An Empirical Throughput Analysis of Multimedia Applications with OpenFlow-based Dynamic Load Balancing Approach in WLAN

Syed Mushhad Mustuzhar Gilani[1,3], Tang Hong[2], Guofeng Zhao[2], Chuan Xu[2], Muhammad Azeem Abbas[3]

[1] School of Computer Science and Technology,
[2] School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications (CQUPT) Chongqing, China
[3] University Institute of Information Technology, PMAS-Arid Agriculture University Rawalpindi, Pakistan
mushhad@hotmail.com, {tanghong, zhaogf, xuchuan}@cqupt.edu.cn, azeem.abbas@uaar.edu.pk

Abstract

Load balancing among nearby access points (APs) is an inexorable issue in IEEE 802.11 wireless networks. It can cause the throughput degradation, transmission delay, and packet loss that might reduce the quality of service (QoS) and quality of experience (QoE). The present work presents a SDN-based IEEE 802.11 WLAN architecture to take advantage of OpenFlow protocol for dynamic load balancing (OFDLB) among adjacent APs to enhance quality of multimedia applications. The empirical evaluation of the proposed architecture was carried out by throughput analysis of different real application scenarios under multiple test cases. The results reports several benefits achieved by the proposed approach such as good throughput of multimedia applications, 30% reduction in datagram loss and jitter minimization in the OFDLB-WLAN environment. An apparent improvement in network performance by the proposed solution as compared to traditional methods is deemed to be a contribution of the present work.

Keywords: OpenFlow, Load balancing, Software-defined network, Throughput, WLAN

1 Introduction

Over the last few years, there has been a tremendous growth in the wireless Internet and multimedia applications with the increasing number of the Internet users [1]. Recently, Internet users possess smart devices that are accelerating access to the multimedia applications such as audio/video contents, live video streaming, and others TCP/UDP based applications. These applications are based on rich multimedia contents that requires high-speed Internet for their smooth execution [2-3]. Internet users prefers IEEE 802.11 wireless local area network (WLAN) due to its better throughput as compared to mobile Internet [4]. Moreover, upcoming WLAN claims to have several enhancements such as extensive data rate (upto 10Gb/s), rapid access to data, 99% reliability and expected delivery of gigabit link in crowded areas.

In contrast to the benefits of WLAN, an apparent problem of throughput degradation faced by the associated stations is visible in a crowded WLAN network. The reason is, Internet users connect to an available AP that has highest received signal strength indication (RSSI) as depicted in the Figure 1. However, several other APs (AP2 and AP3 refers to Figure 1) with slighter less RSSI are neglected for connection. The traditional IEEE 802.11 standard do not send AP load information during probing process when any station performs association with an AP. This cause a load imbalance in WLAN [5]. The current WLAN design is inherently fixed structured following the vendor specification [6] or the proprietary closed hardware standards [7].

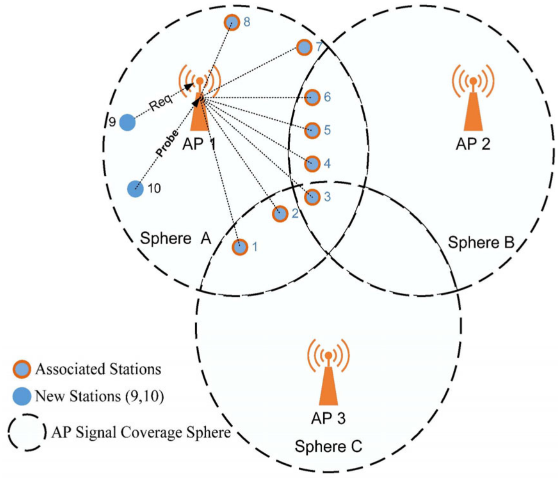


Figure 1. The legacy IEEE 802.11 wireless LAN imbalanced scenario

A load imbalance scenario is shown in Figure 1. For an increased throughput, the stations (STA1, STA2 & STA3) can easily be connected with AP3, while

*Corresponding Author: Syed Mushhad Mustuzhar Gilani; E-mail: mushhad@hotmail.com
DOI: 10.3966/160792642019012001022

stations (STA4, STA5 & STA6) with AP2 respectively. However, AP1 get overloaded due to its strong RSSI resulting a throughput degradation due to imbalance in network load. In an ideal situation, STA1, STA2 and STA3 should associate with AP3 and STA4, STA5 and STA6 should associate with AP2 for an increased throughput. The present work propose a software defined network (SDN) technique for load balancing in IEEE 802.11 WLAN when an overlapping area occurs among multiple APs. A station receives multiple probes for association within an overlapped coverage area [8].

The software-defined network (SDN) [9-10] paradigm is an attractive and most feasible solution for WLAN to enhances the future Internet services and multimedia applications quality of service (QoS) [11]. The SDN introduces the programmable control plane isolated from data plane to enable innovative applications and protocols such as OpenFlow provides a centralized global view of the entire network that formulate the foundations of present proposal for load balance in WLAN [13].

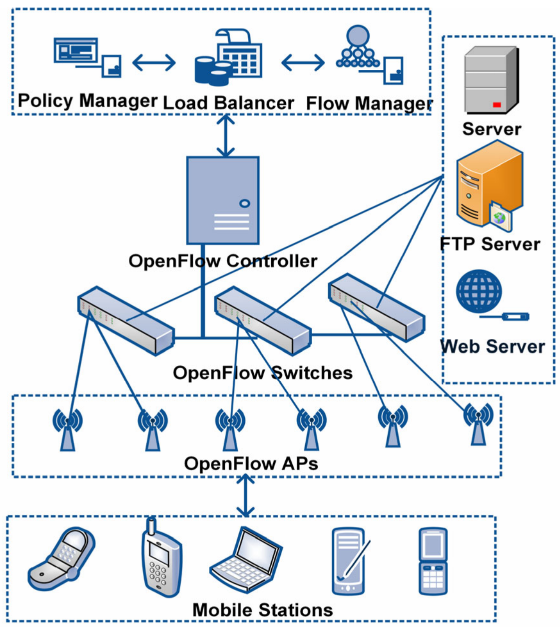
For specifically, the proposal of the present work is an OpenFlow-based dynamic load balancing (OFDLB) approach for WLAN. The proposed technique (OFDLB) includes three main modules: (1) The load balancer module, responsible to collect the APs workload information from the load collector i.e. a component of the OpenFlow AP (OFAP); (2) The flow manager module, responsible for the traffic flow direction or re-direction among adjacent APs and stations (STAs); (3) The policy manager module, it sets rules or policies according to the pre-defined algorithm for the load balancing among adjacent APs. This research work focus on the throughput analysis of TCP, FTP and UDP based applications using OFDLB approach as compared to legacy WLAN environment. The proposed technique is implemented and evaluated the using Mininet-Wifi [14] emulator that provides support for building SDN-based WLAN. The evaluation parameters are jitter amount, packet drop ratio, and the throughput to measure the quality of service (QoS).

The remainder of the paper is planned as follows. A system design detail with the proposed approach is provided in section 2. The implementation procedure is described in section 3. Test cases evaluation and experimental results are presented in section 4. In Section 5, we draw our conclusions.

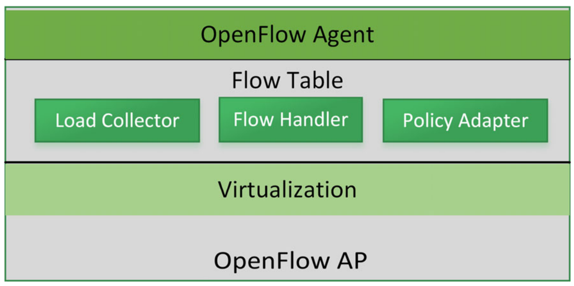
2 OFDLB System Design

The limitations of traditional load balance methodology motivate us to present the OFDLB solution for WLAN by familiarizing the SDN architecture [15]. With the OFDLB, it will be possible to acquire the real-time traffic load balancing of each AP that was previously complicated to achieve in the traditional WLAN.

A system model for load balancing in the OpenFlow wireless network environment as shown in Figure 2 is presented in the present work. The proposed system is composed of three essential parts. The first part contains STAs and servers which represent the terminal devices. The second part is related to the wireless service network which includes OpenFlow-enables devices such as switch and APs to provide the connection to the terminal equipment. The third part is the control plane of the SDN controller which hosts the load-balancing module with the policy manager and the flow manager on the top of the controller as depicted in Figure 2(a). Each individual AP was configured with the OpenFlow protocol that acts as an agent of SDN controller, and is responsible to receive the instructions from the controller and in response delivers the wireless network statistics as per requirement. Figure 2(b) shows OFAP which contains three virtualized entities as an event handler: (a) load collector, (b) flow handler and (c) policy adapter. Each event handler can configure as per installed applications on the top of the controller.



(a) OFDLB system model



(b) Structure of OFAP

Figure 2. OFDLB Environment

The application modules (refers to Figure 2(a)) contain the load balance, policy manager and flow manager. The Load Balancer module is responsible for the real-time traffic load detection to perform the workload comparison among adjacent APs. The load balancer collects the network statistics through load collector which acts as an event handler in OFAP. The load-collector includes the basic configuration parameters: the number of associated STAs, the distance between STA and AP, payload, packet size, channel utilization and pre-defined potential statistics in the load-balancing module. The Policy Manager defines the new policies as per received load information from the load balancer. It sets the rules and forwards to policy adapter for the implementation of the wireless service network**.** The Flow manager is engaged for setting the traffic flow rules and forwards the instructions to the flow handler to collects the flow statistics which includes the transport protocols with the port number, traffic flow rate, and IP address of source and destination, and transmission paths of the associated STAs. Besides, the OpenFlow controller manages a table of each flow transaction record in the wireless service network. Further, a step down procedural details of the OFDLB is as follows:

Step 1: The STA establishes the network access with OFAP which send OF.PACKET_IN message to the controller.

Step 2: The load collector percepts the existing network load information and forwards to the load balancer module.

Step 3: The load balancer detects the current location of STA and calculates the distance between associated AP and others adjacent APs using algorithm 1which executes in the controller.

Step 4: The load balancer module is maintained the real-time traffic load information into a table. It also includes the comparison of adjacent APs through OF.NXST_FLOW message [16]. If detects change then go to step 5 else, go to step 9.

Step 5: The load balancer is classified the obtained information and shares the information with the policy manager to perform an action by step 7.

Step 6: The policy adapter periodically sends the updates to the policy manager of STA signal strength. If the signal strength is becoming weaker, then need to activate the re-association procedure using step 8 else go to step 7.

Step 7: The policy manager matches the current policy as per receives information. If the current associated STA state is imbalanced then selects other AP from the next list of APs then go to step 8, else remains the previous status and executes the step 9.

Step 8: The policy manager defines new rules/policy and communicates with the policy adapter. Further, shares the information with the flow manager to performs step 12.

Step 9: The flow manager sends the AP association information using an encapsulated message in the OF.PACKET_OUT [16] message to OFAP. This information includes service set identifiers (SSIDs), MAC address, and other corresponding address.

Step 10: The flow manager module keeps a record of the previous path information in a table for direct flow control that reduces the computation cost and maintains the seamless flow.

Step 11: In the case of the redirection in traffic flows: Holds the current flow path by sending OF.OFP_ FLOW_MOD [16] message which sets the hard_timeout and OF.OFP_DEFAULT_PRIORITY when it simultaneously matches the two flow paths. (Prefers high transmission path to the lower ones).

Step 12: The flow manager determines the routing track. It generates the corresponding flow entries and forwards to the flow handler for the execution.

Step 13: Delete the previous flow path by sending OF.OFP_FLOW_MOD message that sets the idle_timeout. For updating go to step 4.

Step 14: The flow handler forwards the flow-level control n the wireless service network for the provision of different multimedia applications for delivery on same STA and then goes to step 4.

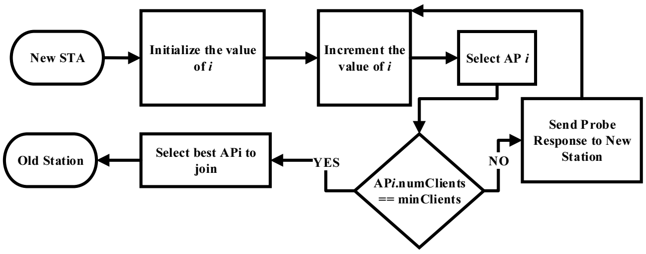


Figure 3. Algorithm for the AP selection

3 Implementation

The empirical throughput analysis was conducted on a computer with 2.30 GHz Intel Core i5-2415 CPU, and 4.0 GB memory, on Ubuntu 14.04. All three OFAPs used in simulation have similar configuration (802.11g, Channel 6, Data Rate 54.0, Range 100m, Pool of Stations 10) with different SSIDs and frequency ranges (2.412 GHz, 2.417 GHz and 2.422 GHz). Iperf [17] and Tcpdump [18] tools are used to collect the network measurements.

3.1 Simulation Setup

The proposed approach has been evaluated over a Mininet-WiFi emulator [14] that provides a virtual platform at large-scale to build the network topologies without inferring cost factors. The modules of the OFDLB shows the implemention in the real network without any modifications. POX controller [19] is selected as the network controller to execute the OpenFlow experiments into the networking devices (switch, AP, and load balancer). Besides, POX is an open source and Python based controller which enables the SDN-based applications with real hardware or Mininet-WiFi emulator to accelerates the development procedure. Our proposed approach modules and applications have developed using the Python to implement in the POX controller. The Algorithm 1 (shown in Table 1) computes the adjacent APs using the pre-defined qualified parameters between associated AP and STA, later it performs a comparison among APs for re-association or maintains the previous AP association.

Table 1. Station association algorithm

Algorithm 1: Procedure of station association with AP.

Input: The set of OpenFlow-enabled APs

Output: MinLoadAP

```
Function userConnecting (userMAC, ap)
    apsInRange
    Station.getStationByMAC (userMAC).getAPsInRange()
    minClients
    dist_newStation=channelParams.getDistance (sta,ap)
    While (APsInRange){
        if dist_newStation>ap.params ['range'] then
            do Association = False => Deny Addlink (ap,sta)
        else
            if ap.numClients() = = minClients then
            do Association = True  => Allow Addlink (ap,sta)
            end if
        end if
    } \\ end while
End Function
```

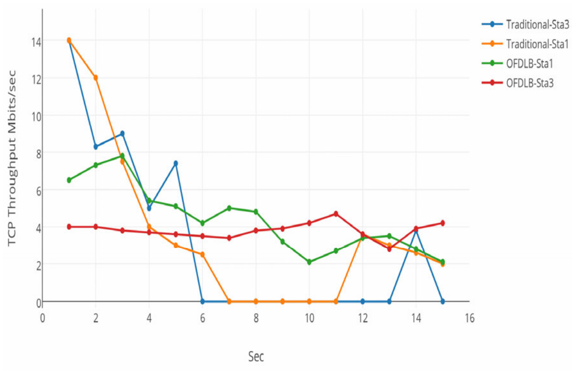
4 Test Cases, Performance Evaluation and Experimental Results

This section evaluates the OFDLB mechanism as compared to the WLAN. For different simulation configurations, a comparison of performance for the load balanced and imbalanced wireless network was performed. The performance comparison was evaluated using the following protocols: TCP, UDP, FTP and HTTP. In the first instance, the aim is to demonstrate the throughput analysis for the uploading and downloading of the TCP, FTP and UDP-based multimedia applications [20]. Further, a video streaming was performed using UDP to analyze the throughput, jitter and datagram loss percentage with the OFDLB approach as compared to the legacy WLAN and DLBA [21] approach. Next, the subsection will divide into the several test cases experiments to analyze the WLAN performance.

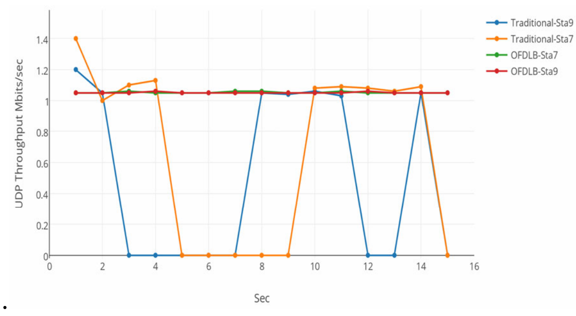
4.1 Case 1: TCP Throughput

Initially, the first experiment setup for TCP performance analysis is based on ten STAs that are associated with three different APs. The deployment of APs in overlapping area provides an analysis of the traffic load balancing. After creating a custom topology, two STAs are taken as HTTP server, and the remaining STAs performed as HTTP client. The centralized controller that configures with a load balancer has defined the policy and rewrites the destination address of incoming packets for forwarding towards less loaded APs. *Ipref* uses to measure the throughput between STAs and server. Moreover, it also permits to perform the various tests that enable the insight view of the current network performance with packet drop ratio, delay, and jitter.

During the first simulation setup, TCP server was configured on Sta2 with port 5566 at default TCP window size 58.3 Kbyte and the TCP clients were set up on Sta1, Sta3, Sta4, Sta5, and Sta6... Sta10, associated with Sta2 for sending the TCP traffic at different transfer rate through the various APs. The first test is performed using the traditional approach in which each station association is based only on RSSI that creates an imbalance environment among APs which effects the bandwidth. The throughput of sta1 and sta3 is degraded and reached to zero Mbits/sec due to their association with overloaded AP (as shown in Figure 4). The same experiment has performed with OFDLB approach, in which STAs association is based on the load of individual OFAP. After adopting the proposed approach, the throughput of sta1 and sta3 is enhanced, the obtained results are illustrated in Figure 4 (a). Meanwhile, the transfer rate and bandwidth of others associated STAs are also improved.



(a) TCP throughput analysis



(b)UDP throughput analysis

Figure 4. Performance analysis

4.2 Case 2: UDP Throughput Analysis

The second experiment is performed using UDP server, which has been configured on the Sta10 with port 5566 and was monitored every second for results. Initially, UDP Clients on Sta1, Sta3……. Sta10 were initiated. Later two stations the Sta7 and Sta9 were analyzed for their performance with a traditional and OFDLB solution.

Figure 4 (b) plots the results of Sta7 and Sta9 that have established a connection with server Sta10 and started data transmission at 1.5 Mbits/sec. Meanwhile, the associated AP received others stations association request due to strong RSSI and connected with them that make overloaded the AP. Due to an unbalanced network, the throughput of Sta7 and Sta9 was gradually decreased and reached to 0 Mbits/sec.

The same experiment is conducted with OFDLB approach to makes the overall balanced WLAN. Figure 4 demonstrates the performance of Sta7 and Sta9 that maintains good average throughput as compared to the traditional method. In this case study, the comparison has performed between the traditional and OFDLB approach concerning congestion level of each station as shown in Table 2.

Table 2. Jitter values during UDP performance analysis

STAs	Jitter (ms)	
Interval (0-15 sec)	Legacy WLAN	OFDLB
Sta1 => Sta10	1.196	0.230
Sta2 => Sta10	2.613	0.182
Sta3 => Sta10	2.087	0.375
Sta4 => Sta10	1.653	0.128
Sta5 => Sta10	1.879	0.448
Sta6 => Sta10	2.275	0.275
Sta7 => Sta10	3.012	0.248
Sta8 => Sta10	0.897	0.190
Sta9 => Sta10	1.673	0.162

4.3 Case 3: Throughput Analysis of FTP Based Applications

For the first FTP-based experimental setup, we consider a legacy WLAN as a baseline. Recall that, in the legacy WLAN, the STAs association is based on RSSI and handover is triggered by the station. During this experiment, Sta1 is randomly selected as the FTP client that activates the file downloading process from IIS web server; the selected file type is audio of size 32.4MB. The file downloading process was accomplished in 50.30 sec with 0.659 Mbps average throughput as illustrated in Figure 5. Further, analyze the uploading throughput, sta1 is selected one audio file for the uploading towards IIS web server. The file uploading process has completed in 96.8 sec with the average upload speed is 0.343 Mbps as shown in Figure 5.

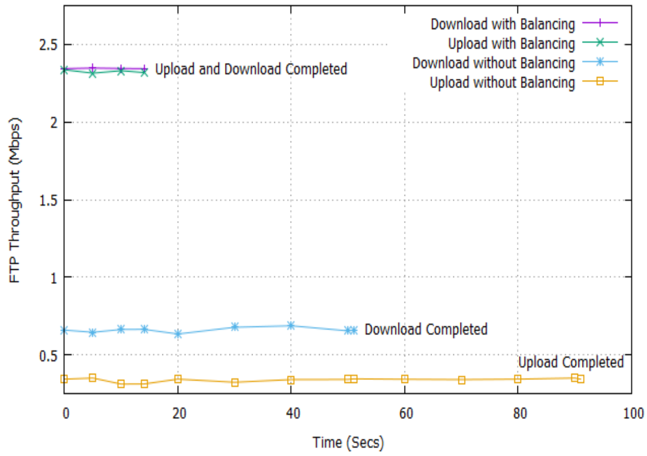


Figure 5. Throughput analysis of FTP uploading and downloading streams between web server and station

For the second experimental setup, the SDN-based WLAN topology has configured with OFDLB implementation. Figure 5 plots that FTP downloading process is acquired for 14.16 sec with 2.34 Mbps average throughput in load balance environment. Similarly, the FTP uploading process has completed in 14.19 sec with 2.33 Mbps average throughput. It shows that the proposed OFDLB enhances the throughput as well as reduces the turnaround time.

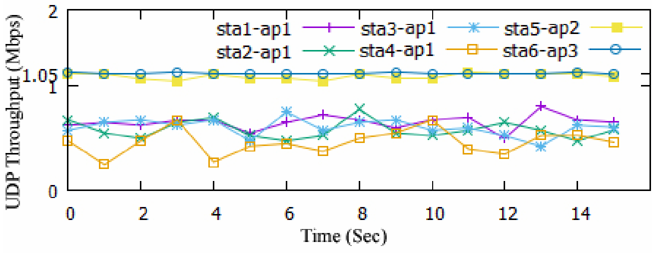
4.4 Case 4: Throughput Analysis of Video Streaming Applications

This case study explores the performance of multimedia applications i.e. video streaming application in the traditional WLAN regarding throughput, jitter or packet loss [2].

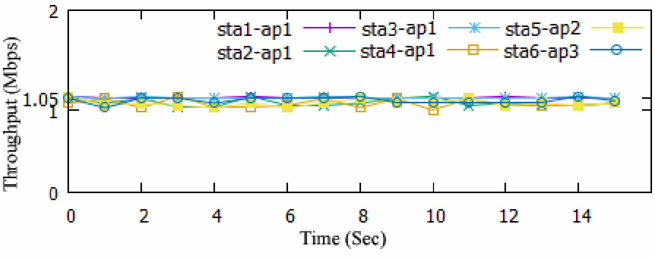
For testing the video streaming scenario with the UDP, we streamed one pre-recorded video file of size 7.27 MB, with Length 3 min 26 sec, and data rate 197 kbps. We configured the UDP-based server at bound port 5001 for the transfer video datagrams. The first video streaming experiment is performed in the legacy WLAN environment, in which four multimedia STAs are connected with AP1 due to the high RSSI value, AP2 and AP3 are associated with only one STA respectively. We selected one STA that is associated with AP1 to analyze the quality of video streaming. In legacy WLAN, the video streaming quality is low and blurred, even some objects were not visible. With OFDLB-WLAN, the video quality has improved, and all objects were clearly displayed.

Figure 6 plots the comparison between legacy WLAN and the proposed approach, from which it can be seen that the UDP throughput improves and sustain bitrate (1 or 1.05 Mbits/sec). That is a major factor for the smooth running of video streaming applications. The figure is the result of 16 sec measurement with the UDP throughput samples taken in every second. As results shown, Sta1 average throughput is increased from 823 Kbits/sec to 1.06 Mbits/sec, Sta2 average throughput is raised from 778 Kbits/sec to 1.02

Mbits/sec, sta3 is also grew from 800 Kbits/sec to 1.05 Mbits/sec.



(a) Legacy WLAN

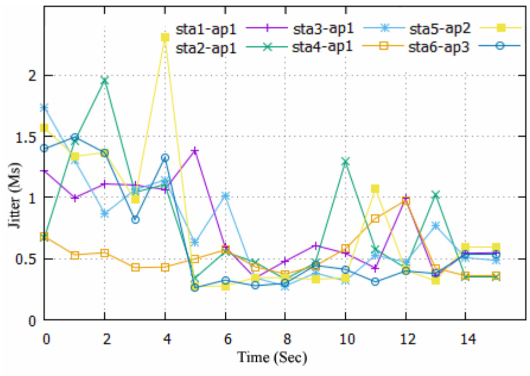


(b) OFDLB-WLAN

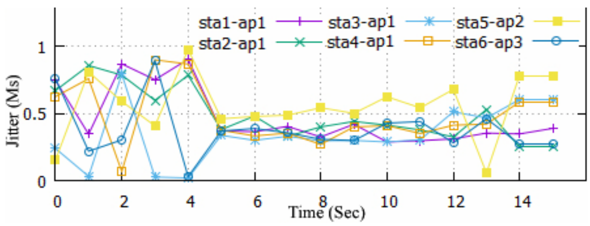
Figure 6. UDP throughput comparison

Figure 7, reports the outcomes of the STAs jitter amount of during video streaming in the legacy WLAN with a comparison to the proposed environment. Note that; the jitter amounts have been taken separately for each associated video streaming STA and followed the random selection method at the beginning of measurements. The increased in jitter values directly degrades the video quality. As it can be seen in Figure 7(a), the jitter rate of sta1 is continuously fluctuated between 0.342ms and 1.384ms, sta2 is reached up to 1.961ms, sta3 jitter values varies from 0.318ms to 1.731ms, sta4 jitter is lower than other stations that above range is less than 1.00ms, while, sta5 jitter rate is passed over the 2.00ms that is the higher jitter value. The sta6 jitter cost is shifting from 0.262ms to 1.494ms. On the other side, Figure 7(b) reports about the jitter amount of all associated STAs that are minimized and sustained at less than 1.00ms.

Figure 8 plots the outcomes of the datagram drop percentage for the video streaming stations. In both experimental setups, the streaming video file size was kept same to take the measurement comparison. During the legacy WLAN experiment, the datagram drop percentage of STAs are higher in comparison of OFDLB-WLAN. As it can be seen, sta1 loss 20% datagram, sta2 loss 24% datagram, sta3 loss 22% datagram and sta4 loss 29% percent datagram that was the higher noted loss percentage. This is because, AP1 workload was exceeded than its normal capacity, in results four videos streaming STAs are associated with AP1 suffered the high loss percentage of datagrams. While the AP2 and AP3 were associated only with sta5

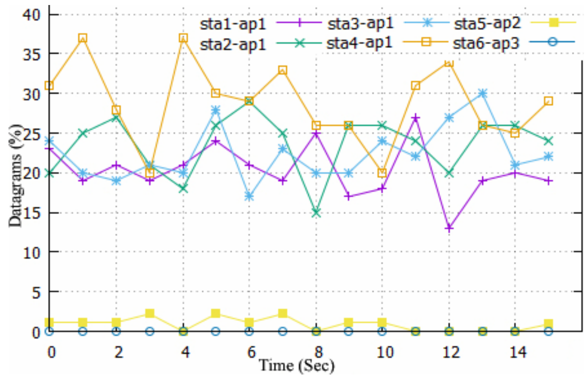


(a) Legacy WLAN

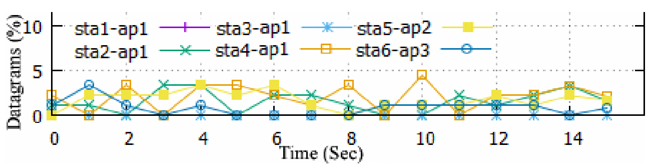


(b)OFDLB-WLAN

Figure 7. Comparison of Jitter cost



(a) Legacy WLAN



(b)OFDLB-WLAN

Figure 8. Datagram drop percentage during video streaming

and sta6 respectively. Therefore, sta5 and sta6 has confronted datagrams drop less than 0.9%. The second experimental setup was identical to the first one, rather than configured POX controller with the implementation of our proposed approach. We ran the simulation setup, in which controller collects information of STAs configurations parameters from

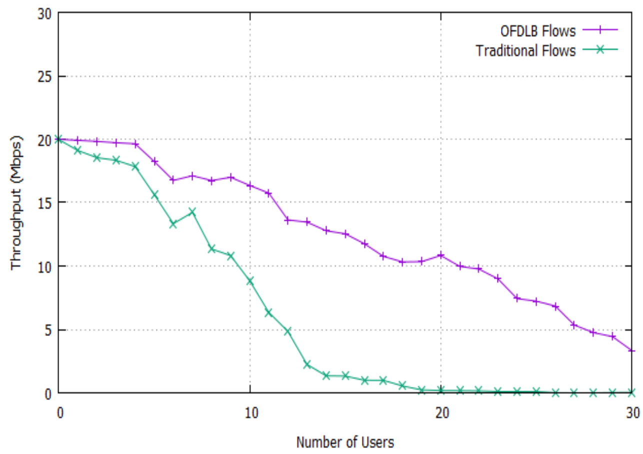
OFAP and evenly distributed the traffic load among adjacent APs. As we see it, in the OFDLB-WLAN, the datagram drop average percentage of associated STAs is less than 2.5 percent.

4.5 Case 5: OFDLB Comparison Test vs. Others Approaches

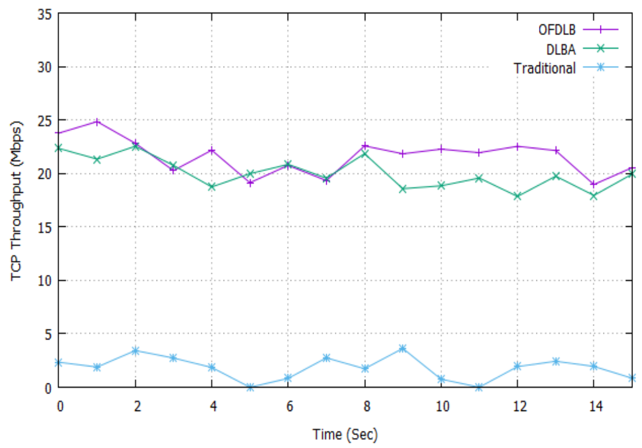
This case experiment elaborates the performance comparison between the OFDLB, traditional WLAN approach, and dynamic load balancing (DLB) approach. We consider an entirely associated wireless network, composed of 30 nodes, arranged in fixed and moveable positions.

The first comparison test is performed between OFDLB and traditional WLAN approach. Initially, throughput flows were started from 20 Mbps, with increasing number of users it gradually reduced, and whenever the number of users limits crossed above 20, the traditional network throughput becomes at zero level. In contrast, OFDLB throughput is considerably better to sustain 30 users without touched the zero level as given in Figure 9(a). The second test is performed to analyze the TCP throughput among OFDLB, dynamic load balancing approach (DLBA) and traditional approaches. Consequently, the TCP throughput is lower in traditional approach than in DLBA and OFDLB as given in Figure 9(b). It is also observed that OFDLB outperforms the DLBA in terms of TCP throughput at a lower cost. The third experiment is executed to examine the UDP throughput comparison among different approaches. Figure 9(c) depicts the obtained results that are shown the performance of OFDLB is slightly higher than DLBA approach and considerably better from the traditional approach.

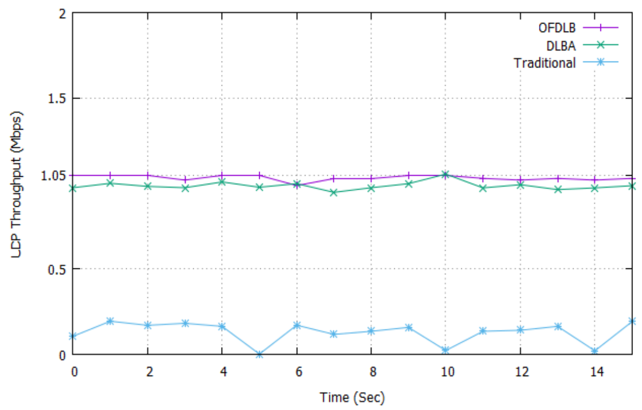
Figure 10 plots the packet delivery cost comparison among the OFDLB, DLBA, and traditional approach. During the first simulation setup, the 30 number of users fixed for measures the packet drop ratio. It is noted that the packet drop is increased with the increment in a number of users as shown in Figure 10(a). Indeed, the OFDLB packet drop percentage is between 0% to 16% while traditional approach led to more packet loss and reached up to 60% packet drop. The second experiment was set up for 14 sec to measure the packet delay among STAs and server. It can see clearly in Figure 10(b), the packet delay is significantly low in the case of OFDLB as compared to DLBA and traditional approaches. Moreover, it is observed that OFDLB has minimum packet drop ratio than other approaches as mentioned in Figure 10(c). One fact is that the traditional WLAN has suffered load imbalanced situation that was raised packet loss up to 15% in 14sec, which is considerably higher than DLBA and OFDLB.



(a) Measure the throughput effect with increasing users

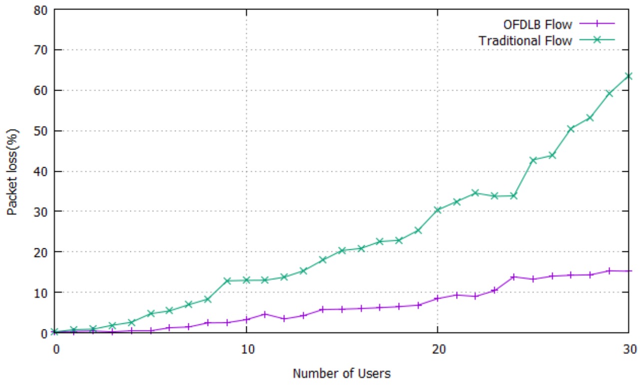


(b) TCP throughput comparison

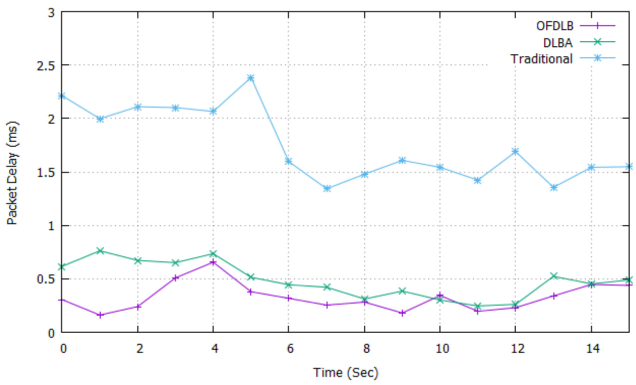


(c) UDP throughput comparison

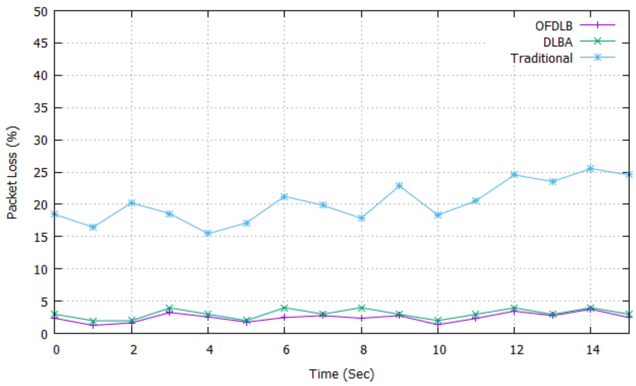
Figure 9. Throughput test



(a) Packet loss rate vs. users



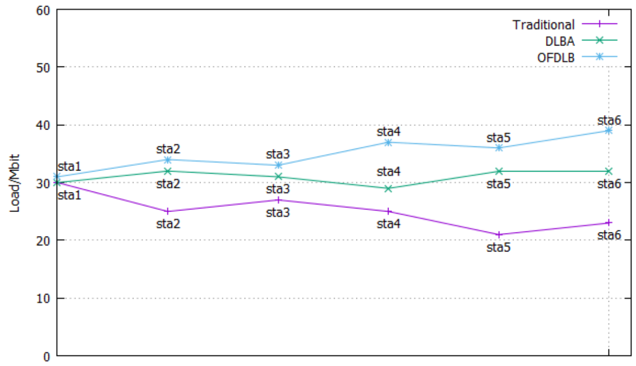
(b) Packet delay



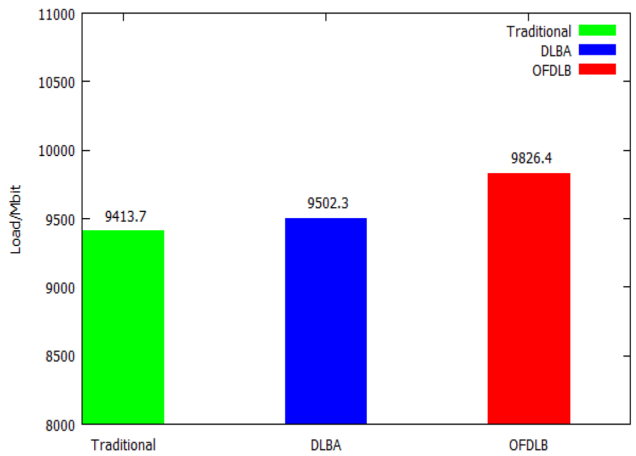
(c) Packet drop percentage comparison

Figure 10. Packet Delivery Cost Comparison

In Figure 11(a), the evaluation has performed for each STA to measures the traffic load with different solutions. It can see clearly in the figure that the average throughput of the traditional approach is 25Mbit/sec, in the case of DLBA approach the average throughput is 30Mbit/sec, while OFDLB average throughput is 35Mbit/s that is higher than the existing solutions. Further, the next experiment was conducted to examine the total throughput of the overall system with each solution as illustrated in Figure 11(b). It is noted that OFDLB throughput is significantly higher than DLBA and traditional approach.



(a) Throughput comparison among STAs



(b) Overall throughput performance

Figure 11. Throughput workload comparison

5 Conclusion

In this paper, the OFDLB approach has introduced to solve the load balancing problem of WLAN. We build an application on SDN controller for decision actions i.e. defined traffic flow rules, set the policies according to the perception of the current network state to achieved load balancing among adjacent APs. To test the proposed approach, an empirical throughput analysis is conducted using multiple test cases under practical application scenarios, which performs using Mininet-Wifi emulator. The collected results ensure the benefit of the OFDLB approach in providing improved throughput of multimedia applications, meanwhile, reduces the 30% datagram loss and minimize the jitter amount in the OFDLB-WLAN environment compared to the legacy WLAN. Therefore, STAs receives a good quality of multimedia applications. The experiment results show that OFDLB approach improves the FTP, TCP and UDP throughput, and reduces the end-to-end delay, datagram drop ratio, and the jitter amount. We shall deploy the proposed architecture with different applications to evaluate the performance of OFAP. It is expected that mix technologies will emerge future network architecture with the various solutions to enables rapid growth of the future Internet.

Acknowledgements

This work was supported by Prospective Research Project on National Nature Science Foundation (61402065, 61501075), Future Networks (BY2013095-2-03) fund by Jiangsu Future Networks Innovation Institute and Project Foundation of Chongqing Municipal Education Committee (No. KJ1500429).

References

[1] CISCO White Paper, Global Mobile Data Traffic Forecast Update, 2016-2021, 2017.

[2] D. Lu, B. Hu, X. Zheng, H. Liu, G. Zhang, Session-Based Cloud Video Delivery Networks in Mobile Internet, *Journal of Internet Technology*, Vol. 18, No. 7, pp. 1561-1571, December, 2017.

[3] Z. Pan, J. Lei, Y. Zhang, X. Sun, S. Kwong, Fast Motion Estimation Based on Content Property for Low-Complexity H.265/HEVC Encoder, *IEEE Transactions on Broadcasting*, Vol. 62, No. 3, pp. 675-684, September, 2016.

[4] V. Jones, H. Sampath, Emerging technologies for WLAN, *IEEE Communications Magazine*, Vol. 53, No. 3, pp. 141-149, March, 2015.

[5] A. J. Nicholson, Y. Chawathe, M. Y. Chen, B. D. Noble, D. Wetherall, Improved Access Point Selection, *Proceedings of the 4th International Conference on Mobile Systems, Applications and Services*, Uppsala, Sweden, 2006, pp. 233-245.

[6] K.-K. Yap, R. Sherwood, M. Kobayashi, T.-Y. Huang, M. Chan, N. Handigol, N. McKeown, G. Parulkar, Blueprint for Introducing Innovation into Wireless Mobile Networks, *Proceedings of the Second ACM SIGCOMM Workshop on Virtualized Infrastructure Systems and Architectures*, New York, NY, 2010, pp. 25-32.

[7] H. Moura, G. V. C. Bessa, M. A. M. Vieira, D. F. Macedo, Ethanol: Software defined networking for 802.11 Wireless Networks, *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, Ottawa, ON, Canada, pp. 388-396, 2015.

[8] L. H. Yen, T. T. Yeh, K. H. Chi, Load Balancing in IEEE 802.11 Networks, *IEEE Internet Computing*, Vol. 13, No. 1, pp. 56-64, January, 2009.

[9] Á. L. V. Caraguay, J. A. P. Fernández, L. J. G. Villalba, An Optimisation Framework for Monitoring of SDN/OpenFlow Networks, *International Journal of Ad Hoc and Ubiquitous Computing*, Vol. 26, No. 4, pp. 263-273, 2017.

[10] D. López, Trends on Virtualisation with Software Defined Networking and Network Function Virtualisation, *IET Networks*, Vol. 4, No. 5, pp. 255-263(8), September, 2015.

[11] S. M. M. Gilani, T. Hong, W. Jin, G. Zhao, H. M. Heang, C. Xu, Mobility Management in IEEE 802.11 WLAN Using SDN/NFV Technologies, *EURASIP Journal on Wireless Communications and Networking*, Vol. No. 1, p. 67, April, 2017.

[12] Home- Open Networking Foundation, https://www.opennetworking.org/.

[13] F. Yang, V. Gondi, J. O. Hallstrom, K. C. Wang, G. Eidson, OpenFlow-based Load Balancing for Wireless Mesh Infrastructure, *2014 IEEE 11th Consumer Communications and Networking Conference (CCNC)*, Las Vegas, NV, pp. 444-449, 2014.

[14] R. R. Fontes, S. Afzal, S. H. B. Brito, M. A. S. Santos, C. E. Rothenberg, Mininet-WiFi: Emulating software-defined wireless networks, *2015 11th International Conference on Network and Service Management (CNSM)*, Barcelona, Spain, 2015, pp. 384-389.

[15] K.-K. Yap, M. Kobayashi, D. Underhill, S. Seetharaman, P. Kazemian, N. McKeown, The Stanford OpenRoads Deployment, *Proceedings of the 4th ACM International Workshop on Experimental Evaluation and Characterization*, New York, NY, 2009, pp. 59-66.

[16] Open Networking Foundation, OpenFlow Switch Specification v 1.3.4, 2014.

[17] iPerf- The Network Bandwidth Measurement Tool, https://iperf.fr/.

[18] "TCPDUMP", http://www.tcpdump.org/.

[19] S. Kaur, J. Singh, N. S. Ghumman, Network Programmability Using POX Controller, *ICCCS International Conference on Communication, Computing & Systems*, Ferozepur, Punjab, India, 2014, pp. 135-138.

[20] Z. Xia, X. Wang, X. Sun, Q. Liu, N. Xiong, Steganalysis of LSB Matching Using Differences between Nonadjacent Pixels, *Multimedia Tools and Applications*, Vol. 75, No. 4, pp. 1947-1962, February, 2016.

[21] M. Collotta, G. Pau, V. M. Salerno, G. Scat, A Distributed Load Balancing Approach for industrial IEEE 802.11 Wireless Networks, *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies Factory Automation (ETFA 2012)*, Krakow, Poland, 2012, pp. 1-7.

Biographies

Syed Mushhad Mustuzhar Gilani recevied his Ph.D. degree from CQUPT, China. He is working as Assistant Professor in University Institute of Information Technology, PMAS-AAUR. His research interests include SDN, wireless networks and future Internet architecture.

Tang Hong is a Professor of computer science at School of Telecommunications and Information Engineering, CQUPT. Her research interests are in future Internet, mobile Internet, network management, etc.

Guofeng Zhao serves as a Professor in the Institute of Information and Network Engineering, and the director of the Research Center of Future Internet (RFI) at CQUPT. His research interests are in future Internet, mobile Internet, network management, network security, etc.

Chuan Xu serves as a Professor in the CQUPT, and Research Center of Future Internet (RFI) at CQUPT. His research interests are in Future Internet, Mobile Internet, Network Management, Network Security, etc.

Muhammad Azeem Abbas has received his Ph.D. from Computer and Information Sciences (CIS) Department, Universiti Teknologi Petronas (UTP), Malaysia. He is working as Assistant Professor in University Institute of Information Technology, PMAS-AAUR.