Dynamic Light Path Establishment in Switch Fabric Using OpenFlow

Sadia Qureshi and Robin Braun

May 13, 2021

Dynamic Light Path Establishment In Switch Fabric Using OpenFlow

Sadia Qureshi and Robin Braun
School of Electrical and Data Engineering
Faculty of Engineering and I.T,
University of Technology Sydney

***Abstract*—In today's optical networks, Light paths are established through the Network Management System, which is a manual and cumbersome process. Light Paths are computed and pre-setup according to the known traffic demands, and provisioning any new service takes time. Several efforts have recently been made to make the path establishment process dynamic, including the Software Defined Networking approach. However all of this work has assumed fully interconnected fabrics of the network devices, which is generally not the case. The task of the end-to-end path establishment between network elements and through the fabrics are interrelated, and this task remains incomplete, without consideration of the switch's limitations. This work highlights for the first time the issue of the path establishment dynamically through a switch fabric. The paper briefly explains the problem and suggests an SDN based solution using the OpenFlow protocol. This work contributes a representation of the basic framework which will be required to make the complete path setup process dynamic. The paper also includes an operational flow description.**

***Index Terms*—Optical switch, SDN, Light path**

I. Introduction

Rapidly growing demands for the Internet bandwidth and applications have provided an impetus for the change in the current complex and inflexible way of controlling optical networks [1]. To cater today's dynamic demands, carrier operators need a mechanism for the fast, simple and dynamic provisioning of bandwidth resources. Currently in the Core Transport Networks, a central Network Management System (NMS) is employed which is manual and inefficient. Operators at the Network Operation Center do the path planning offline and the network is then laid down according to that plan. Two main components of the optical networks are Optical Cross Connects (OXCs) and re-configurable optical add drop multiplexers (ROADMs) which may manually be configured to provide circuit-switched end-to-end Light paths. The connecting paths in successive switches' fabrics, along with the fiber transmission links between them, form the complete Light path across the network. The process for setting up a service through the network may take several months and once done, the connections usually remain there for a few months or even years. In the cases where a new service demand has to be catered for, there may be a need to change the external fiber connections of the switch manually. In the current control methodology although the computations are done prior, the service may still not be provided at times if no path is available through the any of the switch's internal fabric. The switch has its own control mechanism which uses complex algorithms to find a path through the fabric and setup physical cross connects. The tasks of setting up a path through the fabric and between the switches are interrelated [2], thus at the network level a single control mechanism is necessary which can have a view of the entire network connections. For a new connection request, the controller should be able to find a free path dynamically and decide if the connection request has to be accepted or rejected. This method will avoid wasted NMS resources and time.

The Software Defined Networks (SDN) approach has been recently developed and is being researched for various networking fields. The idea of Software Defined Optical Networks is also not new and has been in discussion in the last few years [3], [4], [5], [1]. However SDN has inherent characteristics for the packet networks while optical networks are circuit switched, the task is therefore challenging. A few efforts to mention are [6], [7], [8]. They deal with controlling paths through layer 0 i.e. the photonic (WDM) layer however they all assume fully interconnected fabrics, and the switches' limitations are neglected. In above mentioned works, the OpenFlow protocol extensions have been proposed however none of them have been tested in Mininet which is a popular SDN emulation platform. In our work we propose a way that a single SDN controller can have a view of the fabrics connections together with the links between the switches. We propose to use OpenFlow protocol and Mininet tool to find path through the network. The organization of the rest of the paper is as follows. Section IIgives the background and briefly describes the current control mechanism of optical switches. Section III on the following page is about related work and section IV on page 3 explains our proposed solution. Finally the paper is concluded in section V on page 4.

II. Optical Network and Management

Optical network elements that need to be managed include Optical Line Terminals (OLTs), ROADMs, Optical Amplifiers(OA) and Optical Cross Connects. ROADMs and OXCs are intermediate nodes where the switching or routing occurs.

NMS deal with the setting, releasing and keeping track of the path connections through the intermediate nodes. End-to-end routes through the network are computed and wavelengths for those paths are also assigned. NMS is implemented through another lower layer of systems called Element Management Systems (EMSs). An EMS is connected to a domain of the network elements usually from the same vendor. It has a view of only its associated network elements. Thus for the overall management, all EMSs in turn communicate with and report to the NMS. The EMS communicates to NEs through an existing traditional Data Communication Network using Simple Network Management Protocol or legacy protocol e.g. Transaction Language 1. In addition to the EMSs, a simplified local management system is usually provided to enable the operator to configure individual network elements manually. Timing diagram 1 explains the connection setup process in a conventional management system [9].

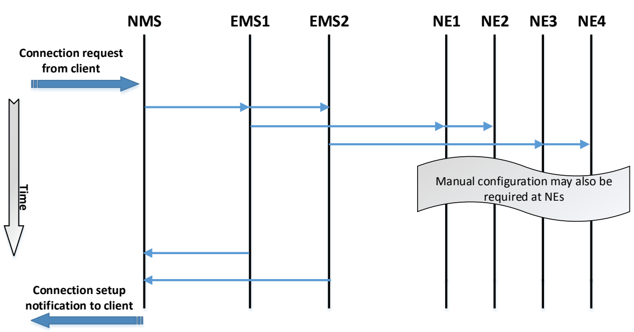

Fig. 1. Connection setup process through conventional NMS

A. Switch Fabric and Control Mechanism

Light paths in the Optical networks are switched at OXCs and ROADMs. Each switching device contains input modules/output modules, a switch fabric, and a control system. See figure 2. Light signals are received by input modules where control/signaling information are passed to the control system. Signals leaving the switching fabric are passed to output modules where control information is again appended. A Light path originates at an electrical-to-optical (E/O) transmitter in the ingress node, and terminates at an optical-to-electrical (O/E) receiver in the egress node. Light signal may or may not be converted into electrical domain at intermediate nodes for the switching purpose depending on the switch type, whether Optical-Electrical-Optical (OEO) or all optical (OOO). The switch fabric is usually a collection of optical domain switches purposed to switch an optical connection (wavelength) from an input port to any idle output port. Most of the optical space switches have been re-used from the rich collection of electronic multistage interconnection network architectures (MINs). Examples of these MINs include the optical crossbar architecture, the Benes and three-stage Clos networks [10]. Some of the switches are equipped with the wavelength converters to reduce the blocking probability due to unavailability of the wavelengths.

The control system inside the switching device controls the switch fabric. It has routing tables which have information of the paths through the fabric and their statuses. When a new request from the manager is received, it sends commands to the controller to setup (release) the cross connects associated with the paths. Depending on the switch fabric design architecture, there may be multiple paths between any pair of the input and output ports. It is also possible that there is no available path due to the limited connectivity or paths being reserved for other requests. The switch local controller uses complex algorithms to find a free connecting path in the switch fabric. If a free path is found, the controller then issues the respective control signals to change the states of the physical switches, and the current state of the switch fabric is updated. The types and details of the control algorithms can be found in [2].

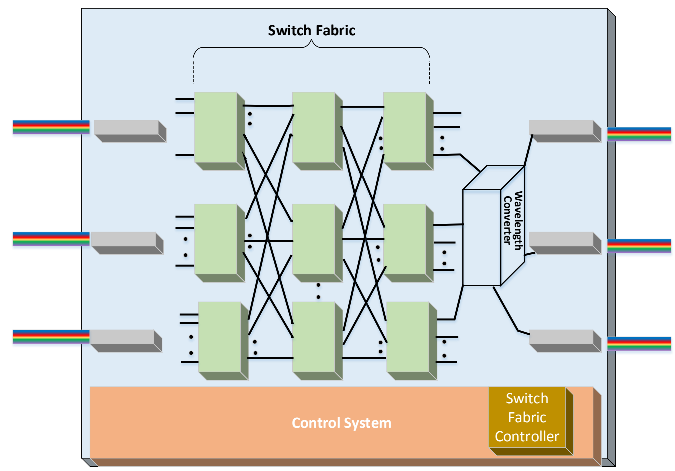

Fig. 2. General Optical Switch

III. Related Work

SDN has started to accommodate optical networks, and over the last decade there have been number of efforts to control Light paths using the OpenFlow protocol. A good survey of previous research work has been discussed in [11] and [12]. These research works propose OF-Extensions for dynamic circuit switching. In [8] the authors have proposed OpenFlow enabled optical switches which can get commands from the NOX controller to establish paths. However the request is generated from the IP router and the switch itself can not invoke requests for path setup. References [7], [13] develop an extended OF controller with an application to compute a Light path and implements it through an agent in the optical devices. It is not clear whether this work has considered the switch fabric's connectivity or not. Authors of [6] use almost same concept and make use of a Path Computation Engine (PCE) and Wavelength Assignment (WA) Algorithm in the controller for the Light path computations while an agent is being used for the communication between controller and the switch. In this work, a port emulation entity is also maintained which gets updates about the ports' statuses but no information about the fabric limitations. Open Networking Foundation's Open Transport Working Group (ONFTWG) is working towards the standardization of OpenFlow protocol for transport networks. The latest work [14] by a group of Fujitsu engineers proposed OF controller extensions which (with minor modifications) are accepted by the ONFTWG.They did port to port connections, however, the switch's connectivity constraints are not discussed. ON.Lab is working towards the control of converged packet and optical switches [15]. Recently in [16], they have presented control method using a dis-aggregated transport network.

IV. Proposed Solution

Previous research works assumed that the fabric is fully connected which is not generally the case. Thus even if the path setup instructions are given to the optical switch there may not be an available path through fabric, and the purpose of dynamic provisioning will fail. Also these works have been represented through experiments and no testing has been done in Mininet. Here we devise a mechanism where optical switch internal configurations are represented as a combination of regular layer two OpenFlow switches to the SDN controller. It is also worth mentioning that although the Open Flow Version 1.4, has some provision for the optical ports, Mininet does not support it, so our proposed mechanism uses the current capabilities of Mininet for path establishment.

A. Structural Overview

Our proposed mechanism dynamically retrieves the switches' internal configuration and links and sets up the Light path in an on-demand manner. when the optical signal arrives. Figure 3represents the overview of the proposed structure. The optical switches in the WDM network can be assumed to maintain their external connections through a topology discovery process, or manually populated lists (Neighbor Discovery is an issue, still to be addressed, as referenced by [15]). Our Resource System consists of two main components and may use SNMP/CLI/JSON to interact with the WDM network. The information collector and mapper component extracts the switches' resources and connectivity limitations and maps this information to create a mirror Mininet network. To the controller this network is a regular layer two OF switches' network. The bigger Mininet network can be considered as a collection of small clusters of the OF Switches where each cluster represents the fabric connectivity of a device, see figure 3.

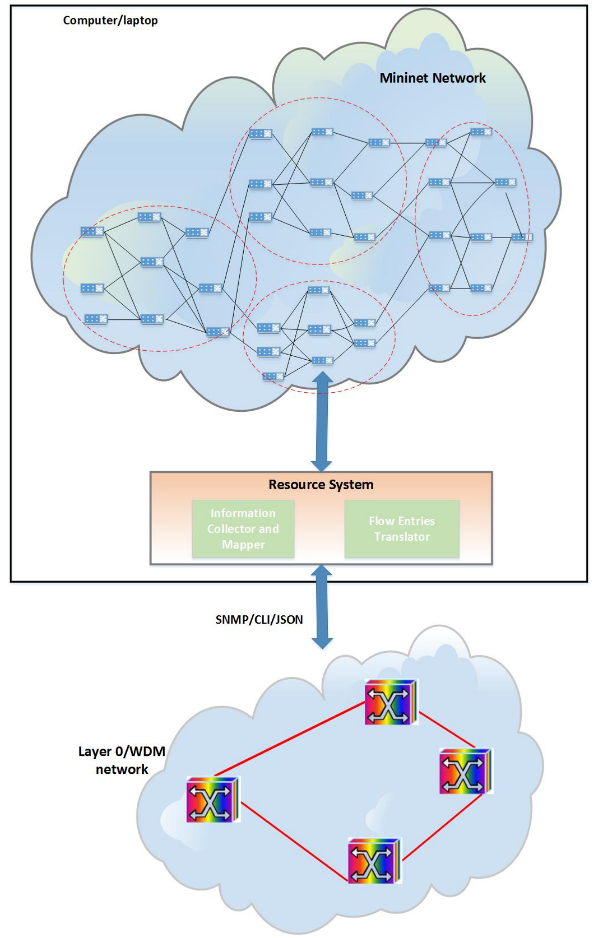

Fig. 3. Structural overview

B. Network Graph Creation

We have assumed that the Optical switch has tables/data sheets of its external connections, either manually populated or through a topology discovery mechanism. Our Information collector and mapper component can query the switch using SNMP/CLI/JSON to obtain the information about the switch's internal and external connections. After getting this information it invokes a Mininet network which replicates the same connectivity which is obtained from the WDM network. The SDN controller can then start its conventional method of network discovery and normal OpenFlow process may start. Figure 4 on the next pagerepresents the network graph creation process.

C. Operational Flow

Once the controller creates the network graph, the network is ready for path setup operation. Figure 5 on the following pageexplains the flow of steps. On the arrival of optical signal, the information about source and destination addresses is extracted and forwarded to the resource system. Here the switch is assumed to have capability of header extraction and conversion to electrical form. The resource system spoofs as a host and sends a frame to the port of the Mininet switch that corresponds to the optical switch's port on which the signal has arrived. The switch checks its flow table entries for matching, if there is no matching then the switch sends the packet to the SDN controller. The controller computes the path and installs the flow entries in all switches. The resource

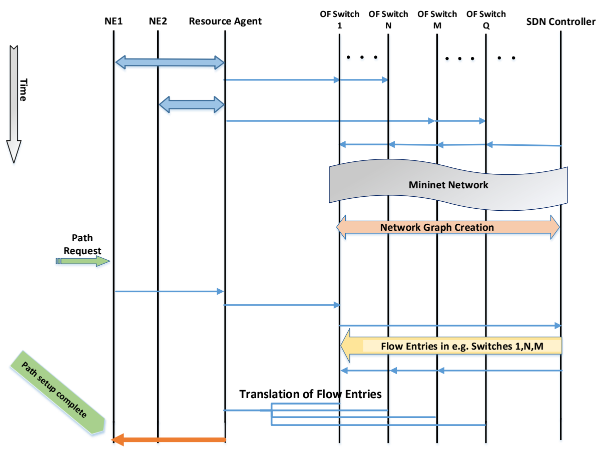

Fig. 4. Network Graph creation and connection setup

system based on the installed flow entries, can translate these into the commands that can setup cross connects at the optical switch and establish Light paths. During the setup phase, the frames that arrive are lost. Obviously this lost needs to be minimized. However, as setups occur relatively seldom, this is not considered to be a problem. However it needs investigation. See figure 5 for the flow of steps.

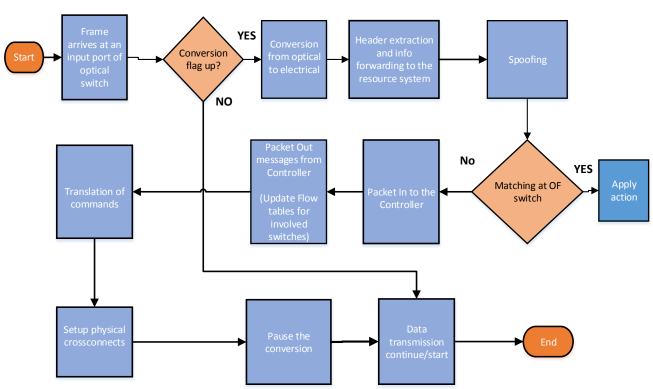

Fig. 5. Operational flow of steps

Step by step explanation of the flow chart:

1) On the arrival of Light signal (wavelength) on any port of the optical switch, the system converts it into electrical signal if needed (the flag for conversion is up).
2) Source and destination addresses are extracted from this electrical signal and forwarded to the resource system (which is running inside for example a computer system). SNMP /CLI is used for this purpose.
3) The resource system makes a packet with the source and destination addresses obtained from the optical device and sends to that port of the openflow switch which corresponds to the input port on the optical switch.
4) OpenFlow switch checks for the match of incoming packet and sends to the controller if there is a table-miss.
5) Controller computes the path through the mininet network and install flows in all switches that are part of the path, using Packet In messages.
6) The translator component of the resource system notices any change in the flow entries and convert these flow entries into suitable commands for physical cross connections in optical switches.
7) Since the path is now setup, the conversion from optical to electrical signals is paused for sometime (conversion flag down) and is started again after a regular interval.
8) The data transmission can be started/continued now.

V. Conclusions

Dynamic Light path establishment in WDM layer of core transport networks is an an important topic at the present time. However the aspect of a switches' limitations during path calculation has been neglected. In this paper we suggest a mechanism for a complete end to end path establishment while taking into account the switches' fabrics connectivity. The mechanism is SDN based and physical layer connectivity is abstracted as a Mininet network. Our solution is in the process of implementation and will be presented in future work with results. The purpose of this paper is to introduce our ongoing work and direct the attention to the issue that switch fabric should also be considered while path planning for the WDM layer.

References

[1] P. Bhaumik, S. Zhang, P. Chowdhury, S.-S. Lee, J. H. Lee, and B. Mukherjee, "Software-defined optical networks (SDONs): a survey," *Photonic Network Communications*, vol. 28, no. 1, pp. 4–18, aug 2014. [Online]. Available: http://link.springer.com/10.1007/s11107-014-0451-5

[2] T. S. El-Bewab, *Optical Switching*, T. S. El-Bawab, Ed. Boston, MA: Springer US, 2006. [Online]. Available: http://link.springer.com/10.1007/0-387-29159-8

[3] S. Das, G. Parulkar, and N. McKeown, "Unifying packet and circuit switched networks," *2009 IEEE Globecom Workshops, Gc Workshops 2009*, 2009.

[4] S. Gringeri, N. Bitar, and T. J. Xia, "Extending software defined network principles to include optical transport," *IEEE Communications Magazine*, vol. 51, no. 3, pp. 32–40, mar 2013. [Online]. Available: http://ieeexplore.ieee.org/document/6476863/

[5] ONF Solution Brief, "OpenFlow-enabled Transport SDN," 2014.

[6] M. Bahnasy, K. Idoudi, and H. Elbiaze, "OpenFlow and GMPLS Unified Control Planes: Testbed Implementation and Comparative Study," *Journal of Optical Communications and Networking*, vol. 7, no. 4, p. 301, 2015. [Online]. Available: http://www.opticsinfobase.org/abstract.cfm?URI=jocn-7-4-301

[7] M. Channegowda, P. Kostecki, N. Efstathiou, S. Azodolmolky, R. Nejabati, P. Kaczmarek, A. Autenrieth, J.-P. Elbers, and D. Simeonidou, "Experimental Evaluation of Extended OpenFlow Deployment for High-Performance Optical Networks," *European Conference and Exhibition on Optical Communication*, p. Tu.1.D.2, 2012. [Online]. Available: https://www.osapublishing.org/abstract.cfm?uri=ECEOC-2012-Tu.1.D.2

[8] L. Liu, T. Tsuritani, I. Morita, H. Guo, and J. Wu, “Experimental validation and performance evaluation of OpenFlow-based wavelength path control in transparent optical networks,” *Optics Express*, vol. 19, no. 27, p. 26578, dec 2011. [Online]. Available: https://www.osapublishing.org/oe/abstract.cfm?uri=oe-19-27-26578

[9] D. Pindarakis and S. Biswas, “Management and control of optical networks,” 2002, pp. 31–47.

[10] H. S. Hamza and J. S. Deogun, “Wavelength-exchanging cross connects (WEX) - A new class of photonic cross-connect architectures,” *Journal of Lightwave Technology*, vol. 24, no. 3, pp. 1101–1111, 2006.

[11] R. Alvizu, G. Maier, N. Kukreja, A. Pattavina, R. Morro, A. Capello, and C. Cavazzoni, “Comprehensive Survey on T-SDN: Software-Defined Networking for Transport Networks,” *IEEE Communications Surveys and Tutorials*, vol. 19, no. 4, pp. 2232–2283, 2017.

[12] D. C. Kilper, “Optical Physical Layer SDN [Invited],” vol. 10, no. 1, pp. 110–121, 2018.

[13] D. E. Simeonidou, R. Nejabati, and M. Channegowda, “Software Defined Optical Networks Technology and Infrastructure: Enabling Software-Defined Optical Network Operations,” *Optical Fiber Communication Conference/National Fiber Optic Engineers Conference 2013*, vol. 5, no. 10, p. OTh1H.3, 2013. [Online]. Available: https://www.osapublishing.org/abstract.cfm?uri=OFC-2013-OTh1H.3

[14] S. Yamashita, A. Yamada, K. Nakatsugawa, T. Soumiya, M. Miyabe, and T. Katagiri, “Extension of OpenFlow protocol to support optical transport network, and its implementation,” *2015 IEEE Conference on Standards for Communications and Networking, CSCN 2015*, pp. 263–268, 2016.

[15] M. D. Leenheer, G. Parulkar, and T. Tofigh, “SDN Control of Packet-over-Optical Networks,” 2015.

[16] M. D. Leenheer, Y. Higuchi, T. Furusawa, and G. Parulkar, “SDN Control of Optical Networks,” *Ecoc*, no. 1, pp. 1–3, 2017.