# Spectrum Defragmentation Algorithms for Elastic Optical Networks using Hitless Spectrum Retuning Techniques

**Mingyang Zhang[1], Yawei Yin[2], Roberto Proietti[2], Zuqing Zhu[1], S. J. B. Yoo[2]**
*1.University of Science and Technology of China, Hefei, Anhui 230027, China, Email: zqzhu@ieee.org*
*2. University of California, Davis, Davis, CA 95616, USA, Email: sbyoo@ucdavis.edu*

**Abstract:** We propose several algorithms to achieve hitless bandwidth defragmentation using spectrum retuning in elastic optical networks. Two retuning techniques, spectrum sweeping and hop tuning, are studied. Simulation results show that the hop tuning technique achieves better defragmentation performance.

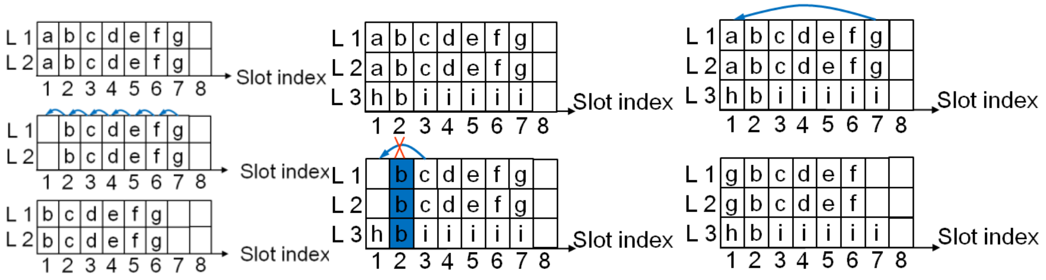
**OCIS codes:** (060.1155) All-optical networks; (060.4251) Networks, assignments and routing algorithms.

## 1. Introduction

Elastic optical networking [1] is a promising approach for achieving efficient spectrum utilization by allocating just-enough bandwidth to each request. However, it also brings new challenges, such as bandwidth fragmentation [2], to robust and efficient network control and management. Bandwidth defragmentation, which aims at consolidating the spectrum usage, is an effective approach to mitigate fragmentation. Previous works have already proposed a few defragmentation methods, including make-before-break rerouting [3] and spectrum sweeping retuning [4, 5]. It is known that rerouting may require additional transponders and make the network less efficient. This limitation can be overcome by the spectrum sweeping based retuning techniques in [4, 5], which do not require additional transponders and can leverage the automatic frequency control (AFC) capabilities of coherent receivers to achieve truly optical layer hitless defragmentation. Nevertheless, due to the device limitations, it is difficult for spectrum sweeping to move spectral slots over other connections' spectra. Therefore, this technique may result in partial defragmentation. Recently, a hop tuning based retuning technique has been experimentally demonstrated in [8], which can support full defragmentation and only introduce a very short retuning latency (<1 μs). In this paper, we investigate the operation principles of spectrum retuning techniques, including spectrum sweeping and hop tuning, and propose several efficient defragmentation algorithms. We design simulations to compare the performance of the proposed algorithms, in terms of bandwidth blocking probability and number of operations. The simulation results indicate that the algorithms based on hop tuning achieve better defragmentation performance.

## 2. Bandwidth Defragmentation Algorithms

Spectrum sweeping achieves bandwidth defragmentation by moving the spectrum slots of a connection to available adjacent slots that have lower indices. For example, as shown in Fig. 1(a), there are seven connections (*a*, *b*, *c*, *d*, *e*, *f*, and *g*) over two fiber links (*L1* and *L2*), and each connection occupies one spectral slot on a link. When connection *a* expires, spectrum sweeping tunes the spectra of connections *b*-*g* to lower-indexed adjacent slots on *L1* and *L2*, one by one. Eventually, the spectrum usage on *L1* and *L2* can be consolidated to the lower end and a slot-block with a larger size is formed for future connections. However, spectrum sweeping may not always be successful. Let us consider the situation in Fig. 1(b), where connection *b* goes across three links. When connection *a* expires, no connection can be retuned. Connection *h* blocks the sweeping of connection *b*, even though connection *c* can fit into the slots emptied by connection *a*. As we will show later in this paper, this type of head-of-line (HOL) blocking can hamper the defragmentation performance of spectrum sweeping.



(a) (b) (c)

Fig. 1. Bandwidth defragmentation schemes in EONs, (a) Successful spectrum sweeping, (b) Unsuccessful spectrum sweeping due to head-of-line blocking, and (c) Hop tuning.

For the hop tuning technique, the local oscillator in the receiver can track a spectrum jump in the transmitter automatically within a short period of time (<1 μs) [8]. Therefore, this technique can move the spectral slots of a connection to any desired location in the spectrum domain without interfering with existing connections, and the end-to-end bi-directional coordination between the transmitter and receiver is conducted automatically. With this

978-1-55752-962-6/13/$31.00 ©2013 Optical Society of America

functionality, hop tuning can resolve the dilemma in Fig. 1(b) by tuning connection *g* directly to the slot released by connection *a* (as shown in Fig. 1(c)). As the hop tuning technique makes spectrum movements more flexible, we can optimize spectrum defragmentation from different perspectives with it. In the rest of this Section, we propose two defragmentation algorithms based on hop tuning to maximize the spectrum rejoins and to minimize the number of operations, respectively. Both algorithms are based on a dependence tree. Specifically, each connection to be retuned corresponds to a node in the dependence tree, and we insert a directed link from a child node to its parent since the child node's corresponding connection can only be tuned after the parent is gone. For example, in Fig. 2(a), if connection *a* expires, we can construct a dependence tree as shown in Fig. 2(d) for potential retuning operations.

The maximum spectrum rejoin (MSR) algorithm operates as follows,

**Step 1:** Set the expired connection as the root node (first level) of the dependence tree.

**Step 2:** For each node in the current level of the tree, search movable connections whose occupied slots are the nearest from the higher-indexed end to the connection corresponding to this node. If such connections exist, add them as child-nodes and connect each child-node to its parent with a direct link.

**Step 3:** Go to the next level, if there are unprocessed nodes, return to Step 2, otherwise continue.

**Step 4:** Examine each leaf-node that has no child in the dependence tree, and select the one that can result in the maximum spectrum rejoins after retuning.

**Step 5:** Perform hop tuning based spectrum retuning for the nodes on the branch of the selected node, one by one from parent to child nodes.

**Step 6:** Check whether other connections can be retuned to consolidate spectrum usage further and perform corresponding retuning, otherwise, stop the defragmentation process.

Fig. 2(b) shows an example of the hop tuning based defragmentation using MSR. When connection *a* expires, we form a dependence tree as shown in Fig. 2(d). Apparently, the movement sequence of *b-e-f* can result in the largest spectrum rejoin after the first round. Then, the sequences of *c-d* and *h-g* are handled and eventually, we consolidate the spectrum usage as shown in Fig. 2(b). Notice that MSR algorithm can be applied to spectrum sweeping technique as well, but may have a premature termination due to the reason that HOL blocking can prevent the growth of the dependence tree's branches.

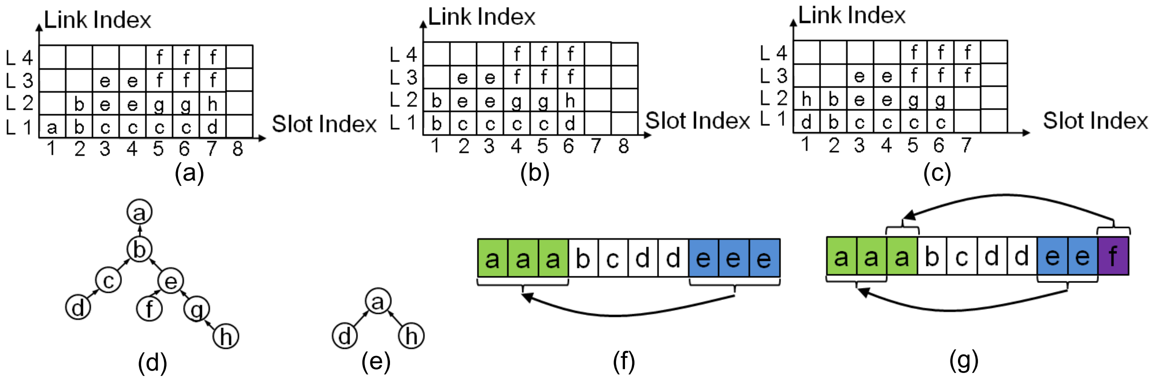


Fig. 2. Examples of MSR and MNO (a) Spectrum before defragmentation, (b) Spectrum after MSR, (c) Spectrum after MNO, (d) Dependence tree of MSR, (e) Dependence tree of MNO, (f) MNO for equal-sized fill-in, and (g) MNO for combined fill-in.

The minimum number of operation (MNO) algorithm operates as follows,

**Step 1:** Same as MSR.

**Step 2:** For each node in the current level of the tree, search movable connections whose occupied slots are the farthest of the connection corresponding to this node from the higher-indexed end. If the bandwidth (*i.e.*, number of slots) of movable connection is equal to the bandwidth of current node, add it as the child-node (*e.g.*, the case in Fig. 2(f)); otherwise find multiple connections whose total bandwidth is the closest to that of current node, add them as child-nodes (*e.g.*, the case in Fig. 2(g)). Connect each child-node to its parent with a direct link.

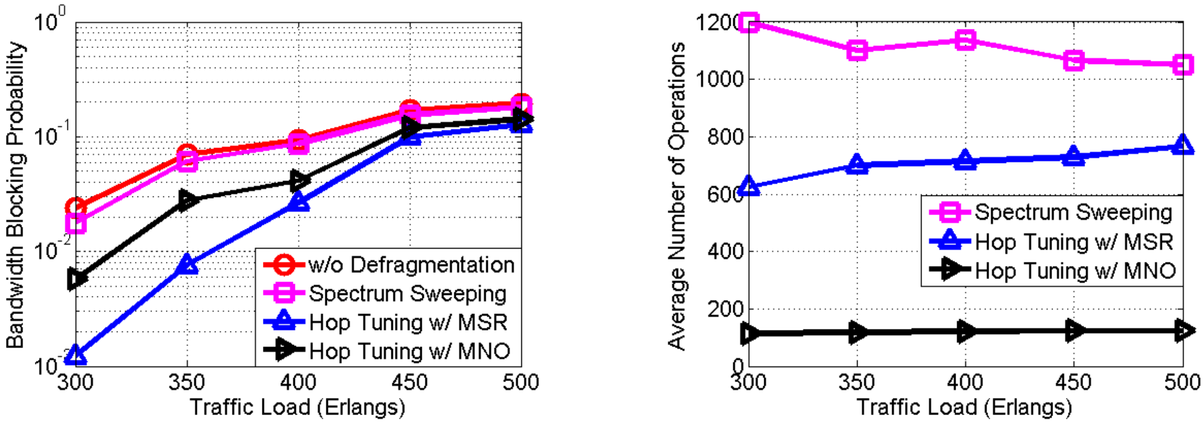
**Step 3-6:** Same as MSR.

Fig. 2(c) shows an example of the hop tuning based defragmentation using MNO. When connection *a* expires, connections *d* and *h* are selected to be tuned directly to the original location of connection *a*, according to the dependence tree in Fig. 2(e). We finally consolidate the spectrum usage to that in Fig. 2(c). Note that the MNO algorithm cannot be applied to spectrum sweeping technique.

## 3. Simulation Results and Discussion

We perform simulations with the 14-node NSFNET topology, and assume that the EON is deployed in the C-Band. Hence, each fiber link has 4.475 THz bandwidth to allocate, which corresponds to 358 12.5-GHz subcarrier slots. The dynamic requests are generated using the Poisson traffic model. For each request, the number of slots is uniformly distributed within 1-16 and the source-destination pair is randomly chosen. The requests are originally served with the K-shortest paths and balanced load spectrum assignment (KSP-BLSA) algorithm [7]. Bandwidth defragmentation is triggered when a request expires. We employ bandwidth blocking probability (BBP), which is defined as the ratio of blocked connection bandwidth versus total requested bandwidth, and average number of operations for defragmentation as the metrics to evaluate the performance of the proposed algorithms.

Fig. 3(a) indicates that when hop tuning is employed, both MSR and MNO algorithms can effectively reduce BBP, compared to the spectrum sweeping based defragmentation. Specifically, the MSR algorithm achieves the lowest BBP for the reason that it maximizes the spectrum rejoins and hence the spectrum usage is consolidated to the greatest extent. Fig. 3(b) shows that the MNO algorithm introduces the smallest number of operations. This is due to the reason that the algorithm always moves connections with highest-indexed slots to fill up the spectrum gap and thus the number of nodes in the dependence tree is smaller. Since the spectrum sweeping technique has to move connections to their adjacent slots step-by-step, the number of operations from it is intrinsically larger than those from hop tuning. Also, it is interesting to notice that the number of operation from MSR increases slightly with the traffic load, while the results from spectrum sweeping indicate an opposite trend. We believe this fact suggests that when the network is crowded with high traffic load, the spectrum sweeping technique becomes less effective due to the high probability of HOL blocking. On the other hand, since the hop tuning technique is more flexible, the MSR algorithm can actually conduct more defragmentation operations when the network is more fragmented with higher traffic load. Putting the results in Figs. 3 (a) and (b) together, we observe a clear tradeoff between BBP and number of operations for hop tuning. Even though the MSR algorithm achieves lower BBP values, it also requires larger number of operations. When the traffic load is larger than 450 Erlangs, MSR only achieves slightly improvement on BBP, while its number of operations can be more than four times of that from MNO. These results suggest that MNO is a better scheme to use when the traffic load is higher than 450 Erlangs. Therefore, network operators should choose different defragmentation algorithms based on the traffic load and/or network status.



(a) (b)

Fig. 3. (a) Simulation results on (a) Bandwidth blocking probability, and (b) Average number of operations.

## 4. Conclusion

We proposed several algorithms to investigate hitless defragmentation using two spectrum retuning techniques, *i.e.*, spectrum sweeping and hop tuning, in elastic optical networks. Simulation results indicated that the defragmentation schemes based on the hop tuning technique achieved better performance.

## Acknowledgment

This work was supported in part by the Project NCET-11-0884 and Project 1208085MF88.

## References

[1] M. Jinno, *et al.*, *IEEE Commun. Mag.*, vol. 47, pp. 66-73, 2009.
[2] Y. Yin, *et al.*, *Opt. Express*, vol. 20, pp. 1798-1804, 2012.
[3] T. Takagi, *et al.*, in *Proc. of ECOC 2011*, paper Mo.2.K.3, Sept. 2011.
[4] F. Cugini, *et al.*, in *Proc. of OFC 2012*, paper JTh2A.40, Mar. 2012.
[5] K. Sone, *et al.*, in *Proc. of ECOC 2012*, paper Th.3.D.1, Sept. 2012.
[6] G. Bosco, *et al.*, *J. Lightw. Technol.*, vol. 29, pp. 53-61, 2011.
[7] Y. Yang, *et al.*, in *Proc. of INFOCOM 2011*, pp. 1503-1511, Apr. 2011.
[8] Roberto Proietti, *et al.*, in *Proc. of Photonics in Switch 2012*, pp. 1-3, Sept. 2012