SDN-enabled Programmable Optical Packet/Circuit Switched Intra Data Centre Network

B. Guo[1], S. Peng[1], C. Jackson[1], Y. Yan[1], Y. Shu[1], W. Miao[2], H. Dorren[2], N. Calabretta[2], F. Agraz[3], J. Perelló[3], S. Spadaro[3], G. Bernini[4], R. Monno[4], N. Ciulli[4], R. Nejabati[1], G. Zervas[1], D. Simeonidou[1]

[1]High Performance Networks Group, University of Bristol, United Kingdom; [2]COBRA Research Institute, Eindhoven University of Technology, Eindhoven, Netherlands; [3]Universitat Politècnica de Catalunya Barcelona, Spain; [4]Nextworks, via Livornese 1027, Pisa, Italy

e-mail: allen.guo@bristol.ac.uk

Abstract: We demonstrated an SDN-enabled optical DCN leveraging AoD, OPS, FPGA-based ToR with extended OpenDayLight controller and extended OF protocol. Experimental results show application-aware OCS/OPS connection provisioning (753/214ms), OPS end-to-end connections latency (252μs), and OCS/OPS switchover.

OCIS codes: (060.4250) Networks; (060.4510) Optical Communications

1. Introduction

Most of current Data Centre Networks (DCN) follow a hierarchical network topology composed of layers of power hungry electronic switches, which causes significant end-to-end latency and limited bisectional bandwidth. Imposed by diverse Data Centre (DC) applications, DCN must support a variety of data flows, including a majority of bursty and short-lived "mice" flows and a few persistent "elephant" flows that carry most of the data. Most importantly, DCN together with its control and management system need to be able to efficiently deliver the flows according to their own characteristics and applications' requirements.

Targeting at these challenges, a plethora of research work have been and are being conducted, e.g. projects LIGHTNESS [1] and COSIGN [2], which aim to provide a holistic solution integrating programmable data plane and SDN-enabled control plane to provide cloud services in multi-tenant DCs. In the LIGHTNESS project, a flattened optical DCN is constructed by plugging multiple passive and active optical devices (e.g. fast and slow switches, Mux/DeMux, splitter) into an optical backplane to form an Architecture-on-Demand (AoD) node [3]. The AoD, orchestrated by the SDN control plane, can provide dynamic DCN connectivity between TORs over different optical technologies such as Optical Packet Switching (OPS) and Optical Circuit Switching (OCS) according to applications' requirements. While OCS can better accommodate high-capacity long-lived smooth data flows, OPS by taking advantage of its statistical multiplexing feature can offer flexible bandwidth to better serve bursty traffic demands. Despite the demonstration of the OCS and OPS as building block of AoD, the overall AoD based DCN operation and performance, orchestrated by SDN control plane, has never been demonstrated and assessed.

In this paper, for the first time, we experimentally demonstrate and assess the performance of the flat optical DCN data plane based on AoD including OPS, OCS switches and FPGA-based ToR for implementing traffic differentiation. The optical DCN is enhanced by an SDN-enabled control plane based on an extended OpenDayLight (ODL) controller supporting optical switch/connection features and OF agents developed for AoD, OPS and ToR. In order to enable the communications between SDN controller and OF agents, OF protocol is significantly extended. Furthermore, to accommodate different applications, we successfully demonstrate dynamic SDN-enabled OPS and OCS connectivity provisioning within 214ms and 753ms respectively. End-to-end communication with OPS shows <252μs latency and <1.5dB penalty. Also, the hitless OPS/OCS connection switchover has been demonstrated.

2. SDN-enabled Programmable Optical Data Centre Network Architecture and Key Technical Enablers

Fig. 1 shows the overall DCN architecture, including the optical data plane and SDN-enabled control plane. In the data plane, instead of a hardwired interconnection of different network elements, a more flexible DCN architecture equipped with an AoD node is adopted, which consists of an optical backplane (i.e. a Polatis fibre switch with a large port number) with switching modules (i.e. OPS, Wavelength Selective Switch) and passive devices (e.g., Mux/DeMux, splitter) attached. With this AoD architecture, different arrangements of input/output and modules can be constructed by dynamically setting up appropriate cross-connections on demand in the optical backplane according to different applications' requirements. OCS connections can be established through configuration of the backplane itself. The OPS consists of a modular SOA-based architecture with highly distributed control for port-count independent reconfiguration time, which has been designed and prototyped in [4]. A hybrid ToR switch is able to parse the input traffic from servers and send it out in different modes/connections (OPS/OCS). In this way, ToR performs traffic aggregation (e.g., traffic destined to the same ToR/Rack) and application-aware traffic classification to initiate either OPS or OCS connection. Fig. 2 (bottom) shows the packet processing workflows in the ToR switch. After the traffic arrives, it will be classified and sent to different buffers according to the matching rules in its look-up-table (LUT) that is updated by the SDN controller. Then, packets in each buffer will go through the OPS/OCS

978-1-55752-937-4/15/$31.00 ©2015 Optical Society of America

mapping module, where the sending mode (OPS or OCS) will be decided first and then other corresponding connection configurations will be proceeded, especially for OPS connections (i.e., optical packet payload size and labels). For the traffic that is classified to OPS, an optical label, generated by a dedicated opto-electronic label generator interfaced with the FPGA, is attached to the optical packet payload at the FPGA output.

To enable the SDN control over this hybrid optical data plane, OF agents are developed for the AoD, OPS, and FPGA-based ToR. The agents interact with the underlying devices through different interfaces (i.e., TL1, Raw socket and Ethernet frame as shown in Fig.1) to perform a set of actions such as capabilities/attributes collection, configurations, and monitoring. The OF protocol is significantly extended to enable the communications between the ODL controller and the optical data plane via the OF agents. Fig. 2 (upper) shows the implemented OF extensions in the fields of *ofp_capability*, *ofp_match* and *ofp_action*. The *ofp_match* is extended for OPS to support the use of labels, while the *ofp_action* is extended for ToR switch to enable the optical packet label configurations.

For the SDN controller, we extended several modules in ODL to let it support optical layer features. In the Service Abstraction Layer, abstracted optical switch and connection features (i.e. optical port, optical cross connection) are extended, while the extensions in Forwarding Rules Manager are for optical flow pushing. Topology Manager is extended for parsing optical port peer information that is used by ODL to build optical network topology.

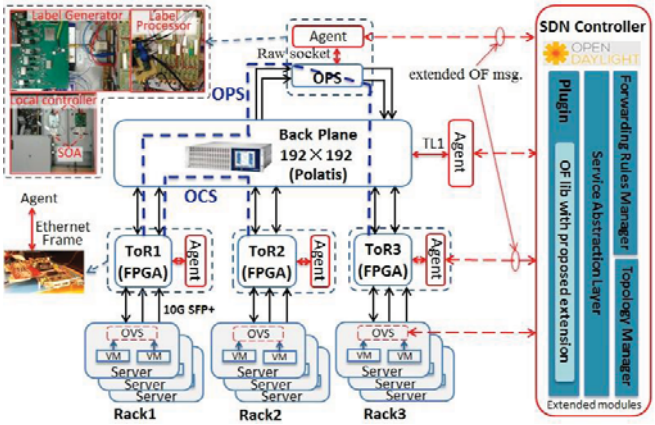


Fig.1 SDN enabled Hybrid OPS/OCS DCN Architecture

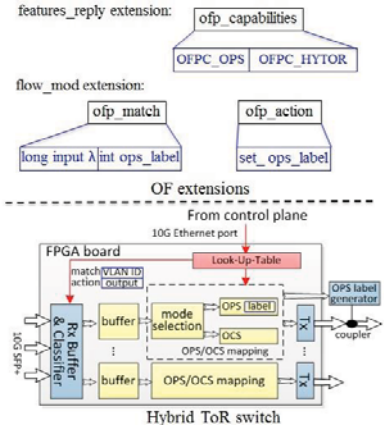


Fig.2 OF extensions and packet processing in ToR

3. Experimentation and Results

In the experimental setup, a 192×192 Polatis fiber switch is used as backplane with a 2×2 OPS switch and other elements such as (De)Mux plugged in to compose an AoD node. ToR switches are implemented with FPGA opto-electronics (HTG Xilinx V6 board) with 12 10GE ports. All the ToR switches are connected to the backplane via 10GE port. Servers are connected to ToRs via 10GE optical links. OPS could forward the traffic to the possible destination with ~10ns switching time, while OCS ensure high-throughput data transmission. Here, we demonstrated the SDN enabled dynamic OPS/OCS connectivity provisioning and hitless OPS/OCS switchover.

3.1 OPS and OCS Connectivity Provisioning

Fig. 3 (a) shows the control messages for the OPS and OCS connection provisioning. Generally, when a new traffic flow arrives and there is no matching rule in Open vSwitch (OVS), OVS will send a packet_in message to ODL through a separated overlay control network to trigger a new connection establishment. An application has been implemented to calculate the configurations of switches i.e. AoD, OPS and ToR. Flows are pushed to different devices via the Representational state transfer (REST) API of ODL, which we have also extended to enable optical flow installation. In this experiment, two services (i.e., short-lived http request and long-lived Rsync service that consumes more bandwidth) are running in two VMs within the same rack. Their packets are tagged with different VLAN IDs according to the flow entries in OVS, which is updated by ODL. As shown in Fig. 3 (b), to establish an OCS connection, the backplane cross-connection (cflow_mod, OF message type 22) and flow entries (flow_mod) in ToR1 (source) and ToR2 (destination) need to be configured. The measured ToR-to-ToR OCS connections provisioning time are 753ms including OF backplane configuration message round trip time (251ms×2), backplane hardware configuration time (16ms) and source and destination ToR LUT update and configuration time (235ms). For the OPS connection, besides a flow entry with label configuration information sent to the source ToR, another flow for the OPS node need to be installed to update the OPS LUT. The measured time required to update the LUTs is 214 ms. It is worth noting that once the LUTs are setup, the underlying OPS connections will operate at its own speed (sub-microseconds) decoupled from the control plane. Fig. 3(b) illustrates the OF protocol extensions to support the OPS configuration. In particular, the figure shows the new *set_label* action added to the *flow_mod*,

which is processed by the ToR. The extended *ofp_match*, which contains the label and the wavelength associated to the flow, carried within the *flow_mod* message to OPS is depicted as well.

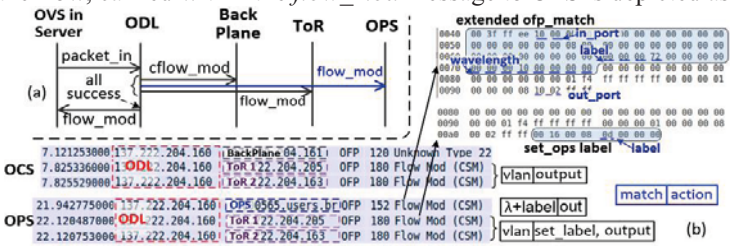


Fig.3 (a) Control message flow and (b) extended OF message

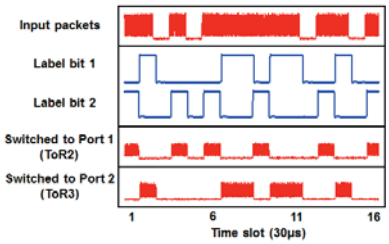
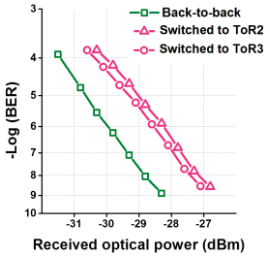
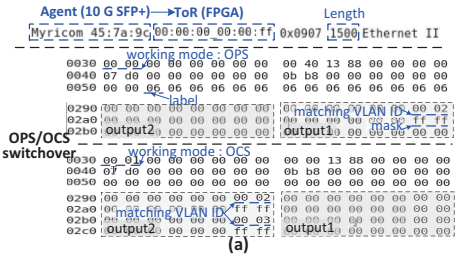


Fig.4 OPS Label Switching





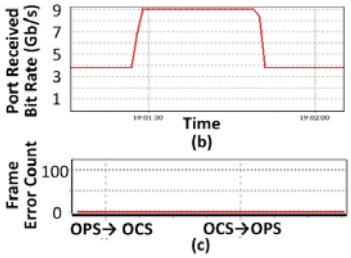


Fig.5 BER vs. Received Power Fig.6 OCS/OPS switchover: (a) Agent-to-ToR msg., (b) Port Received Bit Rate, (c) Error Frame

The SDN controller will push LUT with indicated labels to OPS during the configuration. In Fig. 4, two flows tagged with different optical labels at ToR1 are multiplexed and sent to OPS node as provisioned by ODL. The OPS local switch controller matches the detected label bits with the LUT (populated by the ODL). Based on the LUT, the OPS forwards the packets with label 01 to Port 1 and the ones with label 10 to Port 2 accordingly. It is worth to note from Fig. 4 that, once the connections and LUT are provisioned, the OPS switches each individual 30 us short flow according to the optical label information carried by the flow. The end-to-end latency is less than 252μs including ~100ns propagation delay and 35ns switch control, and most of the time is for traffic (de-) aggregation at ToR. BER of the received packets at the ToR2 and ToR3 are reported in Fig. 5. Power penalty has been observed less than 1.5 dB mainly due to the filtering effect of the label extractor [4] and the noise introduced by SOA gate.

3.2 OPS-based and OCS-based Connection Switchover

In this experiment, one OPS-based connection is established between two servers in different racks running with http service which is tagged with VLAN2, and the working mode of ToR1 is set as OPS as shown in the Agent-to-ToR Ethernet frame in Fig.6 (a). For a new incoming Rsync service (i.e., VLAN3) between the same pair of ToRs, an OCS connection intends to establish between them. To save network resources (e.g., OPS and backplane port), existing VLAN2 and incoming VLAN3 traffic are merged together and switched over to another ToR output port (i.e., output2) with the OCS connection successfully configured by control plane. A hitless switchover (no error frame detected in Fig.6 (c)) is implemented in the ToR switch by changing the working mode to OCS as shown in Fig.6 (a). Fig. 6 (b) shows the received bitrate variation of ToR port induced by the working mode switchover. OCS is able to achieve 9 Gb/s throughput for a 10Gb/s port, while OPS is working under 3.8 Gb/s in this setup due to the 60% packet overhead required for receiving packets in the FPGA (e.g., clock recovery), which can improved by using a fast burst receiver.

4. Conclusion

We experimentally demonstrated an SDN-enabled optical DCN based on AoD, OPS and FPGA-based ToR. With this DCN architecture, OCS/OPS connection provisioning in 753/214 ms and OPS connection with 252μs latency are demonstrated. Moreover, with the dynamically programmable control plane, the hitless OPS/OCS connection switchover has been demonstrated, which further improves the flexibility of optical interconnection in future DCN.

Acknowledgements

The work is supported by EU FP7 LIGHTNESS, COSIGN, EPSRC Hyperhighway, SONATAS.

References

[1] http://www.ict-lightness.eu/
[2] http://www.fp7-cosign.eu/
[3] B. Rofoee, et. al, "Programmable on-chip and off-chip network architecture on demand for flexible optical intra-Datacenters", OE 21 (5), 5475-5480 (2013).
[4] W. Miao, et.al., Novel flat datacenter network architecture based on scalable and flow-controlled optical switch system, OE, 22 (3), 2465-2472 (2014).