SDN-ENABLED DATA OFFLOADING AND LOAD BALANCING IN WLAN AND CELLULAR NETWORKS

by

Aili Kanee Ashipala

A dissertation submitted in partial fulfilment of the requirements

for the degree of

Master of Engineering in Telecommunications

Department of Electrical Engineering, Faculty of Engineering and the Built Environment

University Of Cape Town

November 2021

Supervisor:

A/Prof O. E. Falowo

Dedication

In memory of my brothers Shimpaalishi Ndagwana Ashipala and Klemens Natangwe Morgan. You left fingerprints of grace in our lives.

Declaration

I declare that the work presented in this dissertation is originally my own. Ideas and material generated from other researchers is explicitly stated with appropriate references.

This work is submitted for the Master of Engineering Specializing in Telecommunication at the University of Cape Town. It has not been submitted to any other university for any other degree or examination.

Name: Aili Kanee Ashipala

Date: 21 November 2021

Abstract

Networking is an interesting field that is always evolving as new technologies that connect the world are adopted. There are many distinct types of networks, each of which is classified in a different way. One categorization is based on cellular wireless network generations, which have progressed from 1G to 5G. Several additional interconnected technologies have emerged as networking has progressed. SDN (Software Defined Networking) is a significant networking technology that represents a new paradigm. SDN distinguishes between the data and control planes and allows software-controlled networking for a wide range of applications. Cellular networks are critical for sending digital data from mobile or stationary senders to mobile or stationary receivers in wireless networks.

Cellular networks are currently experiencing a data explosion because of the ever-increasing bandwidth demands of today's mobile applications. This has resulted in traffic congestion and a scarcity of resources. The network must handle a high volume of traffic and serve many customers, which may result in poor service quality for users. A single access network struggles to handle such a tremendous volume of traffic. Operators of cellular networks are attempting to alleviate the problem by offloading mobile data from cellular networks to complementary networks like Wi-Fi. However, without centralized control, traffic offloading may not significantly improve overall network load balancing, network usage, or users' quality of experience. This dissertation proposes a traffic offloading and load balancing algorithm between cellular and Wi-Fi networks, to enhance the overall utilization of cellular network. The proposed algorithm uses an SDN controller for making decisions of offloading users from a cellular network to a Wi-Fi network and to balance the load across access points. The algorithm makes use of the SDN controller's view in making decisions. Simulation results obtained show that the proposed data offloading scheme improves load distribution and throughput.

Acknowledgement

First and foremost, I want to express my gratitude to God for providing me with the strength and sanity of mind to continue working hard toward my goals. My profound gratitude to A/Prof. Olabisi E. Falowo, my supervisor, for his guidance, encouragement, and support.

Finally, I'd want to express my gratitude to my family and friends, for their unwavering support. Their consistent support and encouragement have helped me get through difficult times and endure in the face of adversity — thank you.

Table of Contents

Acronyms

3GPP	Third Generation Partnership Project
1G	First Generation Mobile Communication System
2G	Second Generation Mobile Communication System
3G	Third Generation Mobile Communication System
4G	Fourth Generation Mobile Communication System
5G	Fifth Generation Mobile Communication System
AP	Access Point
API	Application Programming Interface
BGP	Border Gateway Protocol
BS	Base Station
CCK	Complementary Code Keying
CDMA	Code Division Multiple Access
CSMA/CA	Carrier Sense Multiple Access Protocol with Collision Avoidance
DLUX	Daylight User Experience
DSSS	Direct Sequency Spread Spectrum
FDMA	Frequency Division Multiple Access
FHSS	Frequency Hopping Spread Spectrum
GSM	Global System for Mobile Communications
HetNet	Heterogeneous Network
HSUPA	High Speed Uplink Packet Access
ITU	International Telecommunication Union
ISM	Industrial Scientific Medical
IMT	International Mobile Telecommunication
LB	Load Balancing
LTE	Long-Term Evolution
LTE-A	LTE-Advanced
MIMO	Multiple-Input Multiple-Output
MU-MIMO	Multi-user, Multiple-Input Multiple-Output
NETCONF	Network Configuration Protocol
NFV	Network Functions Virtualization
ODL	OpenDayLight
OPNVF	Open Platform for Network Functions Virtualization

ONF	Open Network Foundation
ONOS	Open Network Operating System
ONAP	Open Network Automation Platform
OFDM	Orthogonal Frequency Division Multiplexing
OSGi	Open Services Gateway Initiative
PCEP	Path Computation Element Communication Protocol
PHY	Physical Layer
QoS	Quality of Service
QAM	Quadrature Amplitude Module
RATs	Radio Access Technologies
RAN	Radio Access Network
RSSI	Received Signal Strength Indicator
SDN	Software-Defined Networking
SINR	Signal to Interference and Noise Ratio
SNMP	Simple Network Management Protocol
TCP	Transmission Control Protocol
UE	User Equipment
Wi-Fi	Wireless Fidelity

List of Figures

List of Tables

CHAPTER 1

1 INTRODUCTION

1.1 General Introduction

Global communication technologies have advanced rapidly in recent years and have proven to be critical to modern civilization, particularly in the field of wireless communications. Mobile networks have progressed from 1G to 5G, with considerable increases in coverage, data throughput, and flexibility, from the 1980s to the present [1]. As a consequence of the introduction of smart-phones, tablets, laptops, and wearable devices that support multimedia applications, traditional cellular networks are witnessing unprecedented growth in mobile traffic, and demand for mobile data continues to rise [2]. Furthermore, future wireless communications will have to support a wide range of applications with varying functional needs, this is according to the report in [3]. Automation, live streaming, and interactive gaming, for example, are time-sensitive, but other services, such as the Internet of Things (IOT), need a large number of connections per unit area [3].

According to statistics on mobile data consumption, there has been a massive growth in the use of mobile data traffic [4]. As shown in Figure 1.1, in 2022, global mobile data traffic will increase to 77.5 Exabytes per month, up from 11.5 Exabytes per month in 2017 [4]. In 2017, the telecommunication industry was preparing for a shocking 1000-fold increase in data traffic [5].

Both cellular networks and Wireless Local Area Networks (WLANs) are evolving to meet such huge connection and traffic needs. The deployment of the 5G network has begun, and a new generation of mobile networks has been deployed every ten years or so. To provide higher capacity, lower latency, higher reliability, and greater mobility support than the previous generations of mobile networks, 5G needs to be more efficient and flexible. To support the enormous traffic volume and fulfil the Quality of Service (QoS) expectations of various user applications, future generation of cellular networks will require knowledge of user application requirements and real-time information sharing [1].

The goal of the 5G network is to create a connected society in which drones, sensors, wearables, and medical equipment all communicate with one another, allowing them to provide sophisticated services such as advanced security and telesurgery to end users [1]. It has been discovered that traffic from indoor/hotspot sites can account for up to 90% of overall traffic [6].

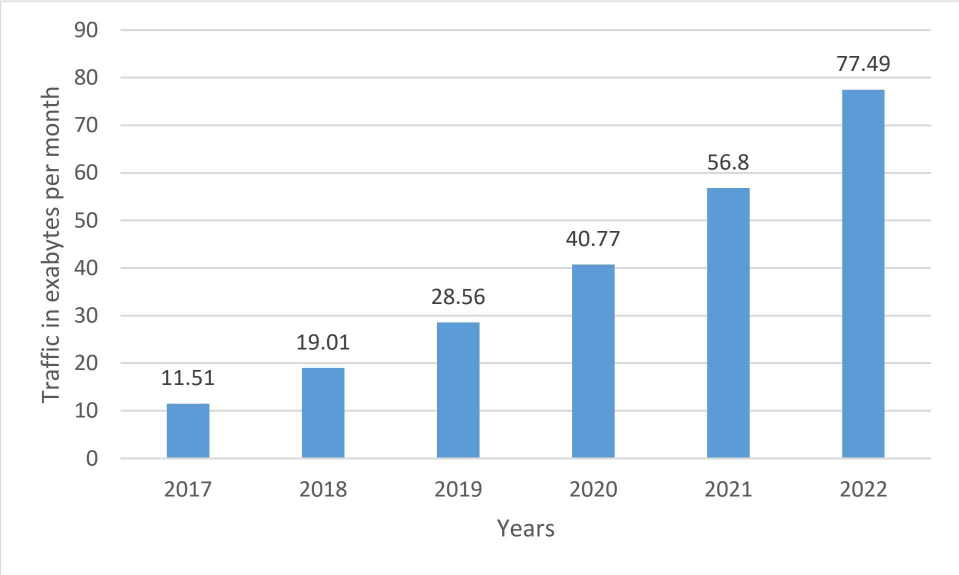


Figure 1.1: Global Mobile Data Traffic from 2017 to 2022

Source: Cisco VNI Global Complete Forecast Highlights, 2017-2022 [4]

Offloading of mobile data to supplemental networks is one viable approach for coping with cellular network congestion caused by the exponential growth of user traffic. To relieve congestion and make better use of existing network resources, mobile data offloading entails the use of complementary network technologies for transmitting data originally targeted to mobile/cellular networks [7]. The objective is to provide network users with high-quality service while decreasing the cost and impact of bandwidth-hungry services on the mobile network. The traditional approach of increasing network capacity by adding more network equipment is always available, but it is neither cost effective nor viable given the increasing demand for data services. As data traffic on mobile networks continues to grow at a rapid rate, mobile data offloading is expected to become increasingly crucial.

In recent generations of cellular networks, several researchers have proposed various strategies for offloading traffic. Offloading using femtocells and offloading with Wi-Fi are the most popular traffic offloading options. This is attributable to several factors. To begin with, it is less expensive for operators to install additional Wi-Fi hotspots than it is to upgrade their cellular network [7]. Second, existing networks have provided a favourable environment for Wi-Fi offloading. For example, some businesses have already established a large number of Wi-Fi access points (APs), implying that mobile users will have little trouble finding an auxiliary Wi-Fi hotspot to send data [7]. Finally, Wi-Fi is attractive to service providers because it operates on unlicensed frequencies (2.4GHz and 5GHz) [7]. According to Cisco, 45 percent of mobile data has been offloaded to a Wi-Fi or femtocell network, and this figure is expected to rise [4]. Furthermore, because smartphones have built-in Wi-Fi capabilities, Wi-Fi is a natural alternative for offloading, and as a consequence, many users in places where cellular coverage is degraded are already using Wi-Fi to access Internet services for a better experience [7].

This study proposes a Software Defines Network (SDN) based data offloading and load-balancing method for cellular and Wi-Fi networks. Based on the number of users connected to the cellular base station and Wi-Fi AP, as well as Received Signal Strength Indicator (RSSI), the scheme will decide on traffic offloading and load balancing. The controller receives load information from the Wi-Fi APs and micro base station through the Open-Flow channels, and then makes offloading and load balancing decisions. For performance analysis, the proposed scheme is implemented in the Mininet-WiFi emulator.

1.2 Problem Statement

Current cellular networks are overburdened by the surge in traffic caused by the proliferation of mobile devices and a variety of bandwidth-hungry smartphone applications (e.g., video streaming applications). This has resulted in cellular network congestion and, as a result, poor service quality for users.

Cellular data traffic is efficiently offloaded to the Wi-Fi network to alleviate the congestion problem in cellular networks and improve the quality of service for end users. In the literature, data offloading and load balancing techniques have been developed to solve the problem of mobile network congestion. However, existing data offloading and load balancing algorithms must be further explored and improved, as offloading without centralized management may not significantly increase overall network utilization and end users' quality of service. In a heterogeneous wireless network, flexible and programmable network management is required to ensure uniform policy and management across several access networks. As a result, SDN has been introduced as a technology that enables high-level administration, flexibility, and control.

1.3 Objectives

The main aim of this research is to address the congestion problem caused by rapid traffic increase in cellular networks, using WLAN offloading.
The following are the specific objectives of this research:

- Develop a scheme for traffic offloading and load balancing based on SDN to enhance the performance of the cellular network.
- Evaluate the performance of the data offloading and load balancing algorithm using some network parameters.

1.4 Research Contributions

The main contribution of this dissertation is the development of a data offloading and load balancing scheme for reducing congestion in an SDN-based heterogeneous networks.

1.5 Research Scope and Limitation

This study focuses on data offloading and load balancing between cellular networks and Wi-Fi access points to alleviate congestion concerns in wireless networks. To imitate a software defined network, the Mininet-Wi-Fi simulator is employed. Because there is no cellular network in this simulator, only Wi-Fi, a 300m Wi-Fi AP has been used in place of a micro-base station. Again, instead of programming the controller to incorporate offloading and load balancing functions, a default controller has been used, which basically makes offloading and load balancing decisions based on signal strength only. As a result, manual simulations using the simulator's mobility and handover modules has been employed to test the concept.

1.6 Dissertation Outline

The following is how the rest of the dissertation is organized. The essential ideas of software-defined networking (SDN), the evolution of wireless cellular networks, and Wi-Fi technology are briefly discussed in Chapter 2. In Chapter 3, a data offloading and load balancing scheme based on SDN is proposed for cellular and WLAN networks. Chapter 3 contains a description of the design and specifications for the proposed scheme. Chapter 4 describes the implementation of the proposed SDN-based data offloading and load balancing scheme. Chapter 4 also covers the software tools used for SDN data offloading and load balancing implementation, as well as the technologies that underpin the setup. The proposed design's results and performance analysis are presented in Chapter 5. Finally, Chapter 6 concludes with recommendations for further research.

CHAPTER 2

2 LITERATURE REVIEW

The background materials for this dissertation are presented in this chapter. The following is how the chapter it is organized. Section 2.1 provides an overview of the evolution of wireless cellular networks. In Sections 2.2, Wi-Fi technology, performance of Wi-Fi technology, and evolution of Wi-Fi technology are discussed. In Section 2.3, the fundamentals of SDN technology are discussed, as well as related research. Section 2.4 discusses SDN-based traffic offloading and load balancing. Finally, Section 2.5 gives the summary of the chapter.

2.1 Evolution of Wireless Cellular Networks

When the cellular idea was originally introduced in the 1970s, no one could have predicted how widespread mobile networks would become. The mobile network has experienced tremendous expansion since the mid-1990s [8]. The fast increase of mobile network customers throughout the world has proven beyond a shadow of a doubt that the mobile network can provide voice and data services. As the demand for mobile network services grows, newer and more efficient generations of mobile networks are being developed.

2.1.1 First Generation (1G)

The first generation of cellular telephone network is referred to as 1G. It was an analog telecommunications standard that began in 1979 and lasted until the development of 2G technology [9]. The earliest wireless mobile phone handsets employed this technology. Japan was the first country to deploy 1G technology, and it swiftly spread to other nations. The analog radio signal was employed in 1G technology. Voice call was supported on the network using the Frequency Division Multiple Access (FDMA) [9].

2.1.2 Second Generation (2G)

Second-generation cellular telephone network is referred to as 2G technology. It was primarily based on the Global System for Mobile Communication (GSM) [9]. Network services such as telephony and instant text messaging were provided by this technology. In Finland, the second-generation network was launched in 1991. Digital encryption was used to encrypt all phone conversations. GSM allowed users to send and receive short message services (SMS) from anywhere and at any time. SMS was created as a low-cost and simple means to communicate with anybody. Time Division Multiple Access (TDMA) or Code Division Multiple Access (CDMA) were the two 2G technologies [9].Time is divided into time slots in TDMA. CDMA assigns a unique code to each user for them to communicate over a multiplexed physical channel. 2G technology increased privacy of telephone network. 2.5G and 2.75G are improved versions of the 2G network.

2.5G technology [9] **has the following features**

- 2G cellular technology with GPRS.
- Data rates range from 56 to 115 kilobits per second.
- Use of e-mail
- Use of the internet.

2.75 G technology [9] **has the following features**

- Enhanced Data Rates for GSM Evolution (EDGE).
- Maximum speed 384 Kbps.

2.1.3 Third Generation (3G)

The third-generation cellular network is referred to as 3G. In 2001, Japan was the first country to commercially launch 3G [9]. In Japan, the transition to 3G was completed in 2005-2006 [9]. In 2005, 23 networks throughout the world were using 3G technology. Some 3G networks were primarily for testing purposes, while others offered consumer services. The 3G network was designed to facilitate growth, increase bandwidth, and support varied applications. The International Telecommunication Union (ITU) set the specifications for 3G in the International Mobile Telecommunication IMT-2000. 3.5G and 3.75G are the improved versions of this technology [9].

3.5G Technology [9] **has the following features**

- HSDPA (High-Speed Downlink Packet Access)
- It provided 3G networks with a smooth evolutionary path that allowed for higher data transfer speeds.

- Up to 8-10Mbps (20Mbps for some systems) data transmission

3.75 G Technology [9] **has the following features**

- The 3.75G refers to the High-Speed Uplink Packet Access (HSUPA) technology.
- HSUPA is a mobile telecommunications technology that is closely related to HSDPA and these two are complementary to one another.
- HSUPA enabled data applications such as real-time gaming with higher and symmetric data rates.

2.1.4 Fourth Generation (4G)

The 4G mobile networks is a packet switched network with high throughput. It is intended to be both cost-effective and spectrally efficient. The Orthogonal Frequency Division Multiple Access (OFDMA) technology is used in the 4G network [9]. Device mobility of up to 350km/hr is possible with 4G. 4G is described using the acronym as MAGIC where M is Mobile multimedia, A= Anytime, anyplace, G = Global mobility support, I= Integrated wireless solution, C= Customize personal service. A 4G network provides laptop, computers, smartphones, and other mobile devices with secure all-IP based mobile internet services [9]. Users may use the 4G network to access services like IP telephone, gaming, and multimedia streaming.

2.1.5 Fifth Generation (5G)

The 5G mobile network is the successor of the 4G (LTE-A). High data rates, low latency, energy conservation, cost reduction, increased system capacity, and enormous device connection are all the goals of 5G performance [9]. The performance of the 5G mobile network is much superior to that of earlier generations of mobile networks. Because of its flexibility and capacity to support a wide range of vertical businesses, the 5G network has a bright prospect [9].

2.2 Wi-Fi Technology

With drastically growing data traffic in recent decades and in the foreseeable future, there is need for wireless networks with the densified deployment of small cells base stations or access points [1]. Wi-Fi technology has been popular because of its ease of deployment and the benefits of operating in the unlicensed band, and it is projected to play an important role in 5G network traffic offloading and serving network users with high data rates in Wi-Fi hotspots [1].

The IEEE standard for Wi-Fi is 802.11 [10]. It contains information and specifications for the physical and MAC layers, which are used for data transmission via wireless networks [10]. Wi-Fi networks employ a variety of frequency bands for transmission, including 2.4, 5 and 60 GHz.

The use of Wi-Fi to connect to the internet has increased as a results of the growing number of smartphones and other mobile device. With the growing demand for data services, network operators have been exploring ways to supplement data service delivery by adding unlicensed spectrum [10].

2.2.1 Evolution of Wi-Fi

IEEE 802.11, popularly referred to as Wi-Fi, has experienced increased throughput from 2 Mbps to over 1 Gbps, a 1000-fold increase in throughput over the past 20 years [10]. By releasing additional standards like 802.11n, 802.11ac, and 802.11ax (Wi-Fi 6), the standard has continuously evolved. Higher order modulation methods, such as 64 QAM, 256 QAM, and 1024 QAM, are supported by the new standards [10]. These new standards also allow multiple streams to be transmitted to a single client or several clients at the same time. In addition to raising peak data rates, efforts have been undertaken to increase spectral efficiency, which describes how efficiently the system utilizes the available spectrum [10]. To increase network efficiency and capacity, multi-user approaches such as Multi-User Multiple-Input Multiple-Output (MU-MIMO) and Orthogonal Frequency Division Multiple Access (OFDMA) have been introduced [10]. Each new standard builds on the preceding one, improving speed and reliability.

2.2.2 Wi-Fi Standards

Since its initial distribution to customers in 1997, Wi-Fi standards have been constantly evolving, often resulting in higher speeds and increased network/spectrum efficiency. As new features are introduced to the original 802.11 standard, they are referred to as standards

(802.11b, 802.11g, etc.). The most used standards are 'b', 'g', 'n', 'ac', and 'ax'. Table 2.1 shows the various standards and the maximum theoretical data rates that can be obtained with them.

Table 2.1. Different IEEE 802.11 Standard [10]

IEEE 802.11 Protocol	**Release Date**	**Frequency Band(s)**	**Bandwidth**	**Max Throughput**
802.11-1997	1997	2.4	22	2 Mbps
11b	1999	2.4	22	11 Mbps
11a	1999	5	20	54 Mbps
11g	2003	2.4	20	54 Mbps
11n (WiFi 4)	2009	2.4/5	20/40	600 Mbps
11ac (WiFi 5)	2003	2.4/5	20/40/80/160	6.8 Gbps
11ax (WiFi 6)	2019	2.5/5	20/40/80/160	10 Gbps

2.2.2.1 802.11-1997 Standard

The earliest wireless standard in the family was 802.11-1997, which was introduced in 1997 but is now obsolete. Using Carrier Sense Multiple Access Protocol with Collision Avoidance (CSMA/CA), this standard defines the protocol and suitable for connecting communication devices via the air in a Local Area Network (LAN). This protocol supports three physical layer technologies: infrared at 1 Mbps, a Frequency Hopping Spread Spectrum (FHSS) at 1 Mbps and an optional 2 Mbps data rate, or a Direct Sequence Spread Spectrum (DSSS) at both 1 and 2 Mbps data rates [10]. The protocol was not widely accepted due to interoperability difficulties, expense, and a lack of sufficient throughput [10].

2.2.2.2 802.11b Standard

In mid-1999, 802.11b standard was commercialized. This standard has a theoretical maximum data rate of 11 Mbps and uses the CSMA/CA media access technique. Because of the exceptional improvement in throughput and significant price reduction, it was widely accepted as a wireless technology. IEEE802.11b uses the Industrial Scientific Medical (ISM) unlicensed frequency band of 2400-2500 MHz. Furthermore, 802.11b is a direct expansion of Direct Sequence Spread Spectrum (DSSS), and its modulation technique is Complementary Code Keying (CCK) [10]. In a point-to-multipoint arrangement, 802.11b is used to connect with mobile clients within the access point's range [10]. This range is determined by the radio frequency environment, output power, and receiver sensitivity. 802.11b has a 22 MHz channel bandwidth and can operate at 11 Mbps but can also scale back to 5.5, 2, and 1 Mbps (adaptive rate selection) to reduce the number of re-transmissions caused by errors. Because it shares the same frequency spectrum as other wireless standards, it can create interference with home wireless devices such as microwave ovens, Bluetooth devices, and cordless phones.

2.2.2.3 802.11a Standard

The 802.11a standard uses 52-subcarrier Orthogonal Frequency Division Multiplexing (OFDM) with a theoretical data rate of 54 Mbps and operates at 5 GHz [10]. This results in a throughput of mid 20 Mbps in practice. It also supports data rates of 6, 9, 12, 18, 24, 36, and 48 Mbps [10]. Because they operate in different unlicensed ISM frequency bands, 802.11a and 802.11b are incompatible. The 5 GHz spectrum gives 802.11 a considerable advantage because the 2.4 GHz band is becoming crowded.

2.2.2.4 802.11g Standard

In the summer of 2003, 802.11g became available. It uses the same OFDM technology as 802.11a. It enables a maximum theoretical rate of 54 Mbps, just like 802.11a. However, like 802.11b, it operates in the crowded 2.4 GHz band, making it more susceptible to interference. 802.11g and 802.11b are compatible (i.e.,802.11b devices can connect to an 802.11g access point). By using 802.11a and 802.11b/g, 802.11g could support dual-band or dual-mode access points [10].

2.2.2.5 802.11n Standard

The 802.11n standard uses MIMO and 40 MHz channels at the physical layer (PHY), and frame aggregation at the MAC layer. MIMO is a technique for increasing the capacity of a radio link by utilizing multipath propagation by using multiple transmit and receive antennas. These antennas are spatially separated so that the signal from each transmit antenna to each receive antenna has a distinct spatial signature and can be divided into parallel independent channels by the receiver. When compared to a single 20 MHz channel, channels having a bandwidth of 40 MHz double the channel bandwidth and deliver double the PHY data rate. Up to four spatial streams can be supported by 802.11n standard, with a theoretical throughput of 600 Mbps [10].

2.2.2.6 802.11ac Standard

802.11ac standard supports gigabit rates per second by extending the 802.11n features such as having wider bandwidth (up to 160 MHz), additional MIMO spatial streams (up to eight), downlink multi-user MIMO (up to four clients), and high-density modulation (up to 256 QAM) [10]. 802.11ac supports 256 QAM at 3/4, 5/6 coding rate (MCS8/9). The first batch of 802.11ac devices only use 80 MHz channels and up to three spatial streams, with physical layer speeds of up to 1300 Mbps. More spatial streams, and MU-MIMO are supported by second batch 802.11ac devices. While MIMO sends multiple streams to a single user, MU-MIMO sends spatial streams to multiple users at the same time, resulting in increased network efficiency [10]. In addition, 802.11ac employs a beamforming technique. The antenna basically sends radio signals that are aimed at certain devices via beamforming.

2.2.2.7 Wi-Fi 6 or 802.11ax Standard

802.11ax is the sixth generation of Wi-Fi technology, which builds on the capabilities of 802.11ac to provide increased wireless capacity and dependability. Denser modulation (1024 QAM and OFDMA) and shorter subcarrier spacing (78.125 kHz) are used in 802.11ax to accomplish the gains. 802.11ax is a dual-band (2.4 and 5 GHz) technology, unlike 802.11ac. 802.11ax is compatible with 802.11a/g/n/ac standards. 802.11ax employs OFDMA, which allows sharing of bandwidth based on users' needs.

2.2.3 Performance of Wi-Fi

Interference is a concern in Wi-Fi networks due to the popularity and large number of Wi-Fi devices that use unlicensed frequency spectrum. In WLAN, service quality is also a concern. The technique for sharing Wi-Fi radio resources necessitates devices competing with one another. As a result, QoS is not always guaranteed.

2.4 GHz and 5 GHz are the most used Wi-Fi bands. Three non-overlapping 20MHz channels are available in the 2.4GHz frequency [1]. There is no standard for assigning these bands in a Wi-Fi network. As shown in Figure 2.1, the three non-overlapping channels at 2.4 GHz are usually 1, 6, and 11. Wi-Fi does not provide any guarantees about latency or bandwidth.

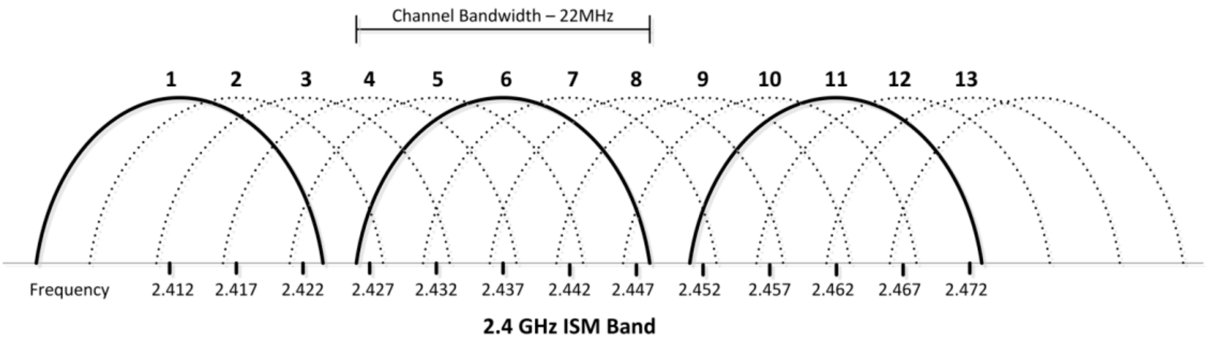


Figure 2.1: Wi-Fi network data pattern [1]

2.3 Software Defined Networking (SDN)

2.3.1 Fundamental Idea of SDN

This research requires a thorough understanding of Software Defined Networks (SDN). SDN is a game-changing upgrade to existing network technologies [11]. The goal of centralizing the control function in networking has long been a desirable goal, and SDN has now achieved it [12]. In most networks, the data and control planes are integrated. SDN, on the other hand, separates the data and control planes for high-level control and coordination. Due to the presence of Application Programming Interface (API) in SDN, such a distinction between data and control plane is feasible. Another feature of SDN is the OpenFlow protocol, which provides for such distinction and efficient data and control plane operation. Traditional networks lacked network programmability, which was remedied by SDN. Programmability enables for flexibility and control without requiring the system's hardware to be changed [11].

In the architecture of SDN, there are three layers that can be generically classified. The topmost layer, known as the application layer, is responsible for defining how a network behaves. The application layer uses an API to verify that the network is connected to the applications. The layer also enables applications to communicate with networks from a distance. The application layer can be used to remotely control and program a network. The control layer is the following layer. APIs are connected to the application layer and control layer, acting as a bridge for control and data communication between the layers [13]. The control layer of SDN manages all network-related policies and traffic flows [14]. The data layer is the third layer in the architecture, and it consists of devices that forward data packets such as switches [15]. The data layer's principal responsibility is data packet forwarding. The API links the control and data layers together. There are different protocols in SDN such as the OpenFlow. The OpenFlow protocol of SDN is the focus of this dissertation.

2.3.1.1 OpenFlow Protocol

The OpenFlow (OF) protocol oversees connecting hardware devices like switches to the SDN controller [13]. The standard southbound communication between the SDN controller and the switch is OpenFlow. In network management, it can also be used to monitor switch and port information. However, a simple and effective way of switching between multiple data sets is necessary. As a result, the controller's application layer must be programmable and flexible. Altering routing tables and other flow settings in the application layer is more convenient than changing hardware [16]. Thus, the flow rules are the highlight of the OpenFlow protocol, which allows the controller to be programmed on the software layer without requiring any hardware changes [17]. The flow table specifies the actions that the switches should take. Dropping the packet and forwarding are two possible actions.

Figure 2.2 depicts key components of an OpenFlow switch: packet matching and forwarding flow tables, and an OpenFlow channel coupled to a remote controller. Each flow table has a collection of flow entries, each with its own set of match fields and instructions for matching packets. The standardized OpenFlow protocol allows the remote controller to create, amend, and delete flow entries in flow tables.

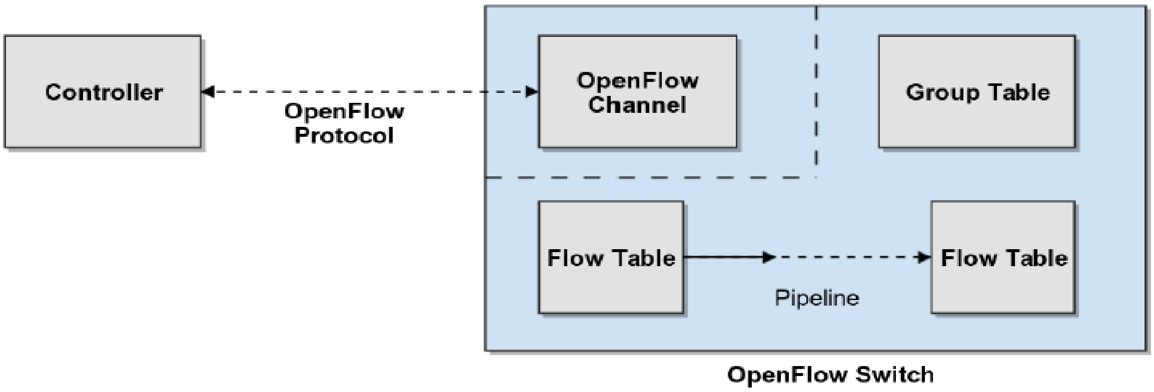


Figure 2.2: Components of an OpenFlow Switch [17]

2.3.2 Architecture of SDN

SDN's design is critical for the load balancing and offloading functionalities to work properly. SDN has been explored from multiple perspectives by both the Open Network Foundation (ONF) and the Software-Defined Networking Research Group [12]. The Open Network Foundation, created in March 2011 by Google, Microsoft, Yahoo, and a few telecom operators, is a non-profit organization dedicated to the advancement of SDN-related technology, standardization, and marketing [18]. In the SDN design, the control and data planes are separated, network intelligence and state are logically centralized, and the underlying network infrastructure is isolated from the applications [11]. Figure 2.3 depicts the architecture of software-defined networking. SDN is a network architecture that separates the control and data planes of traditional devices, which were previously vertically integrated.

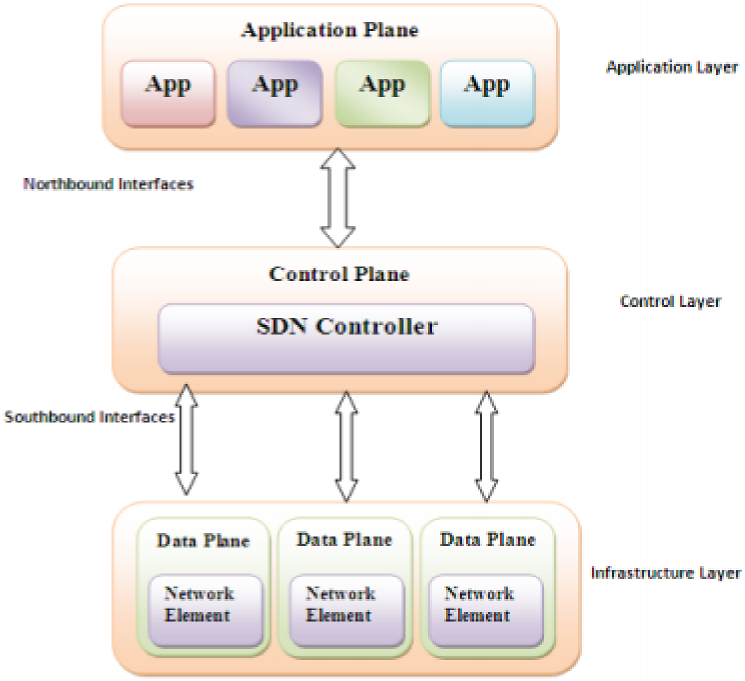


Figure 2.3:The Architecture of Software-defined Networking [16].

2.3.3 Open Source SDN Controllers

The core components that define the functionality of any action in SDN are controllers. Controllers are in charge of determining and changing the system's behaviour as well as efficiently routing data packets to their destinations [18]. Because the services that define network performance can be rationally created and controlled from the control plane, the SDN architecture's centralization is advantageous. Instead of a distributed network, the applications normally operate on the control plane using a controller, just like a traditional one-computer system.

There are a variety of SDN controllers available that handle a variety of Northbound and Southbound interfaces, all of which are based on the control and data plane separation idea. The following are some of the SDN controllers:

OpenDayLight (ODL) is one of the controllers. It has three layers: northbound interfaces such as NETCONF (Network Configuration Protocol) to allow external policy to be implemented, a central core with an OSGi (Open Services Gateway initiative)-based Service Abstraction Layer, and a set of southbound plugins to interface to programmable network devices. Each

ODL cluster maintains an in-memory model-based view of the network for which it is responsible. BGP (Border Gateway Protocol) is used natively by ODL to allow isolated SDN islands to exchange routing topology [19].

ODL has been widely utilized as the SDN controller in a Network Function Virtualization stack due to its simplicity of integration - native integrations with OpenStack, Kubernetes, Open Platform for Network Function Virtualization (OPNVF), and others exist to enable a single Network Function Virtualization (NVF) management stack to control both virtualization and network topology [19].

POX is another controller [20] written in Python for rapid prototyping and development of software. Because it is easy build software for POX, it is mostly used for research, demonstrations, and experimentation. Because it is written in Python, the network's performance is lower than that of other programming languages such as C++ or Java. Furthermore, POX only supports OpenFlow version 1.0 and does not work in a distributed fashion [20].

RYU- Ryu is another SDN controller [19]. Ryu supports a wide number of protocols, including the OpenFlow 1.5, which makes it excellent for research and prototyping. Because Ryu lacks a native clustering mechanism, its scalability is constrained by the capabilities of the platform on which it runs [19].

Another SDN controller, **ONOS (Open Network Operating System)**, provides an open source, distributed network operating system. Because it is written in Java, it takes longer to learn to develop applications for ONOS than POX and Ryu, which are both based on Python. ONOS supports Open Flow versions 1.0 and 1.3 as well as alternative protocols such as OVS-DB as southbound interface [19].

2.3.4 Wireless Networks and SDN

In different ways, wireless networks such as cellular and other networks are strongly linked to SDN. The survey in [21] presents an overview of the application of SDN in four common wireless networks: mesh, cellular networks, sensor networks, and home networks, as well as its benefits. To enable scalable deployments, the author has proposed the use of SDN architecture in wireless cellular networks to address the concerns of effective resource allocation and interference avoidance. A logically centralized controller with decoupled control and data planes can perform global resource allocation and interference management, allowing for more precise judgments and improved performance and stability.

As broadband networks reach theoretical limits in terms of spectral efficiency per link, Tomovic, *et al* have explained the potential benefits and contributions that the SDN paradigm can provide to solving congestion problems in [22] . They have claimed that addressing those issues with the current inflexible, scalable, and sophisticated design is difficult, if not impossible. They have emphasized the significant potential benefits of implementing the SDN idea in mobile networks, including more efficient inter-cell interference management, easier traffic control, and complete network virtualization. Furthermore, by separating the control and data planes, new Radio Resource Management techniques can be easily implemented without requiring networking hardware changes.

Based on the literature cited above, Software defined networking (SDN) is a new networking architecture paradigm that holds great promise for overcoming many of the limitations in wireless cellular networks and delivering needed performance improvements by decoupling control functions from the underlying physical infrastructure. SDN assists in the implementation of resource management schemes such as data offloading and load balancing.

2.4 Mobile Data Offloading and Load balancing Algorithm

2.4.1 Cellular Data Offloading

Mobile data offloading, also known as data offloading or traffic offloading, is a method of reducing congestion by using alternative networks to transport mobile data that was originally intended for cellular networks. An end-user (mobile subscriber) or an operator can set the rules that activate the mobile offloading activity. The algorithm that implements the rules can run on an end-user device, a server, or a combination of both. Users would demand data offloading to reduce data service costs and gain access to more bandwidth. The main complementary network technologies used for mobile data offloading are Wi-Fi and femtocells .

2.4.2 Load Balancing in Wireless Networks

When it comes to sustaining quality of service in mobile networks, data traffic congestion is a problem. The available bandwidth is limited, and the number of devices connected to the network is increasing by the day. As a result, network congestion should be dealt with effectively, as it affects service quality. In a mobile network, load balancing is thus an essential contributor to network quality of service. Load balancing ensures that many devices may be accommodated while network resources are utilized efficiently.

In WLANs without load balancing, a mobile device will scan all available APs and select the AP with the best signal strength to associate with by default, without considering the AP's traffic load status. This approach frequently results in imbalanced traffic load on APs, causing poor QoS.

2.5 Chapter Summary

The background concepts employed in the dissertation were reviewed in this chapter. A brief history of wireless cellular networks and Wi-Fi technology evolution and standards were provided at the start of this chapter. Software-defined networking was explored in Section 2.3. Both Concepts and architecture of SDN and related works on the use of SDN in wireless networks were provided. The concepts of data offloading and load balancing were also discussed in section 2.4. The chapter focused on addressing congestion issue in cellular networks by employing data offloading and load balancing.

CHAPTER 3

3 SDN-ENABLED DATA OFFLOADING AND LOAD BALANCING IN WLAN/CELLULAR NETWORKS

3.1 Introduction

Cellular networks have seen an unparalleled increase in data traffic over the previous decade, putting a significant strain on mobile network. The emergence of smart and media-rich mobile devices is primarily responsible for this growth [1]. These smart devices make use of high-traffic mobile applications and cloud-based services [1]. Mobile data is predicted to exceed 77.5 Exabytes per month by 2022, according to a Cisco networking visual index analysis issued in February 2017 [4].

Existing cellular networks are unable to handle the enormous volumes of data generated by smart devices. Furthermore, cellular network infrastructure upgrades cannot keep up with the surge of mobile data traffic. As a result, 5G is being touted as the next-generation cellular standard, and operators have been installing the new wireless network to deliver greater data rates as well as small cell solutions to satisfy rising traffic demand [1].

Data offloading and load balancing are viewed as key components in resolving the congestion issue. The usage of complementary network technologies for conveying data originally intended for cellular networks is referred to as mobile data offloading [7]. This approach reduces the amount of data sent across the cellular network, freeing up radio bandwidth for other users [23]. An end-user (mobile subscriber) or an operator can set events that trigger the mobile offloading action. The load balancing algorithm may exist in an end-user device, a network server, or a combination of both [24]. End users may use data offloading to save money on data services and take advantage of larger bandwidth. Wi-Fi and femtocells are the most common complementary network technologies used for mobile data offloading [23].

This research investigates data offloading and load balancing between cellular and Wi-Fi networks utilizing software-defined networking technology. Offloading is utilized in conjunction with load balancing to alleviate network congestion and enhance the resource utilization in cellular networks.

3.2 Related Work

The related work on data offloading and load balancing in wireless communication networks is covered in this section. In wireless networks, there has been an increasing interest in the study of data offloading and load balancing based on SDN.

3.2.1 Data Offloading

Duan *et al* [1] proposed an SDN-based partial data offloading and load balancing method to relieve spectrum shortage concerns and network congestion issues. The algorithm uses the SDN controller's global view of the network to fulfil the objectives while taking network conditions and end-user QoS requirements into account. Their goal, though, was to use Wi-Fi whenever possible to limit cellular network usage. This is problematic because cellular networks are sometimes more suitable for some services than Wi-Fi.

Ahn and Chung proposed a traffic offloading technique between femtocells and Wi-Fi networks, based on software-defined networking (SDN) technology in [25]. SDN ensures network flexibility and a global view of the whole network. However, offloading from a femtocell to a Wi-Fi AP is not a viable option because femtocell has a smaller coverage than Wi-Fi.

To relieve network congestion and enhance traffic load balancing, Bo Fan and Zhengbing presented an intelligent software defined cellular vehicle-to-everything (C-V2X) in [26]. By separating the network data plane from the control plane, their approach offers flexible and low-complexity traffic offloading. In comparison to existing systems, their architecture was able to dramatically enhance network speed, load balancing, and user service ratio.

Zhou *et al* implemented a Reverse Auction-based Incentive Mechanism (RAIM), to motivate nodes in Opportunistic Mobile Networks (OMNs) to provide data offloading services in [27]. Because OMN nodes are rational and selfish, they do not provide data offloading services if they are not adequately rewarded. As a result, the authors devised incentive mechanisms to encourage nodes to provide data offloading services. However, they did not incorporate software defined networking in their scheme.

3.2.2 Load Balancing

The topic of load balancing using SDN have been tackled by many researchers in the past. For example Kiran *et al*, in [28] developed an SDN handover algorithm based on AP load. In their scheme, they used Mininet Wi-Fi emulator to construct topologies for their experiments. The simulation results showed a successful handover from an overloaded AP to a lightly loaded AP. Moreover, the throughput and latency parameters that they tested showed a significant improvement when the stations were connected to least loaded AP than to loaded AP.

In [29] , Shafi *et al* developed a handoff decision scheme for Wi-Fi systems. In traditional handoff in Wi-Fi networks, handoff decisions are made based on signal strength which results in poor connectivity specifically when an access point is overloaded. They developed a decentralized approach for selecting the best access point for a user in an overlapping region. The scheme aimed to switch users to the least loaded AP to prevent an access point from getting overloaded. The simulation results showed an improved data rate per user when handoff decisions were made based on achieved throughput and load per access point rather than based on RSSI only.

In [30] , Sounni *et al* proposed a load balancing algorithm for IoT devices. The scheme was proposed to tackle the issue of mobility in communication devices in a dense network. They developed an algorithm to select the best AP for a user in an overlapping area. Their results revealed that the proposed system balanced the network's load and improved device throughput.

Khan *et al* [31] proposed a framework for Software Defined Cellular Networks (SDCN). They developed an algorithm for user scheduling, load estimation, handover decision, admission, and rate control. They used the NS3 simulator to evaluate the SDN based network. The proposed system was assessed and compared to other cell association algorithms. Their algorithm outperformed the other algorithms.

Kiri *et al* proposed an SDN-based multi-RAT Radio Access Network architecture in [32]. They presented an architecture supporting network flexibility and load balancing. In comparison to existing algorithms, their algorithm enhanced network performance.

Shiwei proposed a motion state estimation-based load balancing algorithm for heterogeneous network [33]. The simulation results showed that the proposed algorithm effectively balanced the load in the heterogeneous network while also improving system efficiency. However, the however, did not incorporate software defined networking.

Torres *et al* [34] proposed a load balancing scheme combining a fuzzy logic controller algorithm with social awareness. However, their scheme did not incorporate software-defined networking.

The works reviewed above showed performance improvements regarding traffic management in existing networks. Thus, this dissertation builds on the previous works by developing an algorithm for traffic offloading and load balancing between a cellular network and a Wi-Fi network.

3.3 SDN Based Data Offloading and Load Balancing Algorithm

The offloading and load balancing algorithms are described in depth in this section. The combination of offloading and load balancing methods provides a network congestion solution. The traffic congestion problem is reduced if the algorithm moves devices between cellular and Wi-Fi networks and balances the load. The load should be distributed equally over the available networks as much as possible.

3.3.1 Parameters for Making Offloading and Load Balancing Decisions

To address the issues of load balancing and traffic offloading, the scheme proposed in this research includes requirements for making appropriate offloading decisions. The algorithm makes traffic offloading decisions based on signal strength and total traffic loads on the Wi-Fi AP and micro-base station.

3.3.1.1 Signal Strength

The RSSI, or "Received Signal Strength Indicator," shows the relative quality of a device's received signal. The stronger the signal, the higher the RSSI value. The signal strength is expressed in decibels-milliwatts (dBm) (0 to -100). This is the decibel (dB) power ratio of the measured power when compared to one milliwatt. The stronger the signal, the closer the value is to 0. For example, -50dBm is a pretty good signal, -75dBm - is reasonable, and -100 is no signal at all [35].

Because the signal received at a particular point travels through several pathways (multipath effects), it is critical to consider the following elements that influence the received signal strength [35].

- Obstacles between the access point and the device in the environment (trying to talk through doors and walls)
- Interference from other electronic equipment that create electronic signals (trying to chat in a room with other people conversing)
- Distance between the AP and the device (trying to talk while far away)
- The mobility of the user

3.3.1.2 Load as Number of Users

In this dissertation, the number of users/stations connecting to each access point is referred to as the load. We assume that all stations have the same traffic pattern, actively transmitting and hence the same bandwidth requirement.

3.4 System Model

The proposed system model is based on SDN. The system model consists of one micro cell base station, an overlay of N number of Wi-Fi APs, and M number of stations (users) within the micro cell's coverage area, as well as users connected to the Wi-Fi APs and the micro base station. The logically centralized controller is connected to the microcell base station and Wi-Fi APs. Figure 3.1 shows the proposed SDN system model. The controller has a full view of the whole network topology and is aware of every flow in the network. The controller communicates with the access points using Open Flow to collect information such as received signal strength and the number of stations connected to each Wi-Fi AP and micro-base station. The controller maintains this information.

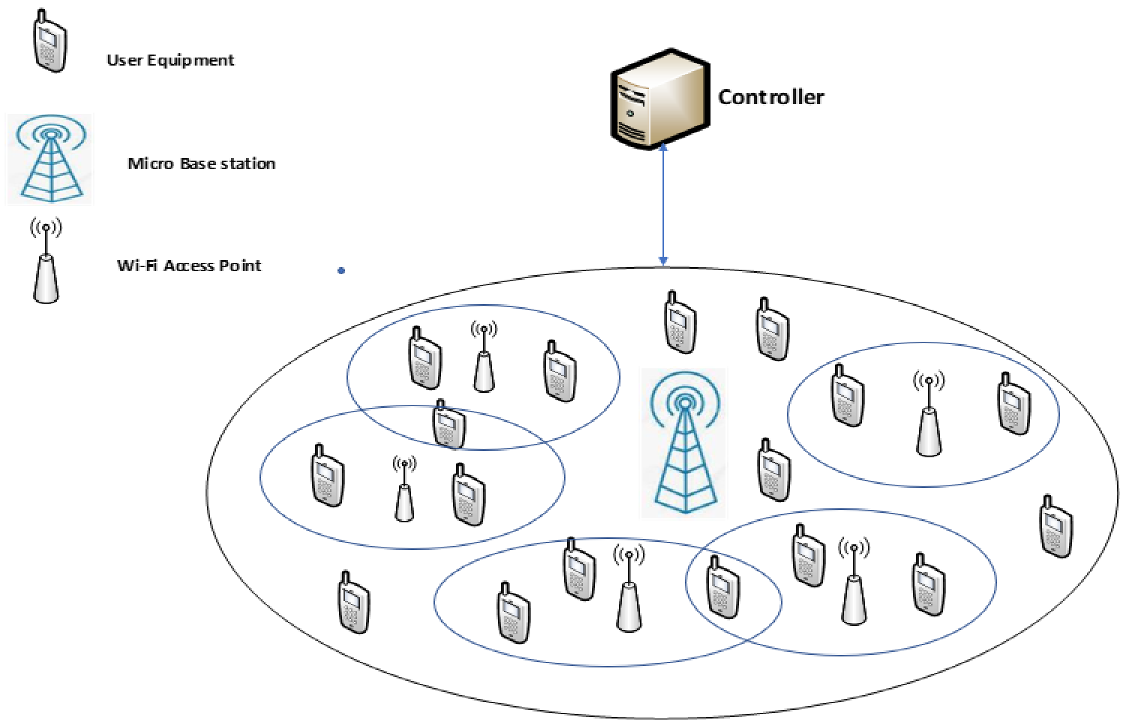


Figure 3.1: Proposed System Model

The controller is responsible for various functions. Firstly, it is responsible for collecting the signal strength received by the devices from each WiFi AP, and the number of users that are connected to each AP and base station. Secondly, it is responsible for running the offloading and load balancing algorithm, which makes offloading and load-balancing decisions among under-loaded APs and loaded AP. Lastly, it is responsible for associating users with either micro cell or Wi-Fi Ap. The offloading procedure is outlined in the following steps.

Step 1

The Wi-Fi APs and the micro-base station periodically send information about the number of users they are associated with and the signal strength of each of the associated devices, which is indicated by the RSSI, to the controller. The periodic information sent by the Aps and micro-base station gives the controller a global view of the entire network. Thus, the controller has a list (which is updated constantly) of APs along with the number of users per AP and their signal strength, which enables it to make offloading and load-balancing decisions.

Step 2

When a new user joins the network and requests for service, the Wi-Fi network is given priority. The goal is to reduce cellular network utilization by first determining if a Wi-Fi AP is accessible to which a mobile user may connect, hence alleviating cellular network congestion. The algorithm assesses each Wi-Fi AP's state and makes an association decision based on signal strength and the number of devices connected. The proposed method will first determine if the RSSI matches the minimal threshold criterion. The number of connected users is next checked to see whether it is fewer than the minimal number of users that each AP should support. New users will be connected to any Wi-Fi AP with the least number of users and the acceptable signal strength. If no Wi-Fi AP meets the minimal requirements, the controller checks to see if the number of users connected to the base station is fewer than the threshold; if it is, the user is assigned to the micro-base station. The user will be refused service if the microcell base station is unable to accommodate its request.

In the following scenarios, the load balancing procedure is explained.

Scenario A

The system handles load balancing in addition to offloading. Consider the case depicted in Figure. 3.2, with two APs, AP1 and AP2, having overlapping coverage. The number of users associated with each AP is shown in the Figure 3.2. There are six users connecting to the APs: usr1, usr2, usr3, usr4, usr5, and usr6. Three users are initially connected to each of the AP. Usr2, and usr3 that are connected to AP1 decide to terminate their sessions. Assuming that all these users are active and generate equal amounts of traffic; AP2 is considered overloaded in comparison to AP1 based on this assumption and the number of stations connected to each AP. To balance the number of stations on both APs, the controller decides to disconnect one of the stations from AP2 and transfer it to AP1. The result of AP2 handing over one user to AP1 is shown in Figure 3.3.

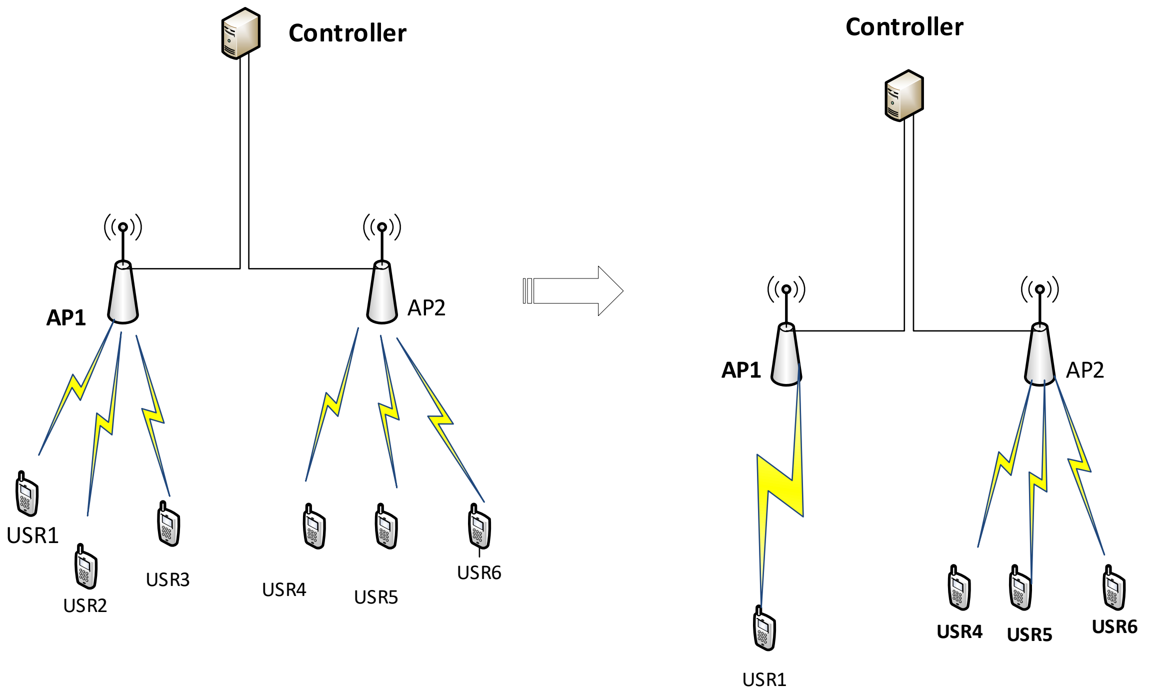


Figure 3.2: Two Users on AP1 Ended their Sessions

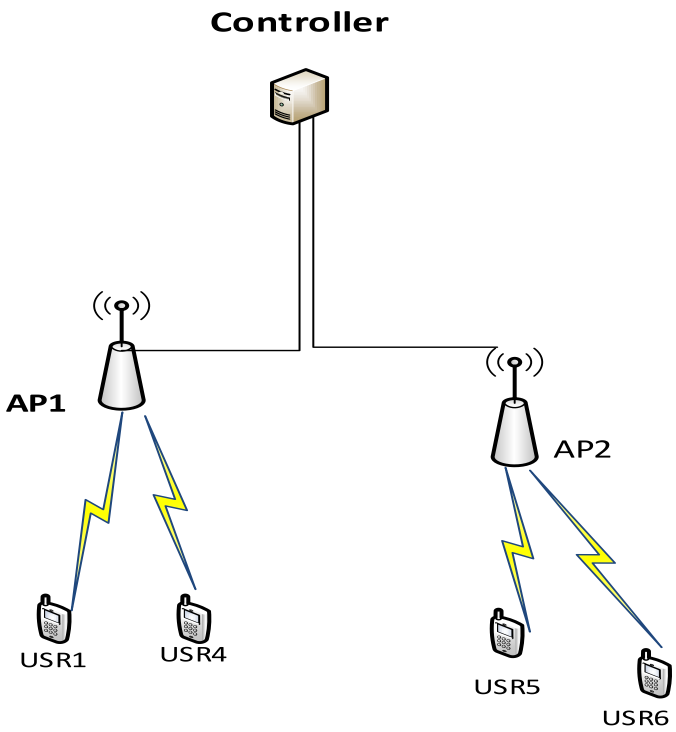


Figure 3.3: Controller Moves One User (USR4) from AP2 to AP1

Scenario B

Consider another scenario, as illustrated in Figure 3.4, with two APs, AP1 and AP2. There are four users connected to the APs: usr1, usr2, usr3, and usr4. The two APs have overlapping area. Three users are initially connected to AP2, whereas one user is connected to AP1. If all users generate the same amount of traffic and all users are active, AP2 is deemed highly loaded in comparison to AP1. If usr2 moves into the overlapping zone of the two APs, the controller will reassign usr2 to AP1. The result is shown in Figure 3.5.

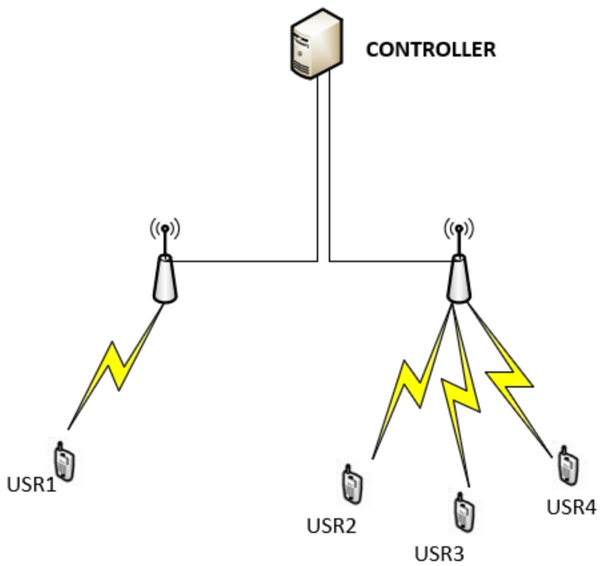


Figure 3.4: Users Connected to AP1 and AP2

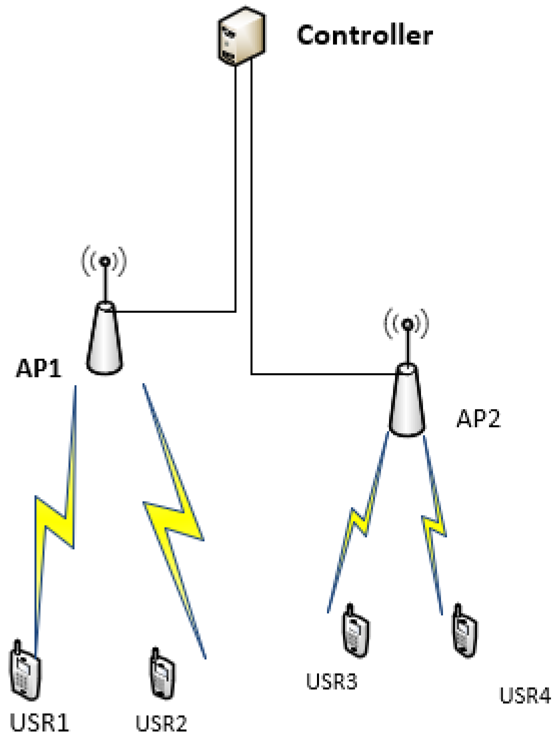


Figure 3.5: Controller Re-Assign Usr2 to AP1

The flowchart of the algorithm is shown in Figure 3.6.

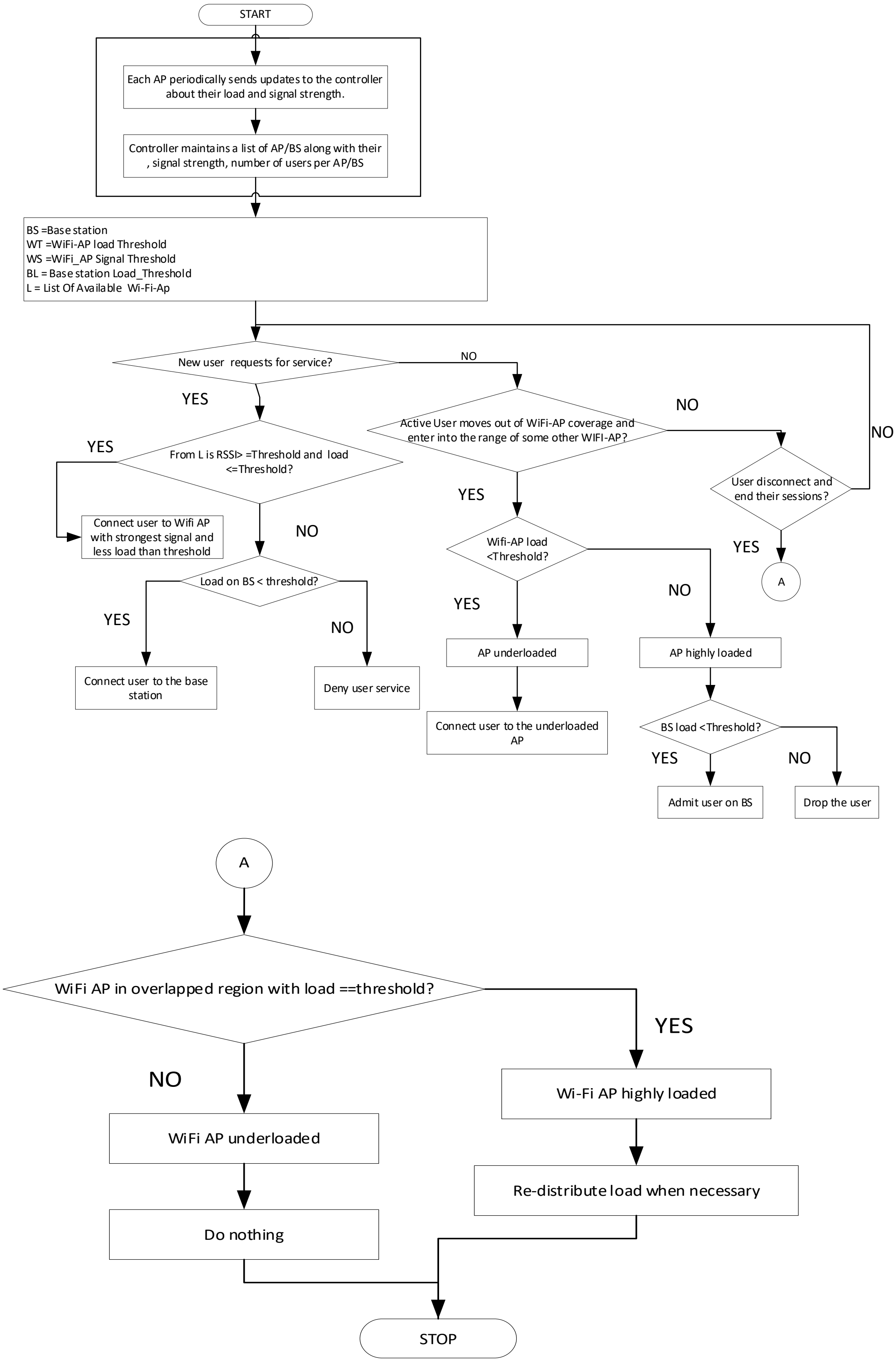


Figure 3.6: Flowchart of the Offloading and Load-Balancing Algorithm.

3.5 Chapter Summary

In cellular networks, efficient congestion control management is a critical concern due to rising data traffic. This chapter discussed an SDN-based scheme for data offloading and load balancing between microcell base stations and Wi-Fi. The SDN idea was introduced to make network control easier. The proposed scheme uses the controller's global view of the network to make more informed decisions about data offloading and load balancing. The system model and detailed information on the operation of the scheme were presented in the chapter. The next chapter explains how the offloading and load-balancing scheme was realized.

CHAPTER 4

4 IMPLEMENTATIONS OF THE PROPOSED ALGORITHM

4.1 Introduction

This chapter details the implementation of the proposed SDN-based data offloading and load-balancing algorithm. The algorithm is intended for data offloading from cellular networks to Wi-Fi networks, as well as load balancing between WLAN and cellular networks. The Mininet-Wi-Fi emulator has been used to implement the proposed system. The chapter is structured as follows. The software tools used for the implementation of the scheme are discussed in detail in Section 4.2. Section 4.3 describes the design and implement of the network (using the tools mentioned in Section 4.2). Finally, Section 4,4 contains the summary of the chapter.

4.2 Associated Software and Technologies

This section describes the software platforms and technologies used to evaluate the SDN-based data offloading and load balancing scheme. The Mininet-Wi-Fi emulator is the main network tool used. The advantage of using the Mininet WiFi is that it mimics the real-world network situation.

4.2.1 Virtual Box

VirtualBox is an Oracle-supported platform used as a virtual testbed for research. It is intended to improve academic and corporate research without the need for actual hardware. On a Linux operating system (OS), the proposed SDN-based data offloading and load balancing solution was implemented. VirtualBox 5.2 was used in this dissertation.

4.2.2 Mininet-WiFi

Mininet-WiFi arose as a Mininet dependency, an OpenFlow/SDN emulator capable of adding virtualized stations and access points using normal Linux wireless drivers and the 802.11 hwsim wireless simulation driver. The SDN paradigm is used in this emulator. It enables network administrators to specify network behaviour in a logically controlled manner using a controller that communicates with forwarding devices via the OpenFlow protocol [36].

4.2.2.1 Installation of Mininet -WiFi

Step 1: Down the Virtual Machine Mininet-WiFi

First and foremost, download the virtual machine. For this research, Lubuntu virtual machine with preinstalled Mininet –Wi-Fi was downloaded.

Step 2: Import the Virtual Machine into Virtual Box

Next, import the Lubuntu Mininet virtual machine into the VirtualBox application to produce a version of the Mininet-Wi-Fi virtual machine in VirtualBox. Figure 4.1 shows the Mininet Wi-Fi virtual machine.

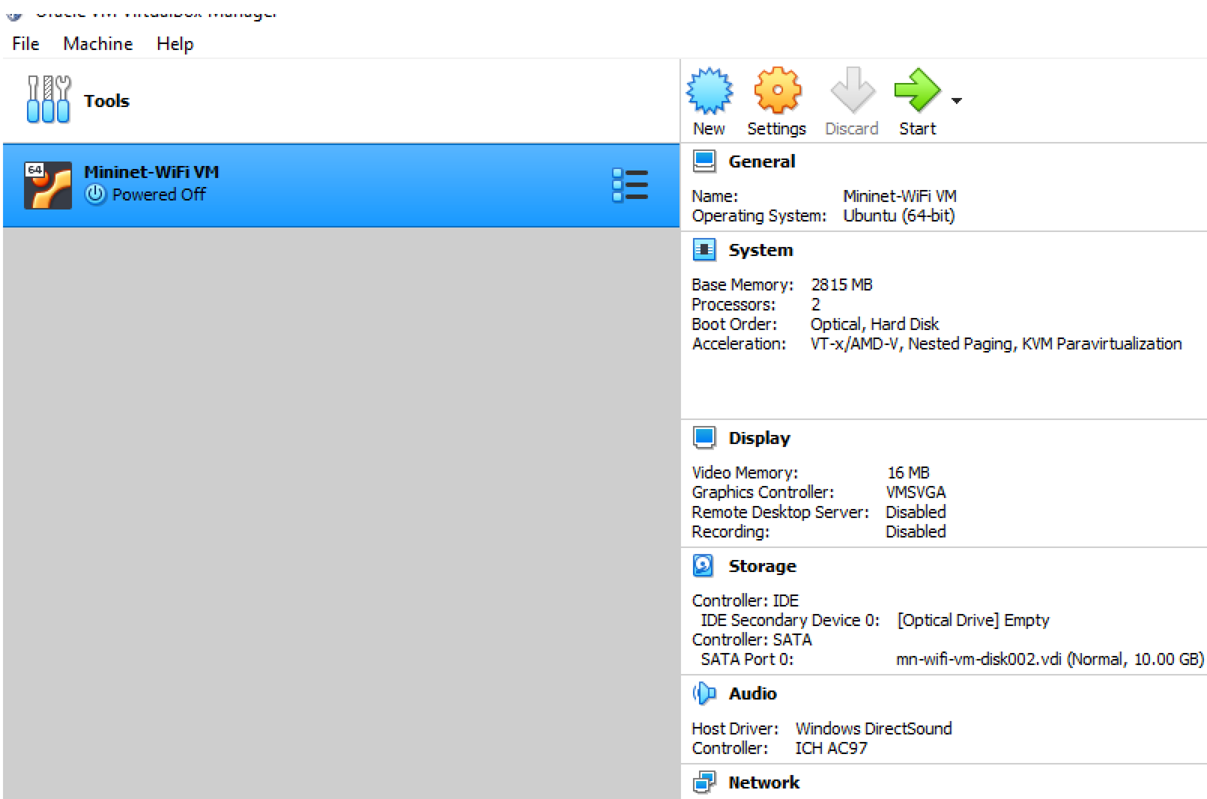


Figure 4.1: Importing Mininet-WiFi into Virtual Box

4.2.2.2 Mininet-WiFi Python API

Mininet may be used with Python-based user programs. Because of its readable syntax and numerous helpful libraries, Python is one of the easiest computer languages to grasp, learn, and use [37]. Mininet-Wi-Fi Python APIs also make it easier to add components to Mininet-WiFi and build network topologies. To mimic the SDN-based data offloading and load balancing system, python-based scripts were written to simulate the scheme.

4.2.2.3 Mininet-WiFi Limitation

Although Mininet-WiFi is great, it does have some limitations. Mininet uses a single Linux kernel for all virtual hosts; this means that a user cannot run software that depends on BSD, Windows, or other operating system kernels. [38].

4.2.3 POX Controller

The control plane of a networking device (switch/router) is separated from the data plane by SDN. This allows a network to be controlled, monitored, and managed from a single location. By separating the data plane from the control plane, Software Defined Networking aims to make the creation of new applications easier [39].

4.2.3.1 POX Components

POX components are Python applications that may be run from the command line when POX is started. In a software defined network, these components implement network functions. POX comes with several pre-installed stock components [39].

The code for each POX stock component may be found in the /pox/pox directory on the Mininet 2.2 VM image, which is detailed in the POX Wiki. However, the POX part of this dissertation has not been realized because Mininet Wi-Fi has modules such as handover and mobility that were extended to accomplish the dissertation's goal.

4.3 Design and Implementation

The design and implementation of the proposed model is discussed below.

4.3.1 Network Setup

The Mininet WiFi was used to evaluate the performance of the proposed offload and load-balancing scheme. The Mininet-WiFi is a wireless OpenFlow/SDN emulator that enables high-fidelity experiments by replicating real-world networking environments. The Mininet-WiFi includes virtualized Access Points (APs) and stations (sta) based on the mac80211/SoftMac Linux wireless device driver, allowing for Wi-Fi emulation. In this dissertation, topologies were generated using Python scripts, which were then used to test the scheme. The handover and mobility modules in Mininet Wi-Fi were used to test the concept. The class addAccessPoint() were used to add Wi-Fi access points to the network.

4.3.2 Load Balancing and Data offloading Algorithm Evaluation

For the experiments, scripts were written in Mininet-WiFi to create network topologies containing stations, access points, base station, and a remote controller for the experiments.

4.4 Chapter Summary

This chapter discussed the design and implementation of the proposed data offloading and load balancing scheme. It described the software platform used for evaluating the performance of proposed scheme. Lubuntu with pre-installed Mininet-WiFi was installed on a VirtualBox-hosted Virtual Machine.

CHAPTER 5

5 PERFORMANCE EVALUATION AND RESULTS

5.1 Introduction

This chapter discusses the results of the experiments described in the previous chapter. A series of experiments were conducted to evaluate the performance of the proposed data offloading and load-balancing scheme in cellular and WLAN networks.

5.2 Network Set up in Mininet-WiFi

As indicated in Chapter 4, the experiments were conducted using the Mininet-WiFi. In the experiments, one microcell base station, two Wi-Fi access points, nine mobile stations, and one controller were used, and the network dimension was set to 700 m x 700 m. The simulation parameters are listed in Table 5.1.

Table 5.1: Simulation Parameters

Parameters	**Description**
Number of Wi-Fi Access Points	2
Number of Micro Base station	1
Number of Users	9
Number of Controllers	1

Table 5.2 shows the specifications for the cellular base station and Wi-Fi access points while Table 5.3 shows the specifications for mobile stations (stas).

Table 5.2: Micro Cell and Wi-Fi Specification

SSID	**Range**	**Mode**	**Frequency**	**Position**
Ap1-ssid	100 m	g	2.4 Ghz	‘260,350,0’
Ap2-ssid (base station)	300m	g	2.4 Ghz	‘350,350,0’
Ap3-ssid	100m	g	2.4 Ghz	‘440,350,0’

Table 5.3: Specifications for Mobile Stations

Stations/users	IP Address	Interface	range
Sta1	10.0.0.1	WLAN 0	40m
Sta2	10.0.0.2	WLAN 0	40m
Sta3	10.0.0.3	WLAN 0	40m
Sta4	10.0.0.4	WLAN 0	40m
Sta5	10.0.0.5	WLAN 0	40m
.			
.			
.			
.			
Sta9	10.0.0.9	WLAN 0	40m

5.3 Evaluation Metrics

This section discusses the metrics used in the evaluation of the data offloading and load balancing algorithms.

5.3.1 Throughput

TCP provides a congestion control mechanism. An important measure of the performance of a TCP connection is its throughput—the rate at which it transmits data from the sender to the receiver. Throughput is measured in kbps, Mbps, or Gbps. In the simulation, TCP throughput per user was measured by performing iperf test between a host and another host in a client and server mode.

5.3.2 Load Distribution Measurement

In the simulation, load balancing was evaluated based on the distribution of users among access points.

5.4 Evaluation Scenarios

The performance of the proposed scheme was evaluated considering different scenarios as described below.

5.4.1 Performance Evaluation of Data offloading and Load Balancing

Experiment 1: Connection Scenario Considering Signal Strength Only

This experiment was conducted to measure the stations' throughput while they were connected to the highly loaded base station. The experiment comprised two Wi-Fi APs (AP1 and AP3), one base station (AP2), and nine stations. The stations were positioned within the range of the base station. Thus, all the stations were connected to the base station, as shown in Figure 5.1. None of the stations connected to AP1 or AP3. Figures 5.2 to 5.5 show the connection of sampled stations to the base station (AP2). Iperf test was performed to measure throughput per user between a host/user and another host/user in a client and a server mode. The TCP traffic between each pair of sampled stations, sta1 and sta9, and sta3 and sta7 was investigated. On one station, the command *'iperf -s -i 1 -p 5001'* was used to make it behave as a server and listen to another station (client). On another station, the *'iperf -c IP Address -p 5001 -t 10'* command was used to send data to the server. Figures 5.6 to 5.9 show the throughput of stations while connected to a highly loaded base station.

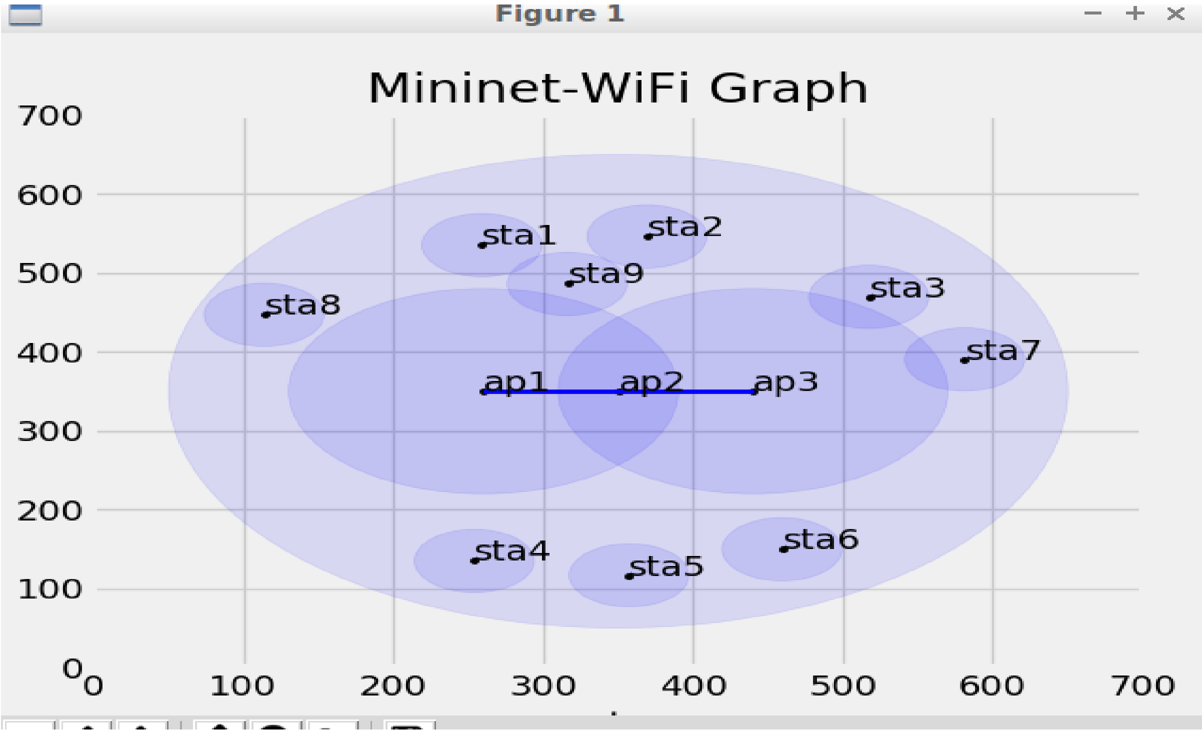


Figure 5.1: All Stations Connected to AP2 (BS)

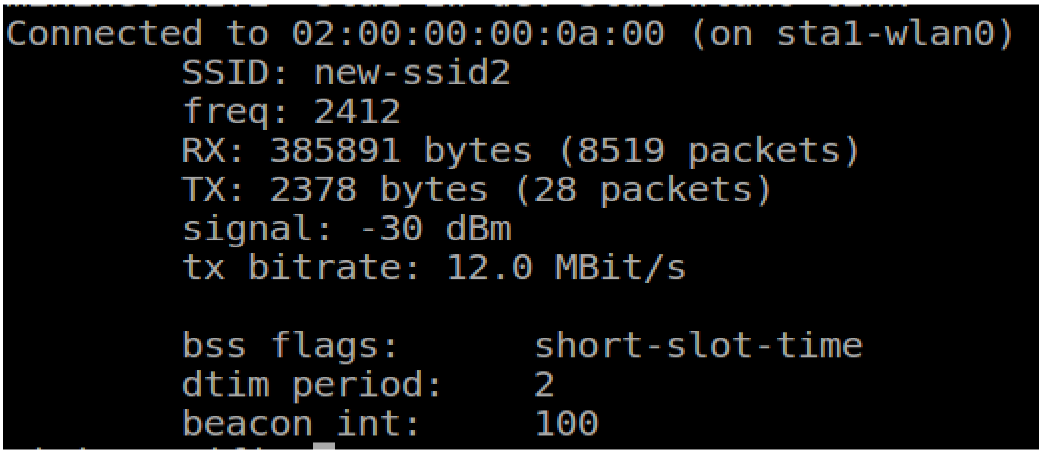


Figure 5.2: Association of sta1 to the Base Station

```
Connected to 02:00:00:00:0a:00 (on sta3-wlan0)
        SSID: new-ssid2
        freq: 2412
        RX: 182936 bytes (3969 packets)
        TX: 1737 bytes (20 packets)
        signal: -30 dBm
        tx bitrate: 24.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon_int:     100
```

Figure 5.3: Association of sta3 to the Base Station

```
mininet-wifi> sta7 iw dev sta7-wlan0 link
Connected to 02:00:00:00:0a:00 (on sta7-wlan0)
        SSID: new-ssid2
        freq: 2412
        RX: 235968 bytes (5167 packets)
        TX: 1983 bytes (23 packets)
        signal: -30 dBm
        tx bitrate: 48.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon_int:     100
```

Figure 5.4: Association of sta7 to the Base Station

```
Connected to 02:00:00:00:0a:00 (on sta9-wlan0)
        SSID: new-ssid2
        freq: 2412
        RX: 337792 bytes (7452 packets)
        TX: 2300 bytes (26 packets)
        signal: -30 dBm
        tx bitrate: 5.5 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon_int:     100
```

Figure 5.5: Association of sta9 to the Base Station

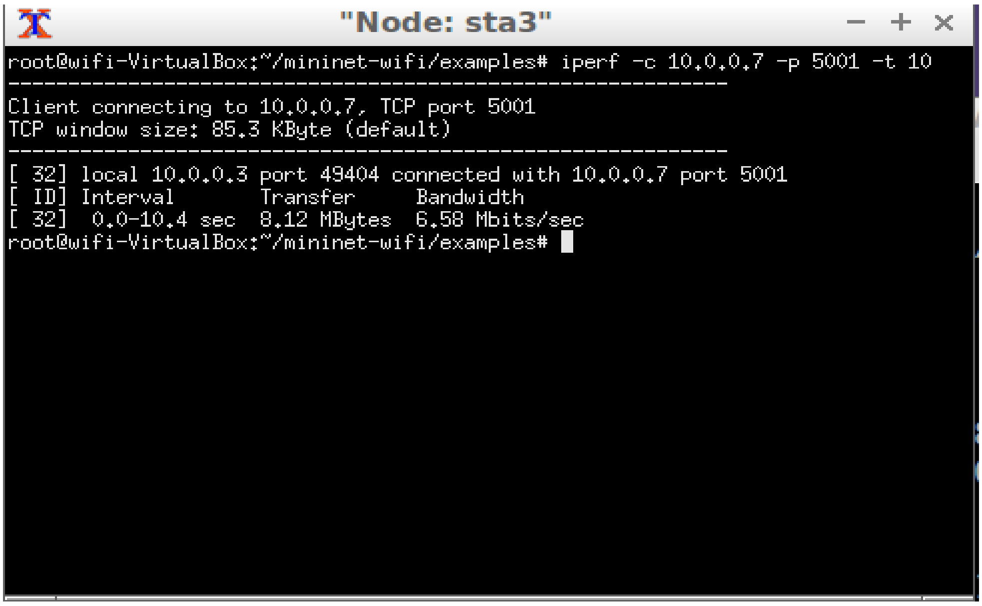


Figure 5.6: sta3 as a client

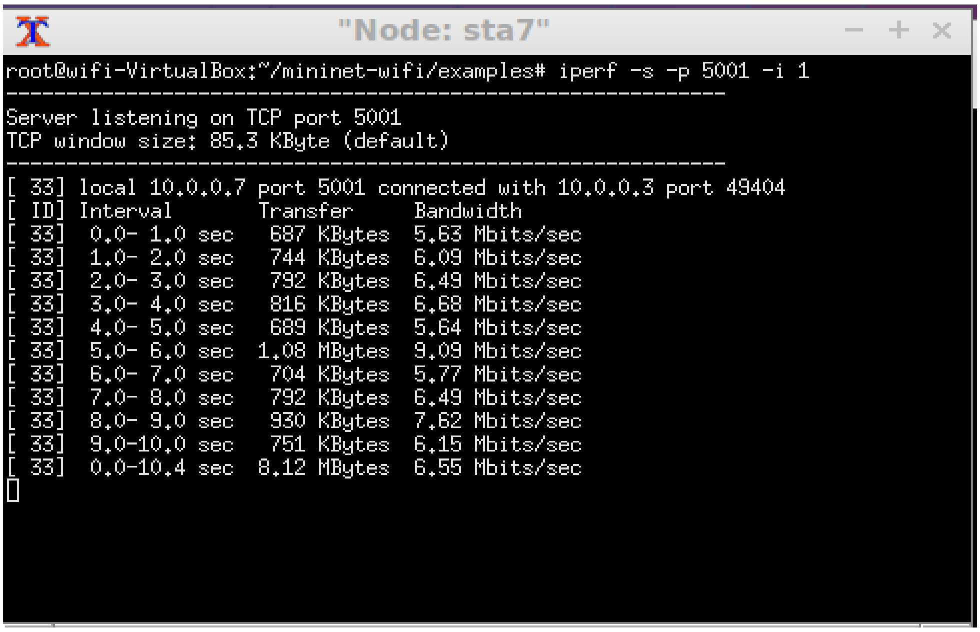


Figure 5.7: sta7 as a server

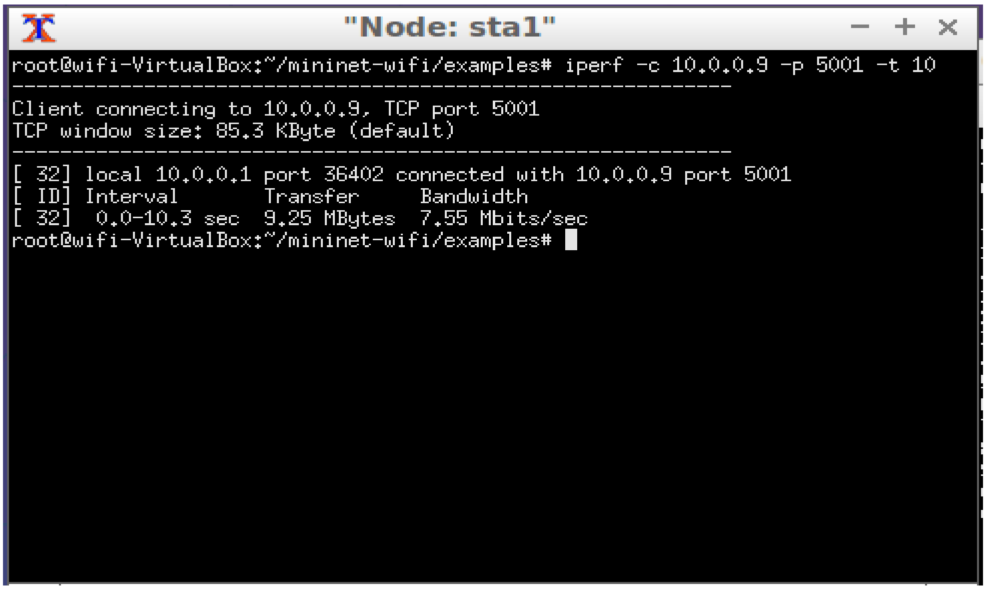


Figure 5.8: sta1 as a client

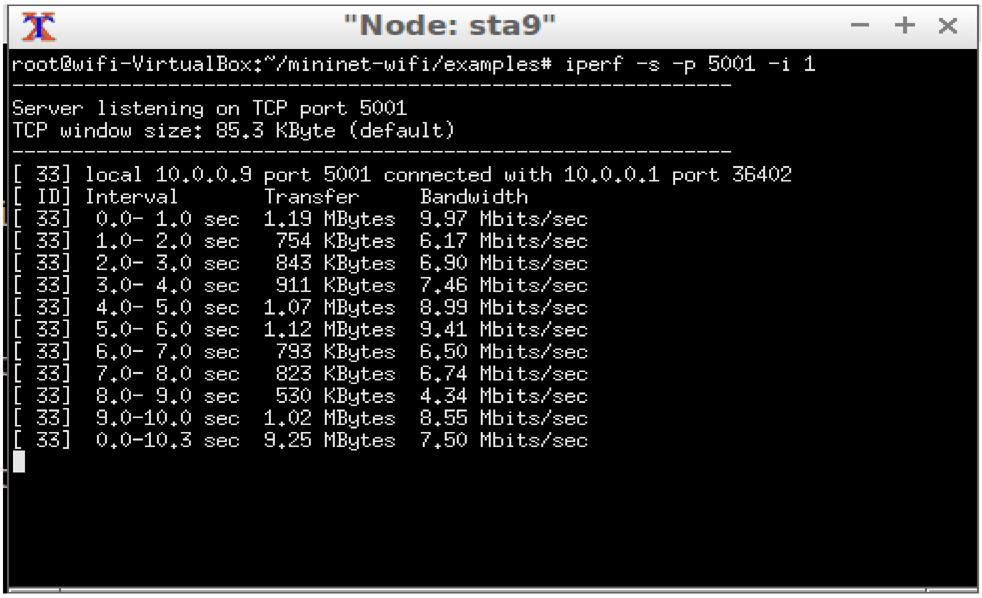


Figure 5.9: sta9 as a server

Experiment 2: **Connection Scenario Considering Signal Strength and Load per AP**

This experiment measured throughput in a scenario where stations were connected to the base station (AP2) and the APs (AP1 and AP3) based on the signal strength and load (number of stations) per AP. As shown in figure 5.10, stations sta1, sta3, sta6, sta7, sta8, and sta9 roamed into the coverage of AP1 and AP3 and were associated with them since they had the best signal strength and the least load. Mobility starting and ending points were set in such a way that the stations moved towards those APs and connected to them. Sta3, sta6, and sta7 connected to AP3, sta1, sta8, and sta9 connected to AP1, and sta2, sta4, and sta5 connected to the base station. Throughput for sampled stations sta1 and sta9, and sta3 and sta7 was investigated. Figures 5.11 and 5.12, depict the association of stations sta3, and sta7 to AP3, where the transmitted bitrate was much lower than the expected maximum value of 54 Mbps. Figures 5.13 and 5.14 show the results of testing the throughput between test stations sta3 and sta7 that are associated with AP3. Figures 5.15 and 5.16 show the association of sta1, sta9, to AP1.The results of the AP1 throughput test between sta1 and sta9 are presented in Figures 5.17 and 5.18.

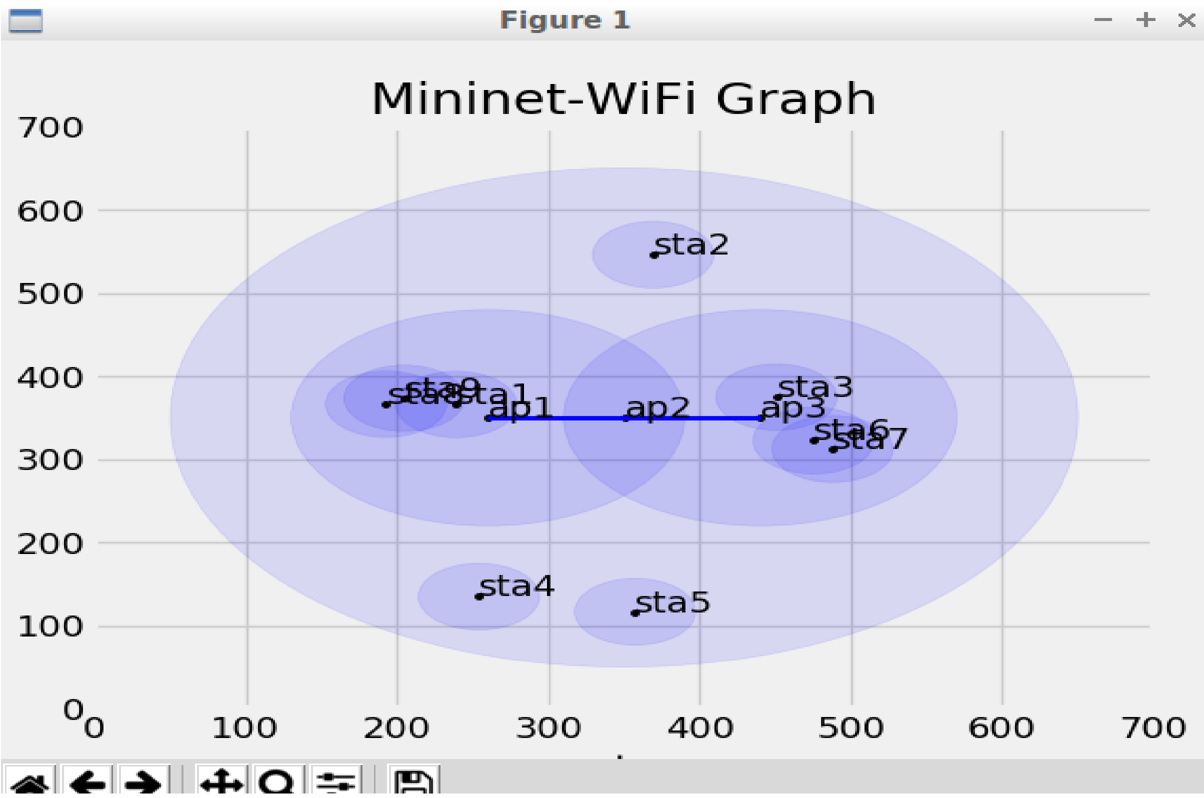


Figure 5.10: Connection Considering Number of Users and Signal Strength

```
Connected to 02:00:00:00:0b:00 (on sta3-wlan0)
        SSID: new-ssid3
        freq: 2412
        RX: 113617 bytes (2549 packets)
        TX: 627 bytes (8 packets)
        signal: -30 dBm
        tx bitrate: 1.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon_int:     100
```

Figure 5.11: sta3 Association with AP3

```
Connected to 02:00:00:00:0b:00 (on sta7-wlan0)
        SSID: new-ssid3
        freq: 2412
        RX: 122640 bytes (2745 packets)
        TX: 627 bytes (8 packets)
        signal: -30 dBm
        tx bitrate: 1.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon_int:     100
```

Figure 5.12: sta7 Association with AP3

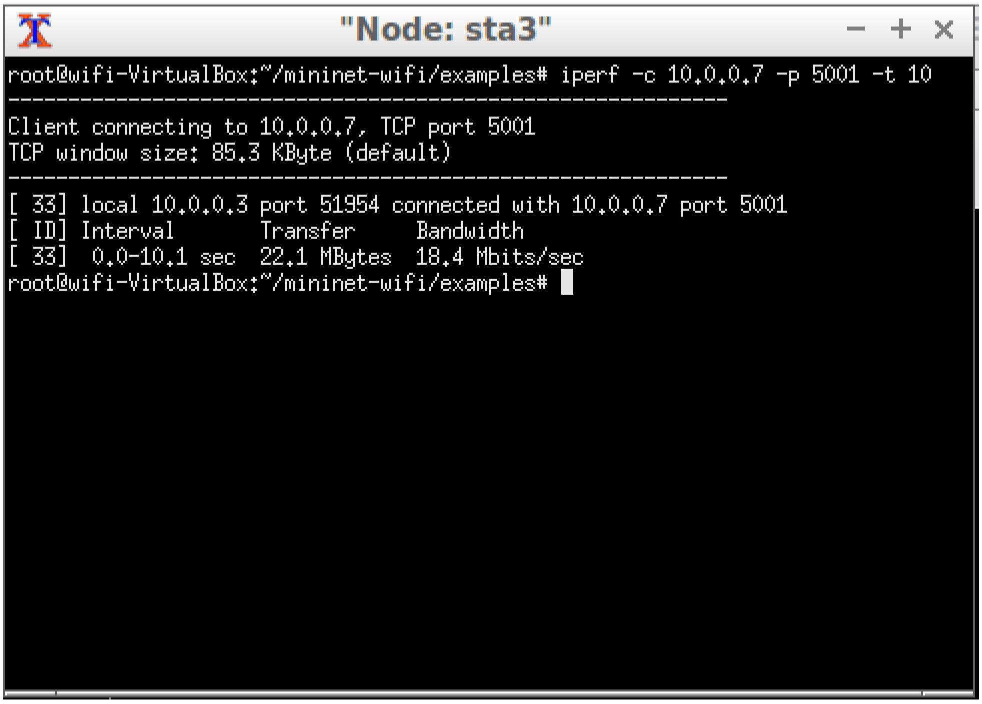


Figure 5.13: sta3 as a client

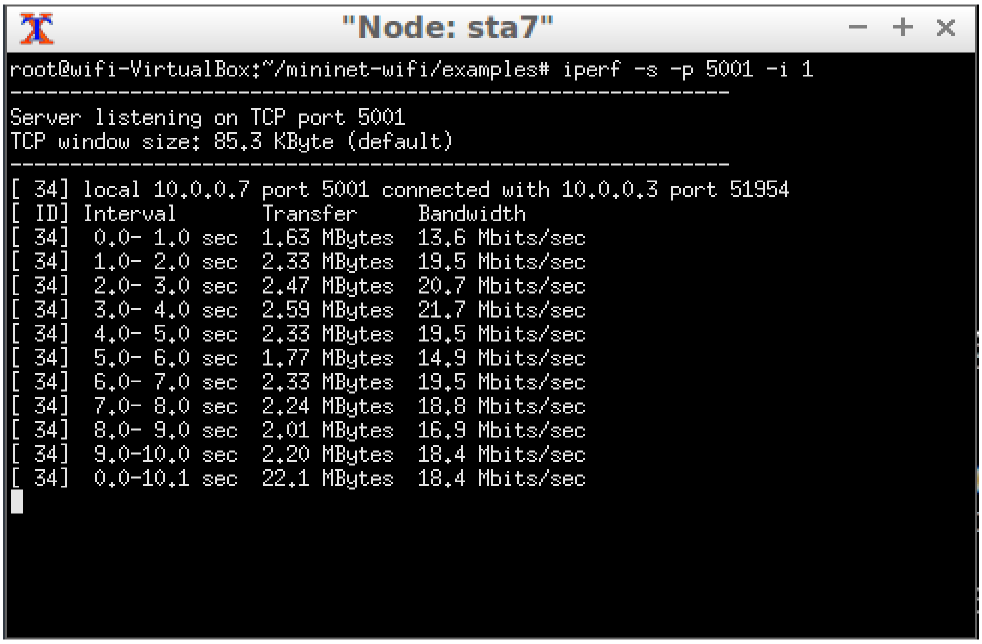


Figure 5.14: sta7 as a server

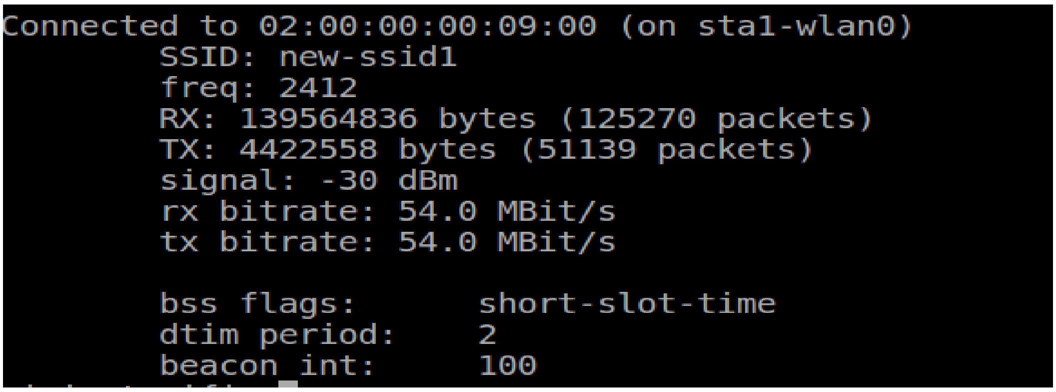


Figure 5.15: Association of sta1 with AP1

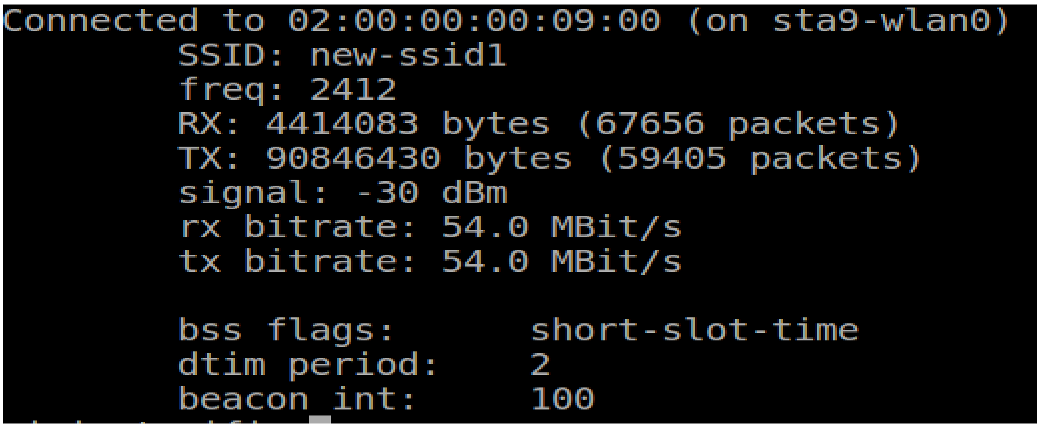


Figure 5.16: Association of sta9 with AP1

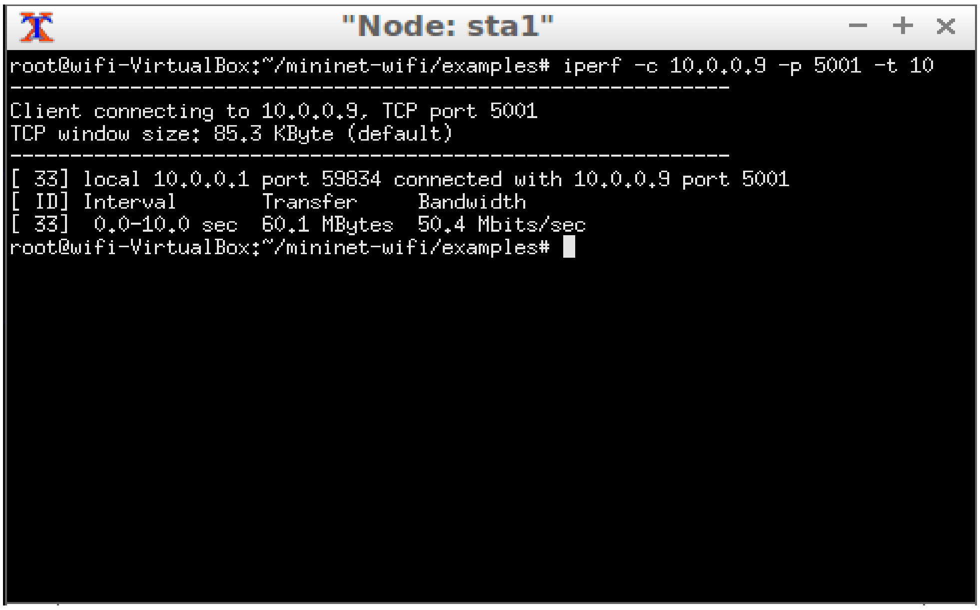


Figure 5.17: sta1 as a client

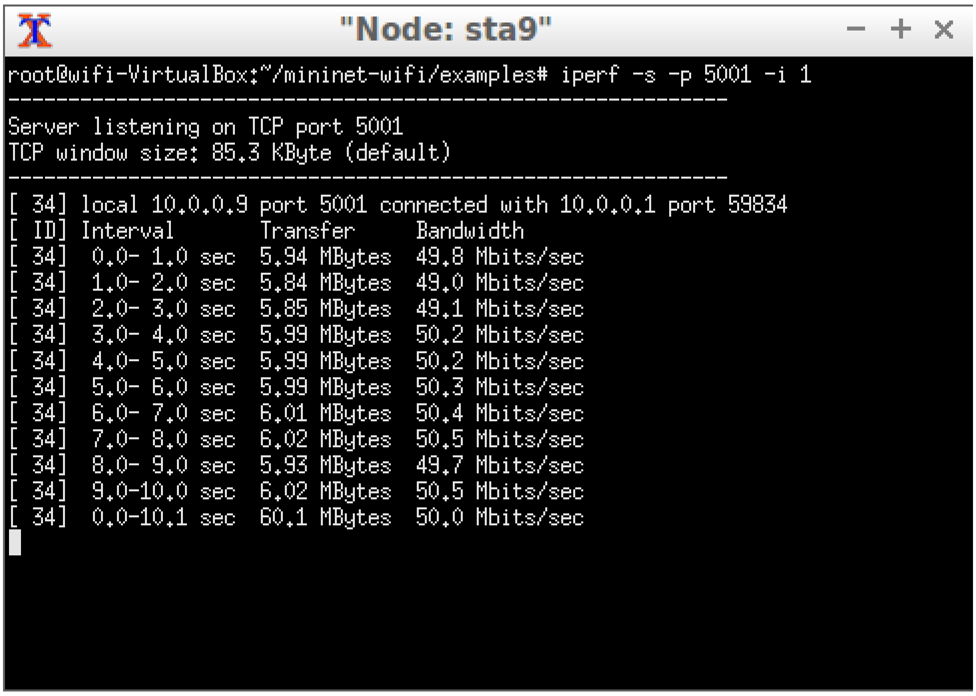


Figure 5.18: sta9 as a server

5.4.1.1 Performance Analysis of Experiment 1 and Experiment 2 with the Number of Connected Devices

Figure 5.19 depicts the number of stations associated with each access point and base station when the stations were connected to the base station based on signal strength. The graph illustrates that all the stations were connected to the base station since they were in the range of the base station, and it provided the best signal strength. Figure 5.20 depicts the results of stations that were connected to APs based on signal strength and load on each AP. When the load per AP was considered, the load was more evenly distributed than when stations were connected only based on signal strength.

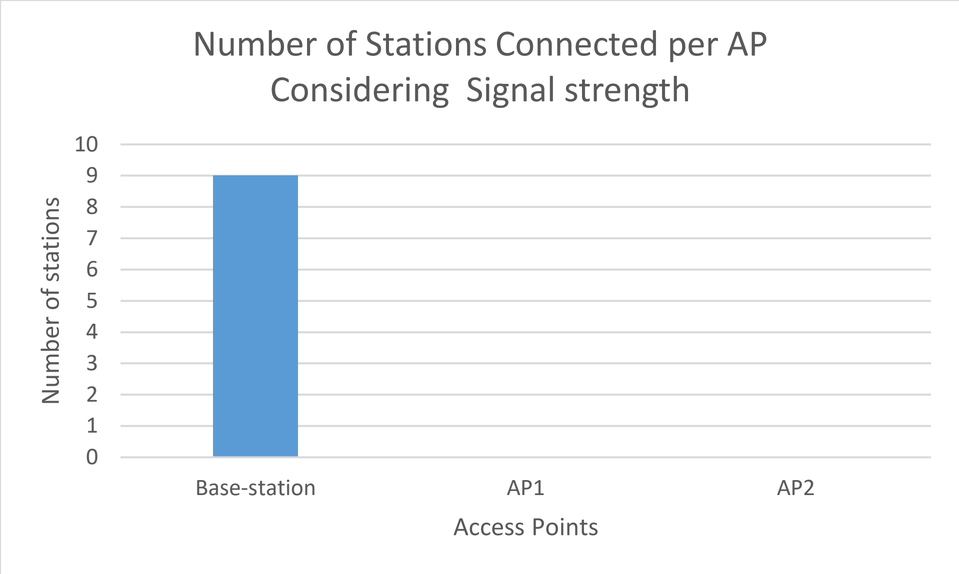


Figure 5.19: Number of Connected Devices Based on Signal Strength

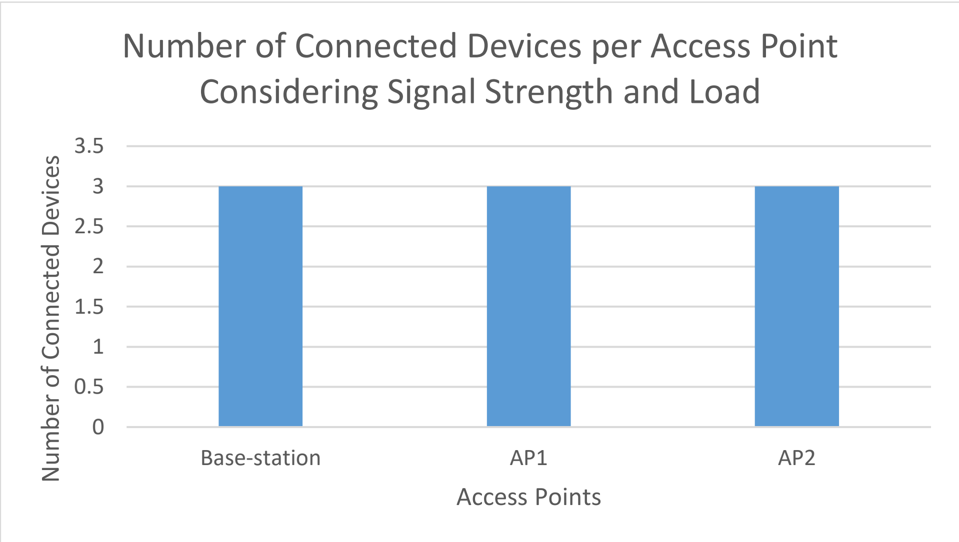


Figure 5.20: Number of Connected Devices Based on Signal Strength and Load

5.4.1.2 Performance Analysis of Experiment 1 and Experiment 2 with Throughput per User

The figures below compare the throughput of test stations sta4, sta7, and sta9 while connected to a highly loaded base station with when they were offloaded to the least loaded APs. Figure 5.21 compares the throughput of sta7 when connected to a highly loaded base station to after it was offloaded to a Wi-Fi AP3 with the least number of stations. When connected to the base station, sta7's throughput was 6.58 Mbits/sec, but it increased to 18.4 Mbits/sec when connected to the least loaded AP3. Figure 5.22 indicates that while connected to AP1, sta9's throughput increased from 7.55 Mbits/s to 50.5 Mbits/s. Throughput increased when some stations were offloaded to the APs with the lowest number of users.

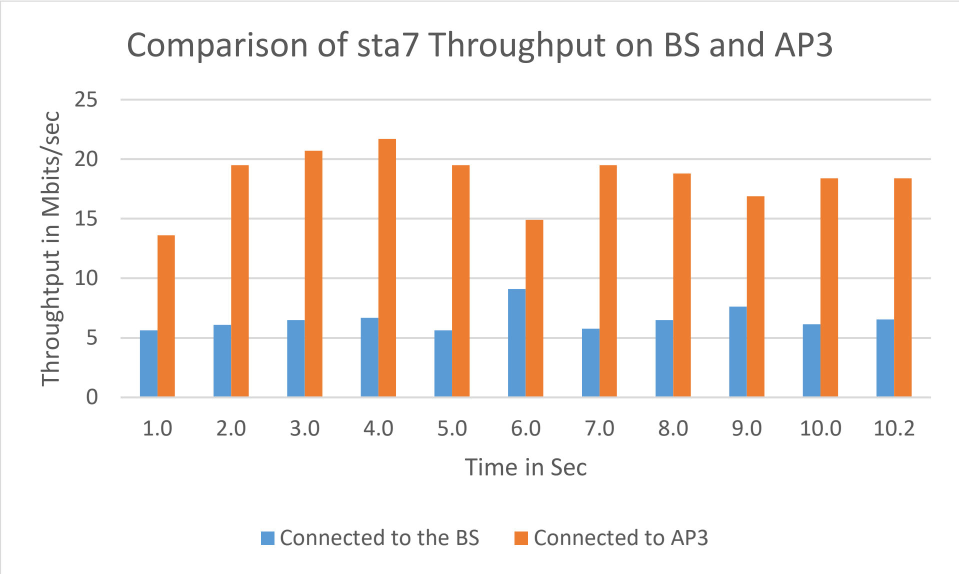


Figure 5.21: Throughput for sta7 Before and After offloading

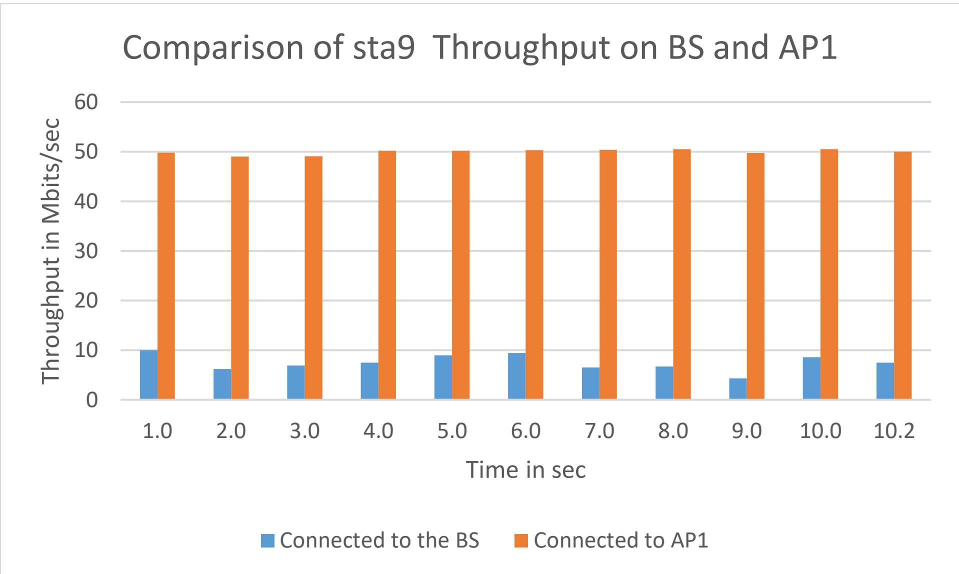


Figure 5.22: Throughput for sta9 Before and After Offloading

5.4.2 Performance Evaluation of Load Re-distribution Scenarios

Load re-distribution scenarios are presented in this section. A network with two access points, one base station (AP2), one controller, and nine stations is shown in Figure 5.23. Three stations connected to the base station and the two access points. Sta1, sta9 and sta8 connected to AP1, sta3, sta6, sta7 connected to AP3, and sta2, sta5 and sta4 connected to the base station (AP2). The aim was to demonstrate a load re-distribution scenario between two access points.

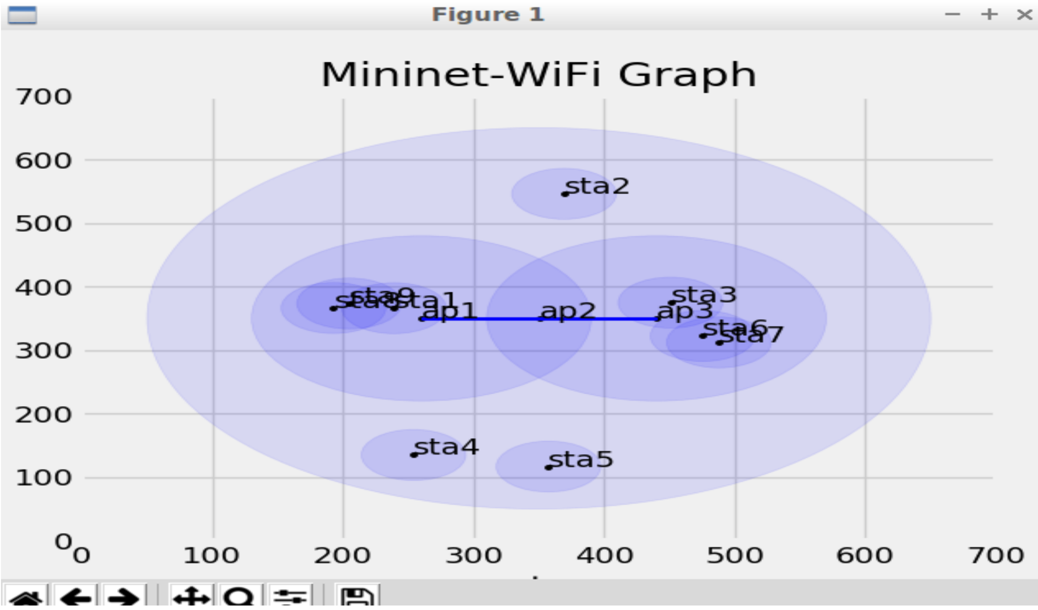


Figure 5.23: Network for Load Distribution Scenario

Experiment 3: Disconnection Scenario

The goal of this experiment was to demonstrate the throughput of stations connected to a highly loaded AP. As shown in Figure 5.24, sta8 and sta9 were first connected to AP1 and later disconnected. As a result, AP3 was highly loaded in comparison to AP1, necessitating network load redistribution. Figures 5.25 and 5.26 illustrate that Iperf tests between sta9 and sta1, and between sta1 and sta8 were unsuccessful because the stations were not connected. Figure 5.27 depicts the association of sta1 with AP1, whereas Figures 5.28, 5.29, and 5.30 depict the association of sta3, sta6, and sta7 with AP3. Throughput between test station sta3 and sta6 connected to AP3 was measured; Figures 5.31 and 5.32 show the throughput results between sta3 and sta6.

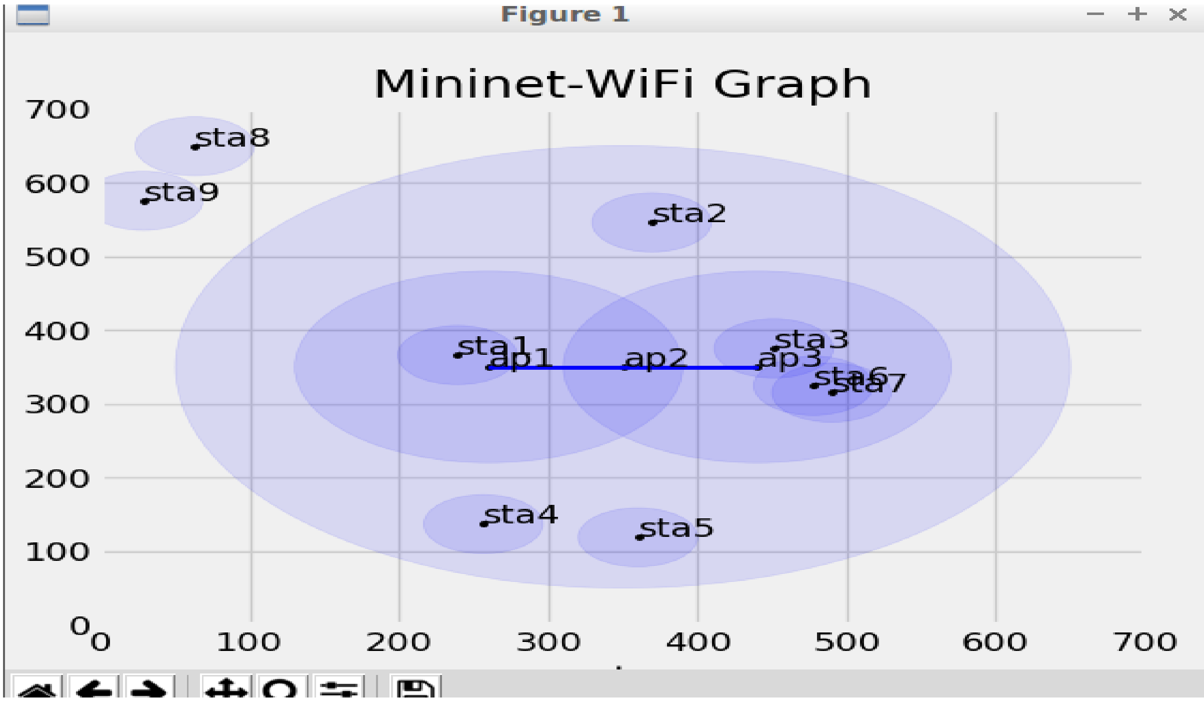


Figure 5.24: sta9 and sta8 Disconnected

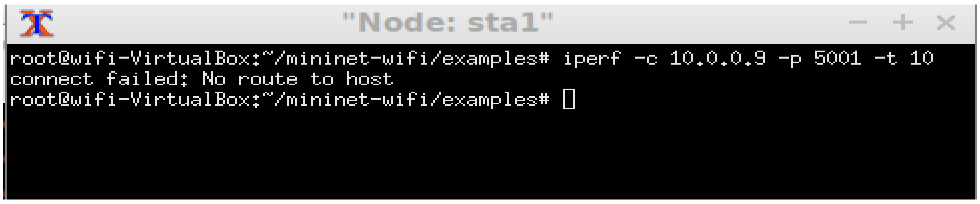


Figure 5.25: Unsuccessful Iperf Between sta1 and sta9

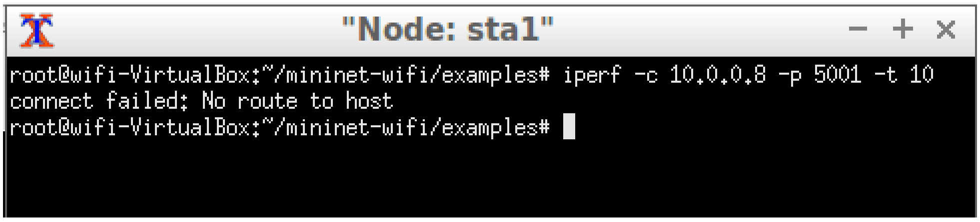


Figure 5.26: Unsuccessful Iperf Between sta1 and sta8

```
mininet-wifi> sta1 iw dev sta1-wlan0 link
Connected to 02:00:00:00:09:00 (on sta1-wlan0)
        SSID: new-ssid1
        freq: 2412
        RX: 77737852 bytes (240592 packets)
        TX: 211193302 bytes (161165 packets)
        signal: -30 dBm
        rx bitrate: 54.0 MBit/s
        tx bitrate: 54.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon int:     100
```

Figure 5.27: Association of sta1 to AP1

```
mininet-wifi> sta3 iw dev sta3-wlan0 link
Connected to 02:00:00:00:0b:00 (on sta3-wlan0)
        SSID: new-ssid3
        freq: 2412
        RX: 218047093 bytes (305092 packets)
        TX: 108141175 bytes (136951 packets)
        signal: -30 dBm
        rx bitrate: 54.0 MBit/s
        tx bitrate: 54.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon int:     100
```

Figure 5.28: Association of sta3 to AP3

```
mininet-wifi> sta7 iw dev sta7-wlan0 link
Connected to 02:00:00:00:0b:00 (on sta7-wlan0)
        SSID: new-ssid3
        freq: 2412
        RX: 1886687 bytes (42394 packets)
        TX: 5771 bytes (73 packets)
        signal: -30 dBm
        tx bitrate: 12.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon int:     100
```

Figure 5.29: Association of sta7 to AP3

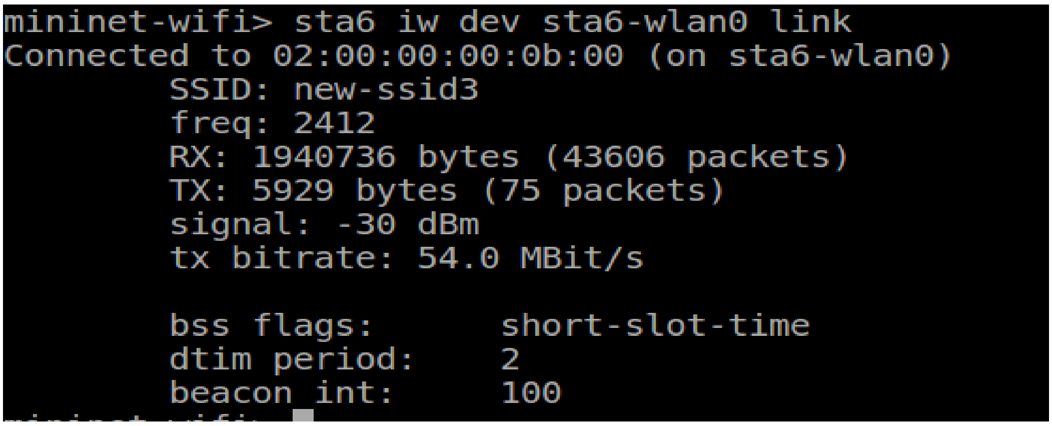


Figure 5.30: Association of sta6 to AP3

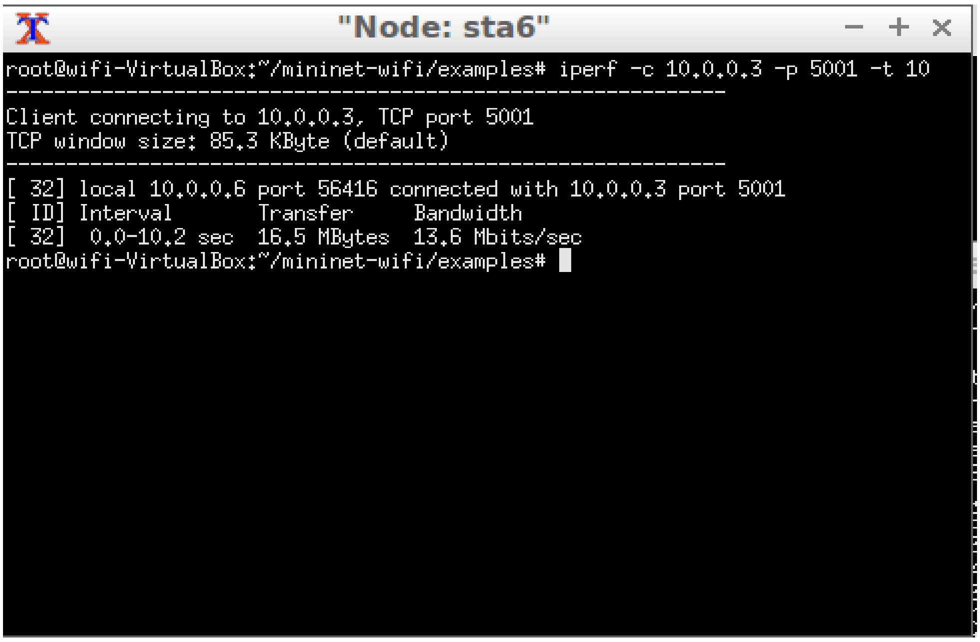


Figure 5.31: Iperf Between sta6 and sta3 with sta6 as client

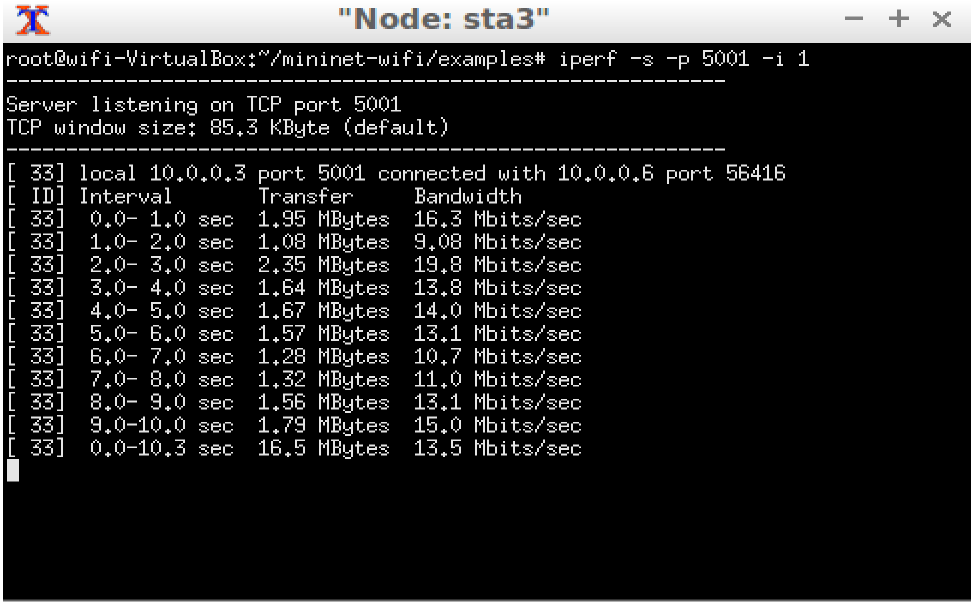


Figure 5.32: Iperf Between sta6 and sta3 with sta3 as a server

Experiment 4: sta3 In Overlapped Region

The throughput of test station sta3 in the overlapping zone of AP1 and AP3 was measured in this experiment. Sta3 roamed into the overlapping region of AP1 and AP3 in Figure 5.33, making it accessible to both AP1 and AP3. It then connected to AP1 since it was the least loaded AP, even though AP3 still provided the best signal strength. Figures 5.34 and 5.35 depict the sta3 and sta1 associations with AP1. As illustrated in Figures 5.36 and 5.37, Iperf was performed between test station sta3 and sta1 to determine the throughput of sta3 when it connected to AP1.

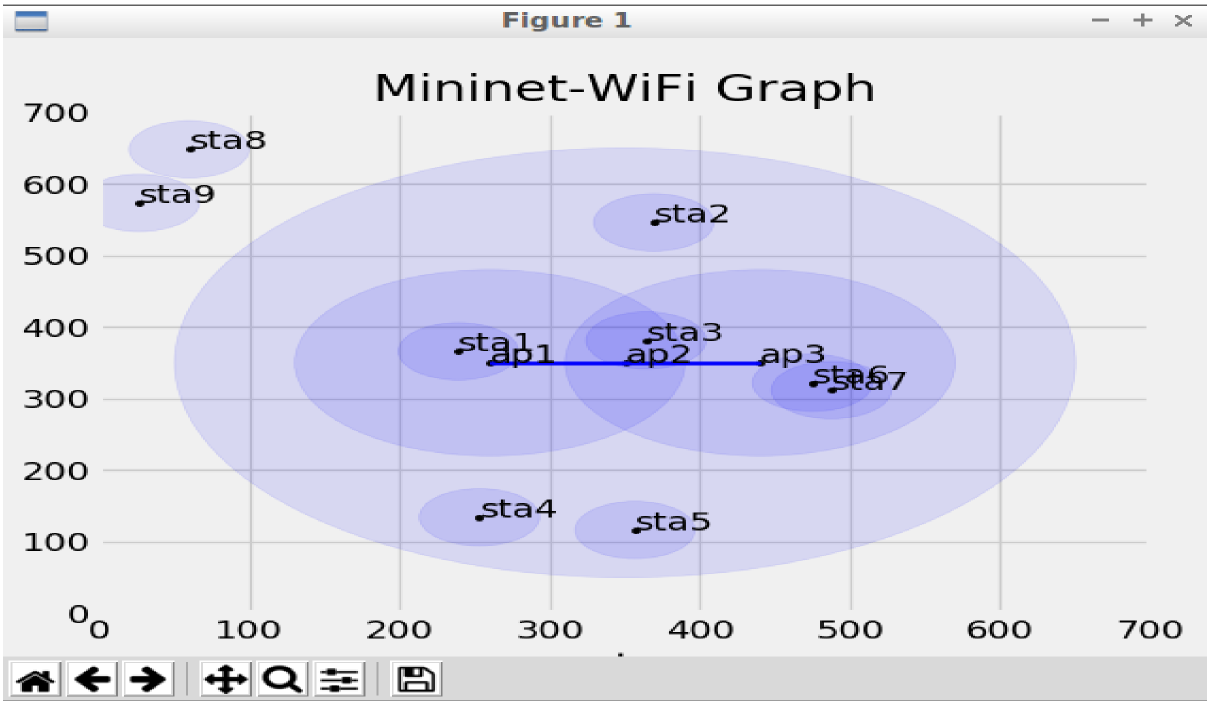


Figure 5.33: sta3 Roamed into Overlapping Area

```
mininet-wifi> sta1 iw dev sta1-wlan0 link
Connected to 02:00:00:00:09:00 (on sta1-wlan0)
        SSID: new-ssid1
        freq: 2412
        RX: 89449 bytes (1909 packets)
        TX: 1333 bytes (15 packets)
        signal: -30 dBm
        tx bitrate: 11.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon int:     100
```

Figure 5.34: Association of sta1 to AP1

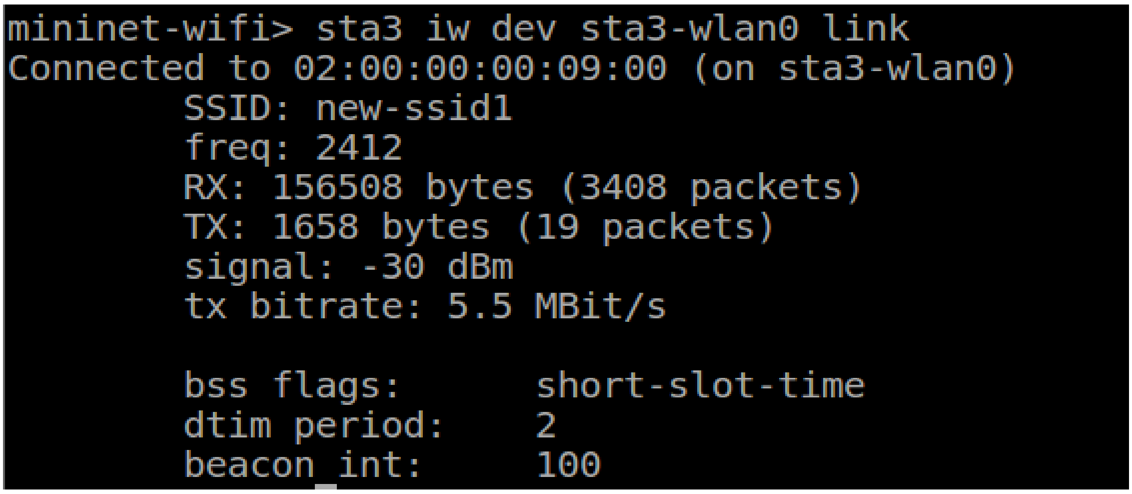


Figure 5.35: Association of sta3 to AP1

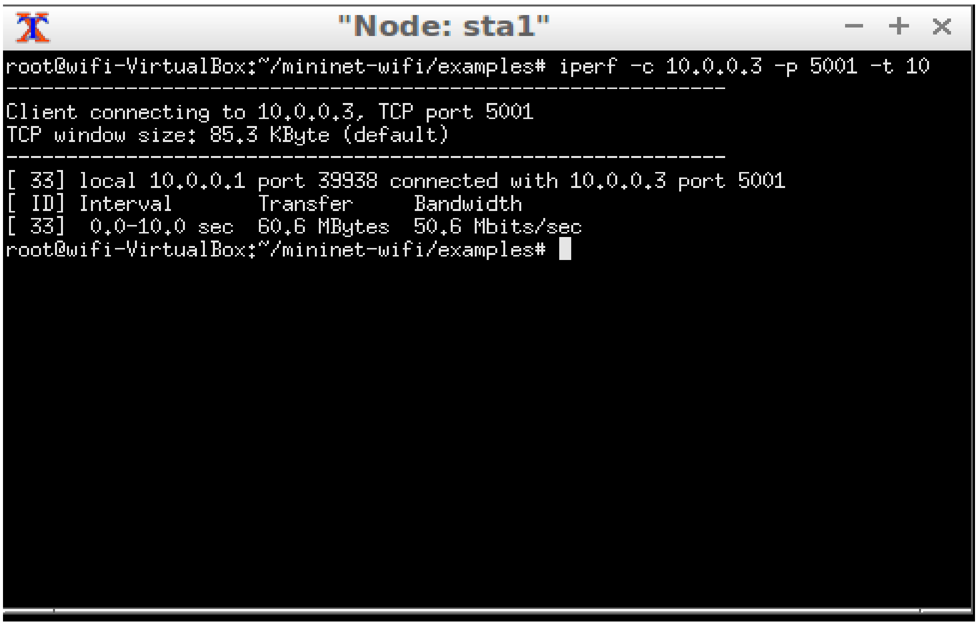
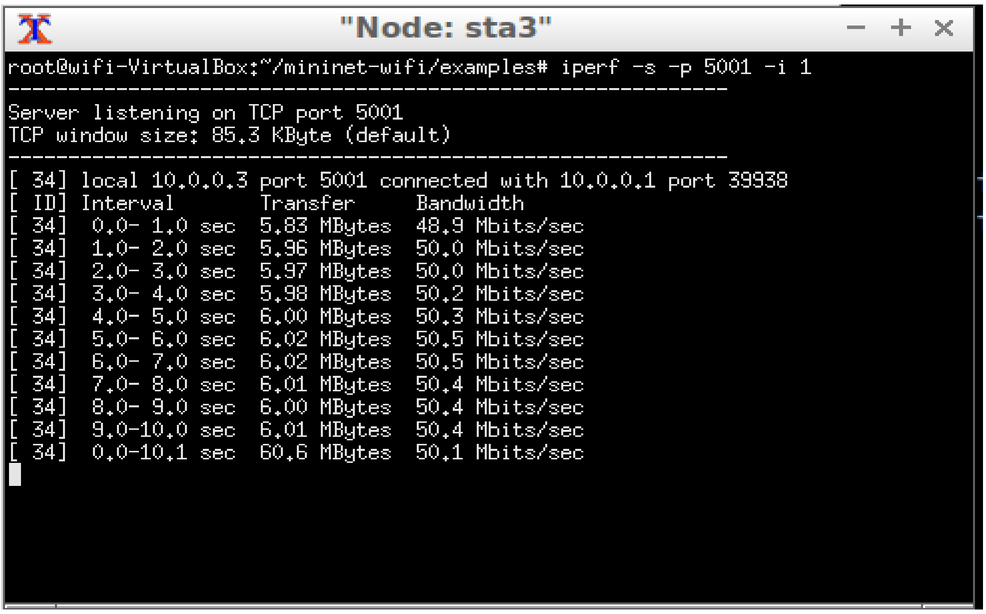


Figure 5.36: Iperf Between sta1 and sta3 with sta1 as a client



```
"Node: sta3"
root@wifi-VirtualBox:~/mininet-wifi/examples# iperf -s -p 5001 -i 1
------------------------------------------------------------
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
------------------------------------------------------------
[ 34] local 10.0.0.3 port 5001 connected with 10.0.0.1 port 39938
[ ID] Interval       Transfer     Bandwidth
[ 34]  0.0- 1.0 sec  5.83 MBytes  48.9 Mbits/sec
[ 34]  1.0- 2.0 sec  5.96 MBytes  50.0 Mbits/sec
[ 34]  2.0- 3.0 sec  5.97 MBytes  50.0 Mbits/sec
[ 34]  3.0- 4.0 sec  5.98 MBytes  50.2 Mbits/sec
[ 34]  4.0- 5.0 sec  6.00 MBytes  50.3 Mbits/sec
[ 34]  5.0- 6.0 sec  6.02 MBytes  50.5 Mbits/sec
[ 34]  6.0- 7.0 sec  6.02 MBytes  50.5 Mbits/sec
[ 34]  7.0- 8.0 sec  6.01 MBytes  50.4 Mbits/sec
[ 34]  8.0- 9.0 sec  6.00 MBytes  50.4 Mbits/sec
[ 34]  9.0-10.0 sec  6.01 MBytes  50.4 Mbits/sec
[ 34]  0.0-10.1 sec  60.6 MBytes  50.1 Mbits/sec
```

Figure 5.37: Iperf Between sta1 and sta3 with sta3 as a server

5.4.2.1 Performance Analysis with Throughput Before and After Load Re-distribution

As illustrated in Figure 5.38, the throughput of test station sta3 increased after it roamed and connected to a less loaded AP1 compared to when it was connected to a highly loaded AP3.

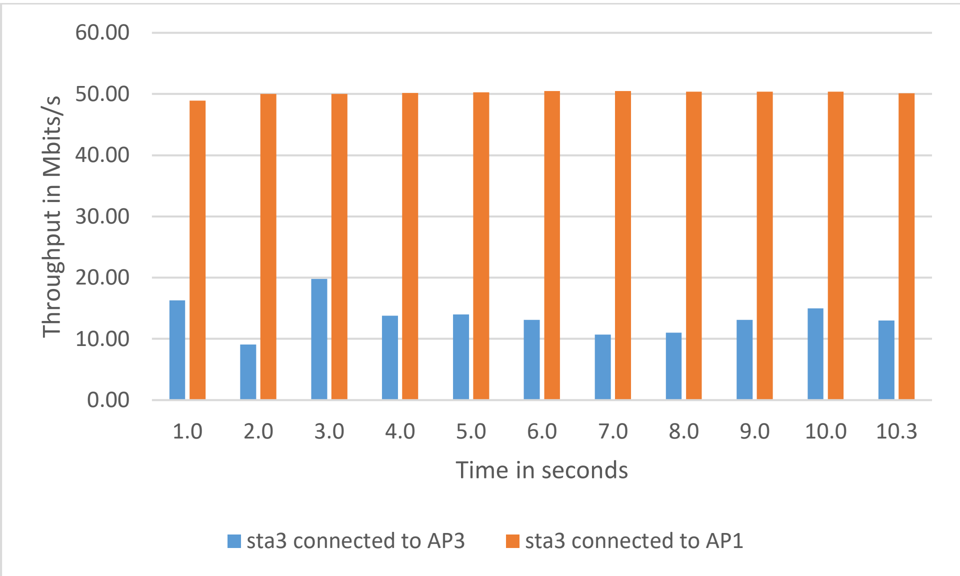


Figure 5.38: sta3 Throughput

Experiment 5: Roaming Scenario

In this experiment, the aim was to test the throughput of test station sta7 when connected to a highly loaded AP3. Two Wi-Fi APs, one base station, and seven stations were used to build a network. The stations associated with the APs as shown in Figure 5.39. Three stations connected to AP3, one station connected to AP1, and three stations connected to the base station. The association of sta1 to AP1 is shown in Figure 5.40. The associations for sta3, sta6, and sta7 to AP3 are shown in Figures 5.41, 5.42, and 5.43 respectively, where the transmitted bitrate was much lower than the expected maximum bit rate of 54 Mbps. Again, throughput for test station sta7 while connected to AP3 is shown in Figures 5.44 and 5.45.

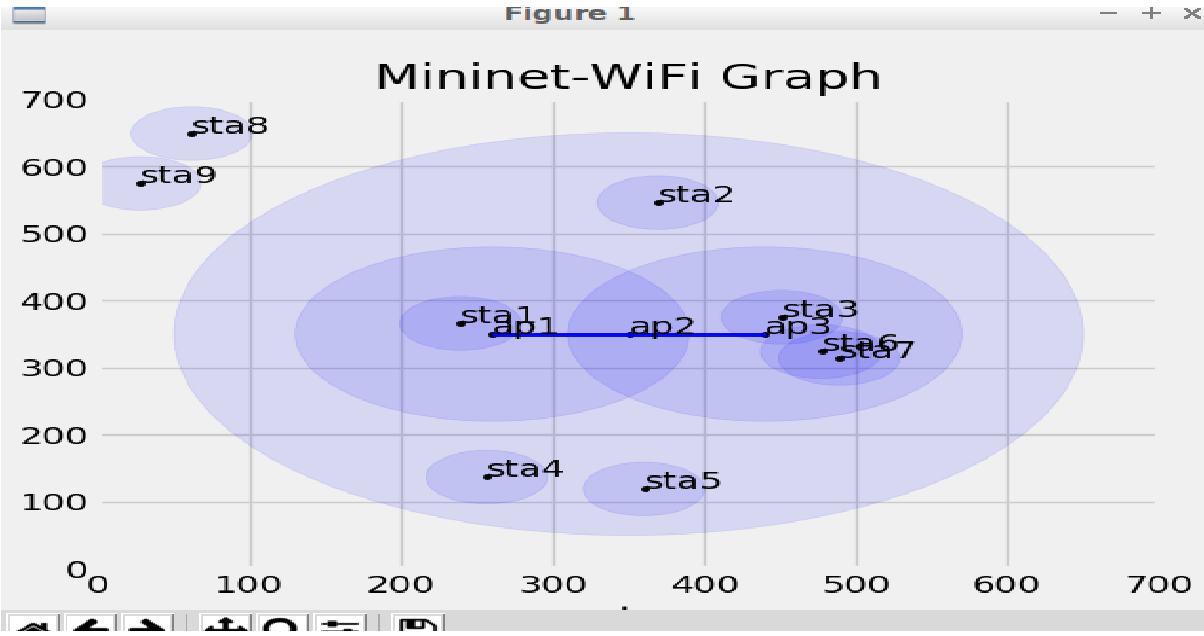


Figure 5.39: Network Setup

```
mininet-wifi> sta1 iw dev sta1-wlan0 link
Connected to 02:00:00:00:09:00 (on sta1-wlan0)
        SSID: new-ssid1
        freq: 2412
        RX: 85504 bytes (1821 packets)
        TX: 1333 bytes (15 packets)
        signal: -30 dBm
        tx bitrate: 48.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon int:     100
```

Figure 5.40: Association of sta1 to AP1

```
mininet-wifi> sta3 iw dev sta3-wlan0 link
Connected to 02:00:00:00:0b:00 (on sta3-wlan0)
        SSID: new-ssid3
        freq: 2412
        RX: 57649 bytes (1292 packets)
        TX: 469 bytes (6 packets)
        signal: -30 dBm
        tx bitrate: 1.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon int:     100
```

Figure 5.41: Association of sta3 to AP3

```
mininet-wifi> sta7 iw dev sta7-wlan0 link
Connected to 02:00:00:00:0b:00 (on sta7-wlan0)
        SSID: new-ssid3
        freq: 2412
        RX: 149953 bytes (3355 packets)
        TX: 706 bytes (9 packets)
        signal: -30 dBm
        tx bitrate: 1.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon int:     100
```

Figure 5.42: Association of sta7 to AP3

```
mininet-wifi> sta6 iw dev sta6-wlan0 link
Connected to 02:00:00:00:0b:00 (on sta6-wlan0)
        SSID: new-ssid3
        freq: 2412
        RX: 89555 bytes (2011 packets)
        TX: 548 bytes (7 packets)
        signal: -30 dBm
        tx bitrate: 1.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon int:     100
```

Figure 5.43: Association of sta6 to AP3

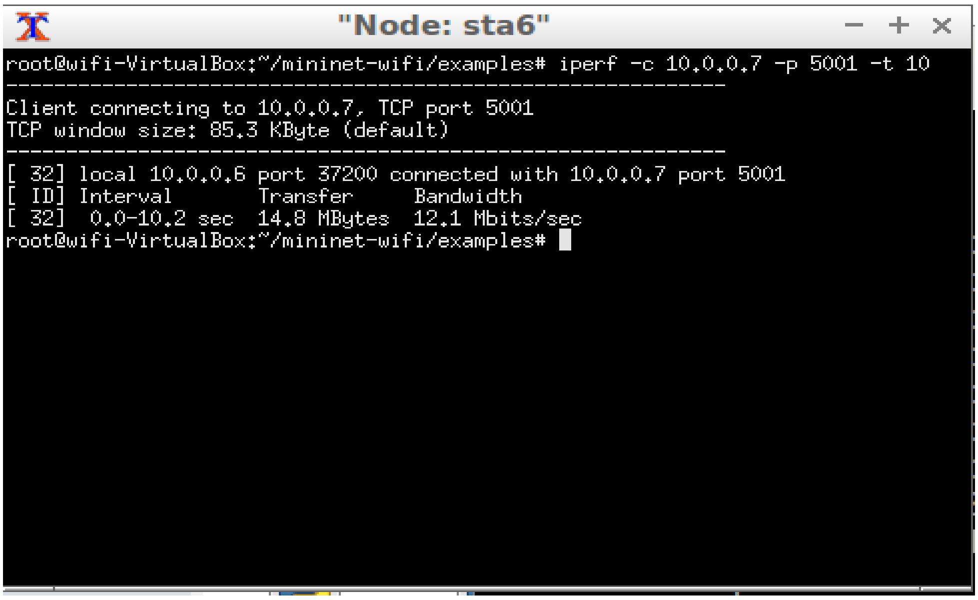


Figure 5.44: Iperf Between sta6 and sta7 with sta6 a client

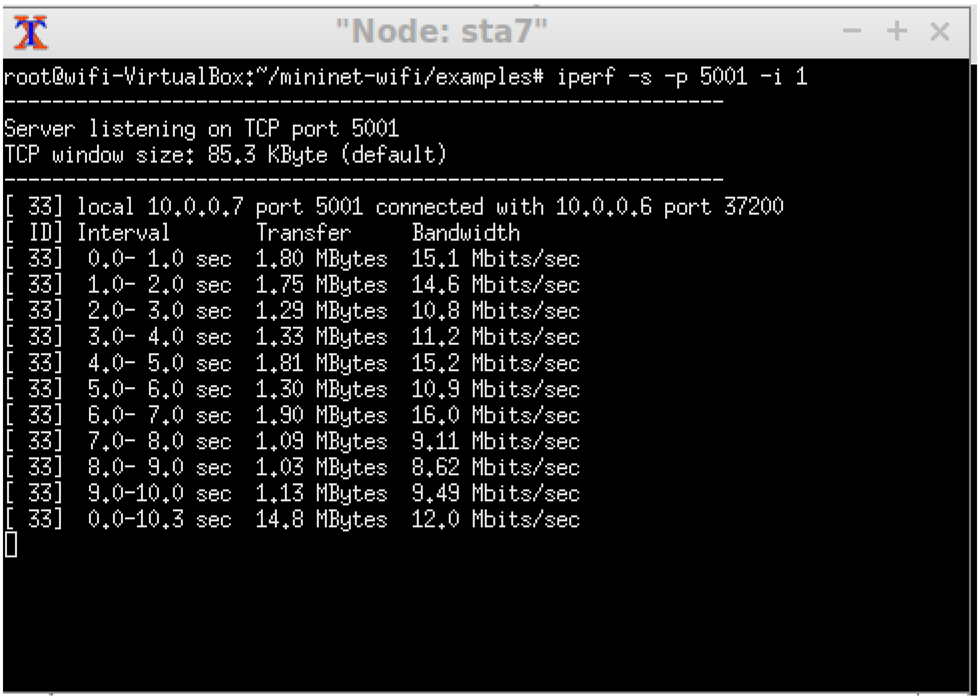


Figure 5.45: Iperf Between sta6 and sta7 with sta7 as a server

Experiment 6: sta7 Roamed and Successfully Connected to AP1

The aim of this experiment was to measure sta7 throughput when connected to the least loaded AP. Station 7 moved into the overlapping region of AP1 and AP3, as shown in Figure 5.46. It connected to AP1 because it was the least loaded AP. The associations of sta1 and sta7 to AP1 are shown in Figures 5.47 and 5.48. Throughput for sta7 after connecting to AP1 was measured by performing iperf test between sta1 and sta7. The results are shown in Figures 5.49 and 5.50.

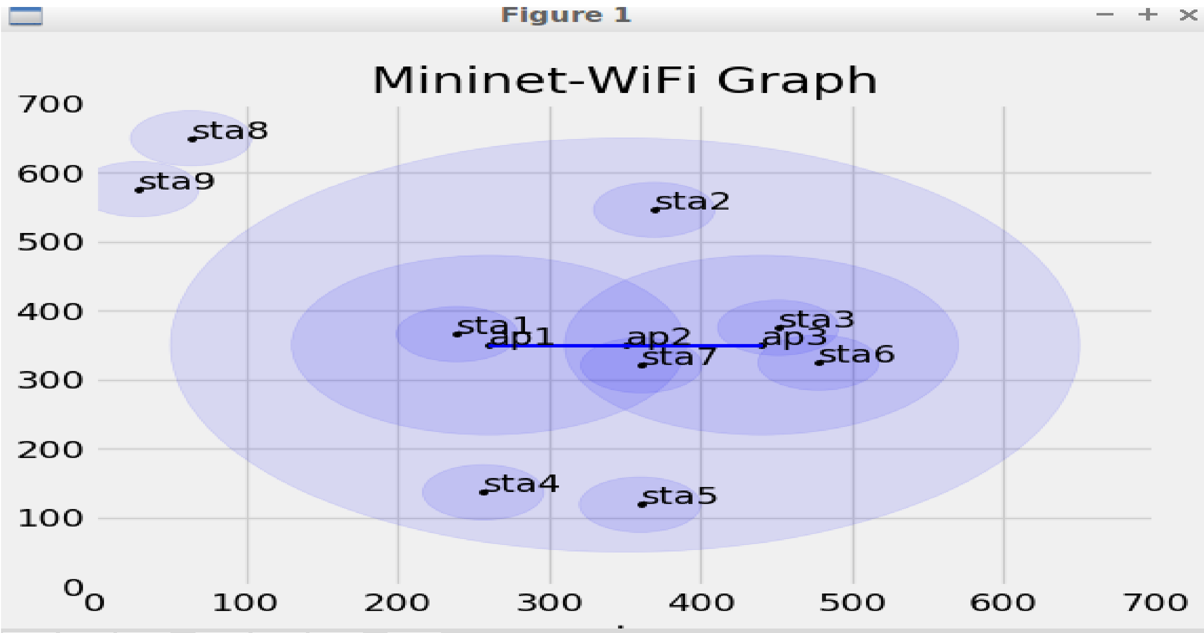


Figure 5.46: sta7 Successfully Connected to AP1

```
mininet-wifi> sta1 iw dev sta1-wlan0 link
Connected to 02:00:00:00:09:00 (on sta1-wlan0)
        SSID: new-ssid1
        freq: 2412
        RX: 2110473 bytes (27066 packets)
        TX: 66708093 bytes (43583 packets)
        signal: -30 dBm
        rx bitrate: 54.0 MBit/s
        tx bitrate: 54.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon int:     100
```

Figure 5.47: Association of sta1 to AP1

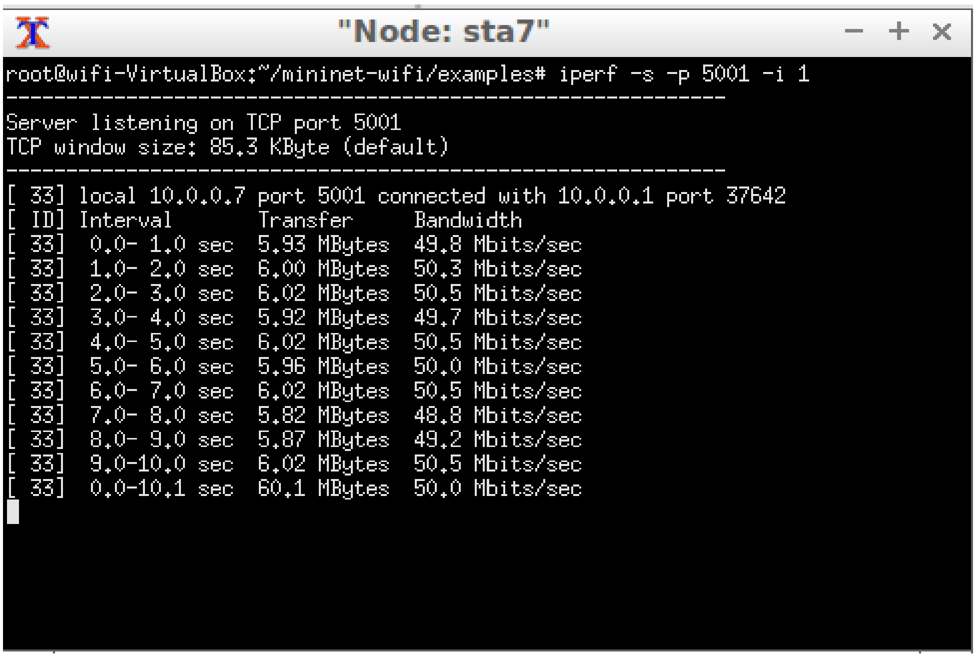
```
mininet-wifi> sta7 iw dev sta7-wlan0 link
Connected to 02:00:00:00:09:00 (on sta7-wlan0)
        SSID: new-ssid1
        freq: 2412
        RX: 112159819 bytes (78396 packets)
        TX: 3301218 bytes (38852 packets)
        signal: -30 dBm
        rx bitrate: 54.0 MBit/s
        tx bitrate: 54.0 MBit/s

        bss flags:      short-slot-time
        dtim period:    2
        beacon int:     100
```

Figure 5.48: Association of sta7 to AP1

```
"Node: sta1"
root@wifi-VirtualBox:~/mininet-wifi/examples# iperf -c 10.0.0.7 -p 5001 -t 10
------------------------------------------------------------
Client connecting to 10.0.0.7, TCP port 5001
TCP window size: 85.3 KByte (default)
------------------------------------------------------------
[ 32] local 10.0.0.1 port 37642 connected with 10.0.0.7 port 5001
[ ID] Interval       Transfer     Bandwidth
[ 32]  0.0-10.0 sec  60.1 MBytes  50.3 Mbits/sec
root@wifi-VirtualBox:~/mininet-wifi/examples#
```

Figure 5.49: Iperf Between sta1 and sta7 with sta1 as a client



```
"Node: sta7"
root@wifi-VirtualBox:~/mininet-wifi/examples# iperf -s -p 5001 -i 1
------------------------------------------------------------
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
------------------------------------------------------------
[ 33] local 10.0.0.7 port 5001 connected with 10.0.0.1 port 37642
[ ID] Interval       Transfer     Bandwidth
[ 33]  0.0- 1.0 sec  5.93 MBytes  49.8 Mbits/sec
[ 33]  1.0- 2.0 sec  6.00 MBytes  50.3 Mbits/sec
[ 33]  2.0- 3.0 sec  6.02 MBytes  50.5 Mbits/sec
[ 33]  3.0- 4.0 sec  5.92 MBytes  49.7 Mbits/sec
[ 33]  4.0- 5.0 sec  6.02 MBytes  50.5 Mbits/sec
[ 33]  5.0- 6.0 sec  5.96 MBytes  50.0 Mbits/sec
[ 33]  6.0- 7.0 sec  6.02 MBytes  50.5 Mbits/sec
[ 33]  7.0- 8.0 sec  5.82 MBytes  48.8 Mbits/sec
[ 33]  8.0- 9.0 sec  5.87 MBytes  49.2 Mbits/sec
[ 33]  9.0-10.0 sec  6.02 MBytes  50.5 Mbits/sec
[ 33]  0.0-10.1 sec  60.1 MBytes  50.0 Mbits/sec
```

Figure 5.50: Iperf Between sta1 and sta7 with sta7 as a server

5.4.2.2 Performance Analysis of Roaming Scenario with Throughput

Figure 5.51 shows the throughput of test station sta7 when it connected to AP3 compared with when it connected to AP1. As shown in Figure 5.51, when sta7 was connected to the least loaded AP1, it had a higher throughput than when it was connected to the highly loaded AP3.

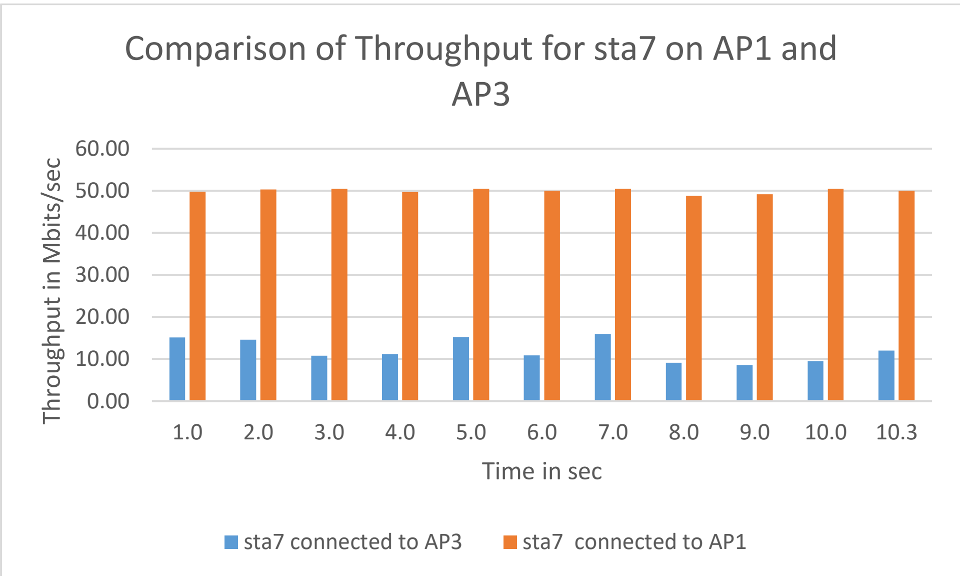


Figure 5.51: Throughput for sta7 when connected to AP1 and AP3

The diagrams that show the association of stations to access points may show varying results depending on whether the test was done before or after performing any iperf test. In addition to that, if you run iperf tests repeatedly, you may get varying results.

5.5 Chapter Summary

An SDN-based data offloading and load balancing scheme was evaluated in this chapter. For this purpose, Mininet Wi-Fi scripts were used to implement the network topologies made up of Wi-Fi APs, stations, and a controller. Network metrics such as throughput and load distribution were used to evaluate the performance of the proposed SDN-based data offloading and load balancing scheme. Simulation results showed that the proposed scheme enhanced the distribution of network's load and improves device throughput.

CHAPTER 6

6 CONCLUSION AND FUTURE WORK

6.1 Introduction

This chapter gives a conclusion to the research presented in the dissertation. It also highlights future works regarding data offloading and load balancing.

6.2 Conclusion

An SDN-based traffic offloading and load-balancing algorithm was developed for a heterogeneous network comprising cellular and WLAN to decrease congestion in cellular networks. The proposed algorithm was simulated using the Mininet-Wi-Fi emulator. The performance of the proposed algorithm was evaluated using throughput and load distribution. The results showed that the proposed algorithm improved network's load distribution and device throughput.

6.3 Future Work

The scheme implemented in this dissertation was designed and simulated for a simple network with basic network parameters.

The following points are to be considered in future work.

- Other network parameters such as the amount of data offloaded, and the number of handover instances should be considered in future work.
- The aspect of using Pox controller for load balancing and offloading has not been implemented in this dissertation. This should be considered in the future work.

References

[1] X. Duan and X. Wang, “Software-defined Networking enabled Resource Management and Security Provisioning in 5G Heterogeneous Networks,” The University of Western Ontario, 2017.

[2] M. Pozza, C. E. Palazzi, and A. Bujari, “Mobile data offloading testbed,” *Proc. Annu. Int. Conf. Mob. Comput. Networking, MOBICOM*, vol. 2015-Septe, no. 2, pp. 212–214, 2015, doi: 10.1145/2789168.2795159.

[3] 3GPP, “3لrd Generation Partnership Project; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network,” vol. 11.5, no. Release 11, 2013.

[4] Cisco, “Cisco visual networking index (VNI) global mobile data traffic forecast update, 2017-2022 white paper,” *Ca, Usa*, pp. 3–5, 2019.

[5] Cisco Systems Inc., “Cisco Visual Networking Index : Global Mobile Data Traffic Forecast Update , 2016 – 2021,” *Growth Lakel.*, vol. 2011, no. 4, pp. 2010–2015, 2017.

[6] T. O. Olwal, K. Djouani, and A. M. Kurien, “A Survey of Resource Management Toward 5G Radio Access Networks,” *IEEE Commun. Surv. Tutorials*, vol. 18, no. 3, pp. 1656–1686, 2016, doi: 10.1109/COMST.2016.2550765.

[7] F. Rebecchi, M. Dias De Amorim, V. Conan, A. Passarella, R. Bruno, and M. Conti, “Data offloading techniques in cellular networks: A survey,” *IEEE Commun. Surv. Tutorials*, vol. 17, no. 2, pp. 580–603, 2015, doi: 10.1109/COMST.2014.2369742.

[8] A. Singh, “A Review of Different Generations of Mobile Technology,” *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 4, no. 8, pp. 3404–3408, 2015.

[9] Q. K. U. D. Arshad, A. U. Kashif, and I. manzoor Qureshi, “A Review on the Evolution of the Cellular Technologies,” *Acad. Perspect. Procedia*, vol. 1, no. 1, pp. 1146–1156, 2019.

[10] J. Sharma, “The Wi-Fi Evolution,” *Qorvo*, no. March 2020, pp. 1–6, 2020.

[11] X. Foukas, M. K. Marina, and K. Kontovasilis, “Software Defined Networking Concepts,” in *Software Defined Mobile Networks (SDMN): Concepts and Challenges*, 2015, pp. 21–44.

[12] A. Hakiri, A. Gokhale, P. Berthou, D. C. Schmidt, and T. Gayraud, “Software-defined networking: Challenges and research opportunities for future internet,” *Comput. Networks*, vol. 75, no. PartA, pp. 453–471, 2014, doi: 10.1016/j.comnet.2014.10.015.

[13] D. Kreutz, F. M. V. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, “Software-defined networking: A comprehensive survey,” *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, 2015, doi: 10.1109/JPROC.2014.2371999.

[14] J. Xie, D. Guo, Z. Hu, T. Qu, and P. Lv, “Control plane of software defined networks: A survey,” *Comput. Commun.*, vol. 67, no. May, pp. 1–10, 2015, doi: 10.1016/j.comcom.2015.06.004.

[15] B. A. A. Nunes, M. Mendonca, X. N. Nguyen, K. Obraczka, and T. Turletti, “A survey of software-defined networking: Past, present, and future of programmable networks,” *IEEE Commun. Surv. Tutorials*, vol. 16, no. 3, pp. 1617–1634, 2014, doi:

10.1109/SURV.2014.012214.00180.

[16] W. Braun and M. Menth, "Software-Defined Networking Using OpenFlow: Protocols, Applications and Architectural Design Choices," *Futur. Internet*, vol. 6, no. 2, pp. 302–336, 2014, doi: 10.3390/fi6020302.

[17] Y. Liu, "An SDN Platform for Traffic Offloading," UNIVERSITY OF HELSINKI, 2015.

[18] O. N. F. Tr-, "ONF SDN Evolution," pp. 1–47, 2016.

[19] D. Lake, N. Wang, R. Tafazolli, and L. Samuel, "Softwarization of 5G Networks-Implications to Open Platforms and Standardizations," *IEEE Access*, vol. 9, pp. 88902–88930, 2021, doi: 10.1109/ACCESS.2021.3071649.

[20] A. L. Stancu, S. Halunga, A. Vulpe, G. Suciu, O. Fratu, and E. C. Popovici, "A comparison between several Software Defined Networking controllers," *2015 12th Int. Conf. Telecommun. Mod. Satell. Cable Broadcast. Serv. TELSIKS 2015*, pp. 223–226, 2015, doi: 10.1109/TELSKS.2015.7357774.

[21] I. T. Haque and N. Abu-Ghazaleh, "Wireless Software Defined Networking: A Survey and Taxonomy," *IEEE Commun. Surv. Tutorials*, vol. 18, no. 4, pp. 2713–2737, 2016, doi: 10.1109/COMST.2016.2571118.

[22] S. Tomovic, M. Pejanovic-Djurisic, and I. Radusinovic, "SDN based mobile networks: Concepts and benefits," *Wirel. Pers. Commun.*, vol. 78, no. 3, pp. 1629–1644, 2014, doi: 10.1007/s11277-014-1909-6.

[23] K. Sarfo and S. Lv, "Challenges in Wi-Fi and Femto Offloading and Coexistence Issues," 2019, doi: 10.1088/1757-899X/507/1/012025.

[24] D. H. Hagos, "The Performance of WiFi Offload in LTE Networks," Luleå University of Technology, 2012.

[25] C. W. Ahn and S. H. Chung, "SDN-Based Mobile Data Offloading Scheme Using a Femtocell and WiFi Networks," *Mob. Inf. Syst.*, vol. 2017, 2017, doi: 10.1155/2017/5308949.

[26] B. Fan, Z. He, Y. Wu, J. He, Y. Chen, and L. Jiang, "Deep Learning Empowered Traffic Offloading in Intelligent Software Defined Cellular V2X Networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13328–13340, 2020, doi: 10.1109/TVT.2020.3023194.

[27] H. Zhou, T. Wu, X. Chen, S. He, and J. Wu, "RAIM: A Reverse Auction-based Incentive Mechanism for Mobile Data Offloading through Opportunistic Mobile Networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 4697, no. c, pp. 1–1, 2021, doi: 10.1109/tnse.2021.3126367.

[28] K. Nahida *et al.*, "Handover based on AP load in software defined Wi-Fi systems," *J. Commun. Networks*, vol. 19, no. 6, pp. 596–604, 2017, doi: 10.1109/JCN.2017.000100.

[29] U. Shafi, M. Zeeshan, N. Iqbal, N. Kalsoom, and R. Mumtaz, "An Optimal Distributed Algorithm for Best AP Selection and Load Balancing in WiFi," *2018 15th Int. Conf. Smart Cities Improv. Qual. Life Using ICT IoT, HONET-ICT 2018*, pp. 65–69, 2018, doi: 10.1109/HONET.2018.8551335.

[30] H. Sounni, N. El Kamoun, and F. Lakrami, "A new SDN-based load balancing algorithm for IoT devices," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 21, no. 2, pp. 1209–1217,

2020, doi: 10.11591/ijeecs.v21.i2.pp1209-1217.

[31] F. Khan and M. Portmann, "Backhaul, QoS, and channel-aware load balancing optimization in SDN-based LTE networks," 2017 11th International Conference on Signal Processing and Communication Systems (ICSPCS), 2017, pp. 1-10, doi: 10.1109/ICSPCS.2017.8270500.

[32] P. K. Taksande, P. Jha, and A. Karandikar, "Dual Connectivity Support in 5G Networks: An SDN based approach," *IEEE Wirel. Commun. Netw. Conf. WCNC*, vol. 2019-April, 2019, doi: 10.1109/WCNC.2019.8886045.

[33] G. Shiwei, "Load Balancing Algorithm for Heterogeneous Wireless Networks Based on Motion State Estimation," *2021 IEEE 9th Int. Conf. Information, Commun. Networks, ICICN 2021*, pp. 175–178, 2021, doi: 10.1109/ICICN52636.2021.9673943.

[34] R. Torres, S. Fortes, E. Baena, and R. Barco, "Social-Aware Load Balancing System for Crowds in Cellular Networks," *IEEE Access*, vol. 9, pp. 107812–107823, 2021, doi: 10.1109/ACCESS.2021.3100459.

[35] Y. Chapre, P. Mohapatra, S. Jha, and A. Seneviratne, "Received signal strength indicator and its analysis in a typical WLAN system (short paper)," *Proc. - Conf. Local Comput. Networks, LCN*, pp. 304–307, 2013, doi: 10.1109/LCN.2013.6761255.

[36] R. R. Fontes, S. Afzal, S. H. B. Brito, M. A. S. Santos, and C. E. Rothenberg, "Mininet-WiFi : Emulating Software-Defined Wireless Networks," in *11th International Conference on Network and Service Management (CNSM)*, 2015.

[37] B. Model, "Software Based Networks : SDN and Integration of Virtualization in Networks Introduction to P4 (Programming Protocol-Independent Packet Processors) Objectives of the tutorial : Prepared by :," Karnataka, 2017.

[38] "Introduction to Mininet," 2018. Available at: https://github.com/mininet/mininet/wiki/Introduction-to-Mininet.

[39] "Using the POX SDN controller," 2015. Available at: http://www.brianlinkletter.com/using-the-pox-sdn-controller/.