2016 International Conference on Frontiers of Information Technology

Understanding the Design Space of a Software Defined WiFi Network Testbed

Tausif Zahid, Xiaojun Hei and Wenqing Cheng
School of Electronic Information and Communications
Huazhong University of Science and Technology, Wuhan, China, 430074
Email:tausif@hust.edu.cn, heixj@hust.edu.cn, chengwq@hust.edu.cn

***Abstract*—Due to the simple technical implementation, low-cost network construction and high-bandwidth wireless link capacity, WiFi networks have become popular in the last decade. Nevertheless, WiFi networks have still facing various technical challenges such as load balancing, mobility support and channel adaptation. The emerging software defined networking (SDN) has now been proposed to address the above issues in WiFi networks. SDN provides flexible network management by separating control and data plane, in which some network management entities are able to control the whole network in a unified way. This centralized management provides potential coordinations between access points. In this paper, we study advantages and drawbacks of different approaches for controlling a WiFi network testbed. Our testbed consists of a single controller, multiple access points with up to 24 end-hosts. The evaluation scenarios include single and multiple basic service sets. The performance metric focuses on network throughput with different loads. Our results demonstrate the trade-off between network performance and control flexibility in the current software defined WiFi testbed. Our study may lead to deeper insights into understanding the design space of the software defined WiFi network and the potential directions for the performance optimization.**

***Keywords*-Software Defined WiFi Networks, Performance Evaluation, Throughput, Handover, Mobility**

I. Introduction

Due to rapid proliferation of new services and network devices, IEEE 802.11 wireless local area networks (also known as WiFi networks) have become one of the key network access approaches in the last decade. WiFi services are available everywhere at all the times, but the provision of these services with high-speed connectivity to dozens of users at the same time is challenging. The WiFi access was commonly used as a simple last-hop mechanism. In recent years, we have witnessed increasingly dense deployments of WiFi networks. A large-scale WiFi network consists of multiple basic service sets (BSSs), where a number of wireless stations transmit to or receive from a single access point (AP). The sharing of the WiFi network infrastructure complicates network due to which network performance degradation is one of the core issues. WiFi networking has been migrating from a local and private infrastructure to a large-scale public infrastructure for universities, restaurants, shopping centers and enterprise network etc. It is quite a challenge to provide networking services with full coverage, low delay and minimum interruption. In outdoor environments an AP can cover areas with radius of 200 to 300 meters. In indoor environments the coverage usually shrinks to the radius of around 50 meters. Due to this small cell size frequent handover may occur. Clients who are connected to a far-away AP can only obtain 10 to 50 percent of the network bandwidth as compared to the client which is connected to a nearby AP. It would be immensely beneficial to address this frequent handover problem. Due to different implementations of vendor specific protocol management of traditional WiFi networks, a small change in network policy requires all the devices to be configured, which is very difficult if still possible. Traditional networks cannot accommodate services and applications in accordance to the need of users. Many voice communication based applications need uninterrupted connection without any major delay. Mobility is natural consequence of wireless technology and in a traditional WiFi network, a delay usually occurs due to handover, within the same or different networks when a clients who is already connected with one access point moves from one location to another location. After this handover the client gets better signal strength as compared to the signal strength from the previously connected physical access point and when this signal strength reaches the minimum threshold, the user can then join another network. This handover is one of the serious problems faced by a traditional network. Re-association from one access point to another takes significant time due to the exchange of re-association frames. This handover mechanism is not specified in the 802.11 protocol, though. On the basis of signal strength, the signal-to-noise ratio and other parameters the decision of handover is made by vendor specific protocols. There has emerged a strong need for a unified control of the whole network without significant performance degradation. Traditional WiFi networks can not support unbalanced traffic load, physical channel variation and low-delay handover, in which a WiFi terminal or AP makes decisions of handover on the basis of received signal strength when a user moves from the range of one access point to another.

Software defined networking is a new paradigm which can provide these services with flexible and cheap hardware. With SDN, it is easier to manage the network in a unified way. Recently SDN has been attracting research attention to reshape wireless and cellular networks. SDN separates the intelligence from the network devices through separation of

978-1-5090-5300-1/16 $31.00 © 2016 IEEE
DOI 10.1109/FIT.2016.36

control and data plane. Forwarding decisions are controlled by a centralized controller which is responsible for the whole network. This controller is logically centralized but physically it can be composed of many controlling devices. This centralization provides a full view of network and the whole network can be easily managed. The data plane becomes a simple forwarding element as instructed by the controller. This separation of control and data plane provides opportunity to researcher for experimenting with different protocols and applications without creating any disturbance to production network by slicing the network [1]. The software defined wireless local area network is now under active research and there still are many research challenges such as enhancing the performance.

OpenFlow becomes the de-facto south-bound protocol for deployment of SDN by providing a practical approach for controlling a switch without requiring vendors to expose the codes of their devices [2]. Nevertheless, it has not provided support for 802.11 protocols. Many simulators are available in order to investigate the performance issues but most of them only support software defined wired networks. It is difficult to investigate the performance issues of wireless networks because channel interference and others factors are difficult to replicate in simulated scenarios. Odin [3] is an approach towards a practical deployment of software defined WiFi network. The virtual access points provide management capabilities and virtualize the association mechanism of access points, which authorizes the network administrator to program the network and deploy infrastructure as a network application. The Odin architecture consists of controller and multiple physical access points. The controller and access points communicates through a TCP connection. This deployment doesn't need any client side modification. Due to one-to-one mapping between client and virtual access point there is no need for WiFi clients to re-associate another AP during mobility. The application which is running on the controller will migrate a virtual access point from one physical access point to another and the client can move freely within the network without any noticeable delay of handover.

Fig. 1 shows different scenarios of SDN-based WiFi networks including smart home networks and enterprise networks. These smart devices connect to the Internet by accessing SDN-enabled WiFi APs. These APs can then be controlled by a centralized controller, which provides a global view of network devices. It is possible that APs are no longer required to implement many different WiFi protocols, because the forwarding behaviors can be coordinated by the centralized controllers rather than using customized configurations. By switching from the conventional network platform to programmable entities, SD-based WiFi network offers high degree of flexibility in order to manage APs and to improve the performance of deployed infrastructure. It is also possible to support fast deployment of new services while enabling key features such as virtualization and storage.

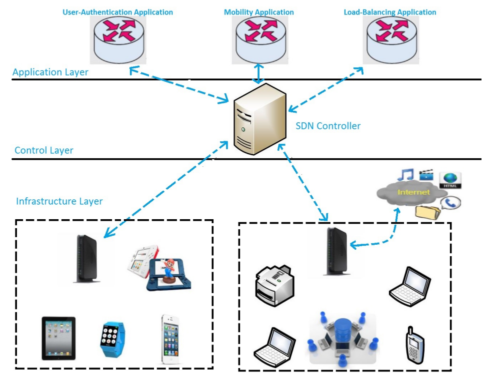

Figure 1: Software defined WiFi networking architecture

In this paper, we have studied different approaches of controlling a WiFi network testbed. We instrumented a software defined WiFi testbed based on the Odin infrastructure with necessary update and customization. Our testbed consists of single controller, multiple access points with up to 24 end clients. The evaluation scenarios include single and multiple basic service sets. The performance metric focuses on network throughput with various different loads. Our results demonstrate the trade-off between the network performance and control flexibility in the current software defined WiFi testbed.

Table I: Specifications of WNDR3700v4

Number	Specification Type	Detail
1	Version	v4
2	SoC	Atheros AR9344
3	CPU (MHz)	560
4	Flash(MB)	128 NAND
5	RAM (MB)	128
6	WLAN Hardware	AR9582[an 2x2:2] + AR9344[bgn 2x2:2]
7	WLAN2.4	b/g/n
7	WLAN5.0	a/n
7	Gigabit Ports	5
7	USB Ports	1*2.0

The remaining of this paper is organized as follows: In Section II, we report the settings of our experimental testbed. Section III describes the measurement results. In Section IV, we summarizes the related work on software defined wireless networking. Finally, we conclude this paper in Section V.

II. Testbed

This testbed consists of three parts, including different types of switches, clients and an SDN controller in order to investigate the network throughput, delay and handover time. Without explicit specifications, we applied the following

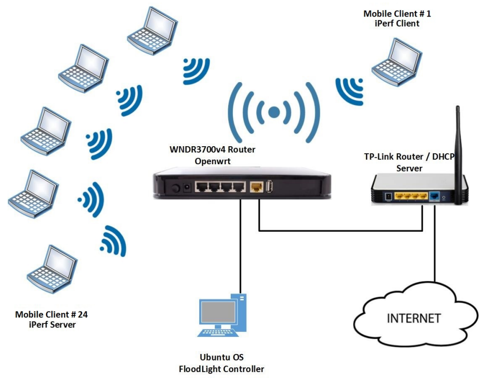

Figure 2: Network topology for measuring the maximum achieved throughput

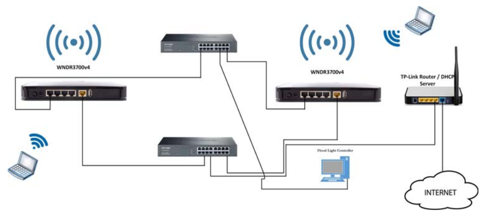

Figure 3: Network topology for measuring the handover time

default testbed settings. A PC with core i3 processor and 4 GB of RAM is utilized as the OpenFlow controller. The NetGear switch 3700v4 with Atheros AR8327, 560 MHz CPU and 128 Mbit RAM is used for different implementations of the access points in this measurement study. Further details are provided in Table I. The OpenWRT version 15.05 chaos calmer is used for the kernel image of the switch. The packages used include the Openvswitch version 2.3, the ath9k Linux driver, and the user-level click modular router. TP-Link TL-SG1024DT switches are used in order to provide the Internet access and the SSH utility. In this paper, the 802.11 network is operated in the 2.4 GHz range.

III. Results

The main objective of this testbed is to enable programmability in wireless local area network without any client-side modification. Interference, distance and other parameters have negative impact on the network performance. We are interested in understanding the performance of different architectures for the home network in the same environment. Our testbed utilizes the same hardware but different implementations including the off-the-shelf NetGear access point with the default settings, OpenWRT and an Odin implementation as a software defined WiFi network testbed.

Fig. 2 shows the topology for performance evaluation of network throughput. This evaluation is conducted under following operational settings. The first scenario consists of a single access point, having different number of clients connected with a wireless connection with access point.

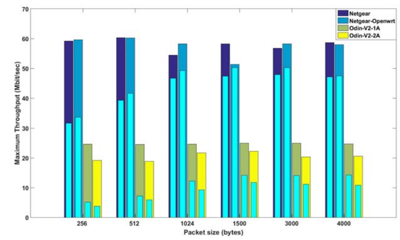

Figure 4: Maximum network throughput evaluation for one client

One host works as an iperf client and sends the traffic using the iperf traffic generator towards another host which behaves as the iperf server. The traffic generated is the total network traffic of this network topology as these two clients are the only hosts in this network. Subsequently, we will increase the number of clients with the same topology up to 24 clients. In case of more than one clients, one host becomes receiver or the iperf server and all the other hosts will generate the traffic or behave as iperf clients. The first scenario shows the comparison of the maximum network throughput for UDP and TCP flows with respect to packet size. The outer bar is the throughput of UDP flow and the inner one is the TCP flow performance. This figure shows the comparison of the same router with different implementations. The first topology results show the performance of traditional router with a customized software running on it. The next performance is for the same router but with OpenWRT based operating system. The third and fourth comparison of throughput consists of software defined approach. The difference between Odin-V2-1A and Odin-V2-2A is the number of applications running on controller side. In this software defined approach we can run multiple applications like load-balancing, authentication, mobility etc on controller which can generate multiple instances on a single or different routers. We also make a comparison to see the effect on overall network throughput by generating multiple slices. Fig. 4 shows the performance of one client, which shows that in most of the cases the performance of OpenWRT-based router and NetGear router is almost same. As compared to UDP flows, the performance of TCP flows degrade due to the acknowledgments between the sender and the receiver. For small and large packet size UDP flow throughput remains almost the same but in case of TCP flow the throughput increases as the size of packet increases. The performance of Odin-V2 is almost half as compared to traditional network and also remains almost constant for UDP based flows.

Fig. 5 and 6 comprise of the performance of 6 and 12 clients respectively. In this case we want to investigate the performance of network with respect to packet size with an increasing number of clients. These two figures show

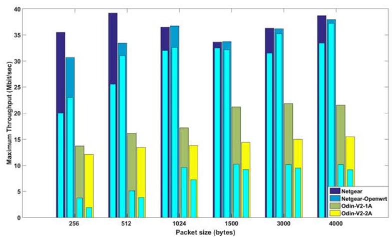

Figure 5: Maximum network throughput evaluation for six clients

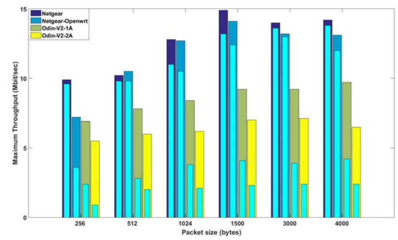

Figure 7: Maximum network throughput evaluation for twenty four clients

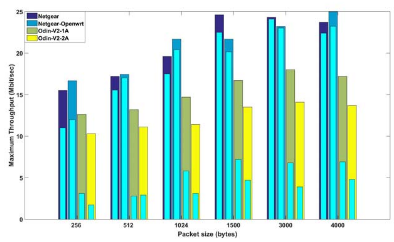

Figure 6: Maximum network throughput evaluation for twelve clients

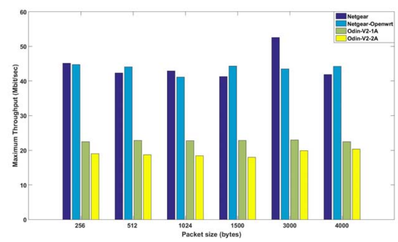

Figure 8: Maximum network throughput evaluation for three active and three passive clients

the dropping trend of performance for UDP based packets as compared to TCP flows. As we increase the packet size and number of clients, the margin of performance of TCP flows get closer to UDP flows as compared to single client performance in the traditional network. As we increase the number of clients, the ratio of performance degradation of the traditional network is much higher than software defined approach.

One access point can handle around 25 clients but this number also depends on the architecture of access point. In order to increase the traffic stress on the AP, we utilized around 24 clients. Fig. 7 shows that our SDN testbed is still working by handling around 24 clients. Although the TCP based throughput is less than 5 Mbps and UDP based throughput is less than 10 Mbps but still many multimedia services can work within this range because all 24 devices are sending traffic at the same time. The main reason for this performance difference is the software implementation of switches and high overload of control traffic in SDN network as it generates separate instances of LVAP for each user in the access point. Another reason for degradation of throughput is the implementation of Openvswitch and the Click modular router which is running at the user level.

Usually in WiFi networks all of the connected devices don't work in active mode all the time. Some of the devices still remain connected as a passive client. Fig. 8 and Fig. 9 shows the performance of 6 and 24 clients in which half of the clients remains connected with the network as passive client. In these experiments we want to investigate the effect of passive clients on performance. These experiment shows the dropping trend of around 6 to 10 Mbps while the devices are connected in passive mode.

Fig. 3 shows the topology for handover. In this experiment, a client was moved from one access point to another in order to investigate the handover latency. Initially the client is connected to one access point then the client moves towards another access point. A laptop is used as a mobile client for these experiments. Dynamic IP is assigned to the mobile client from the TP-link router as shown in figure. For displaying and maintaining connectivity negligible periodic ICMP messages were sent from one client to another. Fig. 10 shows the throughput and the re-association delay during handover. Usually the handover decisions are made based on the received signal strength from APs. When a client reaches to the minimum threshold then the client re-associates itself towards another access point on the basis of receiving a higher signal strength. This handover mechanism is not defined in 802.11 protocol so it depends on the client architecture. The handover delay time also variates with respect to client architecture. The other reason for variation of delay time is, either handover is due to the access point or client initiates this step. In our scenario the handover is due to the mobility of client. The mobile client moves from one access point towards another around 25 seconds. Here we want to investigate the performance of different implementations during handover. Our results show that although the software defined approach is giving around fifty percent throughput as compared to traditional network but during handover the fluctuation in throughput is negligible. The traditional network takes more than 3

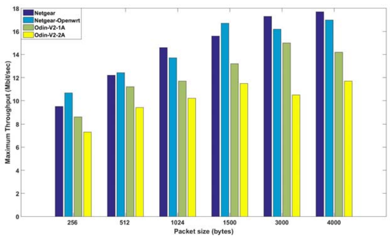

Figure 9: Maximum network throughput evaluation for twelve active and twelve passive clients

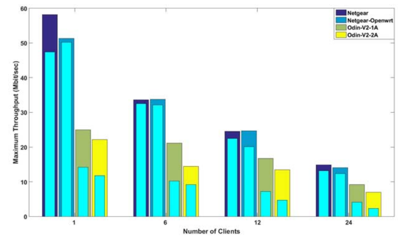

Figure 11: Network throughput evaluation for 1500 packet size

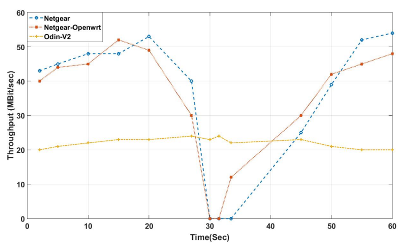

Figure 10: Impact on throughput during handover

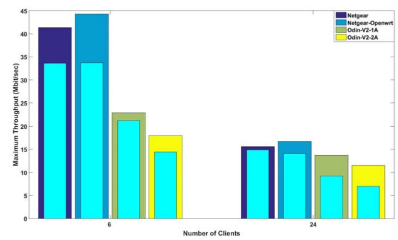

Figure 12: Impact on throughput with clients active and passive mode

seconds in order to re-associates with network and also takes more time to achieve throughput on which it was working before handover.

Fig. 11 shows the dropping trend of Performance of UDP and TCP flows with respect to constant 1500 Bytes packet size. The major performance drop for traditional network can be noticed when clients were increased from one to 6. With a large number of clients, the performance of Odin-V2 with UDP flow is less than 10 Mbps and with TCP flow it is less than 5 Mbps which makes it questionable for large deployment.

Fig. 12 shows the combination of clients with active and passive mode. In these experiments we want to investigate the performance with active and passive clients because usually all of the clients don't send or receive traffic at the same time. So we investigate the performance with half of the devices in active and passive mode. Fig. 12 shows the combination of 6 and 24 clients. In first section three clients are in active mode and three are passive in passive mode. The second part comprises of the combination with 12 active and 12 passive clients. Then we compare both scenarios with 6 and 24 number of clients. The performance in Fig. 12 with outer bar is the performance of 6 and 24 clients. The inner bar is the performance with fifty percent active and fifty percent passive clients. These experiments shows the impact on throughput due to the connection between the client and access point. These results shows than just in order to maintain a connection, traditional network shows more impact on performance degradation with respect to Odin-V2 in case of a smaller number of clients.

IV. Related Work

With rapid deployment and complexity of wireless technology management of wireless infrastructure has become challenging. Traditional networks cannot accommodate services and applications with respect to need of users. Several research efforts were made in order to deal with handover mechanism. These efforts were made by improvement in algorithms, employment of different mechanisms to use dual wireless network cards or in sense of different measurements of received signal strength. The main disadvantage of these schemes are that they require modification on client side. In [4] authors claims the performance improvement of densely deployed infrastructure by using a centralized controller in order to control client associations. [5] and [6] propose the increase in performance by controlling the parameters like channel, power and association. In [7] flow based hybrid mechanism is used to increase the performance of SDN based network through retransmission of lost frame by another access point with having a better channel quality.

On the other side, research and prototypes have been built by utilizing SDN for wireless network. Although most of practical implementations are limited to wired environment [8],but several efforts are made on the wireless deployment of SDN. In [9] through slicing of traffic user, can integrate new services and experiments different protocol just like flowvisor [10] in wired network. In this framework user can freely move be-tween cellular and WiFi network. This handover mechanism requires decoupling of service providers and network owners. This kind of roaming between cellular and WiFi infrastructure may also affect the economy and regulatory challenges. [11], [12] and WARP were other efforts towards this scenario which provides programmability of PHY and MAC layer. Most of the efforts mentioned earlier provide complete architecture but there still are some issues which are critical to the overall performance of the network. Latency is one of the core issues which need to be addressed. Another effort for deployment of Software defined network in wireless domain was contributed by the Odin project[3]. This project introduces concept of Light virtual access point like VAP used in CloudMAC. Odin infrastructure provides a unique BSSID to each client due to which a client can easily handover between different access point. Behop [13] is another approach towards SDN WiFi framework. This infrastructure allows users to control channel association and power in different environments.

V. Conclusion

In this paper, we conducted a performance evaluation study between traditional and SDN-based WiFi Network. We increase the packet size, the number of clients from 1 to 24 in order to investigate how different WiFi management strategies affect user experience and network behaviors with high traffic load on a single access point. Then, in order to investigate the handover mechanism of software defined network and traditional network we deploy multiple access points. This real deployment shows the behavior of network infrastructure in which different number of applications are running on the controller side. We were also interesting to study the impact on performance due to the deployment of different applications running on controller in our SDN testbed. This logically centralized nature of network provides many benefits but degradation of throughput in SDN at larger number of clients is still an unresolved issue. Software Defined WiFi Network can be seen as one way to solve many issues of the Internet including mobility, managing complexity, multicast, load balancing. Nevertheless, before the wide spread deployment of the SDN WiFi networks, several research challenges like network performance, resource discovery and security need to be addressed. Our study show that latency is significant to degrade the quality of service for many applications especially when we are considering widespread deployment of software defined wireless network. SDN architectures provide a unified control and healthy combination of different applications but on other side traditional networks are providing better performance. By considering different implementations it is clear that no single architecture can provide for all the needs of users, so the operators today have to deploy specific solutions for specific environments.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No. 61370231), and in part by the Fundamental Research Funds for the Central Universities (No. HUST:2016YXMS303).

References

[1] Y. Yiakoumis, K.-K. Yap, S. Katti, G. Parulkar, and N. McKeown, "Slicing home networks," in *Proceedings of the 2nd ACM SIGCOMM Workshop on Home Networks*, 2011, pp. 1–6.

[2] A. Lara, A. Kolasani, and B. Ramamurthy, "Network innovation using OpenFlow: A survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 493–512, 2014.

[3] Lalith and S. et al., "Towards programmable enterprise WLANs with Odin," in *Proceedings of the first workshop on Hot topics in software defined networks*, 2012.

[4] I. Broustis and K. P. et al., "Measurement-driven guidelines for 802.11 WLAN design," *IEEE/ACM Transactions on Networking*, 2010.

[5] A. Akella, G. Judd, S. Seshan, and P. Steenkiste, "Self-management in chaotic wireless deployments," in *Proceedings of the 11th Annual International Conference on Mobile Computing and Networking*, 2005, pp. 185–199.

[6] R. Murty, J. Padhye, R. Chandra, A. Wolman, and B. Zill, "Designing high performance enterprise wi-fi networks," 2008.

[7] H. Oh, J. Lee, and C. Kim, "A flow-based hybrid mechanism to improve performance in nox and wireless openflow switch networks," in *Vehicular Technology Conference (VTC Fall), 2011 IEEE*, Sept 2011, pp. 1–4.

[8] Jain and S. et al., "B4: Experience with a globally-deployed software defined wan," *SIGCOMM Comput. Commun. Rev.*

[9] K.-K. Yap and K. et al., "OpenRoads: empowering research in mobile networks," *SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 1, Jan. 2010.

[10] Sherwood and R. et al., "Carving research slices out of your production networks with OpenFlow," *SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 1, pp. 129–130, 2010.

[11] Bansal and M. et al., "OpenRadio: a programmable wireless dataplane," in *Proceedings of the first workshop on Hot topics in software defined networks*, 2012.

[12] Dely and V. et al., "CloudMAC: An OpenFlow based architecture for 802.11 MAC layer processing in the cloud," in *IEEE Globecom Workshops (GC Workshops)*, 2012, pp. 186–191.

[13] Yiakoumis and Y. et al., "BeHop: A testbed for dense WiFi networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 18, no. 3, pp. 71–80, Jan 2015.