A Proposal of OpenFlow Controller to Improve Transfer Rate in Mesh Network

Seawon Koh, Junho Kim, Sungwon Lee*

Department of Computer Science and Engineering
Kyunghee University
Yongin, Republic of Korea
{dream365, junhoya924, drsungwon}@khu.ac.kr

Abstract—**In these days, the Internet is growing and the amount of user to use network increases rapidly. The mesh(ad-hoc) network has studied for many areas where people can't use the Internet. Also, mesh(ad-hoc) network is used for special purposes, and it expands its domain recently. However, low transfer rate and small coverage are a critical issue in mesh(ad-hoc) network. In this paper, it suggests improving transfer rate using BATMAN, Open vSwitch, and Dijkstra algorithm. This paper has three section. In the first section, it suggests network architecture for improving transfer rate. This structure consists of network, controller, and storage. In the second section, it explains the algorithm for finding the optimal path using modified Dijkstra algorithm. Finally, it shows the testbed to realize this proposal. We describe some advantage of controlling traffic flow.**

Keywords—wireless mesh(ad-hoc) network; BATMAN; Open vSwitch; Open Source; Dijkstra algorithm; SDN; SDN controller

I. INTRODUCTION

Due to growing the Internet and penetration rate recently, networks are used by several areas. However, many areas can't supply network service. The wireless mesh(ad-hoc) network, one of the network technology, is getting attention. To solve this problem, the wireless mesh(ad-hoc) network is studied for a difficult environment to use networks, such as war situation or disaster. Recently, the mobile mesh(ad-hoc) network has appeared, which is embedded in a vehicle and called VANET(Vehicular Ad Hoc Network) [1]. Thus, mesh(ad-hoc) network is used for the special purposes like a VANET, so that its usage rate is higher than past usage rate. To transmit packets from one node to another node, mesh(ad-hoc) network doesn't need to have the base network. And, its nodes perform a role as a router to transmit packets from one node to another node. Also, all mesh(ad-hoc) network's nodes have mobility. So, network's status changes steadily, the transfer rate decreases and the packet drop rate increase. In mesh(ad-hoc) network, low transfer rate and small coverage are claimed as a critical issue [2]. This situation gets worse by strong wind and occurring heavy traffic.

In this paper, OpenFlow technology is applied on mesh(ad-hoc) network to improve low transfer rate. It suggests that mesh(ad-hoc) network is managed by the controller on the control plane. Packets move from the path which has low transfer rate to the path having high transfer rate to increase transfer rate.

SDN(Software Defined Networking) is rising in the way of new networking approach [3]. On SDN, control and data plane are separated. So, OpenFlow which is controlled by software is SDN's representative technology. OpenFlow technology provides several network service through the development of new network software module. Also, the user defines the path to move packet by setting flow table of SDN switch with OpenFlow and network controller. And user configures flow by setting several actions.

In this paper, it proposed controller to improve transfer rate using batman-adv. And it makes SDN switch which can receive OpenFlow command by loading Open vSwitch [5, 6]. Then, a path having low transfer rate caused by heavy traffic is controlled by a controller which is combined Dijkstra algorithm. Dijkstra algorithm is the algorithm for finding the shortest path, which is used for OSPF(Open Shortest Path First) protocol.

II. RELATED WORK

A. BATMAN

BATMAN(Better Approach To Mobile Ad-hoc Network) is routing protocol for the wireless mesh(ad-hoc) network. BATMAN knows optimal path's information, and BATMAN sends packets to optimal path. Also, it broadcasts small sized packet called OGM to each node. So, each node knows neighboring node's link information. In OGM packet, it contains the address of the originator, the address of the transmitting the packet, a TTL, and a sequence number.

BATMAN consists of batman-adv, Alfred, batctl, batmand. From among these, batman-adv is an implement of layer 2 protocol in the form of a Linux kernel module operating on layer 2. Other many routing protocols transfer packets by kernel's routing information. In Contrast, batman-adv is formed on layer 2, and it treats data traffic using raw Ethernet frame. Also, batman-adv is configured and debugged through batctl tool. And it contains layer 2 ping, traceroute, and TCP dump.

---

*Corresponding author

B. Open vSwitch

Open vSwitch is virtual software switch based on Linux. It virtualizes the network and controls network resource dynamically. Also Open vSwitch support OpenFlow technology, so that it can be used as SDN switch.
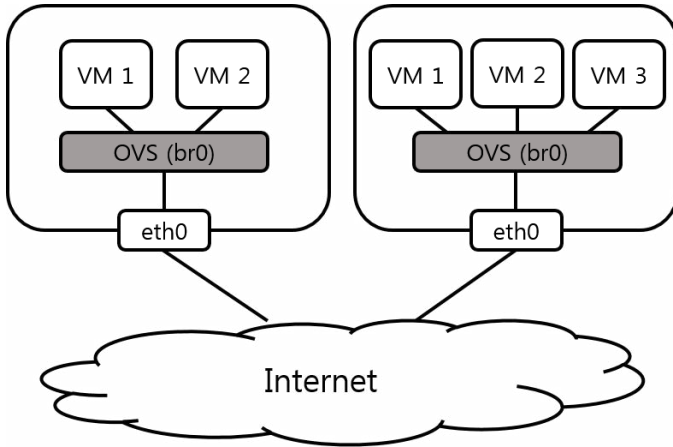

Fig. 1. Open vSwitch's Usage in Hypervisor

Fig. 1 shows Open vSwitch's Usage in the hypervisor. Open vSwitch works as the bridge between outside network and VMs in the hypervisor. It builds bridge and transmitting is available through a link between bridge and Ethernet interface which connects outside network. Also, Open vSwitch provides VLAN isolation, traffic filtering for security.

There are two ways for packet processing. The first is Slow Path. In Slow Path, unknown packet enters from outside network then it makes new flow rule for the unknown packet. The second is Fast Path, In Fast Path, it processes by existing flow rule.

III. PROPOSED SCHEME

In this section, it deals with network architecture for improving transfer rate in mesh(ad-hoc) network and the algorithm to finding the optimal path.

A. Network Architecture

Network architecture consists of three part. The first is a network to be built as mesh(ad-hoc) network. The second is a controller which calculates the optimal path and controls paths. The third is storage which contains transfer rate information about each path and link.
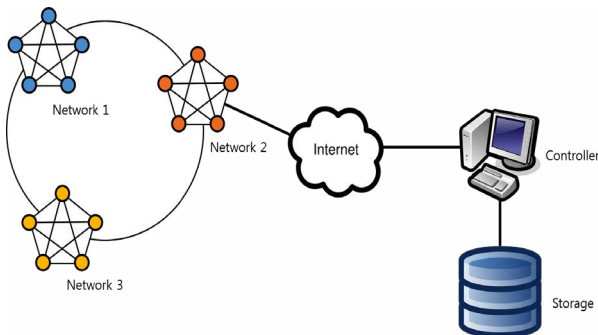

Fig. 2. Proposed Network Architecture

Fig. 2 shows a network to be built using batman-adv which is an implement of layer 2 protocol for mesh(ad-hoc) network. The network consists of three mesh(ad-hoc) network and networks connect each other. Each node loads Open vSwitch then they are used as SDN switch. Therefore, their paths to move packets are controlled by modifying flow table.

Controller checks link's transfer rate between each node steadily. Then if the optimal path which is calculated by modified Dijkstra algorithm is faster than existing path, controller change existing path to the optimal path using OpenFlow command.

Storage saves all link's transfer rate and path's information by controller's order

TABLE I.   DATABASE TABLE

LINK
LINK_ID
TRANSFER_RATE
CHANNEL

PATH
NODE
PATH
TRANSFER_RATE

TABLE I shows tables in storage's database. LINK table contains link's information. Each link connects each node. LINK_ID is each link's identifier. TRANSFER_RATE is each link's transfer rate and it is checked by controller steadily. CHANNEL is the channel of wireless LAN interface. All data in LINK table are elements for finding the optimal path by a controller.

PATH table contains all information about each node's optimal path. NODE is a pair of nodes for the path. PATH is the optimal path which consists of more than one link. TRNSFER_RATE is each path's transfer rate and it is calculated by controller steadily. Using this PATH table, the controller monitors all path's information.

B. Algorithm

Controller finds the optimal path using modified Dijkstra algorithm and path's transfer rate to be checked by controller steadily [4]. Modified Dijkstra algorithm is applied after each link's transfer rate is substituted for the cost which is used for Dijkstra algorithm. Then, if one of the path's transfer rate decreases, the controller finds the optimal path and change existing path to the optimal path.

The algorithm of finding optimal path is presented as below:

Algorithm: Optimal Path Algorithm

Input: Topology, path, transfer rate t[path]

Output: OptimalPath[nodes]

1: for path in Topology

2:   IF t[path] < TransferRateLimit THEN

3:   newpath = dijkstra()

4:   END IF

5:   IF t[path] < t[newpath] THEN

6:   path = newpath

7:   END IF

Line 1 to 7 of the Optimal Path Algorithm shows the entire workflow to finding the optimal path. About all path in topology, algorithm find the path whose transfer rate is lower than transfer rate limitation. Then, it calculates new path using modified Dijkstra algorithm. If the path's transfer rate is higher than existing path's transfer rate, a controller changes existing path to new path using OpenFlow command.

IV. BATMAN-ADV TESTBED WITH OPEN VSWITCH

To modify the path in mesh(ad-hoc) network, it suggests that flow table is modified using Open vSwitch which supports OpenFlow command in this paper. So, we experiment for proving that OpenFlow command can be applied to a mesh network with a small testbed for a mesh network.
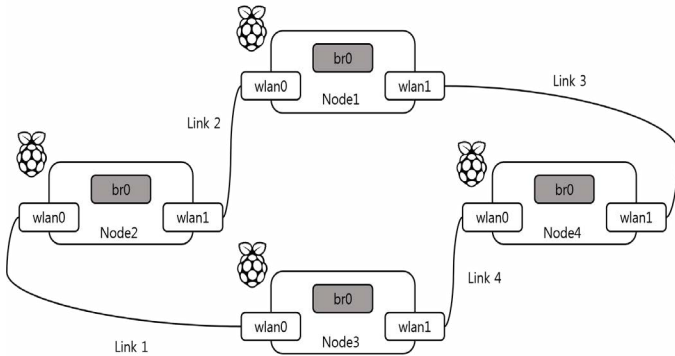

Fig. 3. Mesh(ad-hoc) Network Testbed Applied OpenFlow Technology

Fig. 3 shows testbed for the experiment. Mesh(ad-hoc) network is constructed using raspberry pi 2 and batman-adv. There are four nodes. Each wireless LAN interface loads BATMAN interface. And interfaces are connected by bridge interface. Each link is separated in another network using the different channel. This state, if node 1 sends packets to node 2, packets are transferred via link 2 by using existing flow table.

To apply OpenFlow command, each node is installed Open vSwitch. Then, we proceed experiment. If link 2 has heavy traffic, we change link 2 to another path for improving transfer rate. So, we change each node's flow table by OpenFlow command. As a result, if node 1 sends packets to node 2, packets are transferred via link 3, link 4, and link 1 successfully.

V. CONCLUSION

In this paper, it suggested that the mesh(ad-hoc) network is built using batman-adv and Open vSwitch and find the optimal path to improve transfer rate with the controller. And it controls and manages all of the path and link using controller and storage. As a result, it is predicted that decreasing transfer rate would be solved, which is claimed as a critical issue. Also, we experiment with a small testbed for a mesh network. Then we prove that a mesh network can be controlled by OpenFlow command. Additionally, if it is studied deeper, other problem will be solved such as applying to the large scale network. Also, we will the way to find higher transfer rate and the optimal path rapidly by improving existing algorithm.

In future work, we will develop module and improve the algorithm. And we will proceed the experiment for other issues in mesh(ad-hoc) network.

REFERENCES

[1] Xichen Wang, Weiwei Liu, Lin Yang, Wenli Zhang and Chao Peng, "A New Content-Centric Routing Protocol for Vehicular Ad Hoc Networks," 2016 22nd Aisa-Pacific Conference on Communications (APCC), pp. 552-558, October 2016.

[2] D.G.Anand, H.G.Chandrakanth and M.N.Giriprasad, "Energy Efficient Coverage Problems In Wireless Ad Hoc Sensor Networks," Advanced Computing: An International Journal (ACIJ), vol. 2, no. 2, pp. 42-50, March 2011.

[3] Ramin Khalili, Wint Yi Poe, Zoran Despotovic and Artur Hecker, "Reducing State of OpenFlow Switches in Mobile Core Networks by Flow Rule Aggregation," Computer Communication and Network (ICCCN) 2016 25th International Conference on, pp. 1-9, August 2016.

[4] Liming Wang and Gang Lu, "The Dynamic Sub-Topology Load Balancing Algorithm for Data Center Networks," Information Networking (ICOIN) 2016 International Conference on, pp. 268-273, January 2016.

[5] BATMAN-adv opensource official site [Online], https://www.open-mesh.org.

[6] Open vSwitch opensource official site [Online], http://www.openvswitch.org.