AP LOAD BALANCE BASED HANDOVER IN SOFTWARE DEFINED WiFi SYSTEMS

Nahida Kiran, Yin Changchuan, and Zaid Akram

Beijing Key Laboratory of Network System Architecture and Convergence
Beijing University of Posts and Telecommunications, Beijing 100876, China
kiranbupt@yahoo.com, ccyin@bupt.edu.cn, zaidbupt@yahoo.com

Abstract: In existing wireless systems it's hard for an operator to change the network strategy under heavy load due to its proprietary and inflexible hardware based architectures. This paper explores the concept of software defined networking (SDN) in wireless networks. An access point (AP) load balance based handover scheme in SDN WiFi systems is proposed for overlapping wireless local area network cells to take efficient handover based on AP load. We implemented the proposed load balancing handover scheme in Mininet-WiFi emulator which could handle the APs from a centralized controller and help the test client determine how and when to connect with the right AP and take efficient handover based on AP load. New interface is added with the station, client's traffic is tested at different positions and improved throughput for mobile clients is obtained. Experimental results show that total network throughput is enhanced after a successful handover.

Keywords: Software defined networking; Wireless networks; Mini-net; Mininet-WiFi

1 Introduction

Today's wireless local area networks of IEEE 802.11 standards due to their high capabilities in offering support and throughput for large number of users, are used everywhere in the world to connect various wireless and wired devices to the internet. Mostly, APs (access points) are hugely deployed in such a way that the coverage areas of multiple APs are covered by most of the clients. Due to the increasing number of clients, APs in the hotspot may become overloaded and the throughput of clients connected with such APs drops dramatically because of the non-uniform distribution of WLAN clients. Furthermore, the newly joined clients with lower data rates can severely affect the throughput of the clients that are already associated with the AP. Thus, the most challenging issue in WLANs is how to select the AP to balance load and guarantee high throughput for each client. Hence, load balancing among APs has been the major objective of many proposed AP association schemes [1] in the literature.

In order to address these issues, we can have a central controller having network state information like number of clients associated per AP, traffic load, average data rates, average RSSI (received signal strength indicator) values, operating modes and neighboring APs which can only be realized by the software defined networking paradigm. SDN often referred to as a revolutionary new idea in computer networking, which promises to simplify network control, management and enable innovation through network programmability by separating control plan from data plan. The logically centralized controller has complete overview of the network topology and the underlying physical forwarding devices are doing their job by getting commands from controller.

This paper represents the concept of SDN in wireless networks. SDN implementation in today's wireless networks is a clear indication of the future of networking. The enterprise WiFi networks are moving from stand-alone access points to flexible, scalable and resilient network architecture with a centralized controller and underlying physical forwarding devices. Large number of virtual and physical access points are controlled and managed from a centralized controller. This would give much agility and more efficient manageable architecture.

The paper is organized as follows. Section 2 introduces the related work. Section 3 provides an overview of proposed system model. Section 4 describes the details of implementation of proposed AP load balance based handover in Mininet-WiFi. Section 5 concludes the paper with future work.

2 Related work

In IEEE 802.11 standard, various load balancing schemes have been introduced to resolve the load imbalance problem. Balachandran et al. proposed [2] the idea of associating new user with AP that provides the user minimum required bandwidth. Then associated the user to the AP that has strongest signal strength as compare to other APs.

I. Papanikos et al. and ST.Sheu et al. [3] [4] defined the association of AP based on average RSSI of users that are currently associated to an AP and the number of users that are currently connected to an AP. They also take into account user's RSSI that is intend to associate. These all previous works focus and determine the association of only new arrived users. They don't

978-1-5090-1246-6/16/$31.00 ©2016 IEEE

explain much about the handover of already associated users to the neighboring APs.

To solve the above problems, researchers created SDN [5] at Stanford University and UC Berkeley.

In the SDN architecture [6], the control and data planes are separated, the controller is logically centralized, and the underlying forwarding devices are abstracted from the applications. This decoupling simplifies network configuration and management because network operators can realize complex control logic by designing corresponding programs.

The first successfully deployed programmable SDN-based WLAN architecture is named as OpenRoads [7] which is used for testing various WLAN algorithms. It is a three layer based architecture consist of physical layer, a network virtualization/slicing layer and a control layer.

In [8], Meru proposed the idea of virtual cell. They let APs operate on the same channel and form a large coverage area. In Meru's "One big AP" illusion every client believes that there is only one physical AP in the service area and there is no any other option for the client to select any other AP. But in actual the client is communicating with many other different virtual APs. But Meru's this solution doesn't give more details of triggering handoffs and is proprietary.

The project CloadMAC converts functionalities of WLAN like MAC (medium access control) management functionality to a cloud server based virtual machine. Multiple VAPs (virtual access points) are hosted by these virtual machines and these VAPs are assigned to the physical APs. The paper [9] proposed a technique to move the VAPs from one physical AP to another. This represents the migration of associated mobile users from one AP to other AP. The drawback of this proposed mechanism is that it doesn't support individual handover of mobile devices and it also disconnect some mobile users that are not in the coverage area of second AP.

The programmable enterprise WLAN Odin [10] is proposed where some MAC functionalities are transferred to controller. The controller creates LVAPs (light virtual access points) with unique BSSID (basic service set identifier). The LVAP stores the basic information that is unique BSSID given to mobile device and the mobile device address. The LVAP of physical AP unicast the beacon frames after specific interval to mobile device and communicates the mobile device with the unique BSSID assigned to it. Number of beacon frames of the physical AP depends on the number of mobile devices associated with it. With the increase of mobile devices AP also increases the beacon frames. Due to which collision between frames may occur. This paper also provides a mechanism for seamless handover by removing the LVAP from old AP to new AP but in this mechanism traffic for mobile device can't be re-routed from the old AP to new AP. This may cause interruption and packet loss unless the session is re-established with the server application. This approach is feasible only and handover occurs seamlessly if traffic originates from the mobile device and two APs are in the same subnet.

3 Proposed system model

SDN based system is proposed to solve load balancing and handover problems in WLANs. The architecture of proposed system consist of logically centralized controller, two APs and four stations. The controller has a full view of the whole network topology and is aware of every flow in the network. Some important modules are introduced at controller to maintain necessary information. Controller uses Open-Flow to communicate with the APs and switches and collects all the statistic information (number of stations associated per AP, traffic load on each AP) from the APs and other network switches. Application runs at the top of controller and it will use different interfaces to implement different network services.

There are three important modules inside controller to get the necessary information and to take appropriate action. The first is load control module that is responsible to take load sharing decisions among under-loaded APs and to run the load sharing algorithm in the controller. The second module is responsible to update the MAC filter and to send disassociation message to the overloaded AP to disassociate any of its client. The third module is metric monitoring module which is responsible for collecting the load information of each AP and number of stations associated with it.

The following Figure 1 represents the proposed SDN based WiFi structure having a controller at the center with different functional modules that are connected to OpenFlow enabled APs through OpenFlow channels.

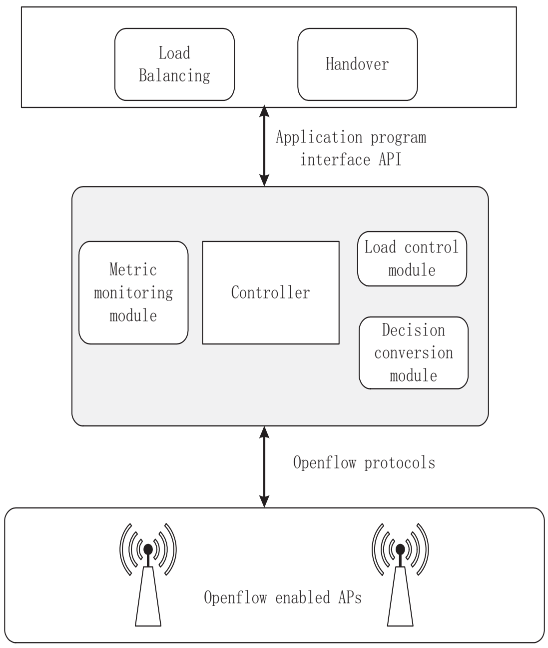

Figure 1 Proposed SDN Based WiFi Architecture

Let's take the WLAN with two APs, AP1 and AP2 with overlapping coverage area. There are four stations sta1, sta2, sta3 connected with AP1 and sta4 with AP2. Suppose the stations are generating static traffic and is equal to 3, 2, 0.7 and 0.3 Mbps approximately. The total traffic for AP1 is 5.7 Mbps and for AP2 is 0.3 Mbps. The controller will compare the received traffic load of AP with the threshold value. If the load is greater than threshold value, it means AP is overloaded and it can't accept further stations. If it's less than threshold value, AP is under-loaded and can accept more stations.

3.1 AP load balance based handover algorithm

Figure 2 below presents the proposed algorithm for AP load balance based handover scheme. Controller is aware of every flow in the network and has full view of the whole network topology. Application runs at the top of controller and it will use different interfaces to implement different network services. When a station comes in the overlapping area, it gets signals from both the APs and broadcast probe request messages. APs get the probe message and stores the stations MAC and exchange this information with controller. If any of the AP gets overloaded then controller will check whether the station of this overloaded AP can be transferred to neighboring AP. If neighboring AP can adjust this station then controller will send a message to AP1 to disconnect the station. This message contains AP's current state, channel of neighboring AP, disassociation notification. AP will send notification to the station through beacon frame. Now the disassociated station connects to AP2 without any handshake messages again because this is the same station whose MAC is stored previously. This will reduce the load of overloaded AP and improve the throughput of the system, no station modification is needed too.

Algorithm 1 AP load balance based handover algorithm

1: Controller gets load information from APs in WLAN systems
2: Controller compares the load of each AP with the threshold value
3: Loop
4: if (APload > Threshold value)
    AP overloaded
    Deny new stations
5: else if (APload < Threshold value)
    AP under-loaded
    Accept new stations
6: When station starts moving to the overlapping region of APs, handoff the station to under-loaded AP
    Check the neighboring AP if it can take the station
    Send disassociation message to old AP to disassociate its station
    Transfer station to new AP
7: else
8: No handoff
9: end if
10: end loop

Figure 2 AP Load Balance Based Handover Algorithm

4 Implementation of AP load balance based handover in Mininet-WiFi

We have conducted the following experiments in Mininet-WiFi to show that the proposed work improves the network throughput in WLANs. The iperf TCP tests are conducted in Mininet-WiFi. The load offered by clients is generated by using TCP client/server tools.

Topology with four stations and two APs are proposed in Mininet-WiFi environment. First of all positions of APs are set in such a way that their coverage areas overlap each other. The ssid of AP1 is set as new-ssid1, channel is set as channel 1, mode is set as g and position is set as (30, 40, and 0). The ssid of AP2 is set as new-ssid2, channel is set as channel 2, mode is set as g and position is set as (80, 40, and 0). Now the stations are added to the topology and each station is given a unique ip address and mac address. Range of stations can also be set as per our requirement. The ip of sta1 is set as "ip=10.0.0.2/8" and range is set as "range= 3". The ip of sta2 is set as "ip=10.0.0.3/8" and range is set as "range= 3". The ip of sta3 is set as "ip=10.0.0.4/8" and range is set as "range= 3". The ip of sta4 is set as "ip=10.0.0.5/8" and range is set as "range= 3". Now controller is added to the topology and this whole topology is under the control of default controller in Mininet-WiFi.

Experiment 1: Purpose of this experiment is to show the throughput of the client when it is at static position and out of the range of both the APs as shown in Figure 3. The test setup involves two OpenFlow enabled APs and four clients. This experiment runs on a total duration of 100 seconds. Initially two clients are connected to AP1 by placing them near to it and AP2 has only one client connected to it. This causes AP1 is highly loaded based on the number of clients associated with it as compare to AP2 because AP2 has only one client connected with it. Test client sta3 is out of coverage area of AP1 and AP2 and is not connected with any AP. So it can't send or receive any traffic to other stations. The results show no route to the host because the client is out of the range of both the APs.

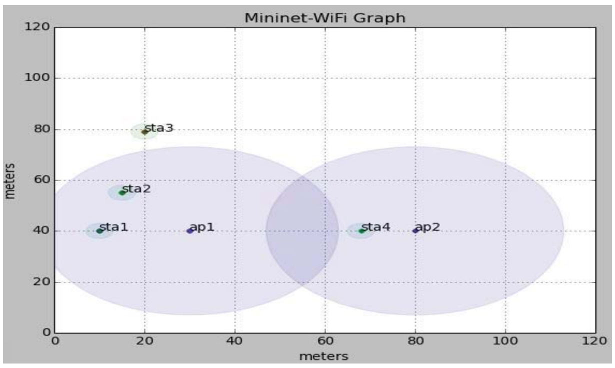

Figure 3 Test Client at Position 1

The throughput result in Figure 4 shows that connection failed and there is no route to host.

Figure 4 Throughput of Test Client at Position 1 in Mininet-WiFi

We use Wireshark to check the throughput graph. The graph in Figure 5 shows a constant line because there is no traffic between the server sta3 and test station sta1.

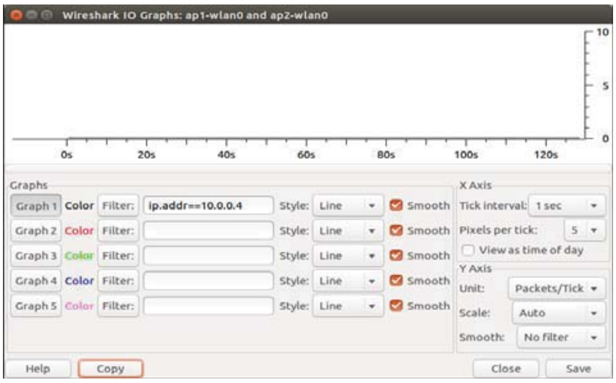

Figure 5 Throughput of Test Client at Position 1

Experiment 2: Now we perform the second experiment by giving mobility to the test client sta3 as shown in Figure 6. Test client is moving to the coverage area of AP1 and get connected with it through interface wlan0. At this point wlan1 interface of sta3 is off because through this interface it can only communicate with AP2. In this experiment throughput is tested for 100 seconds duration between the test client and server that are connected to AP1. The results show that client and server is exchanging data without any throughput drop.

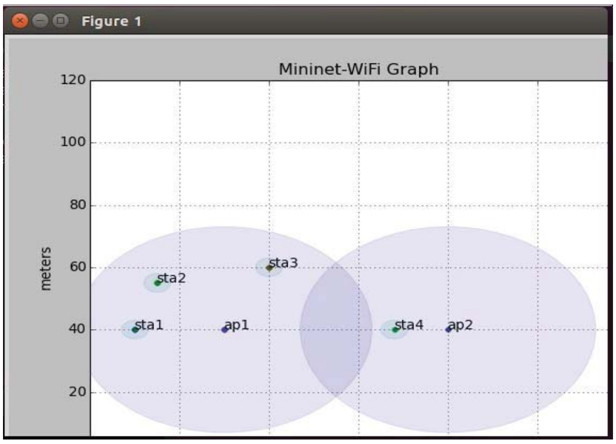

Figure 6 Test Client at Position 2

By taking two stations we can check the throughput in Figure 7. As sta3 is the test station and sta4 is server. The throughput between them is illustrated below.

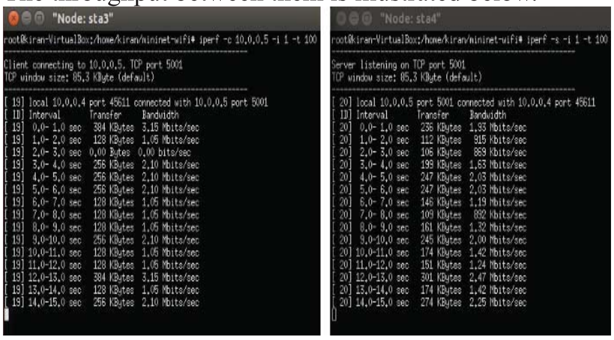

Figure 7 Throughput of Test Client at Position 2 in Mininet-WiFi

Below Figure 8 represents traffic graph between test station sta4 and the server sta1.

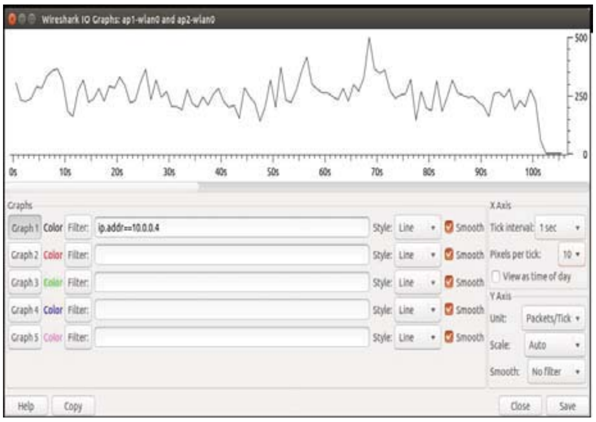

Figure 8 Throughput of Test Client at Position 2

Experiment 3: In this experiment we show the mobility of test client to the overlapping area and throughput results before and after handover as shown in Figure 9. Once sta3 is in overlapping area it is connecting with both the APs through two interfaces. Now test client sta3 uses wlan0 interface to communicate with AP1 and wlan1 to communicate with AP2. The throughput results demonstrate that it is higher after successful handover to AP2 as compared with the throughput before handover.

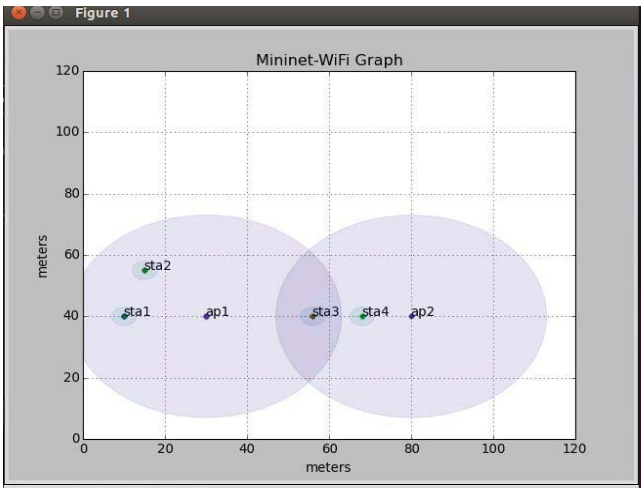

Figure 9 Test Client at Position 3

Figure 10 shows throughput result when test client is going to the overlapping area.

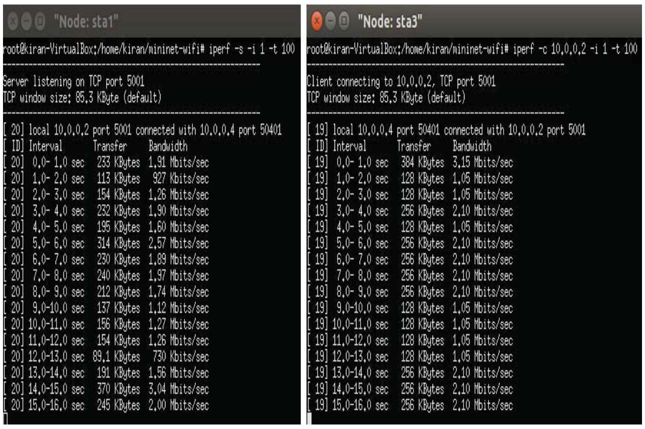

Figure 10 Throughput of Test Client before Handover at Position 3 in Mininet-WiFi

In Figure 11 throughput of test station is tested when it is taking handover from AP1 to AP2. The results show that throughput is higher after handover as compare to before because AP is not overloaded.

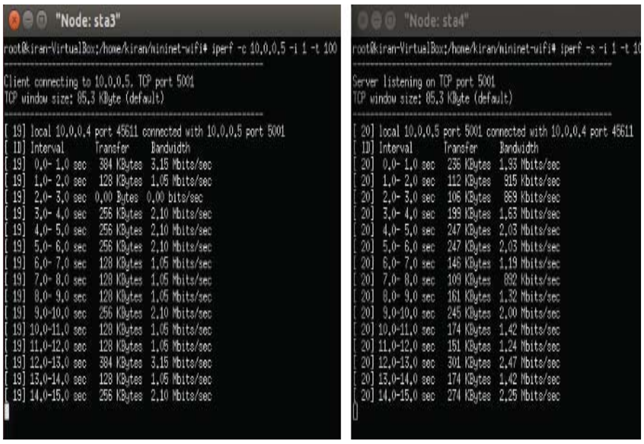

Figure 11 Throughput of Test Client after Handover at Position 3 in Mininet-WiFi

Below graph in Figure 12 shows total throughput result in wireshark.

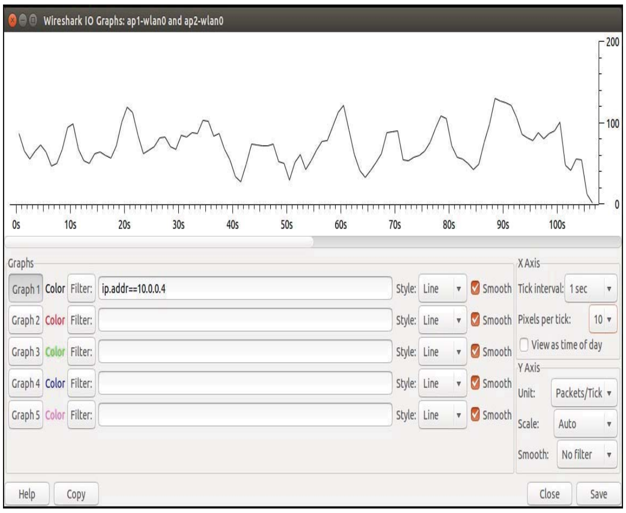

Figure 12 Throughput of Test Client at Position 3

Experiment 4: This experiment shows the efficient handover of sta3 from the overlapping region to the AP2 as shown in Figure 13. During this time sta3 is handed off to the lightly loaded AP2. This time sta3 changes its interface to wlan0. Only one interface works in the coverage area of this AP that is wlan0. This experiment clearly shows that there is no throughput drop at any handoff point in the entire duration of experiment.

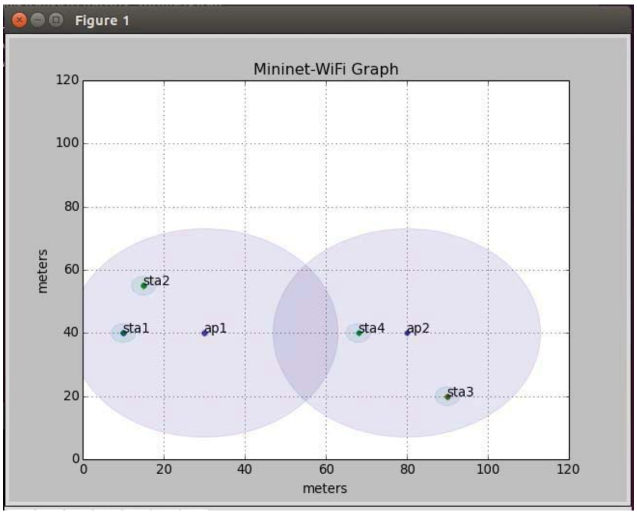

Figure 13 Test Client at Position 4

Figure 14 shows result of throughput when sta3 is connected with AP2 and sta1 connected with AP1.

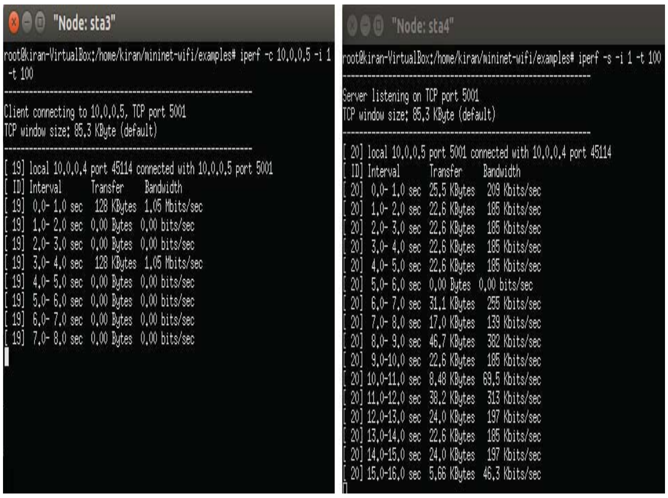

Figure 14 Throughput of Test Client at Position 4 in Mininet-WiFi

This picture in Figure 15 represents the throughput of test station in Wireshark at position 4.

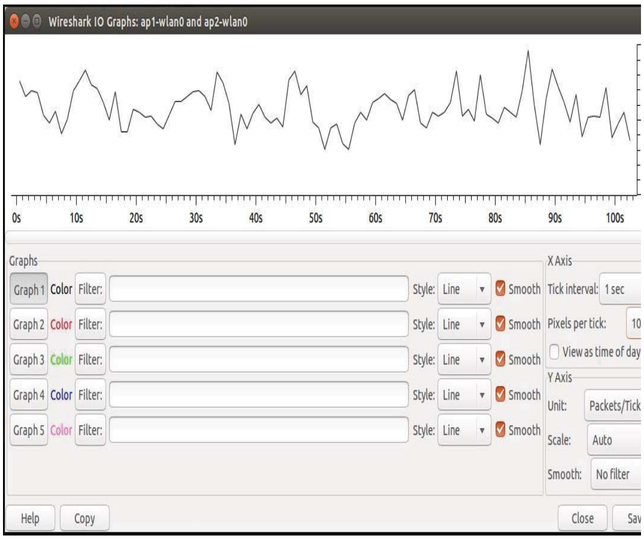

Figure 15 Throughput of Test Client at Position 4

5 Conclusions and future work

A new SDN based WLAN architecture is proposed and some important modules in the controller are introduced to collect necessary information from OpenFlow APs. A novel handover algorithm is proposed in Mininet-wifi emulator with four stations and two APs. The iperf TCP tests are conducted and throughput of the test client is tested at different positions throughout the experiment during the handover from one AP to another AP. Experimental results show that client is taking efficient handover based on AP load where the total network throughput is higher after a successful handover.

To improve the SDN based WiFi architecture, the next future goals are to design efficient handoff and load balancing algorithms and testing those algorithms in controller. To build up a real network and examine the system performance under practical circumstances might improve management and configuration of the enterprise WLAN. This work can be extended to LTE (long term evolution) architecture to further explore the notion of ubiquitous connectivity and network programmability in both WLAN and LTE.

Acknowledgements

This work is supported by the National Basic Research Program of China (973 Program) under Grant 2012CB315801 and NSFC under Grant 61271257.

References

[1] D.Gong and Y.Yang, “AP association in 802.11 n wlans with heterogeneous clients,” in INFOCOM, 2012 Proceedings IEEE, 2012, pp. 1440-1448

[2] A. Balachandran, P. Bahl, and G. M. Voelker, "Hot-spot congestion relief in public-area wireless networks," in Mobile Computing Systems and Applications.

[3] S-T.Sheu and C-C.Wu, "Dynamic Load Balance Algorithm (DLBA) for IEEE 802.11 Wireless LAN," Tamkang Journal of Science and Engineering, vol. 1, pp. 45-52, 1999.

[4] I.Papanikos and M.Logothetis, "A Study on Dynamic Load Balance for IEEE 802.11b Wireless LAN," in 8th International Conference on Advances in Communication and Control (COMCON), 2001.

[5] S. Sezer, S. Scott-Hayward, P. K. Chouhan, B. Fraser, D. Lake, J. Finnegan, N. Viljoen, M. Miller, and N. Rao, "Are we ready for SDN? Implementation challenges for software-defined networks," IEEE Communications Magazine, , vol. 51, pp. 36-43, 2013.

[6] M.-K. Shin, K.-H. Nam, and H.-J. Kim, "Software-defined networking (sdn): A reference architecture and open apis," in 2012 International Conference on ICT Convergence] (ICTC), , 2012, pp. 360-361.

[7] K.-K. Yap, M. Kobayashi, R. Sherwood, T.-Y. Huang, M. Chan, N. Handigol, and N. McKeown, “Openroads: Empowering research in mobile networks,” ACM SIGCOMM Computer Communication Review, vol. 40, no. 1, pp. 125–126, 2010.

[8] Meru, “Virtual Cells: The only scalable multi-channel deployment,” Meru Networks White Paper, Meru, Tech. Rep., Aug 2009. [Online]. Available http://me.westcon.com/documents/36960/Virtual Cell White Paper.pdf.

[9] P. Dely, J. Vestin, A. Kassler, N. Bayer, H. Einsiedler, and C. Peylo, “CloudMAC - An OpenFlow based Architecture for 802.11 MAC Layer Processing in the Cloud,” in Proc. of IEEE Globecom Workshops. 2012, pp. 186–191.

[10] L. Suresh, J. Schulz-Zander, R. Merz, A. Feldmann, and T. Vazao, “Towards programmable enterprise WLANS with Odin,” in Proceedings of ACM HotSDN workshop, 2012, pp. 115–120.