Applying a Resource-pooling Mechanism to MPLS-TP Networks to Achieve Service Agility

Tomoyuki Iijima, Toshiaki Suzuki, and Kenichi Sakamoto
Central Research Laboratory
Hitachi, Ltd.
Kanagawa, Japan
{tomoyuki.iijima.fg, toshiaki.suzuki.cs, kenichi.sakamoto.xj}@hitachi.com

Hidenori Inouchi and Akihiko Takase
Telecommunications & Network Systems Division
Hitachi, Ltd.
Kanagawa, Japan
{hidenori.inouchi.dw, akihiko.takase.wa}@hitachi.com

***Abstract*— A concept called "software-defined networking" (SDN) is applied to carrier networks as one way to add flexibility to those networks. To apply SDN to carrier networks, a resource-pooling mechanism in MPLS-TP networks is proposed. The feasibility of the proposed resource-pooling mechanism applied to Multi-Protocol Label Switching – Transport Profile (MPLS-TP) networks was evaluated in terms of service agility. Moreover, a controller, which utilizes this mechanism to allocate a pooled resource to IP traffic, is prototyped and evaluated. The time required for the controller to allocate pooled resources in MPLS-TP networks to IP traffic is sufficiently short. This result indicates that the proposed mechanism will help to flexibly change carrier networks and reduce manual configurations spanning multiple layers. Consequently, the proposed mechanism will help assure service agility.**

Keywords-Cloud computing; SDN; MPLS-TP; service agility;

I. INTRODUCTION

As cloud computing continues to grow, the number of cloud services is increasing dramatically [1]. Cloud computing is a technology that enables users to access "a large pool of data and computational resources" located far afield via the Internet [2]. These resources are mostly deployed in data centers and are provided to users by making full use of "virtualization." Through these resources, "dynamically composable services" can be deployed through "Web service interfaces [3]." Users can easily and flexibly start their own services using these resources.

For example, online-game providers can use a large amount of computational resources during the launch of their service in order to attract a large number of users. They can thus reduce the amount of investment that they would otherwise have needed if they prepared the resources themselves.

People are now using these resources as if they were located on local computers. A cloud-computing environment is largely supported by the rapidly increasing bandwidth available on the Internet. Even so, it is hard to say that network resources are virtualized enough. Although traffic patterns produced by cloud computing are volatile because of sporadic increases and decreases in the number of Virtual Machines (VMs), networks are not necessarily changed flexibly enough to keep up with such volatility.

In light of this background, Software-Defined Networking (SDN) is getting wide attention. SDN is a concept that separates the control plane of network devices from their data plane and puts the control plane in one place. The controller put in that place then controls the entire network [4]. This centralized controller is expected to have the capability to control network resources virtually. It is also expected to provide flexibility and reliability on behalf of network devices by making full use of virtualization.

These technological trends in networks are often mentioned within the context of data-center networks. However, they are not limited to data-center networks. Carrier networks, i.e., the backbone-network infrastructure managed by telecommunication-service providers, are also affected by the volatile nature of traffic resulting from the increasing number of services provided on the cloud-computing platform. In this regard, it is necessary that carrier networks support SDN; in other words, carrier networks must be virtualized and flexible [5].

There are, however, several issues concerning current carrier networks, and these issues are described in Section II. To address these issues, as described in Section III, "Multi-Protocol Label Switching – Transport Profile" (MPLS-TP) networks have been proposed as a field where SDN is introduced in carrier networks. A resource-pooling mechanism in MPLS-TP networks and collaboration between MPLS-TP and IP networks have also been proposed. The application of the proposed mechanism is described in Section IV, and the proposed mechanism is evaluated in Section V. Finally, future work concerning the mechanism is mentioned in Section VI.

II. ISSUES CONCERNING CARRIER NETWORKS

As mentioned in Section I, network traffic produced by cloud computing is putting a heavy burden on carrier networks. The volatile nature of traffic generated by VMs

 ISBN: 978-1-61208-338-4

and various kinds of services, such as video streaming and gaming on-line, that are provided on a cloud-computing platform are also negatively affecting carrier networks. According to Manzalini et al. [6], for example, moving a VM across a Wide-Area Network (WAN) requires at least 622 Mbps of bandwidth throughout the WAN. On top of that requirement, due to accelerating globalization, carrier networks are expected to deal with more-frequent requests to make network changes spanning long distances. In this regard, introducing SDN into carrier networks will help to satisfy these expectations.

However, current carrier networks have at least two issues concerning introducing SDN. As for the first issue, carrier networks must be tightly controlled to provide high reliability. As for the second issue, they must be multi-layered and allow communications between operators of different layers. These issues are described in detail as follows.

A. Rigidness of carrier networks

In contrast to the "routed-packet network [7]," which is highly distributed, and therefore, not necessarily expected to provide high reliability, a carrier network is expected to ensure high reliability. For example, transport technologies used by carrier networks, such as Synchronous Optical NETwork/Synchronous Digital Hierarchy (SONET/SDH) [8], are equipped with high-reliability functions such as guaranteeing bandwidth, path protection within 50 milliseconds, and "Operation, Administration, and Maintenance" (OAM). Provisioning in the transport layer is, therefore, rigid and requires significant human intervention [9].

B. Multi-layers in carrier networks

Current carrier networks are mostly multi-layered, and their core is made from optical transport networks, consisting of Wavelength-Division Multiplexing (WDM) [10] devices and SONET/SDH devices. Routed-packet networks, namely, IP networks consisting of routers and switches, surround the core of the carrier networks. As of now, each type of device is managed by a different operator. When carriers want to change their networks, operators of different layers thus have to communicate with each other, resulting in a long lead time. Setting up or modifying carrier networks, which involves a man-to-man interface, lasts days or even weeks [11].

These issues are making it difficult for carrier networks to meet current demand and to achieve service agility. It is thus acutely necessary to realize flexibility without undermining the current high reliability of carrier networks.

III. PROPOSAL OF RESOURCE POOLING

To address the issues described in the previous section, a packet-transport technology called "MPLS-TP" is applied as one field in which SDN can be introduced. To realize service agility, a resource-pooling mechanism on MPLS-TP networks and collaboration between MPLS-TP and IP networks are proposed. This approach is similar to CloudNet [12] in that a controller assigns a pooled network resource to a user, but it is different in that the pooled resource is in the transport layer of a carrier network.

A. MPLS-TP as a packet-transport technology

MPLS-TP is a protocol being standardized by the Internet Engineering Task Force (IETF). Originally, the International Telecommunication Union Telecommunication Standardization Sector (ITU-T) was working on the preceding Transport – Multi-Protocol Label Switching (T-MPLS) protocol in order to emulate the SONET/SDH protocols by developing a whole new range of carrier-class service attributes [13]. However, the IETF took over the T-MPLS standardization as MPLS-TP in order to provide compatibility with IP/MPLS. Accordingly, it has been reported that the stage is set for replacement of SONET/SDH in packet networks [13].

The architecture of MPLS-TP is shown in Figure 1. The controller manages the entire MPLS-TP network [14], in a sense that the control plane and data plane are separated, MPLS-TP networks are managed in the same manner as SDN. The controller "simply sets up one tunnel," namely, a Label-Switched Path (LSP) and a pseudo wire, "from source to destination [15]." These tunnels, "due to their deterministic nature of bandwidth and delay, provide a carrier-class solution for transport of any payload [15]."

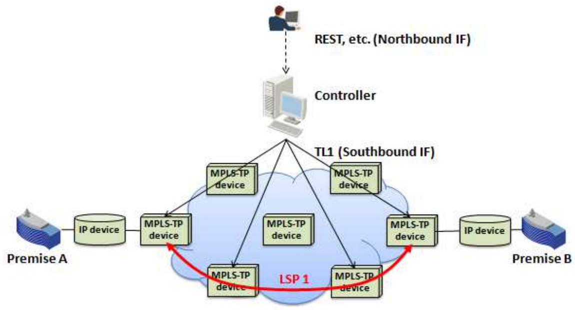

Figure 1. Architecture of MPLS-TP networks.

The functions of MPLS-TP for providing high reliability are well developed. For example, once configured on a MPLS-TP device, a working LSP is required to switch to a standby LSP within 50 milliseconds when it is damaged. The continuity over the LSP is checked before the first packet is sent from the source to destination.

Although one of the advantages of SDN is that the controller ensures reliability on behalf of network devices by making full use of virtualization, it is suggested here that reliability should be ensured by MPLS-TP devices. On top of that suggestion, it is also suggested that flexibility in carrier networks should be attained by introducing a resource-pooling mechanism on MPLS-TP networks.

B. Resource-pooling mechanism in MPLS-TP networks

As a method to flexibly change MPLS-TP networks in a short lead time in accordance with certain traffic patterns or user demands, using a LSP and a pseudo wire as a resource pool is proposed in this section.

 ISBN: 978-1-61208-338-4

According to Wischik et al. [16], "resource pooling" is defined as a technology that "helps robustness against failure, load balancing, and flexibility in the face of bursts of traffic while avoiding problems and limitations." Moreover, according to Gmach et al. [17], "many enterprises are beginning to exploit a shared-resource-pool environment to lower their infrastructure and management costs." Under these circumstances, the proposed resource-pooling mechanism on a carrier network therefore targets achieving flexibility in the face of bursts of traffic.

To create a resource pool composed of LSPs and pseudo wires, first, the resources should be set in advance. The status of these resources, such as "used" and "unused," should be held by the controller. When in need of "flexibility in the face of bursts of traffic," the controller activates the pooled and unused LSPs and pseudo wires. It then allocates the resource to a certain traffic pattern or a user demand.

C. *Resource allocation through collaboration*

When a pooled resource of LSPs and pseudo wires is used to transport traffic sent from "routed packet networks," this traffic sent over the pooled resource is IP traffic. In this case, the controller needs to know the attributes of the IP traffic. Accordingly, the controller must manage not only MPLS-TP networks but also IP networks. As shown in Figure 2, the controller then allocates pooled resources in MPLS-TP networks to IP networks.

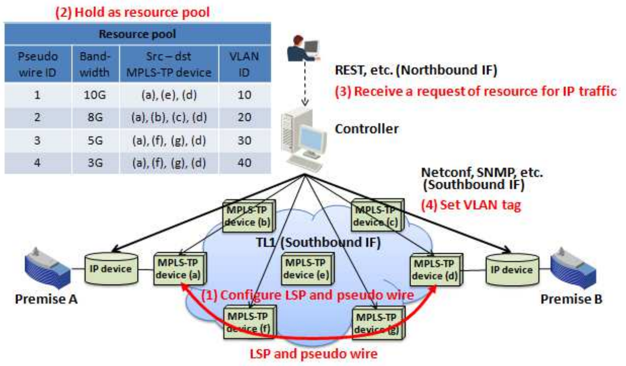

Figure 2. Collaboration between MPLS-TP and IP networks.

There are several possible scenarios about how to use a resource pool on an MPLS-TP network. One of these scenarios is shown in Figure 2. First, the controller configures LSPs and pseudo wires in advance, such as ones that starts from MPLS-TP device (a) through (e) to (d), and holds these resources as a pooled resource. The controller then associates each pseudo wire with a virtual-LAN identifier (VLAN ID) in order to transport IP traffic over the pseudo wire [18]. For example, in the figure, pseudo wire with ID "1" is associated with VLAN ID "10." Therefore, IP traffic with VLAN ID tag "10" in its layer-2 header, which is sent to MPLS-TP device (a), will be transported over the pseudo wire with ID "1." This traffic will thus be transported from MPLS-TP device (a) through (e) to (d).

When a user of IP traffic demands that its traffic is sent between premises A and B with guaranteed bandwidth of 10 Gbps within a certain period of time, the controller determines that the pseudo wire with ID "1" meets this demand. It then allocates that pseudo wire to that IP traffic. For example, when a user demands that its IP traffic from address "100.100.100.0/24" of premise A to address "100.100.200.0/24" of premise B is sent over carrier networks with guaranteed bandwidth of 10 Gbps within a certain period of time, under the assumption that the routing configuration has already been made on the edge IP devices, the controller assigns VLAN ID tag "10" to the IP traffic on the edge IP devices.

A different collaboration scenario is also possible. The controller can retrieve information about IP networks and change the attributes of pooled resources in MPLS-TP networks accordingly. For example, when a user demands that its IP traffic from address "100.100.100.0/24" of premise A to address "100.100.200.0/24" of premise B (with already assigned VLAN ID "50") is sent over carrier networks with guaranteed bandwidth of 10 Gbps, the controller changes the assignment of the pseudo wire from VLAN ID "10" to "50."

The first scenario mentioned above is focused in this study as a way to achieve service agility under the assumption that configuring an IP device is less time-consuming than configuring an MPLS-TP device.

IV. USE CASE AND EFFECTS

A. *Use case*

A use case in which a resource is shared by multiple users in a short interval, namely, data backup, is depicted in Figure 3.

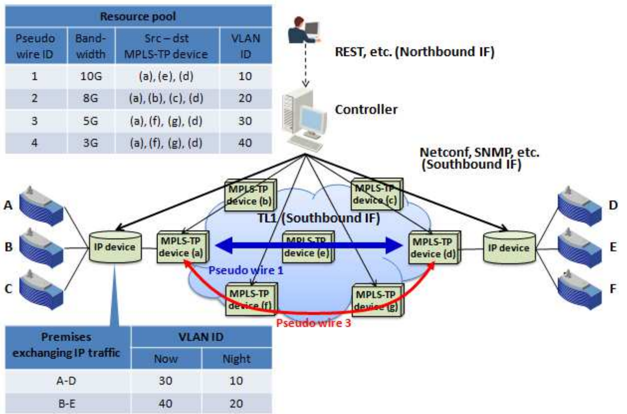

Figure 3. Use case of resource pooling in MPLS-TP networks.

As shown in the figure, IP traffic between premises A and D is already assigned VLAN ID "30," meaning that the IP traffic is transported over the pseudo wire with ID "3" with guaranteed bandwidth of 5 Gbps. In contrast, IP traffic between premises B and E is already assigned VLAN ID "40," meaning that the IP traffic is transported over pseudo wire with ID "4" with guaranteed bandwidth of 3 Gbps.

When a user demands that its IP traffic between premises A and D to be sent over carrier networks with guaranteed bandwidth of 10 Gbps for data backup only at night, the controller acknowledges the demand and assigns VLAN ID tag “10” to this IP traffic at night. Under this configuration, IP traffic between premises A and D is switched from pseudo wire with ID “3,” which has a narrower bandwidth, to pseudo wire with ID “1,” which has a larger bandwidth.

This use case concerns a resource being allocated to a user as a data-backup network in a short lead time. This method achieves flexibility in the face of bursts of traffic.

B. *Expected effects*

The use case described in the previous section is effective in an environment where users do not always use the full potential of guaranteed bandwidth, which is allocated according to the contract between a user and a carrier.

As shown in Figure 4, by enabling the controller to handle a resource pool on MPLS-TP networks and to allocate this resource to IP traffic, resources on MPLS-TP networks will be effectively used by multiple users in a short interval.

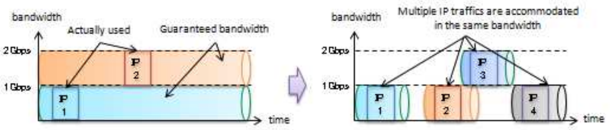

Figure 4. Effects of resource pooling in MPLS-TP networks.

In the left graph, a certain bandwidth is exclusively allocated to one user according to the contract between the user and carrier, and that bandwidth is used by only one type of IP traffic. In this case, bandwidth is underutilized. In contrast, in the right graph, bandwidth is utilized more effectively since it is shared as a resource pool by multiple types of IP traffic in a short interval.

Today, a large amount of lead time is required to change carrier networks. However, thanks to the resource-pooling mechanism and the controller that manages it for IP networks, carrier networks can be flexibly changed in accordance with frequent user demands and volatile traffic patterns resulting from cloud computing. The controller will also reduce the amount of operators’ communications between different layers.

V. PROTOTYPE USING RESOURCE POOLING

A prototype controller, named “Multi-Layer Orchestrator” (MLO), which adopts the resource-pooling mechanism and controls MPLS-TP and IP networks in a collaborative manner, is described as follows.

A. *User interface for configuring MLO*

An overview of the controller, implemented as the MLO, is shown in Figure 5. The MLO provides a REpresentational State Transfer (REST) interface as a northbound interface. Through an application that uses this interface, a user can request a transport network with a guaranteed bandwidth. Manipulating the Graphical User Interface (GUI) of the application, the user can specify a source and destination between which its IP traffic is transported. The user can also request a guaranteed bandwidth. For example, the user can demand that IP traffic from address “100.100.100.0/24” to address “100.100.200.0/24” should be transported in the carrier network with a guaranteed bandwidth of 10 Gbps.

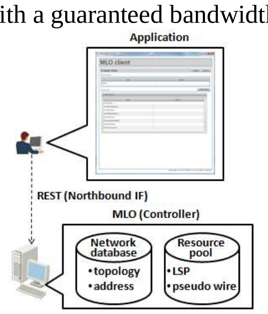

Figure 5. User interface for configuring MLO.

The MLO includes a NetWork DataBase (NWDB) in addition to the resource pool. Topology and address data concerning MPLS-TP and IP networks are held inside this database. When the MLO receives a user demand through the REST interface, it determines which edge IP devices this IP traffic belongs to by referring to the NWDB. It also identifies physical ports of these IP devices through which this IP traffic goes. Then, the MLO identifies MPLS-TP devices that are connected to the physical ports of the edge IP devices. Since the IP traffic will be transported between these MPLS-TP devices, the MLO, finally, searches the resource pool to find the pooled resource that meets the user’s demand.

B. *Data retrieval of MPLS-TP networks through TL1*

For the MLO to hold and manage the resource pool, it needs to obtain the current status of MPLS-TP networks from MPLS-TP devices. Statuses of MPLS-TP networks are retrieved from MPLS-TP devices through Transaction Language 1 (TL1).

If the MLO receives a user demand for guaranteed bandwidth of 10 Gbps for certain IP traffic, it asks the resource pool whether there is a pseudo wire that guarantees a bandwidth of 10 Gbps. If it determines that pseudo wire with ID “1” has the desired bandwidth of 10 Gbps, it searches for the VLAN ID associated with that pseudo wire. If the VLAN ID is “10,” the MLO assigns this VLAN ID to the IP traffic. Then, the MLO configures the IP devices to allocate a VLAN ID to the IP traffic.

C. *Configuration of IP networks through NETCONF*

To assign a VLAN ID to IP traffic, the MLO needs to configure a VLAN ID on IP devices. VLAN ID tags on IP devices are configured through NETCONF [19].

If the MLO concludes that it must assign VLAN ID “10” to a requested IP traffic, it associates VLAN ID tag “10” to the ports of the source and destination IP devices in which the IP traffic is transported. If another VLAN ID is already associated with the IP traffic, the MLO changes the VLAN ID from the previous VLAN ID to “10.”

With this method, the MLO consequently switches the pseudo wire from an old one to a new one over which IP traffic is sent.

 ISBN: 978-1-61208-338-4

VI. EVALUATIONS

To evaluate the feasibility of the MLO in terms of service agility, the configuration time was evaluated as explained below.

A. Evaluation method

The MLO was evaluated in the following environment. Both an application and the MLO were run on a general-purpose computer, whose specification is listed in Table I. The application, through which a user specifies its demand, was implemented with a Java Development Kit (JDK) [20].

The interface, through which the MLO controls MPLS-TP devices, was implemented with JDK. And NETCONF, through which the MLO controls IP devices, was implemented by using the AX – Open Networking – Application Programming Interface (AX-ON-API), which is a Java library [21].

TABLE I. SPECIFICATION OF APPLICATION AND MLO

Specification items	Application	MLO
Operating system	Windows 7 [22]	Ubuntu 12.04 [23]
Processor	2.5 GHz	2.67 GHz
Memory	4 Gbytes	3 Gbytes
Network interface card	1 Gbps	1 Gbps
Runtime environment	Java 7	Java 7
NETCONF implementation	–	AX-ON-API

A testbed composed of an application, the MLO, three MPLS-TP devices, and two IP devices (as shown in Figure 6 with the specification listed in Table II) was constructed. The data plane between MPLS-TP and IP devices was wired by using 10G Ethernet. The control plane between the MLO and all the network devices was wired by using 1G Ethernet.

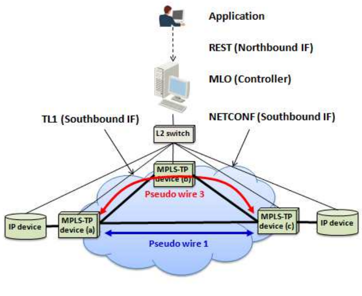

Figure 6. Configuration of testbed.

TABLE II. SPECIFICATION OF MPLS-TP AND IP DEVICES

Type of devices	Product name	Number of devices
MPLS-TP devices	AMN 6400 [24]	3
IP devices	AX 8616 [25]	2

B. Results of evaluation

The time of information retrieval from MPLS-TP devices was evaluated, and the results of the evaluation are listed in Table III. When the MLO retrieved information about a pseudo wire from MPLS-TP devices in order to hold that information as the resource pool, the time needed was 1.5 seconds. In addition, the time to configure the IP devices was evaluated. When the MLO configures the IP devices, the time needed was 13.6 seconds. In total, the time for resource allocation, i.e., the time from the point that the user demanded a network change to the point that a resource of MPLS-TP networks was allocated to the user, was 16.5 seconds.

TABLE III. EVALUATION RESULTS

Evaluation item	Results
Time to retrieve information from MPLS-TP devices	1.5 seconds
Time to configure IP devices	13.6 seconds
MLO's internal processing time	1.4 seconds
Total time to allocate resource to a user	16.5 seconds

On this testbed, the resource pool composed of LSPs and pseudo wires was set in advance. The user then demands that its IP traffic is sent between two MPLS-TP devices with guaranteed bandwidth. After receiving the user demand, the MLO analyzes the demand, searches the resource pool for the demand, and configures IP devices accordingly. Lead time, namely, the time from the point of "user demand," made at "application" in the figure, to the point of "resource allocation," displayed at "application," was evaluated.

The time required for retrieving information from MPLS-TP networks in a carrier network was short. The amount of time is not expected to increase linearly in accordance with the number of MPLS-TP devices since information from each MPLS-TP device is retrieved in parallel. This result indicates that the controller can get timely information from MPLS-TP devices. The resource pool held in the MLO is thus always up-to-date as long as the MLO retrieves information in a short interval.

Currently, it is common to take days or weeks to change the transport technology used in carrier networks. Thus, there is a trend to reduce provisioning time to minutes by placing the service layer on top of the management systems (i.e., controllers) [12]. In this regard, the proposed mechanism aligns with this trend and achieves sufficiently short lead time and service agility.

Consequently, by having a resource-pool mechanism in the MPLS-TP network and making collaboration between MPLS-TP and IP networks, the MPLS-TP network in a carrier network is controlled flexibly. Accordingly, guaranteed bandwidth provided by an LSP and pseudo wire in carrier networks is flexibly allocated to IP traffic according to changes in IP traffic.

VII. CONCLUSIONS AND FUTURE WORK

A growing number of services are provided by cloud computing. The volatile nature of traffic attributed to the behavior of VMs and the increasing number of services provided by cloud computing are, however, negatively affecting carrier networks. To keep up with these trends, carrier networks must be controlled flexibly by SDN.

However, because of a carrier's responsibility to provide high reliability, transport technology in current carrier networks is strictly controlled. Moreover, because carrier networks are multi-layered, communications between operators of different layers are necessary. These issues result in long lead times to change carrier networks.

 ISBN: 978-1-61208-338-4

To control transport technology in carrier networks by SDN, a resource-pooling mechanism on MPLS-TP networks and collaboration between MPLS-TP and IP networks are proposed. More specifically, a controller (named "MLO"), which manages pooled LSPs and pseudo wires and allocates these resources to IP traffic according to user demands, is proposed.

The time to allocate a resource to a user demand was evaluated as 16.5 seconds. From this result, it is concluded that the proposed resource-pooling mechanism on MPLS-TP networks and collaboration between MPLS-TP and IP networks enables SDN in carrier networks in a sense that the proposed mechanism flexibly changes transport technology used in carrier networks. It also reduces communications between operators of different network layers. Consequently, the proposed mechanism can effectively cope with the volatile nature of traffic and thereby achieve service agility. Users utilizing cloud-computing services will thus be able to use network resources in carrier networks on demand.

Future work will be to equip the MLO with a mechanism that makes it work properly with congested networks with few resources, and to equip it with other interfaces of IP devices, such as an "Interface to the Routing System" (I2RS) [26]. By equipping the MLO with these interfaces, it can control not only the VLAN ID tag but also the routing tables of IP devices. This capability will make it possible for the MLO to change routes in layer-3 networks according to resource-pool information of underlying MPLS-TP networks.

ACKNOWLEDGMENT

Part of this research was supported by The Japanese Ministry of Internal Affairs and Communications (MIC) project "Open, Organic, and Optima (O_3)."

REFERENCES

[1] S. Han, M. Hassan, C. Yoon, and E. Huh, "Efficient service recommendation system for cloud computing market," The 2nd International Conference on Interaction Science: Information Technology, Culture and Human, November 2009, pp. 839-845.

[2] D. Nurmi, R. Wolski, and C. Grzegorczyk, "The Eucalyptus Open-source Cloud-computing System," The 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, May 2009, pp. 124-131.

[3] R. Buyya, C. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Generation Computer System, Vol. 25, Issue 6, June 2009, pp. 599–616.

[4] B. Lants, B. Heller, and N. McKeown, "A Network in a Laptop : Rapid Prototyping for Software-Defined Networks," The 9th ACM SIGCOMM Workshop on Hot Topics in Networks, October 2010, pp. 19:1-19:6.

[5] D. Soldani and R. Saracco, "Future carrier networks," IEEE Communications Magazine, July 2013, pp. 24-26.

[6] A. Manzalini, R. Minerva, F. Callegati, W. Cerroni, and A. Campi, "Clouds of Virtual Machines in Edge Networks," IEEE Communications Magazine, July 2013, pp. 63-70.

[7] C. Janz, "Bringing it all together: Multi-layer software-defined networks and automated operations intelligence," http://www.sdnjapan.org/images/2013spdf/0920_05ciane.pdf, SDN Japan 2013, September 2013 [retrieved: March 2014].

[8] ITU-T, "Characteristics of synchronous digital hierarchy (SDH) equipment functional blocks," http://www.itu.int/rec/T-REC-G.783-200603-I/en, October 2006 [retrieved: May 2014].

[9] L. Velasco, et al., "In-Operation Network Planning," IEEE Communications Magazine, January 2014, pp. 52-60.

[10] A. Salehl, "Optical WDM technology for networking and switching applications," Optical Fiber Communication Conference, February 1992, pp. 199.

[11] R. Giladi and E. Menachi, "ETNA's service layer architecture for automatic provisioning of inter-domain Ethernet trasnport services," GLOBECOM Workshops, November 2009, pp. 1-6.

[12] T. Wood, P. Shenoy, K. Ramakrishnan, and J. Merwe, "CloudNet: dynamic pooling of cloud resources by live WAN migration of virtual machines,"7th ACM SIGPLAN/SIGOPS international conference on virtual execution environments, Vol. 46, Issue 7, July 2011, pp. 121-132.

[13] R. Vaishampayan, A. Gumaste, S. Rana, and N. Ghani, "Application Driven Comparison of T-MPLS/MPLS-TP and PBB-TE – Driver Choices for Carrier Ethernet," INFOCOM Workshops, April 2009, pp. 1-6.

[14] B. Niven-Jenkins, D. Brungard, M. Betts, N. Sprecher, and S. Ueno, "Requirements of an MPLS Transport Profile," RFC 5654, September 2009.

[15] S. Bryant and P. Pate, "Pseudo wire emulation edge to edge (PWE3) architecture," RFC 3985, March 2005.

[16] D. Wischik, M. Handley, and M. Bagnulo Braun, "The Resource Pooling Principle," ACM SIGCOMM Computer Communication Review, Vol. 38, Issue 5, October 2008, pp. 47–52.

[17] D. Gmach, J. Rolia, L. Cherkasova, and A. Kemper, "Resource Pool Management: Reactive versus Proactive or Let's be Friends," Computer Networks, Vol.53, no. 17, December 2009, pp. 2905–2922.

[18] R. Martinotti, et al., "Interworking between MPLS-TP and IP/MPLS," http://tools.ietf.org/html/draft-martinotti-mpls-tp-interworking-02, June 2011 [retrieved: March 2014].

[19] IETF, "Network Configuration (netconf)," http://datatracker.ietf.org/wg/netconf/charter/ [retrieved: March 2014].

[20] Oracle Corp., http://www.oracle.com/technetwork/java/copyright-136087.html [retrieved: March 2014].

[21] T. Iijima, H. Kimura, M. Kitani, and Y. Atarashi, "Development of NETCONF-based Network Management Systems in Web Services Framework," IEICE Trans on Communications, Vol. E92-B, no. 4, April 2009, pp. 1104–1111.

[22] Microsoft Corp., http://www.microsoft.com/en-us/legal/intellectualproperty/Trademarks/Usage/Windows.aspx [retrieved: March 2014].

[23] Canonical Ltd., http://www.canonical.com/intellectual-property-rights-policy [retrieved: March 2014].

[24] Hitachi, http://www.hitel.com/solutions/products/optical_transport/6400.html [retrieved: April 2014].

[25] Alaxala, "AX8600R : ALAXALA Networks Corporation," http://www.alaxala.com/en/products/AX8600R/index.html [retrieved: March 2014].

[26] IETF, "Interface to the Routing System (i2rs)," http://datatracker.ietf.org/wg/i2rs/charter/ [retrieved: March 2014].