Consideration of Multi-path Routing over SDN-enabled Network

Sun Uk Baek, Hyun Ho Shin, Dong-Ryoel Shin
Department of Information & Communications Engineering
Sungkyunkwan University, Republic of Korea
E-mail: {sunuk100, shinee, drshin}@skku.edu

ABSTRACT

Since Software-Defined Networking (SDN) technologies have been commonly used in networking components for designing Future Internet, the multi-path routing becomes a fundamental requirement in network deployment. This paper present a prototype of weighted multi-path routing based on OpenFlow protocol. For a stable network operation, in consideration of various factors, it is necessary to select the best path. Bandwidth, load, delay and reliability will be key factors to define the priority of the multi-path.

KEYWORDS

Software-Defined Networking; OpenFlow; multi-path routing; load balancing; weighted routing

1 INTRODUCTION

Software Defined Network (SDN) is a new technology trend of the network. The traditional network equipment vendors have provided all of the hardware, software and management tools. But, SDN is a totally different way to manage the network in a centralized form. It is based on the principle of separating the control and data planes, and the protocol like OpenFlow describes the information exchange between the two planes [1]. Separating control plane makes a wide variety of applications to be applied not dependent on hardware.

In this architecture, an OpenFlow switch contains a flow table consisting of flow entries. A flow entry is made up of fields on which incoming packets are matched, and actions to be applied upon a match. If there is no match, the packet is forwarded to a controller, which runs a program to handle the packet, and decides whether to insert, delete, or update flow entries in the flow table for subsequent packets matching the same fields. Statistics are collected on packet matches which may be used to make decisions.

SDN is applied to the data center, many multi-path routing technologies have been proposed. Multi-path routing was proposed as an alternative to single path routing to fully utilize the multi-path capability of multi-rooted topologies. For example, DevoFlow[2], Hedera[3], Mahout[4], all provide elephant flow management solutions that differentiate between elephant and mice flows.

In addition, when the delay of real-time traffic (e.g. audio and video data) is generated, there is affected by the quality of service. The traffic response time and the error rate of the line will be possible to apply a lot of impact on service. Therefore, the status of link and the port utilization on the path are important factors to improve the quality of service. While maintaining the efficiency of multi-path, it is necessary to introduce a method that can be monitored to reflect these factors.

So in this paper, we propose a method which can be processed stably traffic in consideration of a variety of network variables while maintaining the efficiency of multipath.

2 RELATED WORK

Traditional single path routing algorithms cannot fully utilize the multi-path capability of multi-rooted topologies, such as the fat tree [5], as shown in Figure 1. The spanning tree algorithm [6], for example, always reduces a network topology to a tree. It removes the multi-path efficiency of multi-rooted topologies and causes waste of network resources [7]. In multi-path routing, traffic bound to a destination is split across multiple paths to that destination. Therefore, multi-path routing was proposed as an alternative to single path routing to fully utilize the multi-path capability of multi-rooted topologies.

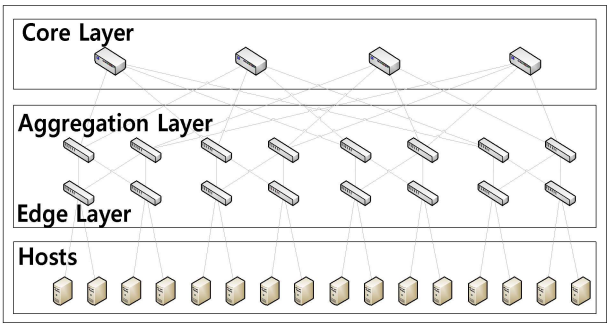

Figure 1. Example of fat tree topology.

Multi-path routing can awfully reduce congestion in hot spots by distributing traffic to unused network links and devices instead of routing all the traffic on a single path. That is, multi-path routing provides load balancing and makes full utilization of network resources [8]. There are many issues in multi-path routing: how to select multiple paths, how many paths are needed, and how to deal with the problem of avoiding shared bottleneck links among multiple paths [8]. The reasons why these issues are very important that the quality of paths selected represents the efficiency of a routing method

3 SYSTEM DESIGN

In general behavior of SDN controllers, if there are multiple paths to send data between hosts, select only one of the best paths to handle the flow. In this case, even though a sufficient margin, using only one path makes inefficient situation. It can be implemented to find multiple paths through various multi-path processing technologies like ECMP [9], MPTCP [10], etc. In addition, the path can be chosen using the port usage, delay, link error rate.

As shown in Figure 2, our mechanism is composed of four modules. Topology module identify the physical link-level connectivity of the OpenFlow switches. Monitoring module monitors the utilization and error rate of each device and link, and it collects and provides the basic information for Path Computation. Path Computation module gives priority of multiple paths in the information collected from the topology module and monitoring module. And Path Install Module serves to apply a prioritized flow table information to OpenFlow switch.

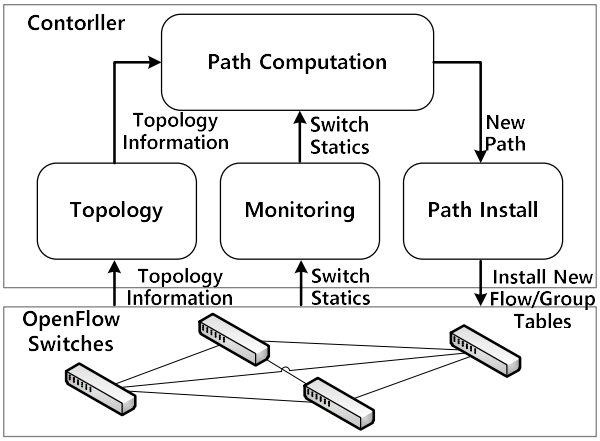

Figure 2. Architecture of the proposed mechanism.

In this architecture, OpenFlow switch uses the flow table that is the path information about packet transmission. When a packet is transferred, the controller, based on the topology information gathered through the topology discovery procedure, and calculates the optimum route, and sets a path off the Flow table in the OpenFlow switch. The communication between the hosts can be set along the path. There is group table specifying

a path, which is able to define different routes for a flow table. Group table can be defined in one of the group to put a large number of paths to the action buckets [11].

Each bucket carries a weight field that defines the share of the traffic processed by the group in the bucket. An example is illustrated in Figure. 3.

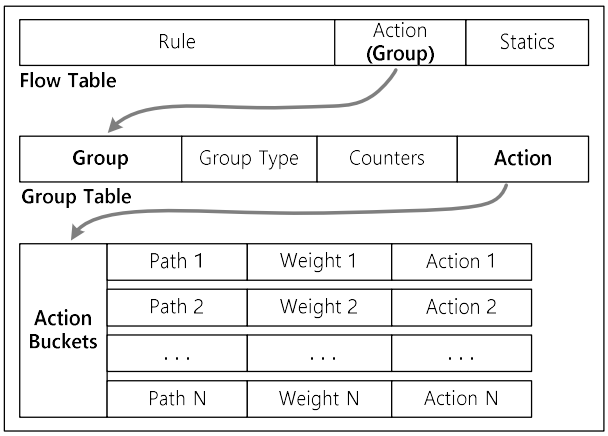

Figure 3. OpenFlow Group Table.

Weight values is existed in the action buckets and the using weight values is possible to adjust the priority. After setting the multi-path list in the action buckets, the monitoring module collects usage and delay of each switch port, and the link error rate. And then the path computation module adjusts the weight value of each path.

3.1 Topology

This module handles generation and parsing/interpreting LLDP packets for all switches in the network. The bulk of the functionality is performed in the following manners. There is a generator function which is called in a timer every timeout period. It iterates over all ports in the network and sends an LLDP packet on each invocation. When receiving and LLDP packet, the module infers the link-level connectivity by querying the LLDP packets. The network links are stored in the instance variable adjacency list.

3.2 Monitoring

The purpose of this module is to query, consolidate and store the statistics from all OpenFlow switches. These information is needed to identify the utilization of multiple paths. It collects regularly per-table, per-flow, per-port statistics, link error rate, and so from each OpenFlow switch. And it saves only the number of recent data to be used to Path Computation.

3.3 Path Install

This module installs the route chosen by the Path Computation module into the series of OpenFlow switches. Each entry in the table contains the information about flow entry attributes like source/destination MAC, source/destination IP, Port and the action installed

3.4 Path Computation

The most important factor in selecting the path, is the ratio of the bandwidth utilization. The lower utilization, need to use the path priority is higher. Furthermore, in consideration of the delay and reliability of the path, it is necessary to select a path. Reliability is a measure that indicates whether the link or port error rate is low. Bandwidth, load, delay and reliability are key factors to define the priority of the multi-path, these values can be obtained from the Monitoring module. Its formula can be expressed as follows.

$$\frac{1}{\left(\frac{Load}{Bandwidth}+Delay\right)\times\left(\frac{1}{1+Reliability}\right)}\times Default\ Priority \quad (1)$$

The formula in (1) is made with reference to the Cisco's EIGRP routing protocol metric calculation formula. The lower traffic amount, the lower delay and the lower link error rate will have a higher priority in the action buckets list.

4 CONCLUTION

Through various experiments, we know that using multi-path, such as ECMP, MPTCP, WMR (Weighted Multi-path Routing) are effective for the use in multi-path environment rather than using only one path, as shown in Figure 4 [12].

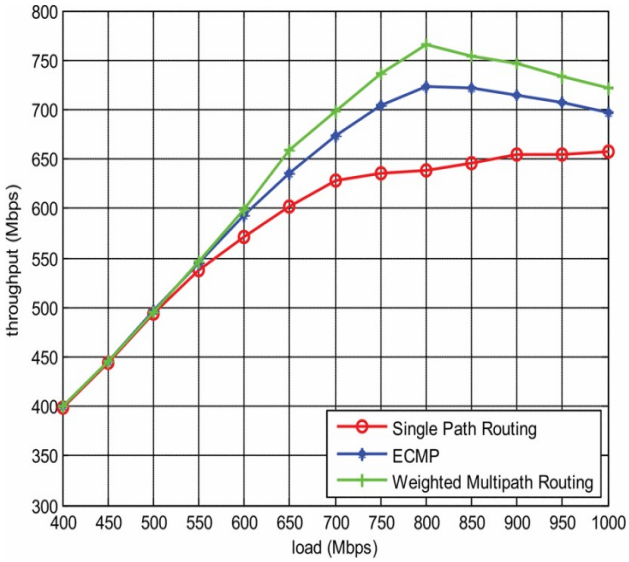

Figure 4. Throughput performance of weighted multipath routing versus single path routing, ECMP

Only the effect of multipath, there is not enough to process a particular traffic, like real-time traffic, depending on the status of various network conditions. To maintain the immediate transmission, or require immediate response to the traffic, it must be considered the bandwidth, load, delay and reliability for each path of the multipath. Consideration of bandwidth, load, delay and reliability makes to operate more stable multi-path network.

5 FUTURE WORK

We are preparing to implement this prototype. In the suggested formula to extend the existing concepts, we plan to proceed an additional experiment what ratio is the better efficiency among bandwidth, load, delay and reliability. For verification, more scenarios with large-scale network topologies will be tested.

6 REFERENCES

[1] Software-Defined Networking: The New Norm for Networks. [Online]. Available: http://www.opennetworking.org/images/stories/downloads/sdn-resources/white-paper/wp-sdn-newnorm.pdf

[2] Mogul, J. C., Tourrilhes, J., Yalagandula, P., Sharma, P., Curtis, A. R., & Banerjee, S. (2010, October). Devoflow: Cost-effective flow management for high performance enterprise networks. In Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks (p. 1). ACM.

[3] Al-Fares, M., Radhakrishnan, S., Raghavan, B., Huang, N., & Vahdat, A. (2010, April). Hedera: Dynamic Flow Scheduling for Data Center Networks. In NSDI (Vol. 10, pp. 19-19).

[4] Curtis, A. R., Kim, W., & Yalagandula, P. (2011, April). Mahout: Low-overhead datacenter traffic management using end-host-based elephant detection. In INFOCOM, 2011 Proceedings IEEE (pp. 1629-1637). IEEE.

[5] Niranjan Mysore, R., Pamboris, A., Farrington, N., Huang, N., Miri, P., Radhakrishnan, S., ... & Vahdat, A. (2009, August). Portland: a scalable fault-tolerant layer 2 data center network fabric. In ACM SIGCOMM Computer Communication Review (Vol. 39, No. 4, pp. 39-50). ACM.

[6] 802. I w - Rapid reconfiguration of spanning tree [online] Available : http://www.ieee802.org/1/pages/802.1w.html.

[7] Fang, S., Yu, Y., Foh, C. H., & Aung, K. M. M. (2012, October). A loss-free multipathing solution for data center network using software-defined networking approach. In APMRC, 2012 Digest (pp. 1-8). IEEE.

[8] Dasgupta, M., & Biswas, G. P. (2012). Design of multi-path data routing algorithm based on network reliability. Computers & Electrical Engineering, 38(6), 1433-1443.

[9] Syrivelis, D., Parisis, G., Trossen, D., Flegkas, P., Sourlas, V., Korakis, T., & Tassiulas, L. (2012, October). Pursuing a software defined information-centric network. In Software Defined Networking (EWSDN), 2012 European Workshop on (pp. 103-108). IEEE.

[10] Barré, S., Bonaventure, O., Raiciu, C., & Handley, M. (2011). Experimenting with multipath TCP. ACM SIGCOMM Computer Communication Review, 41(4), 443-444.

[11] OpenFlow Switch Specification, Version 1.1.0. https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.1.0.pdf.

[12] Liu, J., Li, J., Shou, G., Hu, Y., Guo, Z., & Dai, W. (2014, September). SDN based load balancing mechanism for elephant flow in data center networks. In Wireless Personal Multimedia Communications (WPMC), 2014 International Symposium on (pp. 486-490). IEEE.