Adaptive Spectrum Defragmentation with Intelligent Timing and Object Selection for Elastic Optical Networks with Time-Varying Traffic

Mingyang Zhang[(1)], Changsheng You[(1)], Huihui Jiang[(1)], Zuqing Zhu[(1)*], Yawei Yin[(2)], Lei Liu[(2)], 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 intelligent timing and object selection algorithms for adaptive spectrum defragmentation in EONs with time-varying traffic. The simulation results show that the algorithms can stabilize and reduce bandwidth blocking probability (BBP) effectively with the minimum number of connection reconfigurations.*

Introduction

Recently, elastic optical networks (EONs) based on the optical orthogonal frequency-division multiplexing (O-OFDM) technology have attracted intensive research interests because they can facilitate agile spectrum management in the optical layer[1]. However, as EONs reduce the spectrum allocation granularity down to 12.5 GHz or less, spectrum fragmentation, i.e., the existing of non-aligned, isolated and small-sized spectrum segments in their network spectra, becomes a serious issue. Since the spectrum segments can hardly be used for future connections, spectrum fragmentation can result in frequent request blocking, especially in dynamic network environments with connections being set up and torn down on-the-fly.

In order to overcome spectrum fragmentation, previous works have proposed several defragmentation algorithms for EONs[2–4], to tackle the problems of "*Which objects to defragment?*" and "*How to defragment?*". While the timing of defragmentation is also important, it is still under-explored. Most of the previous works[2–4] either invoke a defragmentation operation when a request blocking happens, or in a periodic manner. However, in practical network operations, the network status can change rapidly and the traffic load can also have time-dependent fluctuations. The operation timing is critical to make defragmentation work cost-effectively in such network environments. Untimely defragmentation will not be able to reduce the bandwidth blocking probability (BBP) to an acceptable level, while invoking defragmentation too frequently will lead to unnecessary operation complexity and cost. Moreover, the problem of "*Which objects to defragment?*", i.e., how to select "defragmentation objects" from the existing connections to reconfigure, should also be revisited for adaptive operations. Our previous work proved that partial defragmentation (i.e., defragmentation objects only include a portion of the existing connections) can achieve comparable blocking probability improvements as in full defragmentation[4], while how to optimize the defragmentation percentage adaptively is still an open question.

In this work, we propose novel defragmentation algorithms that can track the network status intelligently and select the timing and objects adaptively for defragmentation operations. Simulations show that the algorithms can stabilize and reduce BBP effectively with the minimum number of connection reconfigurations.

Adaptive Defragmentation with Intelligent Timing and Object Selection

In this work, we assume that the EON's traffic load varies in an unpredictable but gradual manner, and the network operator can invoke a series of defragmentation operations to reduce BBP. The proposed adaptive defragmentation algorithm determines the defragmentation percentages and intervals intelligently based on the network status. We define the following variables,

- γ_i: Defragmentation percentage for the i-th operation, as the number of connections to reconfigure over the total number of existing ones in the EON.
- T_i: Time interval between the $(i-1)$-th and i-th defragmentation operations.
- $B(\Delta t)$: Instant BBP, as the ratio of blocked bandwidth over total arrived bandwidth during Δt.

The proposed algorithm determines γ_i and T_i adaptively according to $B(\Delta t)$. We set the initial values of γ_i and T_i based on the traffic prediction during network initialization. Then, T_i and γ_i are determined with a double slide windows algorithm that maintains two windows, i.e., the time window T' and the blocking window B'. Its detailed procedure is as follows,

Step1: Obtain the time interval Δt since the last operation and monitor $B(\Delta t)$.
Step2: If $\Delta t > T'$ and $B(\Delta t) > B'$ are both satisfied, we set $T_i = \Delta t$, calculate the adjustment parameter as $\alpha = \frac{B(T_i)-B(T_{i-1})}{B(T_{i-1})}$, get $\gamma_i = \gamma_{i-1}\cdot(1+\alpha)$, update T' and B' as $T^{'} = T_i\cdot(1-\alpha)$ and $B^{'} = B(\Delta t)\cdot(1-\alpha)$, and invoke a defragmentation operation right away. Otherwise, go to **Step 1**.
Step3: Reset $\Delta t = 0$ and go to **Step 1**.

The rationale behind this algorithm is as follows. Firstly, the change on instant BBP α indicates the trend of traffic load fluctuation. When $\alpha > 0$, the network becomes more crowded and requires more frequent defragmentations with larger defragmentation percentages. Secondly, since the ultimate goal of defragmentation is to reduce BBP, we also consider the variation on instant BBP. If the instant BBP is relatively low, we may not need to invoke defragmentations too frequently, and this can be ensured by increasing the blocking window. Note that T' and B' are both updated at each operation, and T_i and γ_i are determined by the one of them that is reached later in time. We also set lower- and upper-bounds for both windows to make sure that they change in reasonable ranges.

Fig. 1 shows an example of the double slide window algorithm. For the period starting at t_1, the defragmentation will not happen until T' is reached at t_2, since between T' and B', T' is reached later. Similarly, for the period of $[t_2, t_3]$, the occurrence of defragmentation at t_3 is determined by B' since it is reached later. With this mechanism, we can avoid invoking defragmentation operations too frequently and can save certain operation complexity and cost.

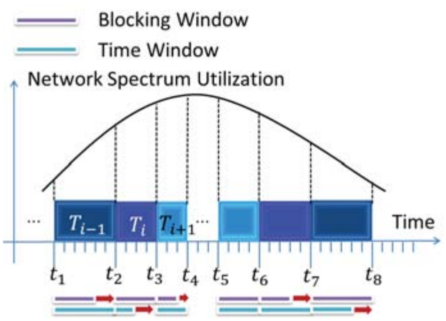

Fig. 1: Illustration of the double slide window algorithm.

For each defragmentation operation, the proposed algorithm selects the defragmentation objects' (i.e., connections) based on γ_i with the highest used FS-index first (HUFSIF) strategy[4]. Specifically, the HUFSIF strategy sorts the existing connections according to the high-end indices of their FS' in descending order, and then chooses γ_i percentage connections as the defragmentation objects. We employ a make-before-break based rerouting-and-retuning-combined method to reconfigure the objects for defragmentation. Basically, with the objective to minimize the spectrum fragmentation in the network, the routing and spectrum assignments (RSA) of the selected objects are re-optimized[4,5], i.e., moving the objects to FS' with lower indices and opening up more upper-portion spectrum for accommodating future connections.

Performance Evaluation

We design simulations to evaluate the performance of the proposed adaptive defragmentation with the 14-node NSFNET topology. We assume that the EON is deployed in the C-band and the bandwidth of each FS is 12.5 GHz, which means that each fiber link carries 358 FS'. The dynamic traffic is generated using the Poisson traffic model with time-varying load. The source-destination pair of each request is randomly selected and its bandwidth requirement in terms of FS' is uniformly distributed within three ranges [1, 6], [1, 10], and [1, 16] corresponding to *Scenarios* 1, 2 and 3, respectively, as shown in Fig. 2. Therefore, we simulate three traffic scenarios with the similar trend but different request sizes in the 24-hour period. The 24-hour period is divided into 500 equal intervals ($\sim$ 3 minutes each) for the defragmentation algorithm to monitor the network status. The requests are originally served with the K-shortest paths and balanced-load spectrum assignment algorithm[6].

We refer the proposed algorithms as the defragmentation with adaptive timing and percentage selection (DF-AT-AP) and the defragmentation with adaptive timing and fixed percentage selection (DF-AT-FP), with the difference that the later one does not include the adaptive defragmentation percentage selection, i.e., γ_i is fixed. In order to verify their performance, we also design two benchmark algorithms, such as the defragmentation with fixed timing and percentage selection (DF-FT-FP) and the defragmentation after fixed number of expired requests with fixed percentage selection (DF-FNR-FP). These four algorithms are evaluated with three performance metrics. The first metric is defined as the percentage of the time intervals whose instant BBP is above a preset threshold. This metric reflects the volume of the network's "congestion spots" along the time axis, and therefore can tell whether the defragmentation is conducted timely or not. The second metric is the overall BBP in a given period. The third metric is the total connection reconfigurations within the 24-hour period, which measures defragmentation algorithm's operation complexity and cost.

We first study the overall BBP's variation along the time axis. With *Scenario* 1, we obtain the results in Fig. 3 for no defragmentation, DF-FT-FP, and DF-AT-AP. In DF-FT-FP, we set the defragmentation percentage at 50% (i.e., $\gamma = 0.5$) for each operation. It can be seen that when there is no defragmentation, the overall BBP is relatively high and can have large fluctuation due to the time-varying traffic load. When we apply DF-FT-FP, the overall BBP is reduced significantly, but noticeable fluctuations appear. DF-AT-AP achieves the lowest overall BBP among the three algorithms, and most importantly, it clamps the fluctuation on overall BBP for most of the provision time. Therefore, the results in Fig. 3 verify that the proposed DF-AT-AP can track the network status intelligently and can perform adaptive defragmentation to stabilize the overall BBP at a reasonable level.

In order to further confirm the benefit of adaptive defragmentation, we repeat the simulations for the other two traffic scenarios and obtain the percentages of the time intervals whose instant BBP is above 20% for no defragmentation, DF-FT-FP, DF-FNR-FT, DF-AT-FP, and DF-AT-FP. Fig. 4 shows the results, and for the algorithms that use fixed defragmentation percentage, we still set the percentage at 50%. It can be seen that the algorithms with adaptive timing selection (i.e., DF-AT-FP and DF-AT-AP) provide smaller percentages than the other three. Hence, we can conclude that adaptive timing selection can effectively reduce the volume of the network's congestion-spots along the time axis, i.e., the defragmentation operations are conducted in a better and more timely way. It is also interesting to notice that the percentage from DF-AT-AP is slightly larger than that of DF-AT-FP. We believe this is because that with $\gamma = 0.5$, DF-AT-FP actually performs more connection reconfigurations than DF-AT-AP. The results on the total number of reconfigurations in Fig. 5 verify this. We observe that DF-AT-FP invokes the most reconfigurations among the four algorithms, while the total number of reconfigurations from DF-AT-AP is the smallest. Fig. 6 shows the overall BBP within the whole 24-hour period from the algorithms.

The simulation results demonstrate that the proposed adaptive defragmentation algorithms, i.e., DF-AT-AP and DF-AT-FP, achieve better performance on BBP reduction than the others that do not have such intelligence. Even though DF-AT-FP achieves slightly smaller BBP than DF-AT-AP, it also involves more connection reconfigurations. In all, DF-AT-AP can effectively reduce BBP with the smallest number of connection reconfigurations, and we therefore consider it as an intelligent defragmentation algorithm for EONs with time-varying traffic.

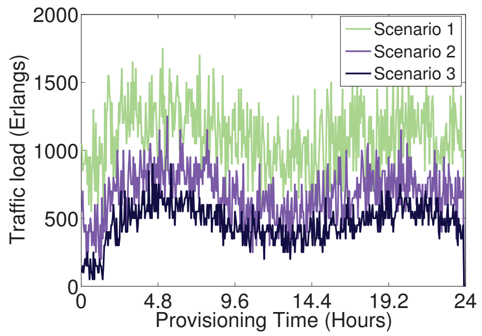

Fig. 2: Time-varying traffic load.

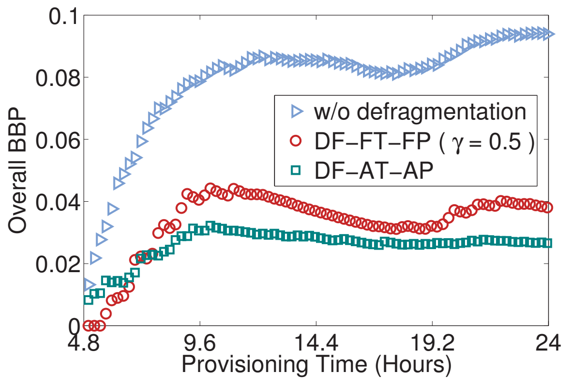

Fig. 3: Overall BBP versus provision time.

Conclusions

We proposed intelligent timing and object selection algorithms for adaptive spectrum defragmentation in EONs with time-varying traffic. The simulation results showed that the algorithms can stabilize and reduce BBP effectively with the minimum number of connection reconfigurations.

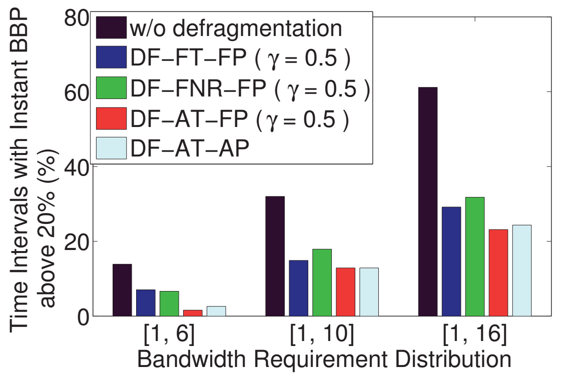

Fig. 4: Results on the percentages of the time intervals whose instant BBP is above 20%.

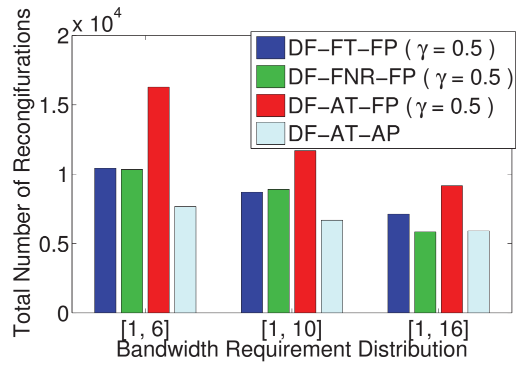

Fig. 5: Total number of connection reconfigurations within the whole 24-hour period.

Acknowledgments

This work was supported in part by the Project NCET-11-0884, and the Natural Science Foundation of Anhui Province under Project 1208085MF88.

References

[1] O. Gerstel et al., IEEE Commun. Mag. **50**, S12 (2012).
[2] T. Takagi et al., Proc. ECOC'11, Mo.2.K (2011).
[3] Y. Yin et al., Opt. Express, **20**, 1798 (2012).
[4] M. Zhang et al., Proc. ICC'13, (2013).
[5] M. Zhang et al., Proc. OFC'13, OW3A.4 (2013).
[6] Y. Wang et al., Proc. INFOCOM'11, 1503 (2011).

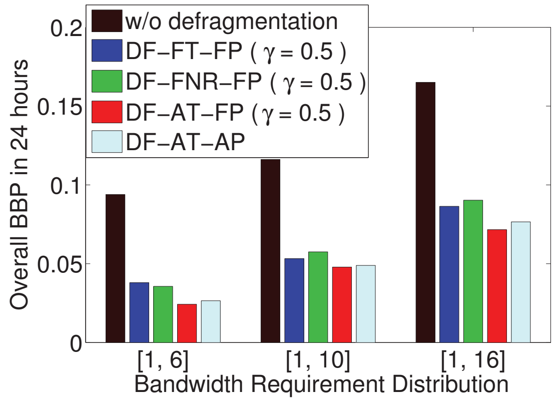

Fig. 6: Overall BBP within the whole 24-hour period.