Design and Implementation of SDN-Based Packet Transport Networks

Wen-Chun Hsu¹², Ping-Fan Ho², and Jyh-Cheng Chen²
¹Loop Telecommunication International, Inc., Hsinchu, Taiwan
²Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan

Abstract - Packet Transport Network (PTN) has been widely used in backbone transmission networks. It plays a key role to simultaneously connect 2G/3G/4G network equipment in cellular networks to reduce operation cost. Meanwhile, the emergence of Software-Defined Networking (SDN) provides an open standard approach to dynamically configure and manage network equipment and data plane through a centralized control plane. A migration from legacy PTN to a software defined PTN is one of the most critical issues for future PTN. In this paper, we first introduce the correlation between the two techniques, and then discuss the integration techniques. To proof the concept, an SDN-enabled PTN prototype is implemented. The SDN-enabled PTN switch provides a bunch of software-defined functions. We demonstrate through extensive experiments that the switch is highly reliable and flexible and can reduce operation cost significantly.

Keywords: SDN, PTN, Mobile Backhaul network, MPLS-TP, OpenFlow

I. Introduction

Packet Transport Network (PTN) plays an important role in the integration of various communication techniques, such as Time-division multiplexing (TDM), Synchronous Digital Hierarchy (SDH), and Ethernet/IP, etc. These techniques are introduced and deployed from time to time. PTN provides the possibility of transparent transmission of data traffic between network equipment with these techniques. Networks are then generally independent of higher layer networks that work with these techniques.

Meanwhile, the emergence of Software Defined Networking (SDN) separates traditional switching functions into a control plane and a data plane. In such design, the control plane has a global view and centralized knowledge of a whole network. It enables the functionalities of dynamic network configuration and packet flow adjustment. Such software defined concept is happened to be coincident with that of Multiprotocol Label Switching – Transport Profile (MPLS-TP). Integrating SDN and PTN is expected as the future evolution of PTN.

In this paper, we first introduce and discuss the feasibility of the integration between SDN and PTN on the demand for future 5G networks. We then propose a novel approach for integrating SDN and PTN. Next, a proof of concept SDN-enabled PTN switch prototype is developed. Through extensive experiments, the results show that the developed SDN-enabled PTN switch is highly reliable and flexible, and can reduce operation cost significantly for mobile operators.

The rest of the paper is organized as follows. Section II gives some background materials on PTN and SDN. Section III introduces the proposed integration scheme for SDN and PTN. Section IV illustrates the experiment results. Section V addresses potential applications. Related work is reviewed in Section VI. Section VII offers conclusions.

II. Background

Packet Transport Network (PTN) is a packet switch based multi-service unified transmission technology. It well provides carrier-class Ethernet service to satisfy four basic requirements: (1) high reliability, (2) flexible-expansibility, (3) strict quality of service (QoS), and (4) perfect operation administer maintenance (OAM). PTN inherits the graphics user interface of synchronous digital hierarchy (SDH), enabling flexible network expansibility, error correction, and channel monitoring capability.

As the mainstream technology of PTN, MPLS-TP (multi-protocol label switching-transport profile) [2] was originally developed by the IETF to establish a cost efficient way of routing traffic in high-performance networks. MPLS-TP enables delivery path independent from the control panel and keeps the end-to-end label forwarding capability of MPLS. The separation of the control plane and data plane leads to higher network stability, reliability, and flexibility. Also, MPLS-TP simplifies MPLS with respect to decreased equipment, operation, and maintenance costs.

Accordingly, MPLS-enabled PTN has the following functions:

1. PWE3 (Pseudo Wire Emulation Edge-to-Edge) [3]: PWE3 is used to be compatible with traditional circuit switch services, e.g., E1/T1 TDM, Ethernet, ATM, SDH, PDH, PPP/HDLC etc., and ensures that the bearer layer is unified with various transmission protocols.
2. VPLS (Virtual Private LAN Service) [4]: VPLS is one sort of Virtual Private Networks (VPN), which provides multipoint to multipoint communication over Ethernet, as shown in Fig. 2. Thus, VPN users can have broadcast capabilities LAN in Layer 2, while end users can get a virtual private Ethernet connection service in the Layer 2 under the VPN. Then messages can be transferred to any point using the VPN.

3. Providing Linear Protection [5] for achieving 1+1, 1:1, and Ring Protection [6], by using packet transport network protection technologies, which enables to achieve less than 50 ms protection reliability as SDH.

4. Utilize complete SDH-like Operations, Administration and Management (OAM) [7] management functions. As shown in Fig. 3, those OAM functions strengthen operations, implementation and management of the transport layer network via Section, Tunnel, and PW OAM levels.

Strengthen PTN network by using SDN technology

Recently, SDN technology characterized as centralized dynamic routing management features becomes an important emerging technology. Compared with traditional Ethernet closed-end architecture, SDN can be programmed through software and centralized controllers for open network architecture. Thereby reducing equipment and maintenance cost for network operators.

To separate control plane and data plane, SDN architecture is divided into three tiers, i.e., application tier, control tier, and data tier. Meanwhile, a new key component, SDN controller, is used to centralized control the entire network behaviors and packet flows. The communication between the SDN controller and SDN switches is based on OpenFlow [1] protocol. The management and configuration of SDN switches is conducted based on OF-Config protocol [1], which locates between management system and SDN switches. In other words, real-time configurations, such as create/modify flow entry and routing path, are performed by OpenFlow protocol. Whereas, OF-Config protocol is responsible for the configuration of IP address of OpenFlow switches, enable/disable ports of the switches.

According to new architecture and protocols, SDN can dynamically configure SDN switches to adjust bandwidth configuration. However, practical commercial systems still include PTN networks and other optical communication equipment, which is the next integration target. Thus, in this paper we propose SDN-based PTN solution to comprehensive integrate PTN and optical communication equipment to enable network slicing, dynamically configuration of transport network, and reduce network deployment cost and time.

III. SDN-based PTN Implementation

In this section, we introduce the implementation details of SDN-enabled PTN.

3.1 SDN Extension Module

To implement SDN-based PTN system, a SDN-enabled module should be added on both controllers and switches, refer to SDN extension module.

The module is to address the integration problems between SDN and PTN. We implement OpenFlow and OF-Config into the module to enable PTN having centralized knowledge of the network and control capability of various network equipment. We detail them as follows.

1. LSP
   - Create and configure Provider (P) and Provider Edge (PE) of LSP.

2. PW
   - Create and configure Pseudo Wire (PW).

3. Tunnel
   - Create and configure Tunnel.

4. VPWS
   - Support for Ethernet Virtual Private Wire Service (VPWS).
   - Support for Attach Circuit (AC).

5. Tunnel APS
Support 1:1 Automatic Protection Switch (APS) for LSP of tunnel
6. VPWS APS
Support 1:1 APS for PW of VPWS
7. LSP OAM CCM
Support MPLS-TP OAM CCM (Continuity Check Message) on LSP level, MEG (Maintenance Entity Group), and local MEP (MEG End Point) and remoted MEP (MEG End Point)
8. LSP OAM LM / DM
Support MPLS-TP OAM LM (Loss Measurement), and OAM DM (Delay Measurement) on LSP level
9. PW OAM CCM
Support MPLS-TP OAM CCM (Continuity Check Message) on PW level, MEG (Maintenance Entity Group), and local MEP (MEG End Point) and remoted MEP (MEG End Point)

Table I shows the items in SDN standard correspondent to the above modules.

Table 1: PTN extension functionalities correspondent to OF-Config, OpenFlow standard.

<table>
<thead>
<tr>
<th>Item</th>
<th>Standard</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. LSP</td>
<td>OF-Config</td>
</tr>
<tr>
<td>2. PW</td>
<td>OF-Config</td>
</tr>
<tr>
<td>3. Tunnel</td>
<td>OF-Config</td>
</tr>
<tr>
<td>4. VPWS</td>
<td>OpenFlow</td>
</tr>
<tr>
<td>5. Tunnel APS</td>
<td>OF-Config</td>
</tr>
<tr>
<td>6. VPWS APS</td>
<td>OF-Config</td>
</tr>
<tr>
<td>7. LSP OAM CCM</td>
<td>OF-Config</td>
</tr>
<tr>
<td>8. LSP OAM LM/DM</td>
<td>OpenFlow</td>
</tr>
<tr>
<td>9. PW OAM CCM</td>
<td>OF-Config</td>
</tr>
</tbody>
</table>

3.2. System Design
We implemented an SDN-based PTN prototype with the functionalities addressed in Section 4.1. Fig. 4 illustrates the system architecture, which consists of switch, controller, and application. We detail them as follows.

3.2.1 Switch
As OVS (Open vSwitch) provides source code, we extend PTN functionalities based on it. As shown in Fig 5, we implement a PTN adapter layer between protocol stack and hardware for extending OpenFlow flow entry. In addition, a PW_FWD engine module is implemented in hardware abstract layer (HAL) to support MPLS-TP packet termination.

3.2.1.1 PTN Adapter Layer

![Figure 5: PTN adapter layer.](image)

Table 1: PTN extension functionalities correspondent to OF-Config, OpenFlow standard.
Monitor the performance of end-to-end connection, such as packet loss, latency, and jitter etc.

- Event module
  Receive event message from HAL, e.g., MPLS-TP OAM CCM packet loss, and then send it to APS (Automatic Protection Switch) module to trigger MPLS-TP APS.

3.2.1.2 OpenFlow Extension

When attaching an Attach Circuits (AC) to a Pseudo Wire (PW) operations, the input port is specified as AC using OpenFlow flow entry, while the output port is set as MPLS-TP virtual interface, as shown in Fig. 7. Then the information of MPLS-TP circuit can be found in the MPLS-TP circuit database by using MPLS-TP virtual interface ID, which forms Ethernet VPWS applications.

![Figure 7: The relationship between OpenFlow flow entry and MPLS circuit.](image)

Fig. 8 illustrates the procedures of creating flow entries. Input interfaces are set as rules for incoming flow entries. If the incoming flow entries match with the rules, both LSP label and PW label are attached to the packets. Then the packets can be transferred in the MPLS-TP networks based on the labels.

![Figure 8: The application of OpenFlow for supporting MPLS circuit services.](image)

3.2.1.3 PW_FWD Engine

Here, we extend the packet processing procedure of OpenFlow. We created a PW_FWD virtual interface to extend the packet processing procedure of OpenFlow. The goal of the PW_FWD virtual interface is to redirect the packet to Flow Table for comparison again to realize the termination function of PTN switch for MPLS-TP packets, as shown in Fig. 9.

![Figure 9: PW_FWD Engine operation procedure.](image)

1. The switch’s network interface receives MPLS-TP packets from virtual private wire service. The packets consist of payload, PW label, and LSP label.
2. When a packet arrives the Flow Table, its LSP label is compared with Rule 1. Then the packet is forwarded to PW_FWD virtual interface.
3. PW_FWD Engine sends the packet to Flow Table again for Rule 2 comparison. This step de-capulate PW label and outputs.
4. The switch outputs the packet to the virtual private wire service corresponding Attach Circuit (AC).

3.2.2 PTN APP

PTN APP provides user interface for PTN operation, which is used to configure MPLS-TP circuit service on SDN-based PTN switches. The PTN APP can also operate Automatic Protection Switch and MPLS-TP OAM function for debugging and diagnostic.

3.2.3 Controller

Since Ryu provides complete source code, we then developed the extension module addressed in Section 4.1 based on it. The controller supports App and send commands to switches.

OpenFlow protocol extension

In our implementation, we extend the OpenFlow protocol, so that it can control LM/DM function module of SDN switches via PTN deployment layer. In addition, we can obtain the MPLS-TP OAM performance monitoring results using the extended protocol.

OF-CFG protocol extension

OF_CFG protocol extension is used to configure the App in the controller. It has the following functions:

- Create MPLS-TP circuit
  Configure LSP label, PW label, Physical output port, MPLS virtual interface ID, and APS group.
- Create MPLS-TP OAM CCM function
  MPLS-TP OAM CCM can conduct error test and trigger APS. It also can configure OAM MEG (Maintenance Entity Group), local MEP (MEG End Point) and remoted MEP (MEG End Point) and CCM Interval.
IV Performance Evaluation

To proof the concept of the proposed SDN-based PTN system, we implemented a prototype and conducted extensive evaluation experiments in terms of APS, transmission latency, and dynamic bandwidth adjustment. The experimental results are compared with the performance of the off the shelf PTN switch, Loop Telecom G7860 [13].

4.1. PTN APS

PTN APS is an importance feature for PTN networks. Here, we conducted an experiment to test if our prototype can conduct APS within 50ms.

Experimental Setting:

As shown in Fig. 10, we designed a PTN APS circuit for testing. Arrival traffic were generated by a TestSet with 1000 packets/sec. We disconnected the line between SW1 and SW2 manually to test if PTN conducts APS or not, and used Wireshark to measure the packet arrival time and APS time.

![Figure 10: An example of PTN and Ethernet virtual circuit service.](image)

Experimental results

Fig. 11 illustrates the impacts of APS on received packets. One can see that the received packets are down to zero when APS is conducted. So the APS switching interval can be measured based on it.

Fig. 12 further shows the experimental results in term of APS switching time. We observe that our SDN-based switch’s APS switching time is only 10ms, which is much shorter than that of Loop Telecom G7860. As compared to Loop Telecom G7860, the SDN-based PTN switch does not occupy many computing resources on control logic, reducing switching time.

4.2. PTN transmission latency

Experimental setting

Low transmission latency is desirable for PTN networks. As shown in Fig 13, the transmission latency is defined as the time interval between when a packet arrives on the switch 1 and when the packet leaves the switch 2. Both MPLS-TP label encapsulate time and de-capsulate time are included.

![Figure 13: Experimental setting for transmission latency.](image)

Experimental results

Fig. 14 shows the transmission latency comparison results between the SDN-based switch and Loop Telecom G7860 switch. One can observe that our switch has lower transmission latency compared with the off the shelf one. The reason is that the off the shelf one have more packets processing functions so that it takes more time to handle the packet, leading to longer latency. Whereas, there is no control logic in the SDN-based switch. Only OpenFlow Flow Table is used to handle the packet, which simplifies the off the shelf switch for more efficiency.

![Figure 12: PTN APS switching time.](image)
4.3. Dynamical Bandwidth Allocation Module

Experimental Setting

Bandwidth allocation is a key feature for PTN network and 5G networks. Here, we tested dynamic bandwidth allocation capabilities of the SDN-based PTN switch. Firstly, the controller monitored the information of flow entries in all switches. Then, in order to control the incoming flow within the total bandwidth, meter table was used for rate limiting function.

Experimental setting is as shown in Fig. 15, the Switch 1 receives two traffics (Traffic 1: 500Mbps and Traffic 2: 700Mbps). Then the controller send a command to the Switch 1 that reduced the Traffic 1 down to 383Mbps and the Traffic 2 down to 616Mbps. So the total output bandwidth was limited to 1Gbps.

Experimental Results

Table 2 shows the experimental results in terms of packet loss rate. As dynamic rate limiting function is disabled in the experiment, we observe that the packet loss rate for the two traffics are different. The Switch 1 could not identify high priority traffics.

Table 2: The experimental results in terms of packet loss rate (rate limiting function disabled).

<table>
<thead>
<tr>
<th>Data Flow</th>
<th>Input (Mbps)</th>
<th>Output (Mbps)</th>
<th>Loss Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traffic 1</td>
<td>500</td>
<td>383.3</td>
<td>23%</td>
</tr>
<tr>
<td>Traffic 2</td>
<td>700</td>
<td>616.7</td>
<td>12%</td>
</tr>
</tbody>
</table>

Table 3 shows the experimental results with rate limiting functions. The Traffic 1 is set as a high priority which is higher than the Traffic 2. Then the Switch 1 identifies the priorities of the Traffic 1 and Traffic 2 and gradually increases the output of the Traffic 1 to 500Mbps. Meanwhile, in order to guarantee that the total output bandwidth does not exceed 1Gbps, the output of the Traffic 2 is decreased to 500Mbps as well. The experimental results show that rate limiting function can dynamically adjust traffic based on their priority.

Table 3: Experimental results in terms of packet loss rate (with rate limiting function).

<table>
<thead>
<tr>
<th>Data Flow</th>
<th>Input (Mbps)</th>
<th>Output (Mbps)</th>
<th>Loss Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traffic 1</td>
<td>500</td>
<td>499.3</td>
<td>0.1%</td>
</tr>
<tr>
<td>Traffic 2</td>
<td>700</td>
<td>616.7</td>
<td>28.5%</td>
</tr>
</tbody>
</table>

V Future Work and Challenges

In the future, 5G networks are expected to integrate various technologies, such as Internet of Things (IoT), vehicle networks, cloud computing, and big data. Here, we list the following research challenges for transport networks:

1. Difficulty for integrating various transport technologies: Existing transport networks such as TDM, E1 / T1, SDH and Ethernet / IP network technology are very complex. In response to the rise of SDN technology, new SDN switches make networks more difficult to integrate.
2. Various 5G network service requirements: Existing IP based networks are not easy to provide the same QoS for all services. As in the future network, networks are divided into network slicing, which are responsible for their corresponding services, respectively. SDN-based PTN technology is expected to help with those needs.
3. Low latency dynamic bandwidth adjustment: Network service providers provide network services to virtual network operators. To increase bandwidth utilization rate, network service providers should be able to dynamic adjust bandwidth between those virtual network operators with low latency.
4. High reliability service requirements: Some special users require high reliability network services. For example, electrical system security monitor service is responsible for detecting dangerous situation for electrical short-circuit event and provides rapid response for alarms. SDN-based PTN technology is expected to fulfil these requirements.

VI Related Work

The SDN-based PTN has been intensively studied [8-12]. In [8]-10], the authors proposed MPLS-TP label switch service. However, these work do not support...
MPLS-TP OAM service and cannot be used for routing performance analysis, debugging, and APS.

Alternatively, the authors in [11] use two flow tables for supporting MPLS-TP label switching services. However, this approach requires that switches have two flow tables for encapsulating/de-encapsulating MPLS-TP label of packets, which is a constraint for off the shelf switches. The authors in [12] introduce SDN-based PTN for optical networks. However, the work cannot support MPLS-TP OAM and cannot be used for routing performance analysis, debug, and APS.

VII Conclusions
In this paper, we implement and test our SDN-based PTN switch prototype. Through extensive experiments, we demonstrated that the switch has much better performance than traditional PTN switches in terms of reliability, flexibility, and operation cost, which is more suitable for future 5G networks. As our future work, we will continue working on the standardization of SDN-based optical networks.

Reference