Demonstration of Online Spectrum Defragmentation Enabled by OpenFlow in Software-Defined Elastic Optical Networks

Shoujiang Ma[1], Cen Chen[1], Shengru Li[1], Mingyang Zhang[1], Suoheng Li[1], Yan Shao[1], Zuqing Zhu[1], 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 and experimentally demonstrate a control-plane framework to realize online spectrum defragmentation (DF) in software-defined elastic optical networks. Experimental results show that the spectrum DF enabled by OpenFlow reduces the blocking probability effectively.

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

1. Introduction

It is well known that by using a flexible grid, elastic optical networks (EONs) can divide the optical spectrum into 6.25 or 12.5 GHz fixed-size frequency slots (FS'), and allocate spectrum resource to each connection request adaptively by assigning the necessary number of contiguous FS' according to the bandwidth requirement [1]. Therefore, agile bandwidth management in the optical layer can be achieved. However, in dynamic network operation, the frequent setting up and tearing down of connections can fragment the optical spectrum into non-aligned, isolated and small-sized spectrum segments. Since these segments can hardly be used for future connections, spectrum fragmentation leads to low spectrum utilization and high blocking probability in EONs [2, 3]. Previously, in order to relieve spectrum fragmentation in EONs, researchers have proposed defragmentation (DF) algorithms [2–4], and Liu *et al.* have demonstrated an OpenFlow framework for realizing fragmentation-aware routing and spectrum assignment (RSA) in the software-defined EONs (SD-EONs) [5]. Since spectrum fragmentation can accumulate in network operations, it is desired to implement the "online" spectrum DF scheme that triggers network reconfigurations dynamically and adaptively to achieve more blocking performance improvement. However, the problem of how to practically implement the online DF scheme has not been studied yet. In this work, by leveraging the software-defined network (SDN) architecture enabled by OpenFlow, we propose a control-plane framework to realize online spectrum DF in SD-EONs. We design the system architecture to support online DF, propose necessary extensions of the OpenFlow protocol to enable efficient reconfiguration of connections, and develop an effective DF algorithm for implementing in the OpenFlow controller (OF-C). The proposed DF framework is then implemented in a control-plane testbed that consists of 14 stand-alone OpenFlow agents (OF-AGs) and one OF-C, which are realized with high-performance Linux servers. Experimental results show that the online spectrum DF enabled by OpenFlow reduces the blocking probability effectively.

2. System Architecture

Fig. 1 shows the overall system architecture of an SD-EON. Above the data-plane that is built with edge routers (ERs) and bandwidth-variable optical cross-connects (BV-OXCs), we have the control-plane consisting of a centralized OF-C and several OF-AGs that each attaches to a data-plane equipment. Figs. 1(a)-(c) show the detailed configurations of the ER, OF-AG, and OF-C, respectively. In the SD-EON, OF-C mainly manages two operations for each request connection, 1) setting it up when it comes in from a client, and 2) reconfiguring it with Re-RSA during online spectrum DF. We first explain how the requests are initially served in the SD-EON.

Step 1: An ER receives a connection request $LR(d,C,\Delta t)$ for client traffic (*e.g.*, IP), where d is the destination address, C is the bandwidth requirement, and Δt is the required service duration.

Step 2: The OF-AG on the ER includes its own address as the source address s and constructs a *Packet-In* message to forward $\{s,d,C,\Delta t\}$ to the OF-C.

Step 3: The resource provision module (RPM) in OF-C receives the *Packet-In* message, and instructs the resource computation module (RCM) to perform the corresponding RSA calculation.

Step 4: RCM requests current network status from the traffic engineering database (TED) and calculate the RSA solution for the request. If a feasible RSA can be obtained, RCM instructs RPM to build the corresponding flow-entries for the source and destination ERs and the intermediate BV-OXCs. Each flow-entry includes information such as input and output ports, starting frequency, number of FS', and etc. Otherwise, if the RSA fails due to insufficient resource, RCM instructs RPM to communicate with the OF-AG on the source ER to block the request.

Step 5: RPM use *Flow-Mod* messages to distribute the flow-entries to the OF-AGs on the source and destination ERs and the intermediate BV-OXCs along the routing path. Each OF-AG parses the flow-entry and configures its data-plane

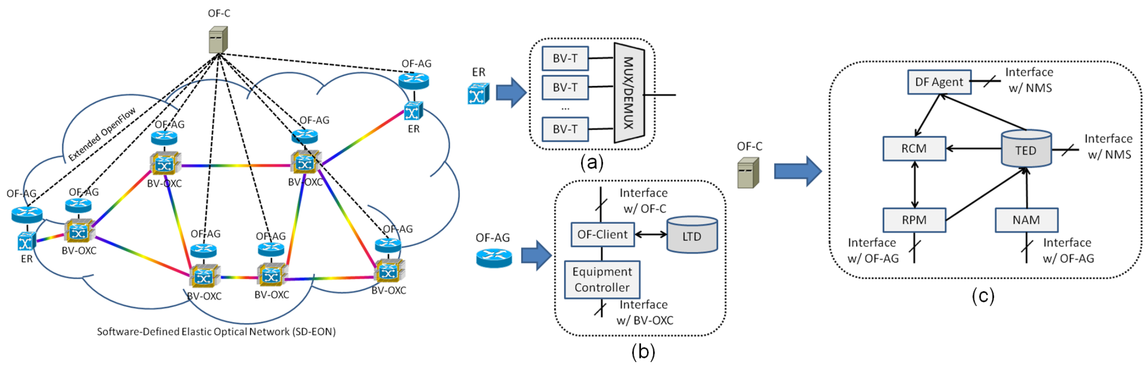

Fig. 1. System architecture of an SD-EON. ER: Edge router, OF-AG: OpenFlow agent, BV-OXC: Bandwidth-variable optical cross-connect, OF-C: OpenFlow controller, BV-T: Bandwidth-variable transponder, MUX/DEMUX: Multiplexer/De-multiplexer, OF-Client: OpenFlow client, LTD: Local traffic database, DF Agent: Defragmentation agent, NMS: Network management system, RCM: Resource computation module, TED: Traffic engineering database, RPM: Resource provision module, NAM: Network abstraction module.

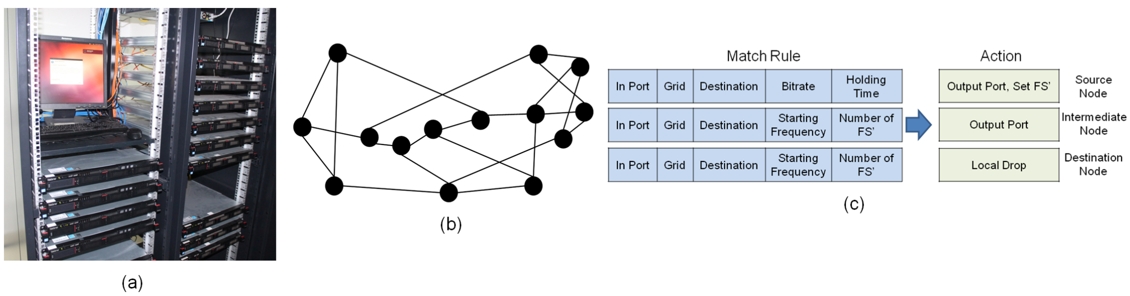

Fig. 2. (a) SD-EON control-plane testbed, (b) Network topology, and (c) Flow-table matching examples.

equipment accordingly, and then, the OF-client in it updates the flow-table in the local traffic database (LTD).
Step 6: Each OF-AG sends its configuration result to OF-C using the *Barrier-Reply* message. If all configurations are successful, RPM updates TED and the request is provisioned. Otherwise, RPM invokes an error-recovery mechanism.
Step 7: When a request expires, OF-C updates its TED and instructs the OF-AGs to remove the related flow-entries.

When an SD-EON node joins or leaves the SD-EON, the network abstraction module (NAM) in OF-C communicates with its OF-AG to update the topology and spectrum information in TED. During network operation, spectrum fragmentation can happen and the proposed control-plane framework works as follows to realize online spectrum DF.
Step 1: The DF agent in OF-C invokes a DF operation, selects existing connections in TED for reconfiguration, and instructs RCM to calculate the Re-RSA solutions for them. Note that a DF operation can be invoked either automatically based on the network status or by the network operator through the network management system (NMS).
Step 2: Based on the Re-RSA solutions, RCM figures out the reconfiguration sequence of the selected connections for minimizing traffic disruptions [3], and instructs RPM to build the new flow-entries accordingly.
Step 3: RPM use the *Flow-Mod* messages to send the flow-entries to the OF-AGs on the original and new routing paths of each selected request. Each OF-AG parses the flow-entry and configures the data-plane equipment accordingly.
Step 4: Each related OF-AG updates the flow-table in its LTD when the reconfiguration of a selected request is done, and in OF-C, RPM updates TED accordingly when all the reconfigurations in a DF operation have been finished.

3. Experimental Setup and Results

We implement the proposed framework in a control-plane testbed built with high-performance servers (ThinkServer RD530). The OF-C is implemented with the POX platform, and the OF-AG is programmed based on the Open-vSwitch running on Linux. We make necessary extensions to the related OpenFlow messages and enable them to support spectrum DF. Fig. 2(a) shows the picture of the control-plane testbed that includes 14 OF-AGs, and the topology of the corresponding SD-EON is in Fig. 2(b), which is based on the 14-node NSFNET topology. Note that in this testbed, the data-plane is emulated. Fig. 2(c) illustrates the flow-table matching examples for the source and destination ERs, and intermediate BV-OXCs. The Wireshark captures for setting up a connection and reconfiguring a connection in a DF

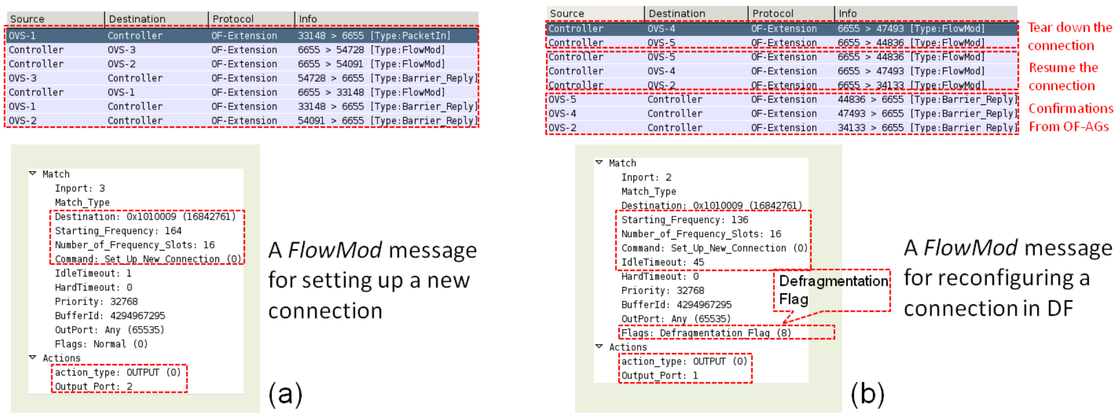

Fig. 3. Wireshark captures for (a) Setting up a new connection, and (b) Reconfiguring an existing connection.

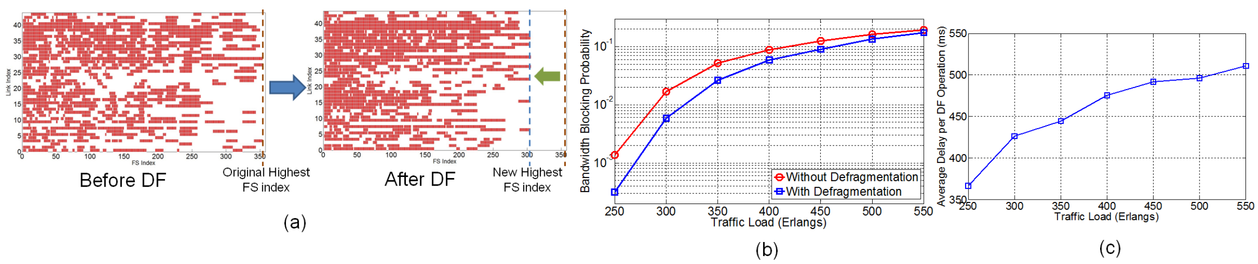

Fig. 4. Experimental results on (a) Spectrum utilizations before and after a DF, (b) Bandwidth blocking probability, and (c) Control-plane latency per DF operation.

operation are in Figs. 3(a) and 3(b), respectively. In this work, we consider the case that the DF agent in OF-C triggers the DF operations automatically based on the network status. More specifically, a DF is triggered when the number of expired requests exceeds 100. In each DF, the DF agent conducts partial reconfiguration for 30% of the existing connections using Re-RSA. The Re-RSA uses the fragmentation-aware RSA algorithm in [5].

We then perform online spectrum DF experiments in the control-plane testbed to investigate the performance of the proposed framework. We assume that each fiber link in the data-plane accommodates 358 FS' that each has 12.5 GHz bandwidth. On each node in the testbed, the OF-AG generates requests according to the Poisson traffic model and selects their destination addresses randomly. The bandwidth requirement C is in terms of FS' and changes within $[2,40]$ FS'. Fig. 4(a) shows the memory dumps of TED for network spectrum utilizations before and after a DF operation. The x-axis and y-axis are for the FS index and link index, respectively, and a red block indicates that the specified FS on a particular link is occupied. It can be seen clearly that the DF consolidates spectrum utilization of the SD-EON effectively. Fig. 4(b) compares the bandwidth blocking probabilities in the testbed with and without DF. As expected, the spectrum DF enabled by OpenFlow reduces the blocking probability in the testbed effectively. Fig. 4(c) shows the control-plane latency per DF operation changing with the traffic load. The latency is around 510 ms when the traffic load is 550 Erlangs. Since when traffic load becomes higher in the experiments, the number of existing connections to configure in each DF is also larger, the latency increases with traffic load in Fig. 4(c). Note that in order to obtain each data point in Figs. 4(b) and 4(c), OF-C serves 14000 incoming requests from the OF-AGs in the experiments.

4. Summary

We proposed and experimentally demonstrated a control-plane framework to realize online spectrum DF using network reconfiguration in SD-EONs. Experimental results showed that the spectrum DF enabled by OpenFlow improved network performance effectively.

References

[1] O. Gerstel, *et al.*, "Elastic optical networking: a new dawn for the optical layer?", *IEEE Commun. Mag.*, vol. 50, pp. S12-S20, Apr. 2012.

[2] T. Takagi, *et al.*, "Disruption minimized spectrum defragmentation in elastic optical path networks that adopt distance adaptive modulation", in *Proc. of ECOC 2011*, paper Mo.2.K.3, Sept. 2011.

[3] M. Zhang, *et al.*, "Bandwidth defragmentation in dynamic elastic optical networks with minimum traffic disruptions", in *Proc. of ICC 2013*, pp. 1-5, Jun. 2013.

[4] R. Wang, *et al.*, "Provisioning in elastic optical networks with non-disruptive defragmentation", *J. Lightw. Technol.*, vol. 31, pp. 2491-2500, Aug. 2013.

[5] L. Liu, *et al.*, "Software-defined fragmentation-aware flex-grid optical networks enabled by OpenFlow", in *Proc. of ECOC 2013*, paper We.3.E.2, Sept. 2013.