Machine Learning for Spectrum Defragmentation in Space-Division Multiplexing Elastic Optical Networks

Silvana Trindade and Nelson L. S. da Fonseca

Abstract

In Elastic Optical Networks with Space Division Multiplexing, the dynamic allocation, and deallocation of frequency slots can generate spectrum fragmentation, which increases the blocking of requests for lightpath establishment. In this paper, we introduce a reactive algorithm and a proactive one that can jointly reduce the spectrum fragmentation. We introduce a novel defragmentation approach based on an unsupervised machine learning technique to rearrange a fragmented spectrum by clustering lightpaths. A Routing, Modulation format, Core, and Spectrum Allocation algorithm uses information on the clustering of lightpaths to establish new lightpaths for incoming requests. Results show that our approach can reduce the blocking of requests and the spectrum fragmentation.

I. Introduction

Recently, advances in optical communication have addressed the exponential increase of Internet traffic. One of these advances is the Elastic Optical Network (EON) technology, which allows the allocation of the spectrum at a much finer granularity of frequency slots of 12.5 GHz. Moreover, the increasing demands for bandwidth can no longer be supported by the Single-Mode Fiber technology. Its capacity will soon be exhausted. In the face of these limitations, Spatial-Division Multiplexing (SDM) has been introduced to EONs to increase the network capacity [1]. SDM uses spatial channels and can be attained by Multi-Core Fiber (MCF) technology in which each core performs as a single-mode fiber.

To establish a lightpath, a block of contiguous slots must be allocated (contiguity constraint), and these blocks must be allocated in all the hops along the lightpath (continuity constraint). In EON with SDM (EON-SDM), a solution of the so-called Routing, Modulation Format, Core, and Spectrum Allocation (RMCSA) problem should be obtained for the establishment of a lightpath. A typical RMCSA algorithm performs the following steps: a) search for a route, b) choice of modulation format, c) calculation of the number of slots needed as a function of the modulation format, and d) allocation of a set of available slots in the links along the chosen route.

Setting up and tearing down lightpaths results in fragmentation of the spectrum, a state in which there are slots available to satisfy a bandwidth request but they are not contiguous and continuous, and, therefore, cannot be allocated to the requested lightpath. The spectrum fragmentation can increase the blocking of incoming requests for lightpath establishment, especially under heavy loads [2].

Figure 1 illustrates a fragmented network spectrum. An arriving request with a 150 *Gb* bandwidth requirement will need four contiguous and continuous slots, and an additional slot for a guard band if the 8-Quadrature Amplitude Modulation (QAM) format is employed. An RMCSA algorithm finds two candidate paths ($P1$ and $P2$). However, because of the state of the spectrum fragmentation, this request cannot be established. The bottom part of Fig. 1 illustrates the fragmented spectrum, in which the available slots are scattered in small blocks, making them non-allocable since they are neither contiguous nor continuous. At the very bottom of Fig. 1, the same path is shown after the spectrum is re-organized (defragmented), now allowing, the allocation of the required bandwidth.

Inter-core crosstalk interference results from the optical power added to the power of a signal by adjacent links. Crosstalk is the most important physical impairment in spectrum allocation in a multi-core fiber and must be considered. If considerable strong crosstalk occurs on a slot, the Quality of Transmission (QoT) can degrade so badly that the slot will have to be considered unavailable for allocation; such unavailability may augment the spectrum fragmentation even more.

There are two types of solutions for handling the fragmentation problem: proactive and reactive. Proactive solutions try to reduce and prevent the occurrence of fragmentation by finding a route and a block of slots for allocation, thus increasing the probability of allocation of slots for incoming requests. Proactive solutions can handle the fragmentation problem for light traffic loads, although under heavy loads, the constant allocation and de-allocation of lightpaths will leave small groups of continuous slots, regardless of how good the algorithm is for low loads.

Reactive solutions [3], [4], [2], on the other hand, focus on the spectrum defragmentation; they reduce fragmentation by reallocating and/or rerouting a set of existing lightpaths. Rerouting reallocates such existing lightpaths to a different route, whereas defragmentation without rerouting changes only the allocation of slots. Defragmentation solutions reorganize the spectrum to

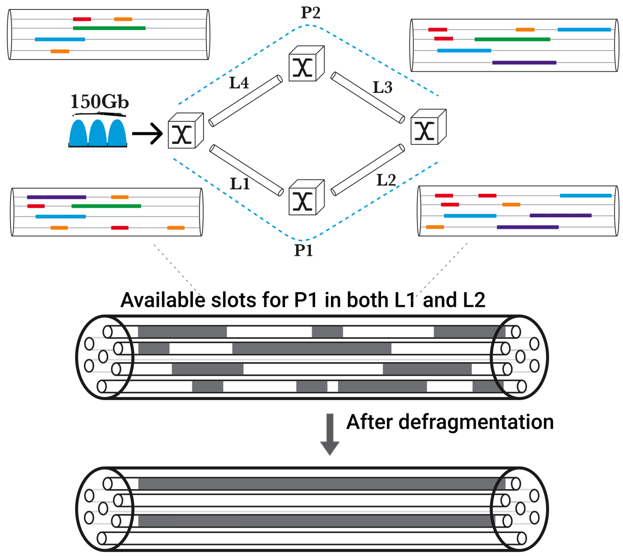


Fig. 1. EON-SDM optical network with high fragmentation.

make continuous and contiguous slots available, thus increasing the chances of the acceptance of incoming requests for lightpath establishment [5].

Most of the existing proposals for the reduction of fragmentation adapt algorithms designed for single-core networks for use in multi-core networks [6], [7], [2]. However, the problem with traditional defragmentation algorithms is that after spectrum defragmentation, the RMCSA algorithm continues to accept requests following no criterion for the avoidance of future fragmentation of the spectrum. Therefore, defragmentation must be repeatedly executed. To mitigate this problem, we propose both a defragmentation solution and a proactive fragmentation-aware RMCSA algorithm, to reduce the need for frequent defragmentation. Our approach capitalizes on the advantages of both proactive and reactive approaches.

The main contribution of this paper is a novel approach that combines an independent defragmentation algorithm with a new RMCSA algorithm. The defragmentation algorithm, called Clusterization-Driven Spectrum Rearrangement (CISUR), uses Unsupervised Machine Learning (UML) to identify lightpaths that can be clustered in the spectrum in the basis of the features of these lightpaths. UML techniques do not require any training, yet they are suitable for dealing with the dynamic arrival of requests, and do not depend on previous knowledge. The CISUR algorithm maps clusters of lightpaths to cores and rearranges the spectrum without performing rerouting. The proactive algorithm, called the Clusterization RMCSA (C-RMCSA) algorithm

TABLE I

FRAGMENTATION-AWARE ALGORITHMS

References	**Techniques**	**Policy**	**Descriptions**
[8]	Proactive	Multi-graph	Allocates slots considering a multi-graph representation of the spectrum.
[7]	Proactive	Index	Allocates requests based on the slot index order.
[10]	Proactive	Partition	Uses Allocates requests using partitioning techniques.
[6]	Proactive	Multi-paths	Reduces the impact of fragmentation on blocking using multiple paths.
[12]	Proactive	Partition	Tries to reduce the fragmentation and crosstalk through traffic prediction.
[3]	Reactive	Defragmentation	Defragmentation of the spectrum using the push-pull technique.
[4]	Reactive	Rerouting	Reduces fragmentation when it occurs by using rerouting.
[2]	Reactive	Rerouting	Tries to reduce crosstalk by rerouting lightpaths.
CISUR	Reactive	Clustering	Performs defragmentation using a UML algorithm to cluster requests and feed the resulting information for the allocation of new requests.

leverages information on the clusterization of the spectrum and the features of the incoming request to allow the acceptance of new lightpaths into one of the clusters formed, therefore helping to prevent future fragmentation of the spectrum.

This paper is organized as follows. First, existing algorithms for the prevention and amelioration of the spectrum fragmentation are described. Then, the CISUR and C-RMCSA algorithms are introduced. Finally, concluding remarks and suggestions for future work.

II. Fragmentation Problem in EON

Various solutions have been proposed for the spectrum fragmentation problem. In [8], the authors use a multi-graph representation of the spectrum to prevent its fragmentation. Their approach models the network spectrum availability as a labeled multi-graph, where the vertices represent Optical Cross-Connect (OXC) switches and the edges are the slots. For each of the graphs generated, a shortest-path algorithm is executed to choose a path, using a set of functions which considers the evaluation of fragmentation metrics. In [7], the authors propose a set of variations on the First-Fit (FF) policy for the selection of slots. A spectrum partitioning criterion is used to prevent crosstalk interference. The spectrum is partitioned into areas, each having the same number of frequency slots.

One of the most common approaches in circuit switching networks is the grouping of channels into partitions and the association of requests to form partitions [9]. The aim is to provide differentiated services and reduce the blocking of requests for lightpath establishment. Although prioritized services can increase revenue, the rigid definition of partitions can lead to blocking of requests, even when slots are available in partitions not associated with a specific request type.

Another common approach is multipath routing, which has been used when the resource demands are greater than the slot availability in a single path. The demand is divided into a set of flows, with slot demands lower or equal to the slot availability along the chosen paths. In [11], multipath routing is adopted to decrease blocking due to fragmentation.

Various other proactive approaches have been proposed. In [12], the authors introduce an algorithm using supervised ML to predict traffic changes to avoid fragmentation. Their algorithm uses the Elman neural network to forecast traffic demands by employing a two-dimensional rectangular packing model to allocate the spectrum. However, proactive solutions cannot fully avoid fragmentation since frequent deallocation of the spectrum can increase its fragmentation.

Various reactive algorithms have been proposed for dealing with the dynamic deallocation of the spectrum. In [4], a defragmentation algorithm is executed after several lightpaths are torn down, which makes the selection of alternative routes and sets of slots available. After that, traffic migration is performed, and the Move-To-Vacancy (MTV) technique is employed to reduce the time during which the transmission is interrupted for reassigning the lightpath to another set of slots. However, this algorithm does not consider any measure of the state of the fragmentation of the spectrum.

In [2], the authors propose a parallel Simulated Annealing algorithm to rearrange light-paths in SDM networks, thus reducing crosstalk. However, parallelization results in significant communication overhead. The authors in [3] propose a technique called push-pull used in defragmentation to improve the QoT. They validated their proposal in two testbeds employing EON. This technique can re-tune a lightpath in few milliseconds, causing no traffic disruption even under high utilization. Moreover, it does not require any additional transponders. The push-pull technique can be employed in seamless spectrum reallocation.

Most of the existing proposals adapt algorithms used for single-core EONs to SDM-EONs

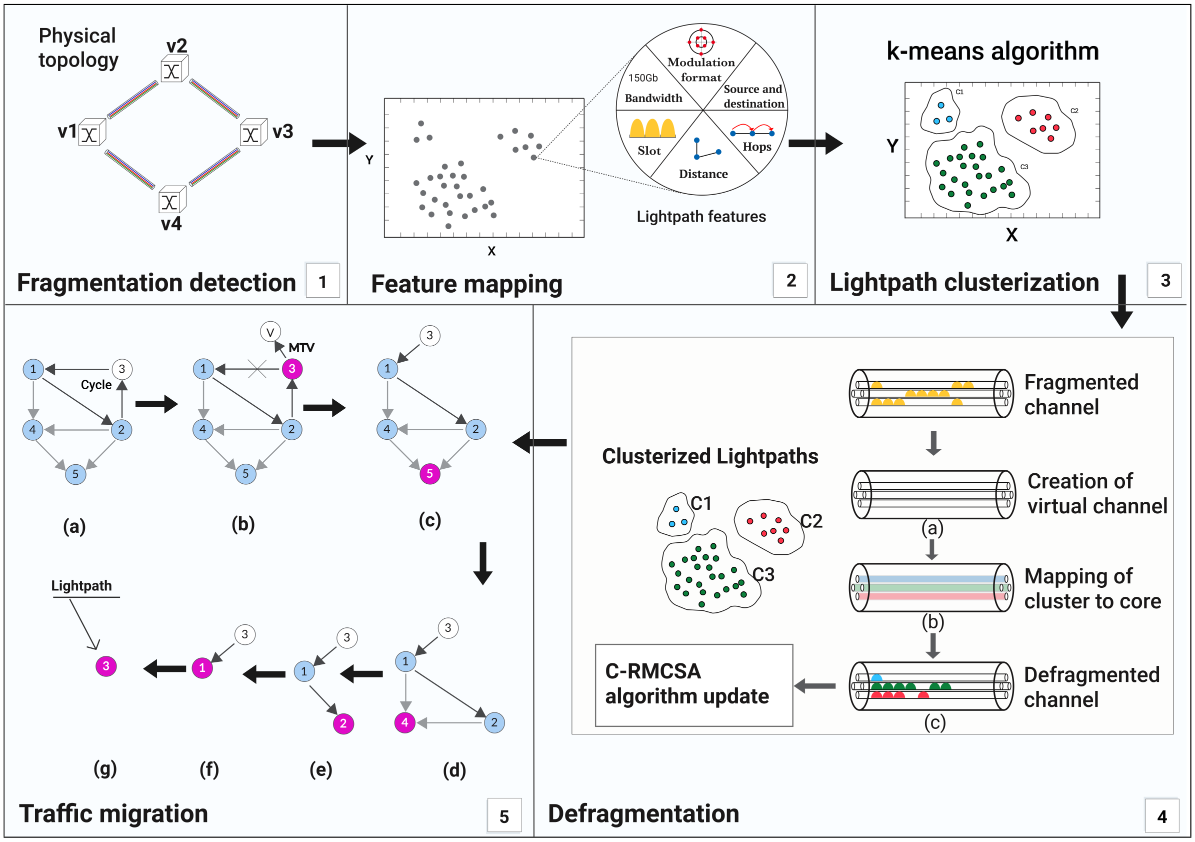


Fig. 2. Defragmentation flowchart in SDM networks using unsupervised learning. The CISUR algorithm is divided in the following steps: (1) verification of the need for defragmentation, (2) features mapping, (3) clusterization of lightpaths, (4) rearrangement, and (5) traffic migration.

based on MCFs. Such an approach does not fully explore the possibility of reducing fragmentation since it ignores the space dimension introduced by SDM. Few algorithms use reactive solutions to reduce fragmentation, and none of them employs information on defragmentation in decisions on routing and spectrum allocation for incoming requests. Moreover, another issue not well addressed by existing algorithms is the decision about when defragmentation should be undertaken.

Our solution capitalizes on both proactive and reactive approaches. The Clusterization-Driven Spectrum Rearrangement (CISUR) algorithm differs from existing ones by employing a UML algorithm. Moreover, information on spectrum clusterization is available to the C-RMCSA algorithm. The joint use of these two algorithms decreases spectrum fragmentation.

III. CISUR Algorithm

In the CISUR algorithm, the spectrum fragmentation is monitored; when monitoring metrics reach certain threshold values, defragmentation is undertaken. Figure 2 shows the steps of the defragmentation executed by the CISUR algorithm. The CISUR algorithm groups lightpaths into clusters, and the clusters are mapped onto cores based on pre-defined features. Fragments are removed, leaving the largest possible contiguous sets of slots available for incoming lightpaths.

The CISUR algorithm groups lightpaths on the basis of their similarities to rearrange them in the spectrum. Supervised learning techniques are not suitable for grouping lightpaths since labeling of the lightpaths would be required, as well as a consideration of a fixed number of clusters for all defragmentation events. Moreover, solutions based on Reinforcement Learning (RL) are not effective for the clustering of the lightpaths problem since RL is based on feedback from actions taken and convergence to an optimal solution may require a long time. However, unsupervised learning algorithms can rapidly find similarities among the lightpaths, without the need for model training.

The CISUR algorithm uses the k-Means clustering algorithm, which converges rapidly and is suitable for scenarios with the dynamic arrival of requests. The effectiveness of the k-Means algorithm was compared with that of other clustering algorithms, such as Mean-shift, Spectral Clustering, Gaussian Mixtures, Affinity Propagation, and Hierarchical Clustering, for the defragmentation of the spectrum. The Manhattan distance metric was used to group the lightpaths into clusters since this metric supports to high dimensionality (large number of features). With this employment the k-Means algorithm produced the lowest blocking of requests than did the other mentioned clustering algorithms.

The Silhouette Coefficient was used to determine the optimal k value. The Silhouette Coefficient is calculated using the mean intra-cluster distance and the mean nearest-cluster distance for each sample. The mean Silhouette Coefficient for all samples are computed and used as a metric to evaluate the proper number of clusters (k value).

Information on the clusters formed is then made available to the C-RMCSA algorithm. New lightpaths are then assigned to the clusters based on the matching of their features with those of the clusters.

A. Feature Extraction

Simulations comprising 10,000 Poisson arrivals of requests for lightpath establishment and using the CHNNET and ARPANET topologies [13] were carried out to produce a dataset for selecting the features relevant to defragmentation. Poisson arrivals are usually assumed for arrivals at the call level when the population generating the requests (users) is much larger than the available resources (maximum possible number of lightpaths in a link). The two topologies differ in the number of links and nodes as well as node connectivity, which increase the diversity of the dataset.

The dataset is composed of selected features that characterize the spectrum fragmentation. The considered set of features was: a) the *bandwidth* requested, b) the *number of slots* demanded by the modulation format chosen, c) the *modulation format* chosen for the lightpath, d) the *arrival* time of the request, e) the *deadline* for the start of the lightpath, f) the expected duration of the lightpath (*lifetime*), g) the pair of source-destination nodes, h) the *physical distance* between the source-destination pair, i) the *set of links* composing the lightpath, j) the *number of hops* along the route chosen, k) the *crosstalk* generated by the establishment of the lightpath and l) the *index of the core* hosting the lightpath. The selection procedure was executed once to define the features to be used by the CISUR and C-RMCSA algorithms. The diverse range of feature values were normalized to fall in the interval [0,1] by using the min-max normalization method.

We employed the ANalysis Of VAriance (ANOVA) technique to select the most relevant features. ANOVA compares the means of the number of features and determines which means are statistically significantly different from the others. We examined the mean of the clusters derived by ANOVA for each dimension to assess how distinct the clusters were. The F ratio was used as an indicator of how well a dimension discriminated the ANOVA clusters. As a result, features producing a similar impact on the spectrum fragmentation were eliminated, and only those translating data diversity were considered. The statistically significant features selected by ANOVA were: a) bandwidth, b) number of slots, c) modulation format, d) arrival time, e) lifetime, f) physical distance and g) the set of physical links. These features are used in the K-Means algorithm to group lightpaths into clusters.

B. Defragmentation

The defragmentation process of the CISUR algorithm is divided into five steps: fragmentation identification, feature collection from the requests, request clusterization, request rearrangement, and traffic migration. These steps are illustrated in Fig. 2, which describes the defragmentation procedure.

The need for defragmentation is identified by monitoring the network fragmentation ratio, the link utilization, the number of blocked requests, and the number of lightpaths torn down (Step 1 in Fig. 2). The fragmentation ratio is defined as the mean ratio between the number of isolated small-sized blocks of slots and the total number of slots in the spectrum [13]. These isolated blocks of slots are neither contiguous in the spectral-domain nor aligned along with the fiber links, which makes the allocation of slots to a lightpath difficult. Link utilization is observed for detecting if the network capacity is close to being fully utilized since in this state defragmentation will have very little impact on the making of slots available.

After evaluating different ranges of values for the mentioned metrics, we concluded that the threshold values for triggering the defragmentation procedure with a high chance of reducing fragmentation are the following: link utilization less than 0.85, more than 20% of lightpaths torn down and fragmentation ratio and blocking ratios greater than 0.2 and 0.3, respectively. These metrics are observed in a period starting with the last execution of the defragmentation procedure. By adopting these criteria to trigger defragmentation, unnecessary overhead can be avoided.

Step 2 in Fig 2 represents the distribution of lightpaths as points on a plane before the identification of clusters. The CISUR k-Means algorithm uses the feature values of the lightpaths and performs the following steps: 1) selection of k arbitrary lightpaths as the initial clusters, 2) computation of the centroids of the k clusters, 3) association of each lightpath with the closest centroid, based on similarity, and 4) verification of potentially significant changes in the clusters. If there is no significant change, the clusterization process ends, with k centroids and lightpaths associated with each cluster.

The k value was determined using the following criteria: k needs to be less than or equal to the number of cores to avoid the generation of crosstalk and k needs to be less than the number of features as in any efficient clusterization. To validate the number of clusters (k), the silhouette value needs to be higher than 0.7 in a range of [0, 1]. When the silhouette value is lower than 0.7, the algorithm selects the current k to be the number of clusters.

Having defined the clusters (Step 3 in Fig. 2), these clusters need to be mapped onto cores and lightpaths onto these clusters. The CISUR algorithm elaborates a ranking of cores based on the number of adjacent cores to reduce crosstalk. Clusters are then assigned to cores according

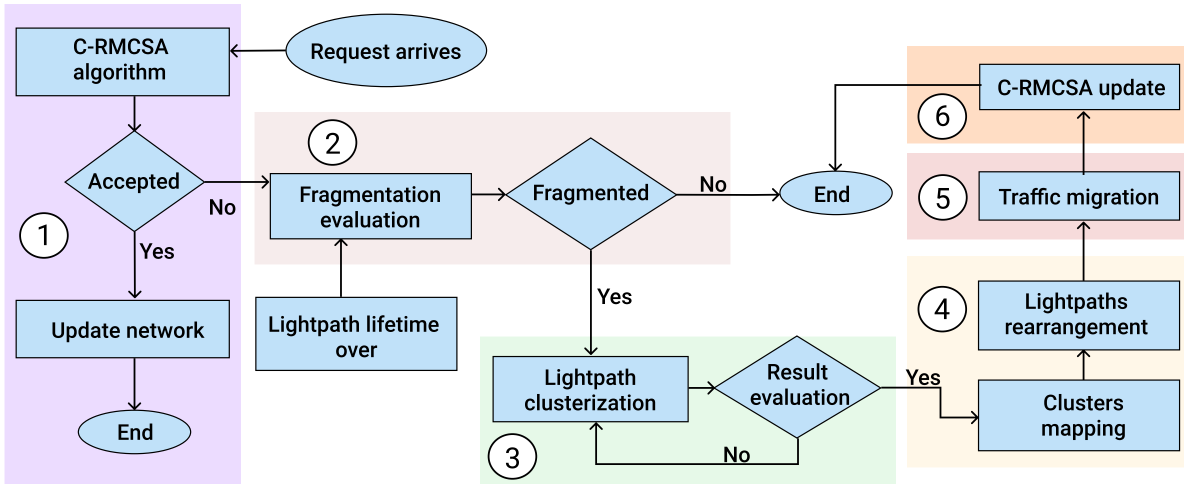


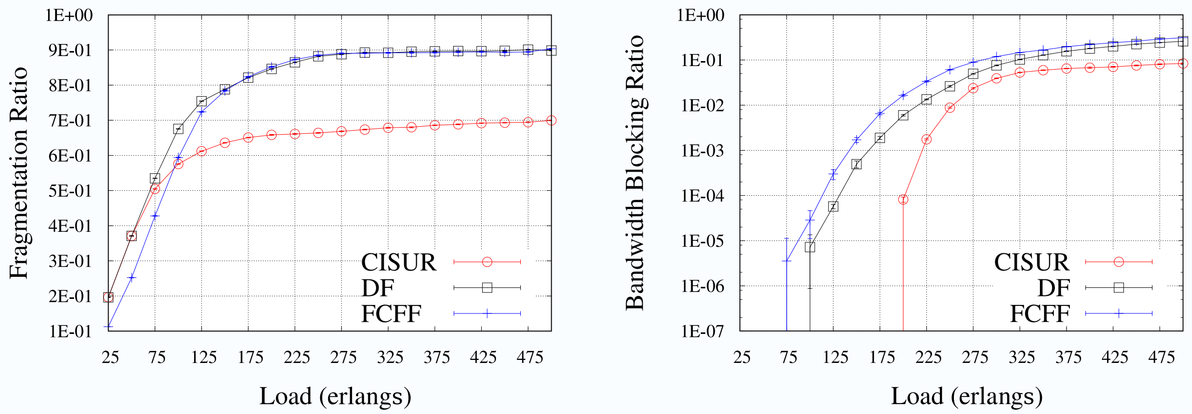
Fig. 3. Illustration of how the CISUR and C-RMCSA algorithms are integrated.

to this ranking [13].

The CISUR algorithm allocates lightpaths into clusters according to the required bandwidth (Step 4 in Fig. 2); it sorts requests in descending order of bandwidth demand. The First-Fit (FF) policy is employed for the rearrangement of lightpaths. Such prioritization decreases fragmentation when lightpaths are torn down in comparison to other criteria evaluated. When more than one cluster is allocated to a core, the FF and the Last-Fit policy are used interchangeably to enlarge the available band in the center of the spectrum.

In the last step of the defragmentation, when lightpaths are indeed reallocated to a cluster, traffic disruption is avoided by the adoption of the push-pull technique [3]. If a lightpath cannot be assigned to a cluster using the same route, the CISUR algorithm uses rerouting. Traffic disruptions due to rerouting can have a strong impact on network performance as well as on the duration of lightpaths.

To circumvent this problem, the Move-To-Vacancy (MTV) algorithm is adopted [4] (Step 5 in Fig. 2). The MTV algorithm creates a directed graph representation to identify cycles in a graph. In this graph, requests are represented by vertices, and their dependencies are represented by edges. A minimum feedback vertex is identified, and an attempt is made to find available resources using the MTV algorithm to set aside the request temporarily. With the graph free of directed cycles, the algorithm then migrates the rest of the requests and finally restores the request which was moved using MTV.



(a) Network fragmentation ratio.

(b) Bandwidth blocking ratio.

Fig. 4. Results extracted from the NSFNET topology; charts on the left side give the fragmentation ratio whereas those on the right side give the BBR.

C. RMCSA Algorithm

The Clusterization RMCSA (C-RMCSA) algorithm determines if a new request is accepted or not. The algorithm is divided into two steps: routing and spectrum selection. When a request arrives, the C-RMCSA algorithm finds a set of candidates paths using the K-Shortest Paths (KSP) algorithm for the pair of source and destination nodes in the request. In the spectrum selection step, for each candidate path, the algorithm tries to find a block of slots for allocation, while respecting the continuity and contiguity constraints. The rounded numbers in Fig 3 represent the steps of the CISUR algorithm, which is illustrated in Fig 2. Figure 3 shows the flowchart of our solution, including the integration of the CISUR algorithm with the C-RMCSA algorithm.

Information on the spectrum occupancy after defragmentation, such as the number of clusters, the mapping of clusters to cores, and the centroids of each cluster is made available to the C-RMCSA algorithm. With this information, the algorithm allocates lightpaths to incoming requests, by determining the cluster in which the lightpath should be allocated. To do this, the features of the request are extracted and normalized, the normalized features are then used to determine the cluster to which the request will be allocated. The algorithm compares the distance between these request features and the centroids of the clusters (calculated on the basis of the same features) using the Manhattan distance metric. The request will be assigned to the closest cluster. The C-RMCSA algorithm then uses the FF policy to allocate the request into the corresponding cluster.

IV. Numerical Results

The performance of our proposal was compared with those of two other algorithms, the FCFF algorithm [14], which is a proactive algorithm, and the Defragmentation (DF) algorithm [4], which is a reactive one. In [14], the author used the FF policy to create several variations of RMCSA algorithms and showed that the FCFF achieved the lowest blocking ratio. In [4], the authors proposed the DF algorithm that performs which defragmentation every 300 requests. Selected lightpaths are then rerouted as a result of defragmentation. To make a comparison with C-RMCSA fair, crosstalk thresholds, and adaptive modulation formats were introduced to the FCFF and DF algorithms [15].

The number of slots depends on the bandwidth requested and the modulation format. Six modulation formats were considered: Binary Phase Shift Keying (BPSK), Quadrature Phase Shift Keying (QPSK), and x-QAM, where $x \in \{8, 16, 32, 64\}$, with the transmission reach and support bit-rates, each with crosstalk thresholds of $\{-14, -18.5, -21, -25, -27, -34\}$ DB, respectively. Since the minimum distance between symbols is most strongly affected by the rotated interference, a strong impact on the crosstalk generated was expected. The values for calculating the crosstalk used the formulation in [15].

The algorithms were implemented in the FlexGrid simulator, (available at https://bit.ly/2OKwwWI)). Each simulation ran 100,000 requests, and the load (in erlang) was increased in steps of 25 erlangs for each simulation from 0 to 500. A confidence interval of 95% was considered using twenty samples for each step. The bandwidth requests were uniformly generated using a distribution between 25 and 400 Gb.

In the simulations, the NSFNET topology was employed [13]. This topology has 14 nodes and 21 fiber links. By using topologies other than those used for the actual simulations, issues of bias in the evaluation were avoided. The node architecture described in [15] was considered. Each fiber link was bidirectional and contained 7 cores, each with 320 slots and a slot spacing of 12.5 GHz.

The Bandwidth Blocking Ratio (BBR) and fragmentation ratio were used to evaluate the performance of the proposed algorithms. Figure 4 shows the BBR and network fragmentation ratio for the NSFNET topology. The fragmentation ratio generated by our proposal was 25% lower than those produced by the other two algorithms. Such results evince the effectiveness of adopting a feature-oriented defragmentation procedure. The DF algorithm [4] performed defragmentation, on average, 330 times per simulation, while CISUR performed 100 times. Moreover, when using CISUR, each defragmentation procedure affected only 10 to 25% of the lightpaths, while the DF algorithm always reconfigured a fixed number of lightpaths. If we had 100,000 requests, each with a lifetime of 1 second. In such scenario, the CISUR algorithm would be executed at most 100 times, with each defragmentation reconfiguring from 100 to 250 lightpaths and traffic disruption occurring to at most only 1% of these lightpaths. Moreover, the CISUR algorithm does not require additional transponders to reconfigure lightpaths, i.e., the effectiveness of our proposal does not depend on additional monetary cost.

The lower fragmentation ratio produced by the CISUR algorithm led to lower blocking of lightpath requests. Results show that the CISUR algorithm successfully decreased the BBR values, starting to block requests only after loads of 200 erlangs, while the DF algorithm starts blocking under loads of 100 erlangs. Using the CISUR algorithm, the BBR values were two orders of magnitude lower than those produced by the DF algorithm. Such results are a direct consequence of the feature-oriented clusterization of lightpaths, which reduces not only fragmentation but also the blocking of requests.

V. Conclusions and Future Directions

Although EON-SDM based on multi-core fibers have the potential to increase the bandwidth availability, spectrum fragmentation may jeopardize this availability. In this paper, we introduced a defragmentation approach based on machine learning that together with a RMCSA algorithm can reduce fragmentation. These algorithms showed to be effective in ameliorating the fragmentation problem, decreasing the blocking of requests for lightpath establishment.

As future work, reducing the incurred overhead in the CISUR algorithm could be tackled. One possibility would be to adopt a mechanism for the detection of regions in the spectrum using image processing algorithms as well as the adoption of classification algorithms to distinguish such regions. Another way of improving the CISUR algorithm is the search for an optimal timing to execute defragmentation. For example, the Principal Component Labeling (PCA) algorithm might be employed before the execution of the clusterization algorithm to reduce the dimensionality of the problem.

ACKNOWLEDGMENT

This work was supported by CAPES, Alumni IC-Unicamp, CNPq, and grant 15/24494-8, FAPESP, BRAZIL.

REFERENCES

[1] Richardson, D.J., Fini, J.M. and Nelson, L.E., 2013. Space-division multiplexing in optical fibres. Nature photonics, 7(5), pp.354-362.

[2] Luo, R., Hua, N., Zheng, X. and Zhou, B., 2018. Fast parallel lightpath re-optimization for space-division multiplexing optical networks based on time synchronization. Journal of Optical Communications and Networking, 10(1), pp.A8-A19.

[3] Cugini, F., Paolucci, F., Meloni, G., Berrettini, G., Secondini, M., Fresi, F., Sambo, N., Poti, L. and Castoldi, P., 2012. Push-pull defragmentation without traffic disruption in flexible grid optical networks. Journal of Lightwave Technology, 31(1), pp.125-133.

[4] Zhang, M., Shi, W., Gong, L., Lu, W. and Zhu, Z., 2013, June. Bandwidth defragmentation in dynamic elastic optical networks with minimum traffic disruptions. In 2013 IEEE International Conference on Communications (ICC) (pp. 3894-3898). IEEE.

[5] Zhao, Y., Hu, L., Zhu, R., Yu, X., Wang, X. and Zhang, J., 2018. Crosstalk-aware spectrum defragmentation based on spectrum compactness in space division multiplexing enabled elastic optical networks with multicore fiber. IEEE Access, 6, pp.15346-15355.

[6] Zhao, Y. and Zhang, J., 2016. Crosstalk-aware cross-core virtual concatenation in spatial division multiplexing elastic optical networks. Electronics Letters, 52(20), pp.1701-1703.

[7] Tode, H. and Hirota, Y., 2017. Routing, spectrum, and core and/or mode assignment on space-division multiplexing optical networks. Journal of Optical Communications and Networking, 9(1), pp.A99-A113.

[8] Moura, P.M., da Fonseca, N.L. and Scaraficci, R.A., 2014, June. Fragmentation aware routing and spectrum assignment algorithm. In 2014 IEEE International Conference on Communications (ICC) (pp. 1137-1142). IEEE.

[9] Fadini, W., Chatterjee, B.C. and Oki, E., 2015. A subcarrier-slot partition scheme with first-last fit spectrum allocation for elastic optical networks. Computer Networks, 91, pp.700-711.

[10] Sugihara, S., Hirota, Y., Fujii, S., Tode, H. and Watanabe, T., 2017. Dynamic resource allocation for immediate and advance reservation in space-division-multiplexing-based elastic optical networks. Journal of Optical Communications and Networking, 9(3), pp.183-197.

[11] Yousefi, F. and Rahbar, A.G., 2018. Novel fragmentation-aware algorithms for multipath routing and spectrum assignment in elastic optical networks-space division multiplexing (EON-SDM). Optical Fiber Technology, 46, pp.287-296.

[12] Xiong, Y., Yang, Y., Ye, Y. and Rouskas, G.N., 2019. A machine learning approach to mitigating fragmentation and crosstalk in space division multiplexing elastic optical networks. Optical Fiber Technology, 50, pp.99-107.

[13] Trindade, S. and da Fonseca, N.L., 2019, May. Proactive fragmentation-aware routing, modulation format, core, and spectrum allocation in EON-SDM. In ICC 2019-2019 IEEE International Conference on Communications (ICC) (pp. 1-6). IEEE.

[14] Klinkowski, M., Lechowicz, P. and Walkowiak, K., 2018. Survey of resource allocation schemes and algorithms in spectrally-spatially flexible optical networking. Optical Switching and Networking, 27, pp.58-78.

[15] Muhammad, A., Zervas, G. and Forchheimer, R., 2015. Resource allocation for space-division multiplexing: Optical white box versus optical black box networking. Journal of Lightwave Technology, 33(23), pp.4928-4941.

VI. Biographies

Silvana Trindade received her M.S. degree from the State University of Campinas, Campinas, Brazil, where she is pursuing her Ph.D. in computer science. Her research interests include optical networks, resource allocation, and artificial intelligence.

Nelson L. S. da Fonseca obtained his Ph.D. degree from the University of Southern California. He is a Full Professor at the Institute of Computing, State University of Campinas, Brazil. He is an active ComSoc volunteer and has served as EiC of the IEEE Communications Surveys and Tutorials, VP Technical and Educational Activities, VP Publications, and VP Member Relations.