Performance analysis of emulated software defined wireless network

Tsehay Admassu Assegie, Pramod Sekharan Nair
Department of Computing Technology, College of Engineering and Technology, Aksum University, Ethiopia

Article Info

Article history:

Received Jan 28, 2019
Revised Apr 27, 2019
Accepted May 19, 2019

Keywords:

Mininet-Wi-Fi
Network
Network performance
Software defined wireless
Wireless network emulation

ABSTRACT

Software defined wireless network is a networking architecture and an emerging networking principle that is based on software defined network. A software defined network is a fundamental networking concept which separates the networking devices used in communication network form the program that runs on the top of these devices. In this paper we will explore the modeling tools used for software defined wireless network in our literature survey and based on the survey we have used mininet-Wi-Fi for modeling software defined wireless network, which is the best and most widely used emulation tool for software defined wireless network. Lastly, we have evaluated the performance of TCP traffic bandwidth for different number of emulated end stations and access points.

Corresponding Author:

Tsehay Admassu Assegie,
Department of Computing Technology,
Aksum University,
Aksum, Ethiopia.
Email: tsehayadmassu2006@gmail.com

1. INTRODUCTION

Software defined wireless network is a software based wireless network. Mininet-Wi-Fi is the most widely used emulation tool for software defined wireless network [1]. This tool provides virtual environment which includes the OpenVswitch, the wireless Access Points and emulated end stations which enables the emulation of the software defined Wireless network.

The number of computers and networking devices used in communication network is growing and the need for high speed, cost effective, energy efficient network is in great demand. The management of the networking systems is becoming a challenge and complex task. As networks become larger in size, traditional static networks are no longer reasonable approaches for implementing new networks that overcome these challenges. Users depend on the new technologies and applications to support their day to day activities. As a result, poor network performance has serious impact on the network speed. In order to evaluate networks, emulators are used as a method to evaluate network performance before they are implemented and the actual system is made. A widely accepted method is performance evaluation with emulation.

Emulators are used to create, network topologies, numerical results that characterize the performance of certain network elements and their functions. In emulation, we observe events happening over time, and collect performance measures to draw our conclusions on the performance of the network, such as link utilizations, response times, and routers' buffer size based on the observation results. In our study, we have used network device sizes and the response time to measure the impact of the changes on the network performance, when new network elements or users' are added to the network.

Mininet-Wi-Fi is extensively used in software defined wireless network emulation researches. This tool is used to evaluate network performance, conduct test beds on network protocols and to conductdiverse research issues before the network is implemented in the real world.

2. LITERATURE REVIEW

In this section, previous works, related to software defined wireless network will be reviewed [1-5]. Many researches has been conducted by researchers on the implementation, challenges and future directions on the real implementation of software defined wireless network, but still the software defined wireless network is in its infancy and further researches are required on this emerging virtualized network infrastructure to provide a cost-effective, dynamic wireless connectivity to the network users.

The number of devices connected to a wireless network is increasing tremendously [1] and the need for high speed cost effective, energy efficient more dynamic wireless network is still in higher demand. To provide such network services to the network users, a software defined network is an ideal solution. This is because; this software defined networking concept is based on highly dynamic networking infrastructure, such as dynamic provisioning of network bandwidth.

A mininet-Wi-Fi is the advanced modeling tool used in emulation of software defined wireless network [2]. It extends the functionalities found in mininet SDN emulation tool. In the study bicasting was used to improve network performance of the software defined wireless network.

One of the problems with the current wireless network is the proprietary nature of the program running on the device used in wireless networks [3]. The control logic, that is the program running on the networking devices, in the wireless network, such as access points to control the data forwarding functionality between the end stations cannot be modified and dynamically controlled as it is embedded into the device by manufacturer on such devices. When the devices come with predefined configuration and less flexible management interfaces, still it is difficult to meet the high demand for network bandwidth and dynamic nature of user requirements.

An OpenFlow protocol, which is used in software defined network to enable connectivity between the data forwarding devices, like switch and router is used in software defined wireless network [4] in the same way it is used in wired network technologies. The OpenFlow controller is used provide connectivity between the data forwarding or network infrastructure layer and the upper layer or the control logic or applications, network services layer, by acting as the standard protocol. Devices from different vendors can be used in a software defined wireless network using this open flow standard. The standard also enables network programmability; through changing the switches flow table entries used in making data forwarding decisions and even determine the path or the route to be used to forward the data can be managed using this protocol.

A software defined wireless network platform provides an effective solution to cost effective high speed ,more dynamic network services when compared with the traditional wireless network [5]. And it is emulated usally by using one of the most powerful tools, the mininet-Wi-Fi. mininet-Wi-Fi have important feature of such mobility modeling, propagation modeling, The mobility model, is the feature in wireless network that allows the network users stay connected while they are in motion, for example the user may move from one building to another building in a campus but still stays connected to the network even if he/she is moving. The mininet- Wi-Fi, along with introduction of OpenFlow enabled flexibility in network implementation, allowing the enterprise to manage the network devices without being restricted on the proprietary software embedded in the traditional networking equipment.

3. SOFTWARE DEFINED WIRLESS NETWORK

Software defined wireless network is programmable network. It is the networking platform that provides an effective solution to cost effective high speed network services when compared with the traditional wireless network. The Software Defined wireless Network (SWDN), which is often denoted as a revolutionary new idea in computer networking, has the capability to radically simplify the network control, configuration, and enable revolution through network programmability. Mininet-Wi-Fi facilitates the modeling and manipulation of software defined wireless network components.

3.1. Mininet-Wi-Fi Software Defined Wirless Network Emulation

To emulate the software defined wireless network, we have used mininet-Wi-Fi, emulation software on UNIX machine, Ubuntu 16.04, 64-bit version. The mininet-Wi-Fi provides an integrated environment to emulate the wireless access points, end stations and controllers. OpenFlow-based software defined wireless network emulation using Mininet-Wi-Fi is explained along with a step-by-step experiment in Mininet-Wi-Fi.

The mininet-Wi-Fi allowed us to create a complete OpenFlow-based software defined wireless network on a single computer. It also allowed us to customize and interacting with OpenFlow, software defined prototype. The emulated devices that we were used in modeling the software defined wireless network are listed as follows:

a) Emulated end stations.
b) LINUX virtual machine, 16.04 LTS, 64-bit version.

c) Emulated Wireless access points.
d) Simulated *hwsim0* wireless driver

We have used wireshark to monitor the wireless network traffic passing between the virtual *hwsim0 interface* and the virtual mobile stations in the Mininet-wifi network scenarios. To capture the traffic, we have enabled the *hwsim0* virtual interface on an access point, and monitored the traffic passing through the interface. The *hwsim0* interface replays communications sent to the the access point's simulated wireless interface. The traffic captured using wireshark on our test environment is shown in Figure 3.

3.1.1. Creating Software Defined Wireless Network

Mininet-Wi-Fi provides different commands for management, creating topologies and contorlling the devices in software defined wireless network. For example the command shown in (1) is used to create a simple wireless network with three stations, an access point and a controller.

sudo mn --wifi --topo single,3 (1)

Starting stations, sta1 in the SDN network with command shown in (2).

Mininet-wifi> xterm sta1 (2)

In the same way as shown in Figure 1 software defined network with four stations are created by using the command shown in (3).

```
*** Creating network
*** Adding controller
*** Adding stations:
sta1 sta2 sta3
*** Adding access points:
ap1
*** Configuring wifi nodes...
*** Adding link(s):
(sta1, ap1) (sta2, ap1) (sta3, ap1)
*** Configuring nodes
*** Starting controller(s)
c0
*** Starting switches and/or access points
ap1 ...
*** Starting CLI:
mininet-wifi>
```

Figure 1. Software defined wireless network created by command issued in (1)

tt@ubuntu~$ sudo mn --wifi --topo linear, 2 (3)

The in (2) creates a software defined wireless network with two emulated access points and two stations.

The location of nodes is an important feature of WiFi networks. The Mininet WiFi offers a graphical demonstration as shown in Figure 2, showing locations of WiFi end stations and access points in a graph. The graph was created by calling the method in the Mininet-WiFi Python API. The graph shows wireless access points and stations, their positions in space and displays the range parameter for each node.

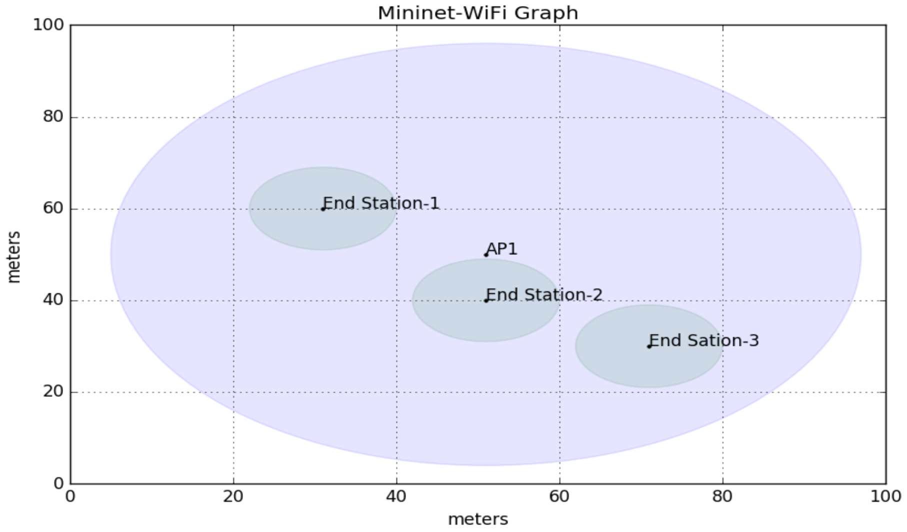


Figure 2. Mininet-Wi-Fi graph

4. PERFORMANCE IMPOVEMENT WITH HANDOVER

Handover is the widely used technique for performance improvement in wireless networks, where multiple access points exist in the network. The concept of handover is realized by creating two or more access points with overlapping regions of signal coverage and moving hosts between the access points based on the number of end stations associated with a particular access point.

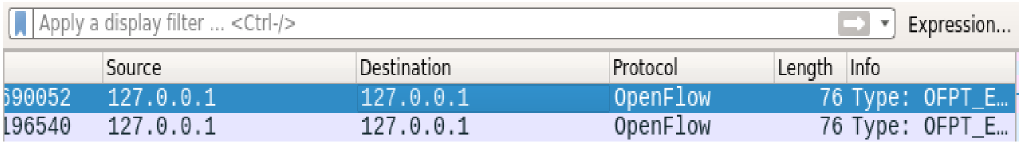
Wireless Access points load balancing based handover Algorithm:

1. Set a network topology
2. Add Access Points, AP1, AP2
3. Define the range, with overlapping coverage between AP1 and AP2
4. Start a controller
5. Controller gets load information of AP1 and AP2
6. Define Threshold, T of AP1 and AP2 on the controller
7. **loop**
8. If AP1_load≥T
9. AP1_overloaded
10. Deny New Stations
11. Define the mobility model to move the New Station to the overlapping region
12. When the New Station reaches the overlapping region of AP2, send de association message to AP1
13. Transfer New Station to AP2
14. else
15. No handover
16. **end if**
17. **end loop**

5. EXPERIMENTAL RESULTS AND ANALYSIS

To evaluate the performance of software defined wirless network, we have used the software defined wireless network topology created using Mininet-Wi-Fi which is shown in Figure 2. This topology includes three end stations connected to an access point through their simulated *hwsim0 interface*. Ensuring the performance of critical applications is a common objective in networking. A Software Defined Wireless Network application can automatically provision the network to allocate bandwidth and apply quality of service to end stations connected to the access point on the network. To ensure these service levels today requires a huge amount of manual configuration across both the wired and wireless networks, but with Software Defined Network, bandwidth provisioning is automated and easy.

To analyze the TCP traffic bandwidth of software defined wireless network we have created the network topology and tested connectivity between the stations in the test topology. Figure 3 shows the captured traffic, which shows the Hello message, feature request/reply in the software defined wireless network. This confirms that the OpenFlow switch in this setup is connected to the OpenFlow controller and the ping reply confirms stations are connected to each other.



Apply a display filter ... <Ctrl-/> Expression...

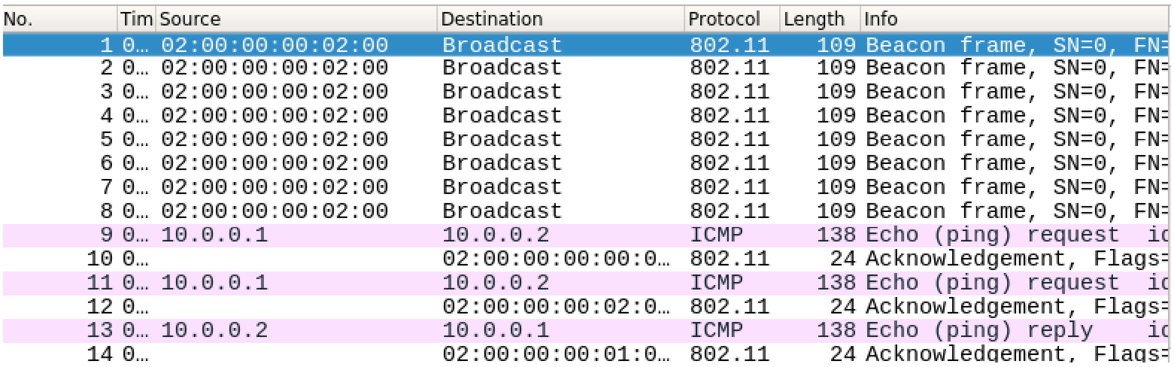
	Source	Destination	Protocol	Length	Info
590052	127.0.0.1	127.0.0.1	OpenFlow	76	Type: OFPT_E…
196540	127.0.0.1	127.0.0.1	OpenFlow	76	Type: OFPT_E…

Figure 3. Wireshark capturing openflow traffic

The connectivity between each emulated end stations in the mininet-Wi-Fi is tested by using the command given in (4).

mininet-wifi> sta1 ping sta2 (4)

The ping sends a ping request packets as shown in Figure 4. A flow entry covering ICMP ping traffic was previously installed in the access point, so no control traffic was generated, and the packets immediately pass through the access point. An easier way to run this test is to use the Mininet CLI built-in pingall command, which does an all-pairs ping. Another useful test is a self-contained regression test. The following command created a minimal topology, started up the OpenFlow reference controller, three end stations, an access point, runs an all-pairs-ping test, and tore down both the topology and the controller.



No.	Tim	Source	Destination	Protocol	Length	Info
1	0…	02:00:00:00:02:00	Broadcast	802.11	109	Beacon frame, SN=0, FN=
2	0…	02:00:00:00:02:00	Broadcast	802.11	109	Beacon frame, SN=0, FN=
3	0…	02:00:00:00:02:00	Broadcast	802.11	109	Beacon frame, SN=0, FN=
4	0…	02:00:00:00:02:00	Broadcast	802.11	109	Beacon frame, SN=0, FN=
5	0…	02:00:00:00:02:00	Broadcast	802.11	109	Beacon frame, SN=0, FN=
6	0…	02:00:00:00:02:00	Broadcast	802.11	109	Beacon frame, SN=0, FN=
7	0…	02:00:00:00:02:00	Broadcast	802.11	109	Beacon frame, SN=0, FN=
8	0…	02:00:00:00:02:00	Broadcast	802.11	109	Beacon frame, SN=0, FN=
9	0…	10.0.0.1	10.0.0.2	ICMP	138	Echo (ping) request id
10	0…		02:00:00:00:00:0…	802.11	24	Acknowledgement, Flags=
11	0…	10.0.0.1	10.0.0.2	ICMP	138	Echo (ping) request id
12	0…		02:00:00:00:02:0…	802.11	24	Acknowledgement, Flags=
13	0…	10.0.0.2	10.0.0.1	ICMP	138	Echo (ping) reply id
14	0…		02:00:00:00:01:0…	802.11	24	Acknowledgement, Flags=

Figure 4. Wireshark capturing WiFi on hwsim0 interface

tt@ubuntu:~$ sudo mn --wifi --topo single, 3 (5)

The command in (5) creates the mininet-Wi-Fi toplogy and the connectiveity between all pairs of the end satations is tested using the command shown in (6).

mininet-wifi> pingall (6)

```
*** Ping: testing ping reachability
sta1 -> *** sta1: ('ping -c1 10.0.0.2',)
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.561 ms
--- 10.0.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
```

rtt min/avg/max/mdev = 0.561/0.561/0.561/0.000 ms
*** Results: 0% dropped (6/6 received)

4.1. Performance Analysis

The performance of software defined wireless network is greatly affected by the number of end stations associated with a particular access point, shown in Table 1. To overcome this problem we have proposed load balancing based handover method as a solution. In this method we have created two access points with overlapping regions of coverage and end stations associated with each access points. The end stations are mobile devices. When one of the access point is overloaded and the station associated with this access point moves into the overlapping region, the station is handed over from the access point and is associated with the other access point, shown in Figure 6.

Table 1. TCP traffic bandwidth vs. number of emulated stations

Number of Stations	TCP Traffic Banwidth(Mbits/Sec)
5	10.5
10	10.2
20	10.4
30	8.81
40	8.3
50	8.2

The wireless network TCP traffic bandwidth varied depending on the number of emulated end stations in the emulation environment as shown in the Table 1. The traffic bandwidth gradually decreased with increasing number of end stations with maximum bandwidth of 10.5 Mbits/Sec' with 5 end stations and minimum of 8.2 Mbits/Sec' with 50 emulated end stations. Figure 5 Shows the TCP traffic bandwidth performance.

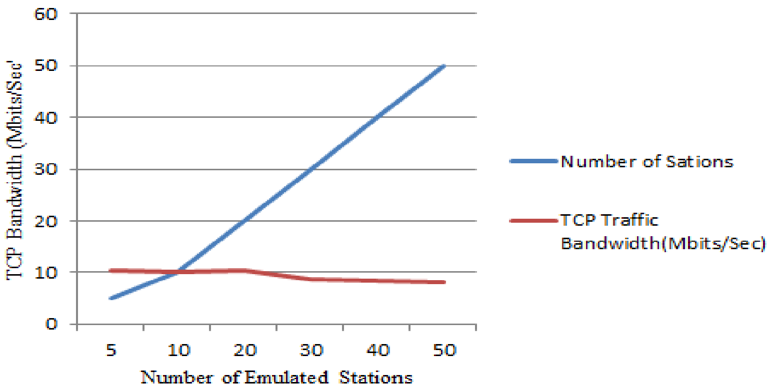
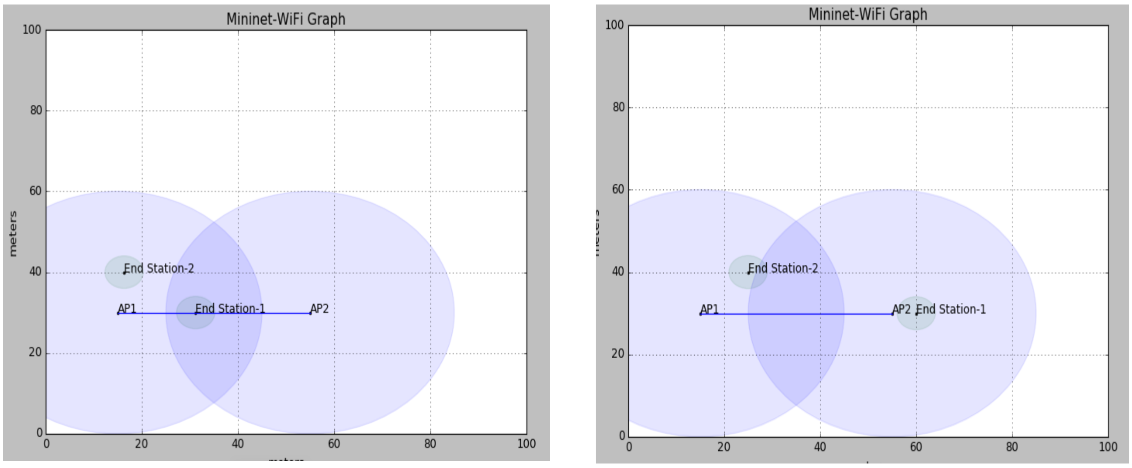


Figure 5. TCP traffic bandwidth performance

Figure 6 shows that, the number of end stations in software defined wireless network has impact on the TCP traffic performance. And the network performs better with limited number of end stations compared to larger number of end stations. The performance of software defined wireless network can be increased by increasing the number of access points, sharing the traffic load between them.

In Figure 6, load balancing based handover is shown. The Figure 6 (a) shows the end stations (end sation-1 and end station-2) were originally associated to access point, AP1. Since access point, AP2 was not associated with any station at this point, as the end sation-1 moves in the overlapping region between AP1 and AP2; it is handed over and associated with access point, AP2 as shown in Figure 6(b).

To ensure high performance, the load is shared between the access points AP1 and AP2. Load balancing is critical to network performance and it can be programmed on the controllers in software defined wireless network. The program will control the number of users connected to a particular access point and if the access point is overloaded the stations will handover and will be associated with another access point programmatically. To achieve handover, the access points should have overlapping coverage, because the stations has to be in the proximity of the access points so that they can be dynamically associated with stations.



(a) (b)

Figure 6. Load balancing handover, Mininet-Wi-Fi Graph, (a) Before handover (b) After handover

6. CONCLUSION

Software Defined wireless Network (SWDN), enables wireless network programmability, like dynamic provisioning of bandwidth and simplifies the network control, configuration, and enable revolution through network programmability. Mininet-Wi-Fi simplifies the emulation and manipulation of software defined wireless network components, like the access points and end stations. It is helpful to explore OpenFlow, which is an open interface for controlling the network elements through their forwarding tables.

In this paper, we have emulated a software defined wireless network using mininet-Wi-Fi and OpenFlow controller, a software platform developed by python which we have used as a controller with emulated stations and we have created the models of software defined wireless networks and discussed the characteristics of software defined wireless network. Lastly, we have analyzed the performance of TCP for network traffic with different numbers of emulated software defined wireless network end stations. And it is shown that, the performance degraded with the increasing number of end stations and as a solution to increase the performance we have proposed load balancing based handover to move stations from overloaded access point to other access point.

REFERENCES

[1] Ramon R. Fontes, Samira Afzal, Samuel H. B. Brito, Mateus A. S. Santos, Christian Esteve Rothenberg, "Mininet-WiFi: Emulating Software-Defined Wireless Networks," *2015 11th International Conference on Network and Service Management (CNSM), IEEE, 2015.*

[2] Dimas Agil Marenda, Galura M. Suranegara, Syamsul Qamar, Rifqy Hakimi, Eueung Mulyana, "Emulating Software-Defined Wireless Network: Bicasting Scenario*," the 3rd International Conference on Wireless and Telematics and Palembang, Indonesia,* Jul 2017.

[3] Kiran Nahida, Changchuan Yin, Ying Hu, Zulfiqar Ali Arain, "Handover Based on AP Load in Software Defined Wi-Fi Systems," *Journal Of Communications And Networks,* vol. 19, no. 6, Dec 2017.

[4] HIDEyuki Shimonishi, Yasuhito Takamiya, Yasunobu Chiba, Kazushi Sugyo, Youichi Hatano, "Programmable Network Using OpenFlow for Network Researches and Experiments," by *Information Processing Society of Japan,* 2012.

[5] Min-Cheng Chan, Chien Chen, Jun-Xian Huang, Ted Kuo, Li-Hsing Yen, Chien-Chao Tseng, "OpenNet: A Simulator for Software-Defined Wireless Local Area Network, *2014 IEEE Wireless Communications and Networking Conference (WCNC),* Apr 2014.

[6] Razvan Beuran, Junya Nakata, Takashi Okada, Lan Tien Nguyen, "A Multi-purpose Wireless Network Emulator: QOMET," *22nd International Conference on Advanced Information Networking and Applications, IEEE*, 2008

[7] Sheetalkumar R. Doshi, Unghee Lee, Rajive L. Bagrodia, "Wireless Network Testing and Evaluation using Real-time Emulation," *Scalable Network Technologies, Los Angeles CA,* 2018.

[8] Abeer A. Z. Ibrahim, Fazirulhisyam Hashim, "An architecture of 5G based on SDN NV wireless network," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS),* vol. 14, no. 2, May 2018

[9] A. Triviño Cabrera, E. Casilari, "Network Simulator: A Learning Tool for Wireless Technologies," *Current Developments in Technology-Assisted Education*, 2006.

[10] Tao Chen, Marja Matinmikko, Xianfu Chen, Xuan Zhou, and Petri Ahokangas, "Software Defined Mobile Networks: Concept, Survey, and Research Directions," *IEEE Communications Magazine*, Nov 2015.

[11] Ramon dos Reis Fontes, Christian Esteve Rothenberg, "Mininet-WiFi: A Platform for Hybrid Physical-Virtual Software-Defined Wireless Networking Research," *SIGCOMM '16*, Aug 2016.

[12] Bruno Nunes Astuto, Marc Mendonça, Xuan Nam Nguyen, Katia Obraczka, Thierry Turletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," *Communications Surveys and Tutorials, IEEE Communications Society, Institute of Electrical and Electronics Engineers*, vol. 16, no. 3, pp. 1617 – 1634, 2014.

[13] Shamik Sarkar, Christopher Becker, Josh Kunz, Aarushi Sarbhai, Gurupragaash Annasamymani, Sneha Kumar Kasera, Jacobus Van der Merwe, "Enabling WiFi in Open Access Networks," *HotWireless'17*, Oct 2017.

[14] Shie-Yuan Wang, Yi-Bing Lin, "NCTUns network simulation and emulation for wireless resource management," *Wireless Communications and Mobile Computing Wirel. Commun. Mob. Comput.*, vol. 5, pp. 899–916, 2005.

[15] Harashta Tatimma Larasati, Rifqy Hakimi, Tutun Juhana, "Extended-LLF: A Least Loaded First (LLF)-Based Handover Association Control for Software-Defined Wireless Network," *International Journal of Computer Engineering and Information Technology*, vol. 9, no. 9, pp. 203–210, Sep 2017.

[16] Murtaza Cicioğlu, Seda Cicioğlu, Bilgisayar Mühendisliği Bölümü, "Performance Analysis of Software-Defined Network Approach for Wireless Cognitive Radio Networks," *2018 26th Signal Processing and Communications Applications Conference (SIU), IEEE*, 2018.

[17] Xenofon Foukas, Mahesh K. Marina, Kimon Kontovasilis, "Software Defined Networking Concepts," *Software Defined Mobile Networks (SDMN): Beyond LTE Network Architecture*, First Edition, 2015.

[18] Daniel King, Charalampos Rotsos, Alejandro Aguado, Nektarios Georgalas, and Victor Lopez, "The Software Defined Transport Network: Fundamentals, Findings and Futures," *2016 18th International Conference on Transparent Optical Networks (ICTON), IEEE*, 2016.

[19] V. Padma, P.Yogesh, "Proactive Failure Recovery in OpenFlow Based Software Defined Networks," *3rd International Conference on Signal Processing, Communication and Networking (ICSCN)*, 2015.

[20] Farzaneh Pakzad, Siamak Layeghyt, Marius Portmann, "Evaluation of Mininet-WiFi Integration via ns-3," *26th International Telecommunication Networks and Applications Conference (ITNAC)*, 2016.

[21] S. M. Shamim, Mohammad Badrul Alam Miah, Angona Sarker, Ali Newaz Bahar, Ananya Sarker, "Simulation of Minimum Path Estimation in Software Defined Networking Using Mininet Emulator," *British Journal of Mathematics & Computer Science*, vol. 21, no. 3, pp. 1-8, 2017

[22] Lindinkosi L. Zulu, Kingsley A. Ogudo, Patrice O. Umenne, "Simulating Software Defined Networking Using Mininet to Optimize Host Communication in a Realistic Programmable Network," *2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD)*, IEEE, 2018.

[23] Casimer DeCusatis, Aparicio Carranza, Jean Delgado-Caceres, "Modeling Software Defined Networks using Mininet," *Proceedings of the 2nd International Conference on Computer and Information Science and Technology (CIST'16)*, Ottawa, Canada, May 2016.

[24] Iulisloi Zacarias, Jana-na Schwarzrock, Luciano P. Gaspary, Andersonn Kohl, Ricardo Q. A. Fernandes, Jorgito M. Stocchero, Edison P. de Freitas, "Enhancing Mobile Military Surveillance Based on Video Streaming by Employing Software Defined Networks," In *Wireless Communications and Mobile Computing*, vol. 2018, pp. 1-12, 2008.

[25] Roberto Corvaja, Andrea Zanella, Michele Dossi, Adolfo Tontoli, Paolo Zennaro, "Experimental Performance of the Handover Procedure in a WiFi Network," University of Padova–Department of Information Engineering, May 2014.

[26] Nazim Abdeddaim, Fabrice Theoleyre, "Implementation of a WSNet Module to Simulate the *IEEE* 802.15. 4 Beacon-Enabled Mode in Multihop Topologies," *An IEEE 802.15.4 package for the wsnet simulator*, 2014.