corresponding edges in the multigraph G have their weight changed to ∞ (Line

22 and 23) meaning that the slots were allocated to the newly established lightpath.

4 Performance Evaluation

To assess the performance of MIFMC algorithm in multi-core networks, simulation experiments were employed using 7 cores. The FlexGridSim [19] simulator was employed. In each simulation, 100,000 requests were generated and simulations for all the algorithms used the same set of seeds. Confidence intervals were derived using the independent replication method with 95% confidence level. The topology used in the simulations were the NSF (Figure 2b), and the USA (Figure 2a) topologies. The NSF topology has 16 nodes and 25 links whereas the USA topology has 24 nodes and 43 links (Figure 2).



(a) USA Topology (b) NSF Topology

Figure 2: Topologies

The spectrum was divided in 240 slots of 12,5 GHz each. In the figures, curves labeled "FIPPMC" show the results for networks using the FIPPMC algorithm [17], curves labeled "SSCA" show the results for networks using the algorithm based in the methods proposed in [2], and curves labeled MIFMC display results for networks using the MIFMC algorithm. The traffic load was increased in units of 0.25 for all the figures in the paper. In the SSCA algorithm, the primary path is treated independently, i.e., the routing problem and the SCA problem. This approach employs pre-computed multiple route. The backup path is created in the same way. However, the backup path uses scheme 1:N.

Figure 3 shows the bandwidth blocking ratio (BBR) as a function of the traffic load for the USA topology. While FIPPMC and SSCA start blocking request under loads of 60 and 80 erlangs, respectively, MIFMC starts blocking only under loads of 100 erlangs. MIFMC produces bandwidth blocking ratio two order of magnitude lower under 100 erlangs. Under loads of 200 erlangs the difference between the BBR produced by the MIFMC algorithm and that given by the SSCA is almost two order of magnitude and 40% when compared with the FIPPMC algorithm. Such lower BBR produced by MIFMC evinces the benefit of considering the minimum interference criteria to create the p-cycles when choosing the backup route. These results show that the MIFMC algorithm produces acceptable blocking for SDM with multi core fibers in despite of the bandwidth reservation for pre provisioning of backup paths.

The use of seven cores generates Intercore crosstalk. Figure 4 shows the "Crosstalk per Slot" (CpS) as a function of the traffic load for USA topology. The crosstalk value for each spectrum slot is defined as the ratio of actual crosstalk index to the maximum value of crosstalk index. The crosstalk ratio is defined by the average of values among all spectrum slots [4]. The CpS is not considered when the slot is reserved but not used. The generated CpS for the FIPPMC algorithm starts at a 0.18 value and increases with the load increase. The same happens with the generated

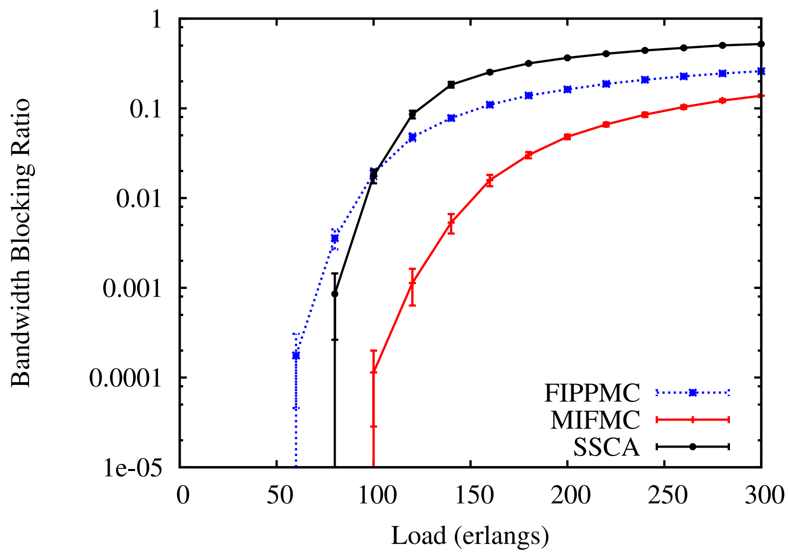


Figure 3: Bandwidth blocking ratio for the USA topology

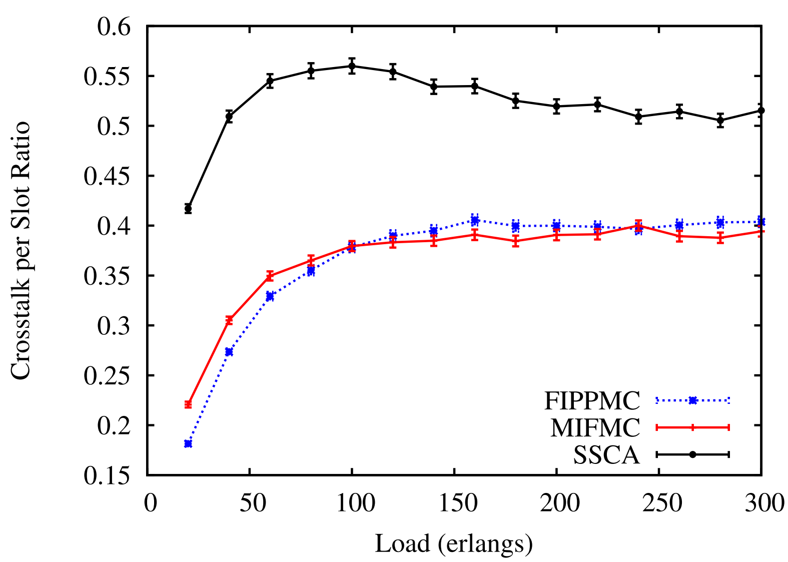


Figure 4: Crosstalk per slot ratio for the USA topology

CpS for the MIFMC algorithm starting at a 0.22 value. The SSCA algorithm has higher CpS than the others algorithms. However, although the MIFMC algorithm leads to low blocking and higher usage of the full network capacity, it has similar CpS than FIPPMC algorithm. The generated CpS for the SSCA algorithm remains between 0.42 and 0.56 under heavy loads. Note that the interleaved use of cores for primary and backup paths decreases CpS generated.

Figure 5 displays the Jain Fairness Index (JFI) of the BBR for different source destination pairs and for the USA topology. The SSCA algorithm applied has high Jain index values due to high blocking produced which affects uniformly all source destination pairs. The MIFMC algorithm produces low Jain Index of fairness since several source destination pairs do not suffer blocking, and therefore there is a greater disparity of BBR values. The FIPPMC algorithm produces high

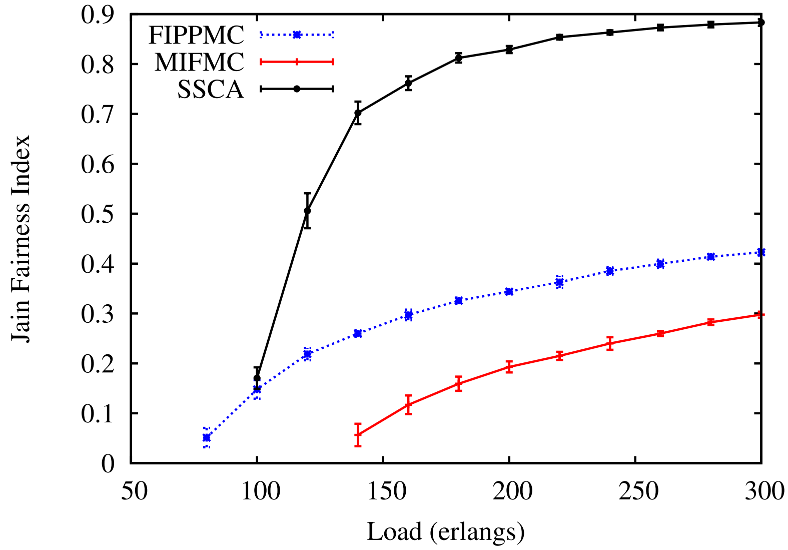


Figure 5: Jain fairness index for the USA topology

Jain Index of fairness than MIMF since several source destination pairs do not suffer blocking.

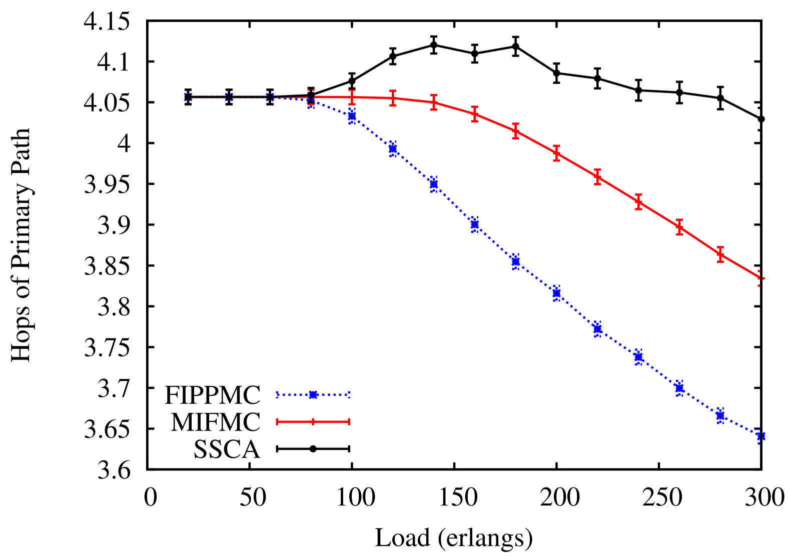


Figure 6: Average number of hops allocated per primary path for the USA topology

Figures 6 shows the average number of hops of primary paths established for the USA topology. The higher the load, the lower is the average number of hops allocated per primary path. Until 80 erlangs loads, the FIPPMC, SSCA and MIFMC demand similar number of hops per primary path. Under higher loads the SSCA demands higher average number of hops allocated per primary path than do the other algorithms. Under high loads the FIPPMC demands low average number of hops allocated per primary path.

Figures 7 shows the average number of hops of backup paths established for the USA topology. The SSCA algorithm produces an almost constant number of hops allocated per backup path regardless of the network load. The number of primary backup allocated by the MIFMC algorithm

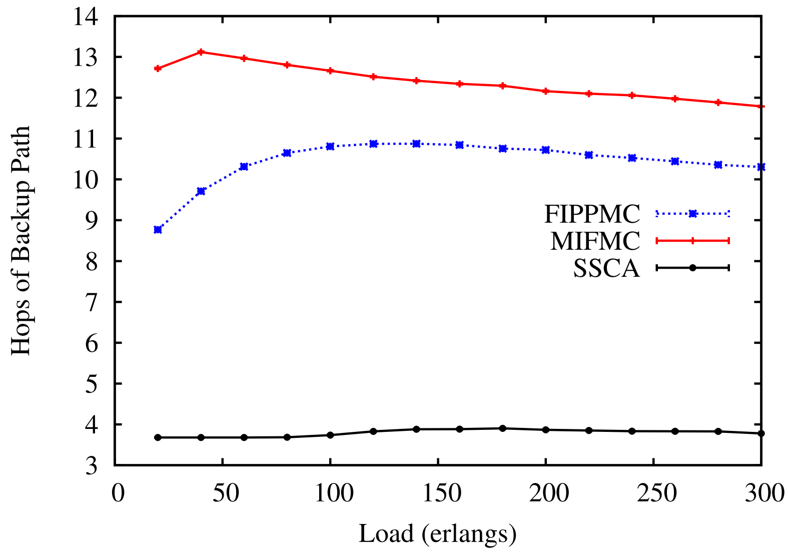


Figure 7: Average number of hops allocated per backup path for the USA topology

is always higher than those demanded by the others algorithms. This is the cost for the creation of p-cycle generating minimum interference. The number of primary backup allocated by the SSCA algorithm is always lower than those of others algorithms.

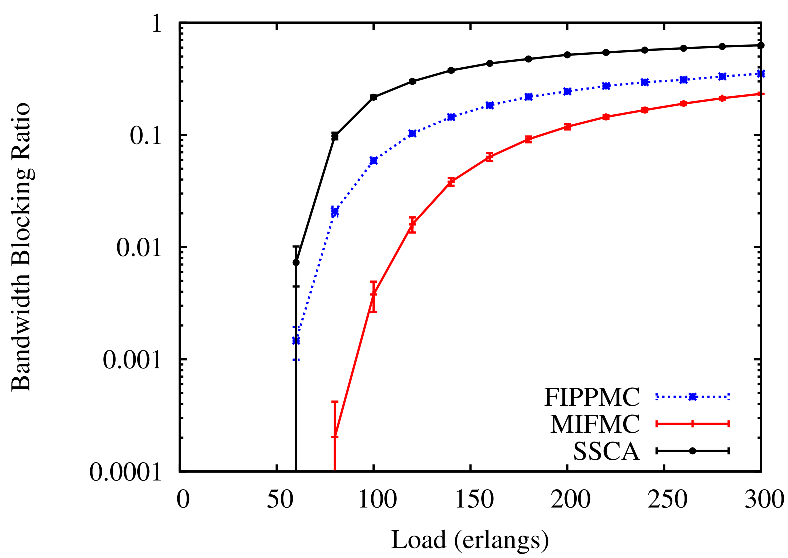


Figure 8: Bandwidth blocking ratio for the NSF topology

Figure 8 shows the bandwidth blocking ratio (BBR) as a function of the traffic load for the NSF topology. While SSCA and FIPPMC start blocking request under loads of 60 erlangs, MIFMC algorithm starts blocking only under loads of 80 erlangs. Under loads of 80 erlangs, the difference between the BBR produced by the MIFMC algorithm and that given by the SSCA algorithm is almost three order of magnitude and one order of magnitude when compared to that produced by FIPPMC. Under loads of 200 erlangs the difference between the BBR produced by the MIFMC algorithm and that given by the SSCA algorithm is one order of magnitude.

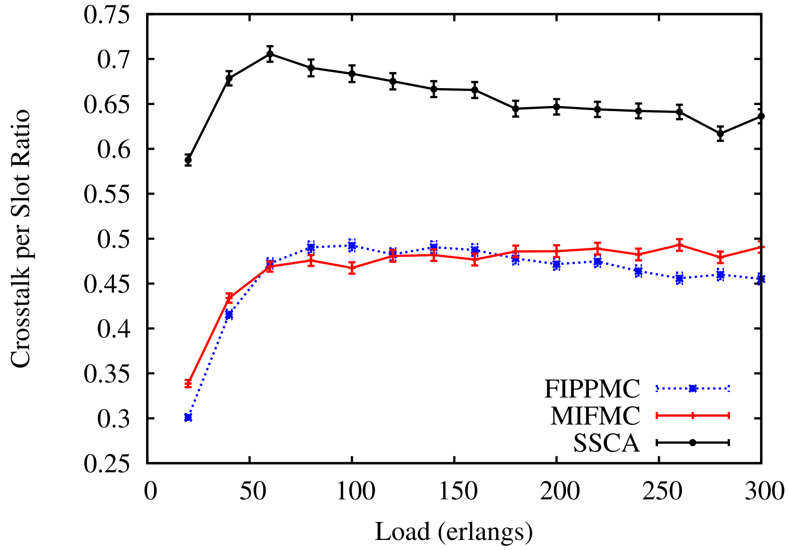


Figure 9: Crosstalk per slot ratio for the NSF topology

Figure 9 shows the "Crosstalk per Slot" (CpS) as a function of the traffic load for the NSF topology. The generated CpS for the MIFMC algorithm is 0.34 and this value increases until 0.49. This also happens with the generated CpS for the FIPPMC algorithm starting at 0.30 until 0.50. The SSCA algorithm produces the highest CpS value. The generated CpS for the SSCA algorithm remains between 0.57 and 0.70. Under heavy, loads the generated CpS by FIPPMC and SSCA algorithms decrease.

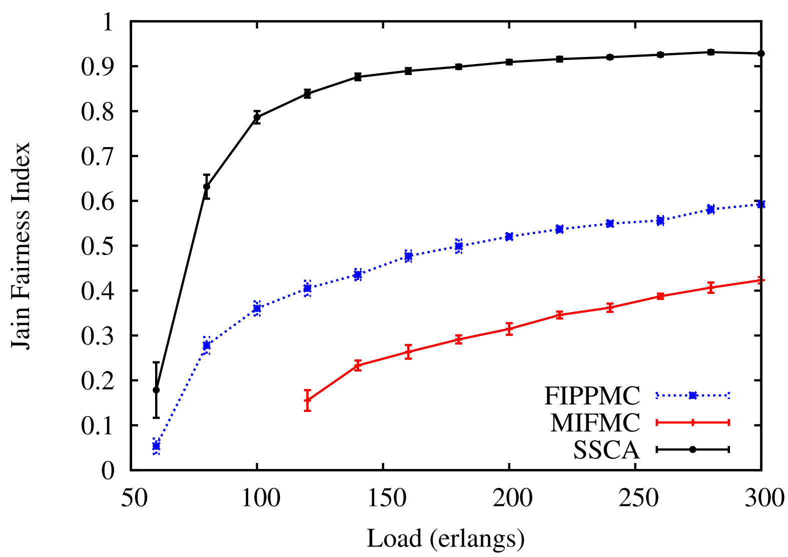


Figure 10: Jain fairness index for the NSF topology

Figure 10 displays the Jain Fairness Index (JFI) of the BBR for different source destination pairs for the topology NSF. The SSCA algorithm has high Jain index values, when compared with MIFMC and FIPPMC algorithms. The SSCA algorithm applied has high Jain index values due to high blocking produced which affect uniformly all source destination pairs. However MIFMC provide lower JFI value than do the FIPPMC algorithm.

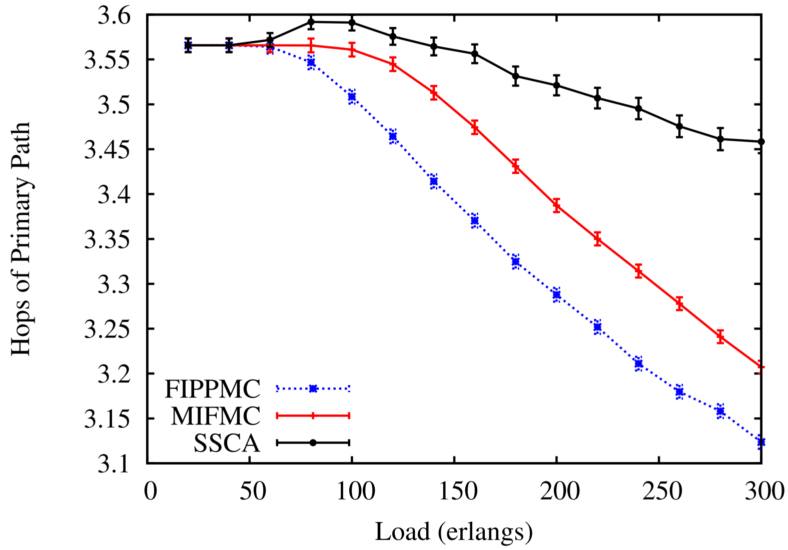


Figure 11: Average number of hops allocated per primary path for the NSF topology

Figures 11 shows the average number of hops of the primary paths established for the NSF topology. Until 40 erlangs loads, the number of hops allocated per primary path by the algorithms are similar. Under loads higher than 100 erlangs, the SSCA demands an average number of hops allocated per primary path higher than do the others algorithms. Under high loads, the MIFMC algorithm demands lower average number of hops allocated per primary path than do the others algorithms.

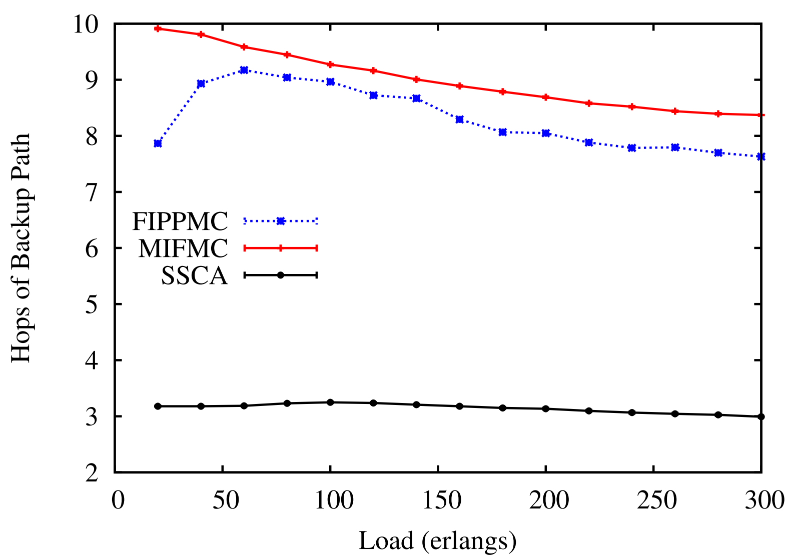


Figure 12: Average number of hops allocated per backup path for the NSF topology

Figures 12 shows the average number of hops of backup paths established for the NSF topology. The SSCA algorithm demands an almost constant number of hops allocated per backup path regardless of the network load. The number of primary backup paths allocated by the SSCA algorithm is always lower than those allocated by the others algorithms. The number of primary backup paths allocated by the MIFMC algorithm is always higher than those allocated by the others algorithms.

5 Conclusion

In elastic optical networks with SDM, large amounts of data can be lost when failure of network links occur, demanding protection mechanisms for connection. This paper introduced an algorithm to support the establishment of lightpaths in elastic optical networks with SDM protected by FIPP p-cycles. The algorithm was evaluated for different topologies and traffic loads. The algorithm was compared with other algorithm in literature. Results indicate that the proposed algorithm can provide efficiently pre-configured protection for SDM in MCF networks. MIFMC algorithm has lower MBBR in topologies with high connectivity. The node degree in a network topology has great influence in the bandwidth blocking ratio and on the length of established paths.

References

[1] H. M. N. S. Oliveira and N. L. S. da Fonseca, "The minimum interference p-cycle algorithm for protection of space division multiplexing elastic optical networks," in *IEEE Latin America Transactions (LATINCOM), 2016 IEEE.* IEEE, 2016.

[2] H. Tode and Y. Hirota, "Routing, spectrum and core assignment for space division multiplexing elastic optical networks," in *Telecommunications Network Strategy and Planning Symposium (Networks), 2014 16th International*, Sept 2014, pp. 1–7.

[3] D. Siracusa, F. Pederzolli, D. Klonidisz, V. Lopezy, and E. Salvadori, "Resource allocation policies in sdm optical networks (invited paper)," in *Optical Network Design and Modeling (ONDM), 2015 International Conference on*, May 2015, pp. 168–173.

[4] S. Fujii, Y. Hirota, H. Tode, and K. Murakami, "On-demand spectrum and core allocation for reducing crosstalk in multicore fibers in elastic optical networks," *Optical Communications and Networking, IEEE/OSA Journal of*, vol. 6, no. 12, pp. 1059–1071, Dec 2014.

[5] X. Chen, S. Zhu, L. Jiang, and Z. Zhu, "On spectrum efficient failure-independent path protection p-cycle design in elastic optical networks," *Lightwave Technology, Journal of*, vol. 33, no. 17, pp. 3719–3729, Sept 2015.

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

[7] A. Kodian and W. Grover, "Failure-independent path-protecting pcycles: efficient and simple fully preconnected optimal-path protection," *IEEE J. Lightwave Technol.*, vol. 23, pp. 3241–3259, Oct. 2005.

[8] H. M. Oliveira and N. L. da Fonseca, "Algorithm for FIPP p-cycle path protection in flexgrid networks," in *Global Communications Conference (GLOBECOM), 2014 IEEE.* IEEE, 2014, pp. 1278–1283.

[9] Y. Wei, K. Xu, Y. Jiang, H. Zhao, and G. Shen, "Optimal design for p-cycle-protected elastic optical networks," *Photonic Network Communications*, vol. 29, no. 3, pp. 257–268, 2015.

[10] X. Chen, S. Zhu, L. Jiang, and Z. Zhu, "On spectrum efficient failure-independent path protection p-cycle design in elastic optical networks," *Journal of Lightwave Technology*, vol. 33, no. 17, pp. 3719–3729, 2015.

[11] G. B. Figueiredo, N. L. S. da Fonseca, and J. A. S. Monteiro, "A minimum interference routing algorithm," in *Communications, 2004 IEEE International Conference on*, vol. 4, June 2004, pp. 1942–1947 Vol.4.

[12] G. B. Figueiredo, N. L. da Fonseca, and J. A. Monteiro, "A minimum interference routing algorithm with reduced computational complexity," *Computer Networks*, vol. 50, no. 11, pp. 1710 – 1732, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128605002483

[13] Y. Yin, H. Zhang, M. Zhang, M. Xia, Z. Zhu, S. Dahlfort, and S. Yoo, "Spectral and spatial 2d fragmentation-aware routing and spectrum assignment algorithms in elastic optical networks [invited]," *Optical Communications and Networking, IEEE/OSA Journal of*, vol. 5, no. 10, pp. A100–A106, Oct 2013.

[14] R. Proietti, L. Liu, R. Scott, B. Guan, C. Qin, T. Su, F. Giannone, and S. Yoo, "3d elastic optical networking in the temporal, spectral, and spatial domains," *Communications Magazine, IEEE*, vol. 53, no. 2, pp. 79–87, Feb 2015.

[15] A. Muhammad, G. Zervas, D. Simeonidou, and R. Forchheimer, "Routing, spectrum and core allocation in flexgrid sdm networks with multi-core fibers," in *Optical Network Design and Modeling, 2014 International Conference on*, May 2014, pp. 192–197.

[16] A. Muhammad, G. Zervas, G. Saridis, E. Salas, D. Simeonidou, and R. Forchheimer, "Flexible and synthetic sdm networks with multi-core-fibers implemented by programmable roadms," in *Optical Communication (ECOC), 2014 European Conference on*, Sept 2014, pp. 1–3.

[17] H. M. S. Oliveira and N. L. S. da Fonseca, "Algorithm for protection of space division multiplexing elastic optical networks," in *Global Communications Conference (GLOBECOM), 2016 IEEE*. IEEE, 2016.

[18] G. Shen, H. Guo, and S. K. Bose, "Survivable elastic optical networks: survey and perspective," *Photonic Network Communications*, vol. 31, no. 1, pp. 71–87, 2016.

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