

A Study of Active Access-Point Configuration Algorithm for Dual IEEE 802.11n and 11ac Interfaces in Wireless Local-Area Network System

September, 2023

Sujan Chandra Roy

Graduate School of
Natural Science and Technology

(Doctor's Course)
OKAYAMA UNIVERSITY

Dissertation submitted to
Graduate School of Natural Science and Technology
of
Okayama University
for
partial fulfillment of the requirements
for the degree of
Doctor of Philosophy.

Written under the supervision of

Professor Nobuo Funabiki

and co-supervised by

Professor Satoshi Denno

and

Professor Yasuyuki Nogami

OKAYAMA UNIVERSITY, September 2023.

TO WHOM IT MAY CONCERN

We hereby certify that this is a typical copy of the original doctor thesis of
Sujan Chandra Roy

Signature of
the Supervisor

Seal of

Prof. Nobuo Funabiki

Graduate School of
Natural Science and Technology

Abstract

Nowadays, the *IEEE 802.11 wireless local-area network (WLAN)* has increased the popularity around the world-wide for the Internet access due to high-speed data transmissions, flexible coverage ranges, low costs, management costs, and easy installations. In WLAN, hosts are connected to the *access point (AP)* wirelessly, which makes it extensible and flexible compared with the wired LAN. As a result, WLAN has been densely deployed by different providers in a lot of fields including offices, schools, and public places, such as hotels, airports, malls, stations, and even in trains and airplanes.

In a network field, multiple APs are often deployed to support a large number of users. In such *dense WLAN* environments, the network performance can be degraded because of *interferences* among nearby APs and hosts. To avoid it as best as possible, the network configuration such as activations/deactivations of APs, associations of hosts with APs, and the channel assignments should be properly optimized, according to traffic demands. The redundant APs should be deactivated to reduce the energy consummation. Therefore, the *elastic WLAN system* and *active AP configuration algorithm* have been studied to properly optimize the active APs and host association in WLAN.

The *elastic WLAN system* has been designed and implemented to dynamically optimize the network topology and the configuration in accordance with the network condition, so that this system can save energy, reduce interference and improve the performance. A *Linux PC* is used for the management server and the host in the *elastic WLAN system* to conduct experiments for this study.

The *active AP configuration algorithm* optimizes the number of active APs, and assigns a channel to every active AP from the limited number of the non-interfered channels, while the *minimum host throughput constraint* is satisfied for every host. The *throughput estimation model* has been used to evaluate the throughput between an AP and a host, where the model parameter values are obtained from extensive measurements by applying the *parameter optimization tool*. The proper parameter setting is essential in the estimation accuracy that is given by the average difference between the measured throughput and the estimated throughput.

However, the current *active AP configuration algorithm* assumes that any AP has only one *IEEE 802.11n* network interface. Besides, it does not consider the concurrent communications of multiple hosts at an AP in the throughput estimation model. It is challenging to estimate the throughput accurately when there are multiple hosts communicating simultaneously.

As the advancement of devices, currently, the simultaneous use of *IEEE 802.11n* and *11ac* interfaces at one AP has become possible. Therefore, a *dual interface* at an AP using *11n* and *11ac* for *2.4GHz* and *5GHz* frequency bands can be a promising approach to enhance the network performance while reducing interferences. The *dual interface* can reduce the necessary number of active APs to meet the traffic demands, which can further reduce the interference and enhance the overall throughput.

In this thesis, I study the *active AP configuration algorithm* for the *dual interface* at the AP

in WLAN. This algorithm selects the associated AP and its network interface to each host. The *throughput estimation model* for 11ac is used together with the model for 11n. Moreover, I introduce the *throughput reduction factor* into the *throughput estimation model* to improve the estimation accuracy when multiple hosts are associated with the same AP. The *dual interfaces* at the AP operate at different frequency bands, which allows the AP to communicate with two hosts simultaneously without interference between them. Therefore, it can greatly enhance the network performance while reducing the number of active APs in the network for smaller power consumptions and less interferences. I confirm the effectiveness of the proposed algorithm through simulations using the *WIMNET simulator*.

I present the implementation of the proposed algorithm using *elastic WLAN system* testbed using dual interfaces. In this implementation, *Raspberry Pi 4B* is used for the software *access point (AP)* by running *hostapd*. The *embedded wireless NIC* is used for the 11n interface, and the *Archer T4U wireless NIC adapter* is used for the 11ac interface. *Linux laptop PCs* are used for the client hosts and the server. Using *Linux commands*, the system collects the necessary information for the extended *active AP configuration algorithm* to determine the activations/deactivations of APs, the host associations, and the channels for the active AP interfaces. Based on the algorithm outputs, the server controls the active APs, their channels, and the associations of hosts automatically.

For evaluations, firstly, I evaluated the proposal when APs are operating under *non-channel bonding (non-CB)*. The experimental results confirm that the estimated throughput is well matched with the measured one, and the proposal achieves the higher throughput with a smaller number of active APs.

Due to the wide channel width, the *channel bonding (CB)* channels can provide higher throughputs. In a CB channel, two adjacent channels are combined into one channel. Therefore, secondly, I evaluated the proposal when the APs are operating under CB. The experimental results confirm that the proposal can reduce the number of active APs and increase the total throughput of the network, while the *minimum host throughput constraint* is satisfied by every host in the network.

In future studies, I will further improve the *active AP configuration algorithm* by considering partially overlapping channels (POCs), transmission power optimizations, and the 11ax protocol. Then, I will evaluate the proposals to confirm their effectiveness in various network fields and topologies.

Acknowledgements

I am truly delighted to extend my heartfelt gratitude to all those who generously dedicated their time and provided unwavering support, making the successful realization of this dissertation possible. I express my sincerest thanks to each and every one of you for playing a significant role in turning my dream into a reality.

I would like to express my sincere gratitude to my supervisor, Professor Nobuo Funabiki for his excellent guidance, valuable suggestions, persistent encouragement, and other helpful assistance throughout my Ph.D. study. He provided me with insightful comments and guidance that helped me to complete my research papers and present them in a productive way. In addition to that, he was always patient and helpful whenever I needed his guidance and assistance in my academic and daily life in Japan. I consider myself incredibly fortunate to have had the opportunity to work with someone as remarkable as him. Indeed, without his guidance and unwavering support, completing this thesis would not have been possible.

I express my gratitude to my Ph.D. co-supervisors, Professor Satoshi Denno and Professor Yasuyuki Nogami, for their invaluable dedication in sparing their valuable time to provide me with advice, guidance, insightful comments, and their assistance in proofreading this thesis. I am sincerely appreciative of their contributions.

I would also like to express my deep appreciation to Associate Professor Minoru Kuribayashi at Okayama University for his invaluable discussions and unwavering support throughout my research. Furthermore, I am grateful to all the course teachers who have imparted their knowledge and ideas during my Ph.D. studies.

I would like to acknowledge the Ministry of Education, Culture, Sports, Science, and Technology of Japan (MEXT) for financially supporting my Ph.D. study at Okayama University.

I would like to convey my appreciation to all the members of FUNABIKI Lab. Especially, I would like to give special thanks to Ms. Keiko Kawabata, Dr. Kwenga Ismael Munene, Dr. Md. Manowarul Islam, Dr. Hendy Briantoro, Dr. Yan Watequlis Syaifudin, Dr. Htoo Htoo Sandi Kyaw, Dr. Md. Mahbubur Rahman, Dr. Hein Htet, Dr. Pradini Puspitaningayu, Dr. Yuanzhi Huo, Mr. Yohanes Yohanie Fridelin Panduman, Mr. Wu Bin, Mr. Sun Yuanshuai, and all members in general.

I would also like to express my special gratitude to my undergraduate and graduate teachers, especially Dr. Md. Ashraful Islam, Muhammad Sajjadur Rahim, and Dr. Dipankar Das for their great support and guidance during the MEXT scholarship application process.

Last but not least, I am grateful to my family, my mother, father, sisters and all my friends for their unwavering friendships, constant inspiration, motivating words, and incredible support. I am deeply grateful for their sincere friendships and the invaluable support they have provided me.

Sujan Chandra Roy
Okayama University, Japan
September, 2023

List of Publications

Journal Papers

1. **S. C. Roy**, N. Funabik, K. I. Munene, M. M. Rahman, and M. Kuribayashi, “An extension of active access-point configuration algorithm to IEEE 802.11n and 11ac dual interfaces in wireless local-area network,” *Int. J. Future Comput. Commun.*, vol. 11, no. 2, pp. 18-26, Jun. 2022.
2. **S. C. Roy**, N. Funabik, M. M. Rahman, B. Wu, M. Kuribayashi, and W. -C. Kao, “A Study of the Active Access-Point Configuration Algorithm under Channel Bonding to Dual IEEE 802.11n and 11ac Interfaces in an Elastic WLAN System for IoT Applications,” *Signals*, vol. 4, no. 2, pp. 274-296, Apr. 2023.

Other Papers

3. **S. C. Roy**, N. Funabik., K. I. Munene, H. Briantoro, M. M. Rahman, F. Akhter, and M. Kuribayashi, “A study of throughput drop estimation model for two-link concurrent communications in 5GHZ IEEE 802.11ac WLAN,” *IEICE Society Conf.*, pp. S22–23, Sept. 14-17, 2021. (Online, Japan).
4. **S. C. Roy**, N. Funabik, M. M. Rahman, Y. Sun, B. Wu, and M. Kuribayashi, “A study of active access-point configuration algorithm under IEEE 802.11n and 11ac dual interfaces with channel bonding in wireless local-area network,” *IEICE Tech. Report*, vol. 122, no. 274, pp. 113-118, Nov. 24-25, 2022.
5. **S. C. Roy**, N. Funabik, B. Wu, Y. Sun, E. E. Htet, and M. Kuribayashi, “A throughput drop estimation model with non-channel bonding for two-link concurrent communications in IEEE 802.11ac WLAN,” *IEICE Society Conf.*, pp. S7–8, Mar. 7-10, 2023.

List of Figures

1.1	Overview of Elastic WLAN.	2
2.1	Components of <i>IEEE 802.11</i> WLANs.	6
2.2	Types of <i>IEEE 802.11</i> networks.	7
2.3	Extended service set (ESS).	8
2.4	Current and future WLAN standards.	10
2.5	WiFi channels in 2.4 GHz band.	12
2.6	WiFi channels in 5 GHz band.	13
2.7	IEEE802.11n channel bonding concept.	14
2.8	Comparison between SISO and 4×4 MIMO technology.	14
2.9	Primary and secondary channels in 802.11ac	17
2.10	Comparison between Single User MIMO and Multi User MIMO	18
3.1	Design of elastic WLAN system.	23
3.2	Sigmoid function for throughput estimation from signal strength.	24
4.1	Experiment field #1.	38
4.2	Testbed topology for dual interface with two hosts.	38
4.3	Group A throughput results for 20MHz.	41
4.4	Group B throughput results for 20MHz.	41
4.5	Group A throughput results for 40MHz.	42
4.6	Group B throughput results for 40MHz.	42
6.1	Example network topology of elastic WLAN system testbed.	50
6.2	Execution flow of elastic WLAN system.	53
7.1	Solution for large topology with 40 hosts and 14 APs.	57
7.2	Field #1 in Engineering Building #2.	58
7.3	Field #2 in Graduate School Building.	59
7.4	Throughput results in Field #1	60
7.5	Throughput results in Field #2	60
7.6	Experimental field #1.	61
7.7	Experimental field #2.	62
7.8	Throughput results for Topology 1.	64
7.9	Throughput results for Topology 2.	65
7.10	Throughput results for Topology 3.	65
7.11	Throughput results for Topology4.	66
7.12	Minimum host throughput result.	67
7.13	Overall throughput result.	67

List of Tables

2.1	<i>IEEE 802.11</i> standards.	8
2.1	<i>IEEE 802.11</i> standards.	9
2.2	Features of common IEEE 802.11 standards.	11
2.3	IEEE 802.11n specifications.	12
2.4	Channel bonding in IEEE 802.11n.	13
2.5	Effects of channel bandwidth and spatial stream's selection towards IEEE 802.11n's throughput.	14
2.6	IEEE 802.11ac specifications.	16
4.1	Devices and software specifications.	39
4.2	Parameter optimization results for Engineering Building #2 (Field #1).	39
4.3	Parameter optimization results for Graduate School Building (Field #2).	40
7.1	Simulation environment.	56
7.2	Simulation parameters in WIMNET simulator.	56
7.3	Performance comparison between proposal and previous for large topology	57
7.4	AP and host location for small topology	59
7.5	Simulation and measurement results for CB cases.	63
7.6	Comparative analysis between 11ac and dual interface.	63
7.7	Simulation and measurement results for non-CB cases [31]	64

List of Abbreviations

WLAN	wireless local area network
AP	access point
CB	channel bonding
non-CB	non-channel bonding
DAP	dedicated access point
VAP	virtual access point
NIC	network interface card
OC	orthogonal channel
WIMNET	Wireless Internet-access Mesh NETwork
RF	radio frequency
BSS	basic service set
IBSS	independent basic service set
ESS	extended service set
SSID	service set identifier
ISM	industrial scientific and medical
OFDM	orthogonal frequency division multiplexing
OFDMA	orthogonal frequency division multiple access
MCS	modulation and coding scheme
MIMO	multiple input multiple output
SISO	single-input single-output
MU-MIMO	Multi-User Multiple Input Multiple Output
RSS	received signal strengths
TCP	transmission control protocol

Contents

Abstract	i
Acknowledgements	iii
List of Publications	v
List of Figures	viii
List of Tables	ix
List of Abbreviations	xi
1 Introduction	1
1.1 Background	1
1.2 Contributions	3
1.3 Contents of Thesis	4
2 Background Technologies	5
2.1 IEEE 802.11 WLAN Overview	5
2.1.1 Advantages of WLAN	5
2.1.2 IEEE 802.11 WLAN Components	6
2.1.3 Types of WLANs	7
2.1.4 IEEE 802.11 Standards for WLAN	8
2.2 IEEE 802.11n Protocol	11
2.3 Features of IEEE 802.11n Protocol	12
2.3.1 Channel Bonding	13
2.3.2 Partially Overlapping Channels	13
2.3.3 Multiple Input Multiple Output (MIMO)	14
2.3.4 MAC Layer Enhancements	15
2.4 IEEE 802.11ac Protocol	15
2.5 Features of IEEE 802.11ac Protocol	15
2.5.1 Channel Bonding	16
2.5.2 Non Channel Bonding	16
2.5.3 Multi-User Multiple Input Multiple Output (MU-MIMO)	16
2.6 Linux Tools for Wireless Networking	17
2.6.1 ‘arp-scan’ - to Explore Currently Active Devices	17
2.6.2 ‘nm-tool’ - to Collect Host Information	18
2.6.3 ‘hostapd’ - to Make AP-mode Linux-PC	18

2.6.4	'ssh' - to Remotely Execute Command	19
2.6.5	'nmcli' - to Change Associated AP	19
2.6.6	'iwconfig' - to Collect Information of Active Network Interface	19
2.6.7	'sed-tool' - to Assign the Channels to the APs	19
2.6.8	'iperf' - to Measure Link Speed	20
2.7	Summary	20
3	Review of Previous Studies	21
3.1	Elastic WLAN System	21
3.1.1	Overview	21
3.1.2	Design and Operational Flow	22
3.2	Throughput Estimation Model	22
3.2.1	Overview of Model	23
3.2.2	<i>Received Signal Strength Estimation</i>	23
3.2.3	<i>Throughput Estimation from Received Signal Strength</i>	24
3.2.4	Throughput Reduction Factor	25
3.3	Parameter Optimization Tool	25
3.3.1	Overview	25
3.3.2	Required Files for Tool	25
3.3.2.1	Parameter Specification File	25
3.3.2.2	Model Program File	26
3.3.2.3	Sample Input Data File	26
3.3.2.4	Score Output File	26
3.3.2.5	Script File for Execution	26
3.3.3	Processing Flow of Tool	27
3.3.3.1	Symbols in Algorithm	27
3.3.4	Parameter Optimization Algorithm	28
3.3.4.1	Algorithm Procedure	28
3.4	Active AP Configuration Algorithm	29
3.4.1	Problem Formulation	29
3.4.2	Algorithm Procedure	31
3.4.2.1	Active AP and Associated Host Selection Phase	31
3.4.2.2	Channel Assignment Phase	33
3.4.2.3	Channel Load Averaging Phase	34
3.5	Summary	35
4	Experimental Observation for Interface Selection	37
4.1	Introduction	37
4.2	Experimental Setup	37
4.3	Throughput Results	38
4.4	Summary	42
5	Proposal of Active AP Configuration Algorithm for Dual Interface	43
5.1	Introduction	43
5.2	Active AP Configuration Algorithm for Dual Interface	44
5.2.1	Problem Formulation	44
5.2.2	Algorithm Procedure	45

5.3	Summary	47
6	Testbed Implementation	49
6.1	Testbed Implementation	49
6.2	Testbed Network Topology	49
6.3	Dual Interface AP Configuration Using Raspberry Pi	50
6.4	Execution Flow of Dual Interface in Elastic WLAN	52
6.4.1	Generation of input for dual interface AP Configuration Algorithm	52
6.4.2	Execution of dual interface Active AP Configuration Algorithm	53
6.4.3	Execution of Channel Assignment Algorithm	53
6.4.4	Application of dual interface Active AP Configuration	54
6.4.5	Application of Channel Assignment	54
6.5	Summary	54
7	Evaluation of Active AP Configuration Algorithm for Dual Interface	55
7.1	Evaluation by Simulation	55
7.1.1	WIMNET simulator	55
7.1.2	Network Environments	55
7.1.3	Simulation Platform	56
7.1.4	Evaluation of Large Topology for Non-CB Case	56
7.2	Testbed Evaluation for Non-CB Case	57
7.2.1	Experimental Setup	58
7.2.2	Network Fields	58
7.2.3	Results and Discussions	59
7.3	Testbed Evaluation for CB Case	59
7.3.1	Experimental Setup	61
7.3.2	Network Fields	61
7.3.3	Results and Discussions	62
7.3.4	Comparisons with Previous Studies	63
7.4	Summary	67
8	Related Works in Literature	69
9	Conclusion	71
	References	73

Chapter 1

Introduction

1.1 Background

Nowadays, *wireless local area networks (WLANs)* have become popular for the Internet access around the world. The main reason behind the popularity of this technology is the simplicity of the IEEE 802.11 protocols, which offer low-cost ownership, flexible coverage ranges, easy installation features, and maintenance combined with the ability to provide high data rates to end users. In WLAN, hosts are connected to the *access point (AP)* wirelessly, which makes it more extensible and flexible than wired LANs. As a result, WLAN services are available in offices, schools, and public places, such as hotels, airports, malls, stations, and even in trains and airplanes [1–6].

The IEEE 802.11 standard was first introduced in June 1997 [5, 7]. Since then, the IEEE 802.11 standards have been updated with newer versions that add new features to improve the performance of WLANs. Currently, the *IEEE 802.11n* [8] and *IEEE 802.11ac* [9] standard are the most widely used, even though its maximum throughput is lower than that of newer standards like *IEEE 802.11ax* [10]. Both standards have several features to improve the network performance including *channel bonding (CB)*, *multiple input multiple output (MIMO)*, and *frame aggregation*, unlike previous standards, namely 11a, 11b, and 11g. The *IEEE 802.11ac* uses more advanced form of MIMO called *MU-MIMO (Multi-User MIMO)*. MU-MIMO allows the AP to transmit data to multiple devices simultaneously, which can significantly improve the performance of the network in areas with a lot of users [11–13].

The WLAN capacity can be enhanced with a relatively low cost, by additionally installing WLAN access points (APs). Therefore, in a network field, multiple APs are often deployed to support a large number of users to ensure seamless Internet access. Such situations are called *dense WLAN environments*, which leads to wide spread of WLAN APs over heavily populated areas (e.g., public, residential, and enterprise areas). In dense WLAN environments, service coverage of neighboring APs can be overlapped with each other. The neighboring APs can commonly use the same channels because of a limited number of available WLAN channels [14–17]. Then, the network performance may be severely degraded due to the interference among neighboring APs. To avoid the performance degradation as best as possible, the network configuration of WLAN, such as activations/deactivations of APs and associations of hosts with active APs, should be properly optimized, according to traffic demands. If redundant APs exist, they should be deactivated to reduce energy consumptions [18–21].

In a WLAN, the distribution of users is usually unevenly distributed [22]. The number of users and traffics fluctuate depending on the time of day and day of the week, which is often unpredictable. Various people use the network for different purposes at different times [23, 24].

For example, at a university, the network is usually the busiest during the afternoon on weekdays, when students use it for classes and activities. The network is generally less occupied in the morning and evening, and on weekends.

In addition, network devices and communication links may be affected by various factors, such as power outages, device failures, bandwidth controls by authorities, and even weather changes [25]. For example, many developing countries including Bangladesh and Myanmar often experience unreliable and slow internet access due to intermittent electricity supplies. These factors can cause network devices to lose power, communication links to become unavailable, and bandwidth to be restricted [26, 27].

To solve the above-mentioned problems, we have studied the *elastic WLAN system* using heterogeneous AP devices [6, 28]. It consist of three types of APs, namely *dedicated APs (DAPs)*, *virtual APs (VAPs)*, and *mobile APs (MAPs)*. Figure 1.1 shows a simple topology of the elastic WLAN system. It has been designed and implemented to dynamically optimize the network topology and the configuration in accordance with the network condition, so that this system can save energy, reduce interference and improve the performance. Additionally, the *elastic WLAN system* has been implemented by running *hostapd* software onto *Raspberry Pi*, which act as an AP. The *Raspberry Pi* is a small card-sized computer. It contains a built-in *wireless network interface card (NIC)* for IEEE 802.11n and runs on a Linux-based operating system called Raspbian. A *Linux PC* is used for the management server and the host in the *elastic WLAN system* to conduct experiments for this study.

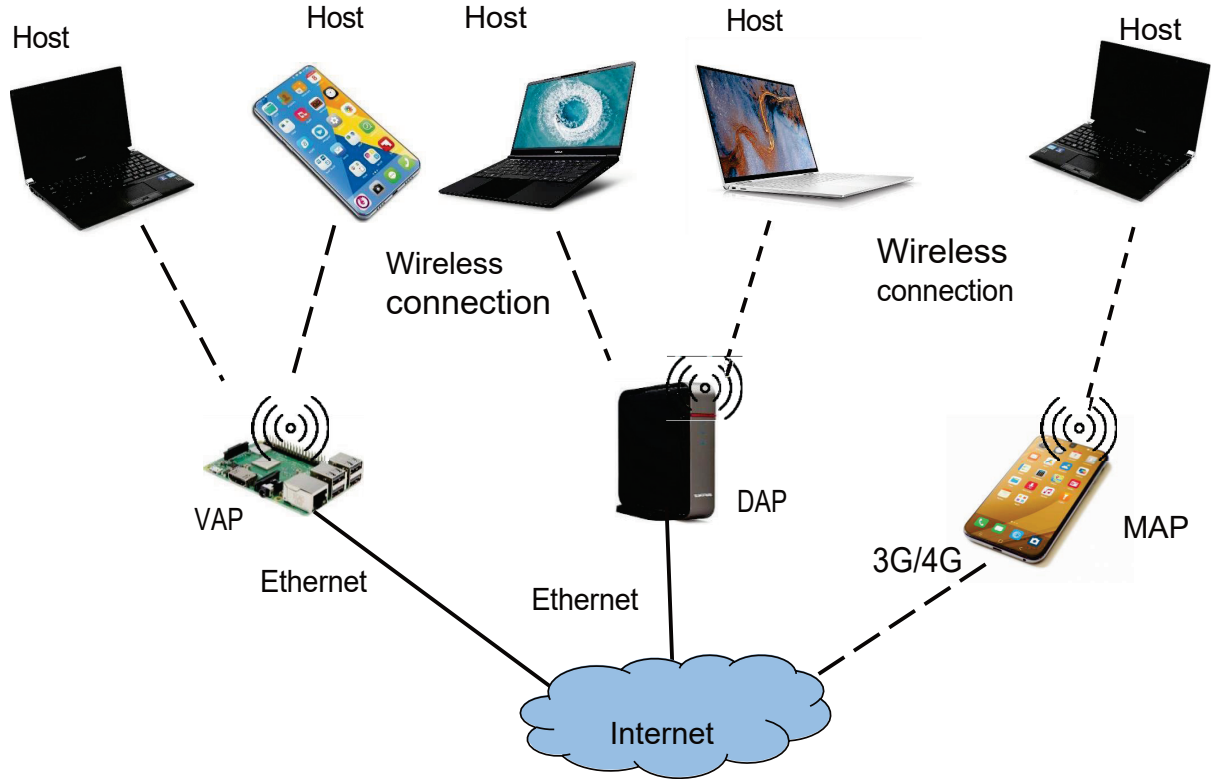


Figure 1.1: Overview of Elastic WLAN.

To optimize the *elastic WLAN system*, we have studied the *active AP configuration algorithm*. It optimizes the number of active APs, and assigns a channel to every active AP from the limited number of the non-interfered channels in a way to minimize the overall interference in the network,

while the *minimum host throughput constraint* is satisfied for every host. Each of the active AP and host in the network uses the default maximum transmission power during the communication.

A *throughput estimation model* [29] has been studied to estimate the link throughput between an AP and a host in a WLAN, which is essential in the active AP configuration algorithm. The parameter values of the model are optimized by applying the *parameter optimization tool* [30] from extensive measurement results with different host locations in the network field. The proper parameter setting is essential in the estimation accuracy that is given by the average difference between the measured throughput and the estimated throughput.

However, the current *active AP configuration algorithm* assumes that any AP has only one *IEEE 802.11n* network interface. Besides, it does not consider the concurrent communications of multiple hosts at an AP to estimate the throughput using the throughput estimation model. It is challenging to estimate the throughput accurately when there are multiple hosts communicating simultaneously.

Currently, the simultaneous use of *IEEE 802.11n* and *11ac* interfaces at one AP has become possible with the advancement of devices. Therefore, a *dual interface* at an AP using *11n* and *11ac* for *2.4GHz* and *5GHz* frequency bands can be a promising approach to enhance the network performance while reducing interferences. The *dual interface* can reduce the necessary number of active APs to meet the traffic demands, which can further reduce the interference and enhance the overall throughput.

1.2 Contributions

The following research contributions are included in this thesis.

A dual interface at the same AP using *11n* and *11ac* at *2.4GHz* and *5GHz* frequency bands is an efficient approach to enhance the network's performance. Therefore, I propose the *active AP configuration algorithm* for the *dual interface* at the AP in WLAN by extending the existing one [31, 32]. This algorithm selects the associated AP and its network interface to each host. The *throughput estimation model* for *11ac* is used together with the model for *11n*. The model parameter values are obtained from extensive measurements by applying the *parameter optimization tool*. Moreover, I introduce the *throughput reduction factor* into the *throughput estimation model* to improve the estimation accuracy when multiple hosts are associated with the same AP. The *dual interfaces* at the AP operate at different frequency bands, which allows the AP to communicate with two hosts simultaneously without interference between them. Therefore, it can greatly enhance the network performance while reducing the number of active APs in the network for smaller power consumptions and less interferences. I confirm the effectiveness of the proposed algorithm through simulations using the *WIMNET simulator* [33].

For the performance evaluation, I present the implementation of the proposed algorithm using *elastic WLAN system* testbed using dual interfaces. In this implementation, *Raspberry Pi 4B* [34] is used for the software *access point (AP)* by running *hostapd*. The *embedded wireless NIC* is used for the *11n* interface, and the *Archer T4U* [35] *wireless NIC adapter* is used for the *11ac* interface. *Linux laptop PCs* are used for the client hosts and the server.

For evaluations, firstly, I evaluated the proposal when APs are operating under *non-channel bonding (non-CB)*. The experimental results confirm that the estimated throughput is well matched with the measured one, and the proposal achieves the higher throughput with a smaller number of active APs.

Due to the wide channel width, the *channel bonding (CB)* channels can provide higher through-

puts. In a CB channel, two adjacent channels are combined into one channel. Therefore, secondly, I evaluated the proposal when the APs are operating under CB. The experimental results confirm that the proposal can reduce the number of active APs and increase the total throughput of the network, while the *minimum host throughput constraint* is satisfied by every host in the network.

The main contributions of this thesis are summarized as follows:

- I propose the *active AP configuration algorithm* to take into account the use of the *dual interface* at each AP, where the interfaces operate at different frequency bands with IEEE 802.11n and 11ac protocols, and the AP can communicate with two hosts simultaneously without any interference between them.
- I incorporate the *throughput reduction factor* into the *throughput estimation model* to improve the estimation accuracy when multiple hosts are associated with the same AP.
- I evaluated the proposal under non-CB and CB through both simulations using the *WIMENT simulator* and experiments using the testbed system in real fields. The testbed system involved a *Raspberry Pi* with an external *Archer T4U* for the AP and *Linux PCs* for the hosts.

1.3 Contents of Thesis

The remaining part of this thesis is organized as follows.

Chapter 2 reviews the IEEE 802.11 wireless network technologies related to this thesis, including features of the IEEE 802.11n and 11ac protocol, software tools in the Linux operating system.

Chapter 3 reviews our previous studies related to this thesis.

Chapter 4 describes the experimental observations for interface selection at the APs.

Chapter 5 presents the proposal of active AP configuration algorithm for dual interface.

Chapter 6 presents the implementation of dual interface active AP configuration algorithm on real testbed.

Chapter 7 describes the evaluations of the proposals.

Chapter 8 reviews relevant works in literature.

Finally, Chapter 9 concludes this thesis with some future works.

Chapter 2

Background Technologies

This chapter provides an overview of wireless network technologies relevant to the background of this dissertation. At first, we describe advantages, components, types, and standards of IEEE 802.11 protocols. Next, we describe the *IEEE 802.11n* and *11ac* protocols and their main features. Finally, we outline *Linux* tools and commands for wireless networking that are used to measure and implement the elastic WLAN system testbed.

2.1 IEEE 802.11 WLAN Overview

The *IEEE 802.11 Wireless local-area network (WLAN)* specifies the *physical (PHY)* and *media access control (MAC)* layers for enabling high-speed data communications. It provides wireless connectivity by extending a wired LAN and supporting mobile user mobility and data communication flexibility. The *Radio frequency (RF)* technology is used to transmit and receive data over the air, which reduces the need for wiring and makes WLANs more cost-effective in homes, schools, and offices. As a result, WLAN technology is widely adopted around the world for different purposes [36].

2.1.1 Advantages of WLAN

WLANs offer several advantages over wired LANs. The list below includes a few of them.

- *Mobility:*
Users can connect to the network and access resources using WLANs from any location within the network's service area. It enables users to move around freely without being restricted by cables and wires.
- *Easy to install:*
Compared to wired LANs, WLANs are simpler to install and configure. It is simpler to install or remove network devices using WLANs because there is no need to lay cables.
- *Cost:*
WLANs are typically more cost-effective to install and maintain than wired LANs. This is because WLANs do not require the installation of cables throughout the building, which can be a time-consuming and expensive process. Additionally, WLANs are easier to move, add, and reconfigure than wired LANs, which can save businesses and organizations money on labor costs.

- *Flexibility:*
WLANs reduce the need to run cables through walls and ceilings. The network coverage area of WLAN can be easily expanded as the network media is everywhere.
- *Scalability:*
WLANs are easily expandable and can be given more coverage by adding more access points. Therefore, WLANs are suitable for expanding enterprises or for companies with varying network requirements.

2.1.2 IEEE 802.11 WLAN Components

IEEE 802.11 WLAN consists of four primary components as shown in Figure 2.1 [36]:

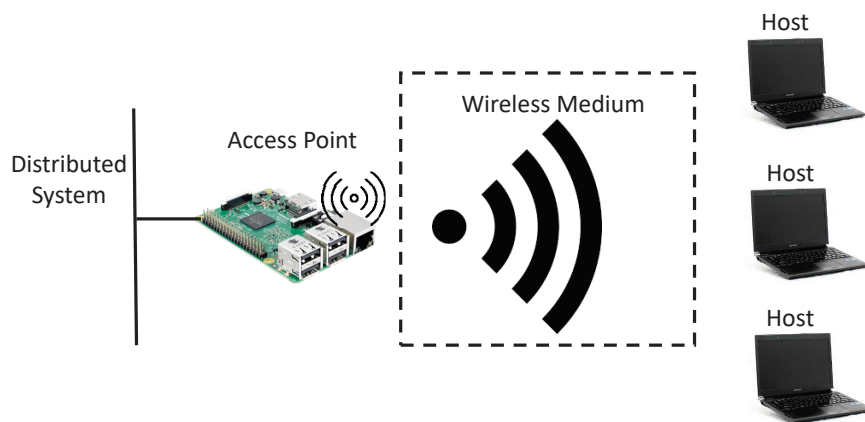


Figure 2.1: Components of *IEEE 802.11* WLANs.

- *Stations or hosts:*
Through WLANs, data is transferred between *hosts* or *stations*. A host or station in a WLAN is an electronic device that has the ability to access the network over a wireless *network interface card (NIC)*. Hosts can include laptops, smartphones, tablets, and other wireless-enabled devices.
- *Access points (APs):*
An *access point (AP)* is a device that acts as a base station for a wireless local area network (WLAN). It provides a connection between wireless client devices and the wired network. APs are similar to hubs and switches in wired networks, in that they connect multiple devices together and allow them to communicate with each other. APs also provide bridging functionality, which allows them to connect two or more networks together.
- *Wireless medium:*
The wireless medium is used by the *IEEE 802.11* standard to transmit information and data from one host to another host in a network.
- *Distribution system:*
When multiple *access points (APs)* are connected together to create a large coverage area, they must communicate with each other to track the movement of devices. The distribution

system is a logical component of a WLAN that serves as the backbone connection between APs. It is often referred to as the backbone network that relays data frames between APs. *Ethernet* is commonly used as the backbone network technology in most cases.

2.1.3 Types of WLANs

The *IEEE 802.11* WLAN functions through a fundamental component called the *basic service set (BSS)*, which consists of a group of hosts that can communicate with each other. Depending on the type of BSS, IEEE 802.11 standard supports two different types of WLAN as shown in Figure 2.2.

- *Independent or ad hoc type:*
This type of WLAN allows stations or hosts to send frames directly to one another without requiring an AP, and is referred as the *independent basic service set (IBSS)*, as illustrated in Figure 2.2(a). However, due to performance limitations and security concerns, this *ad hoc network* is seldom used in practical network scenarios.
- *Infrastructure type:*
This type of WLAN involves exchanges of data between stations through an AP, as depicted in Figure 2.2(b). The AP serves as the central controller for all hosts within its basic service set (BSS) and is referred as the *infrastructure BSS*. To access network services in this type of WLAN, a host must be associated with an AP[37].

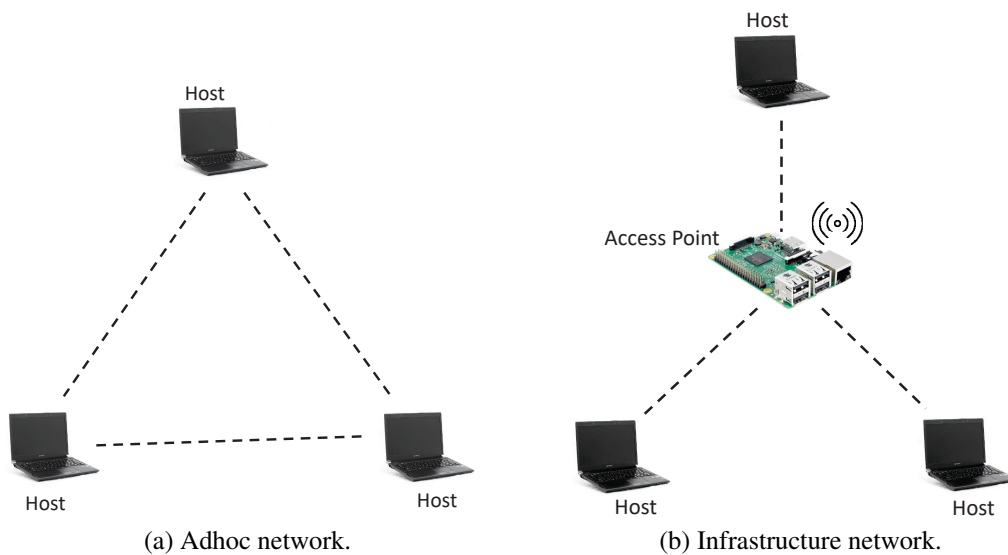


Figure 2.2: Types of *IEEE 802.11* networks.

Furthermore, WLAN can be expanded by linking multiple basic service sets (BSSs) through a backbone network to create an *extended service set (ESS)*, as illustrated in Figure 2.3. Each AP in the ESS is assigned a *service set identifier (SSID)*, which functions as a network name for users. Hosts within the same ESS can communicate with each other, even if they are situated in different BSSs.

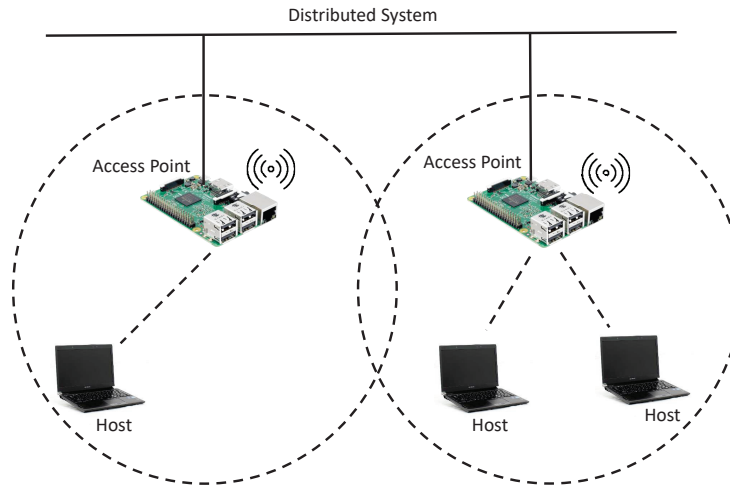


Figure 2.3: Extended service set (ESS).

2.1.4 IEEE 802.11 Standards for WLAN

The IEEE 802.11 working group has enhanced the current specifications of the Physical (PHY) and Media Access Control (MAC) layers to enable WLAN operation at the unlicensed ISM (*Industrial, Scientific and Medical*) frequency bands of 2.4 – 2.5GHz, 3.6GHz, and 5.725 – 5.825GHz, as defined by the ITU-R. Within this working group, numerous types of IEEE Standard Association Standards are available, each of which has a letter suffix and covers various aspects ranging from wireless standards to security, Quality of Service (QoS), and others as shown in Table 2.1 [37–42].

Table 2.1: *IEEE 802.11* standards.

Standard	Purpose
802.11a	Wireless network bearer operating in the 5GHz ISM band with up to 54Mbps of data transmission rate
802.11b	Operates on the 2.4 GHz ISM band, with data rates of up to 11Mbps
802.11c	Includes bridge operation that links LANs with similar or identical MAC protocol
802.11d	Support for additional regulations in different countries
802.11e	An enhancement to the WLAN 802.11a and 802.11b specifications in terms of QoS and prioritization
802.11f	Inter-Access Point Protocol for handover, the standard was withdrawn
802.11g	2.4GHz ISM band, maximum data rate of 54Mbps
802.11h	Transmit power control (TPC) and dynamic frequency selection (DFS)
802.11i	Authentication and encryption
802.11j	Standard of WLAN operation in the 4.9 to 5 GHz band to conform to the Japan's rules
802.11k	Measurement reporting and management of the air interface between several APs
802.11l	Reserved standard, to avoid confusion
802.11m	Provides an integrated view of the 802.11 base standard through continuous monitoring, management, and maintenance

Table 2.1: *IEEE 802.11* standards.

Standard	Purpose
802.11n	Operate in the 2.4 and 5 GHz ISM bands, data rates up to 600Mbps
802.11o	Reserved standard, to avoid confusion
802.11p	Provide wireless access for vehicular environments (WAVE)
802.11r	Fast BSS Transition, supports VoWiFi handoff between access points to enable VoIP roaming on a WiFi network with 802.1X authentication
802.11s	Wireless mesh networking
802.11t	Wireless Performance Prediction (WPP), this standard was cancelled
802.11u	Improvements related to "hotspots" and 3rd party authorization of clients
802.11v	To enable configuring clients while they are connected to the network
802.11w	Protected Management Frames
802.11x	Reserved standard, to avoid confusion
802.11y	Introduction of the new frequency band, 3.65-3.7GHz in US besides 2.4 and 5 GHz
802.11z	Extensions for direct link setup (DLS)
802.11aa	Specifies enhancements to the IEEE802.11 MAC for robust audio video (AV) streaming
802.11ac	Wireless network bearer operating below 6 GHz to provide data rates of at least 1Gbps for multi-station operation and 500Mbps on a single link
802.11ad	Wireless Gigabit Alliance (WiGig), providing very high throughput at frequencies up to 60GHz
802.11ae	Prioritization of management frames
802.11af	WiFi in TV spectrum white spaces (often called White-Fi)
802.11ah	WiFi uses unlicensed spectrum below 1GHz, smart metering
802.11ai	Fast initial link setup (FILS)
802.11aj	Operation in the Chinese Milli-Meter Wave (CMMW) frequency bands
802.11ak	General links
802.11aq	Pre-association discovery
802.11ax	High efficiency WLAN, providing 4x the throughput of 802.11ac
802.11ay	Enhancements for Ultra High Throughput in and around the 60GHz Band
802.11az	Next generation positioning
802.11mc	Maintenance of the IEEE802.11m standard

Figure 2.4 illustrates the current and future WiFi standards, which include IEEE 802.11a, 11b, 11g, 11n, 11ac, and the most recent 11ax. Among these standards, the most widely used and popular ones are 11a, 11b, 11g, 11n, and 11ac. For the physical layer, 11a/n/ac employ *orthogonal frequency division multiplexing (OFDM)* modulation, while 11b uses the *Direct Sequence Spread Spectrum (DSSS)* technology. 11g supports both technologies. The newest standard, 11ax, uses *orthogonal frequency division multiple access (OFDMA)*, which is a multi-user variant of OFDM. Table 2.2 summarizes the characteristics of these common WiFi standards[36, 43–46].

The security level of IEEE 802.11a, b, and g standards is considered moderate, as they employ

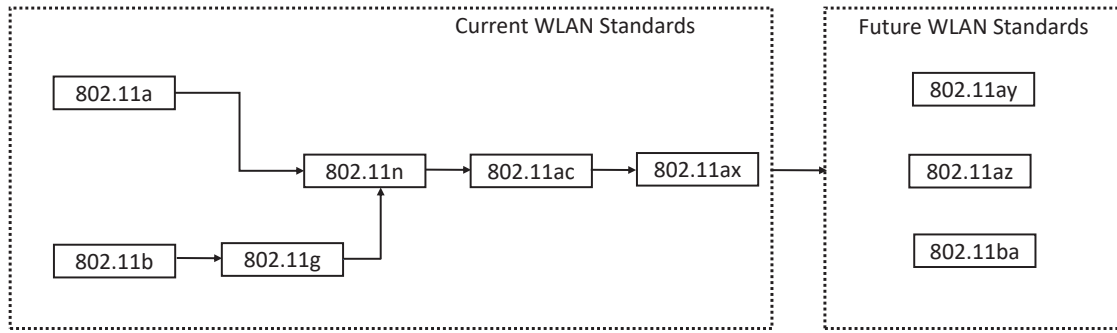


Figure 2.4: Current and future WLAN standards.

the *wired equivalent privacy (WEP)* security technology. The WEP encryption utilizes symmetric RC4 stream ciphers with either 40-bit or 104-bit encryption keys [47]. In contrast, the IEEE 802.11n, ac, and ax standards are considered to have stronger security as they use the advanced *Wi-Fi protected access (WPA)* encryption system called *temporal key integrity protocol (TKIP)* with *message integrity check (MIC)*

- *IEEE 802.11b*: The IEEE 802.11b standard was released in September 1, 1999 and has a maximum data rate of 11Mbps operating on the 2.4GHz frequency band. It is recognized for its robustness and the ability to accommodate the IEEE 802.11 protocol. This standard has not only spurred the development of WLAN products but also promoted competition among WLAN vendors due to its interoperability. However, this standard's drawback is the interference caused by other products utilizing the same *industrial, scientific, and medical (ISM)* band that operates on the 2.4GHz frequency [48, 49].
- *IEEE 802.11a*: The IEEE 802.11a standard also released in September 1, 1999. It operates at the 5GHz ISM band and offers data rates up to 54Mbps through the use of the *orthogonal frequency division multiplexing (OFDM)* coding scheme. However, the main limitations of 11a are the incompatibility of 11a products with 11b products and the unavailability of free 5GHz bands in all countries around the world [48, 49].
- *IEEE 802.11g*: The IEEE 802.11g standard was introduced in 2, 003 to enhance the 2.4GHz 11b standard. It builds on the 11a standard and introduces two modulation techniques: *packet binary convolution code (PBCC)* with data rates up to 54Mbps , and *orthogonal frequency division multiplexing (OFDM)* also with data rates up to 54Mbps . This standard also addresses compatibility issues between 11g and 11b products [48, 49].
- *IEEE 802.11n*: The IEEE 802.11n standard, released in 2, 009, aimed to enhance the range and data rate up to 600Mbps . It supports both the 2.4GHz and 5GHz ISM band as well as the *unlicensed national information infrastructure (UNII)* bands and is backward compatible with earlier standards. The 11n standard brought new technological improvements such as *channel bonding* and *multiple-input-multiple-output (MIMO)* that enable better reception of RF signals and enhance the throughput and coverage range [48, 50].
- *IEEE 802.11ac*: The IEEE 802.11ac standard, released in December 2, 013, which is exclusively designed for the 5GHz band and offers data throughput exceeding 1Gbps . It includes

Table 2.2: Features of common IEEE 802.11 standards.

	IEEE 802.11b	IEEE 802.11a	IEEE 802.11g	IEEE 802.11n	IEEE 802.11ac	IEEE 802.11ax
Release	Sep 1999	Sep 1999	Jun 2003	Oct 2009	Dec 2013	Feb 2021
Frequency Band	2.4 GHz	5 GHz	2.4 GHz	2.4/5 GHz	5 GHz	2.4/5/6 GHz
Max. Data Rate	11 Mbps	54 Mbps	54 Mbps	600 Mbps	1300 Mbps	9608 Mbps
Modulation	CCK ¹ modulated with PSK	OFDM	DSSS ² , CCK, OFDM	OFDM	OFDM	OFDMA
Channel Width	20 MHz	20 MHz	20 MHz	20/40 MHz	20/40/80/160 MHz	20/40/80/160 MHz
# of Antennas	1	1	1	4	8	8
security	WEP encryption	WEP encryption	WEP encryption	WPA encryption	WPA encryption	WPA encryption

¹ CCK: Complementary Code Keying

² DSSS: Direct Sequence Spread Spectrum

support for static and dynamic channel bonding, allowing for up to 160MHz bandwidth, and *multi-user multiple-input-multiple-output (MU-MIMO)* technology [9, 51, 52].

- *IEEE 802.11ax*: IEEE 802.11ax, also known as Wi-Fi 6, is the latest generation of Wi-Fi technology. It was approved in February 2021 and operates in the frequency bands between 1GHz and 7.125GHz. Wi-Fi 6 can support data rates of up to 9.6Gbps, which is four times faster than the previous generation, Wi-Fi 5 (802.11ac). Wi-Fi 6 focuses on improving throughput per area, or the ratio between the overall network throughput and the network area size. It supports channel bonding up to 160MHz, which allows for more efficient use of the wireless spectrum. Wi-Fi 6 also adopts *orthogonal frequency division multiple access (OFDMA)* technique, which is commonly used in cellular networks. OFDMA allows for multiple devices to communicate with an access point at the same time, which can improve overall network performance. However, only a few devices are currently compatible with this standard [10, 53–56].

2.2 IEEE 802.11n Protocol

This section provides an overview of the *IEEE 802.11n* protocol, which has been utilized for measuring throughput, proposing models, and implementing in this thesis. It incorporates various performance enhancement technologies, such as channel bonding (CB), Multiple Input Multiple Output (MIMO), frame aggregation, and security enhancements, surpassing the capabilities of the previous 11a, 11b, and 11g standards. Table 2.3 presents a summary of the main features of the *IEEE 802.11n* standard.

The IEEE 802.11n protocol can operate in both the 2.4GHz and 5GHz ISM frequency bands.

Table 2.3: IEEE 802.11n specifications.

Specification	IEEE 802.11n	
Frequency Band	2.4 GHz	5 GHz
Simultaneous Uninterrupted Channel	2 ch	9 ch
Available Channel	13 ch	19 ch
Max. Speed	600Mbps	
Max. Bandwidth	40 MHz	
Max. Spatial Streams	4	
Subcarrier Modulation Scheme	64 QAM	
Release Date	Sept 2009	

Currently, the 2.4GHz band is the most commonly used. However, it tends to become overcrowded due to the high number of WiFi signals utilizing the same channel or partially overlapping channels. As a result, WiFi signals that operate on adjacent channels will experience interference, leading to a decline in the network's overall throughput [45, 46, 56].

In the 2.4GHz band, the number of non-interfering channels is limited, which are Channel 3 and Channel 11 in 40MHz bandwidth. While for the 20MHz bandwidth, channel 1, channel 6 and channel 11 are free of interference with each other. However, the wider bandwidth will reduce the number of free channels. Figure 2.5 [42] describes the WiFi channels for *IEEE 802.11n* 2.4GHz band.

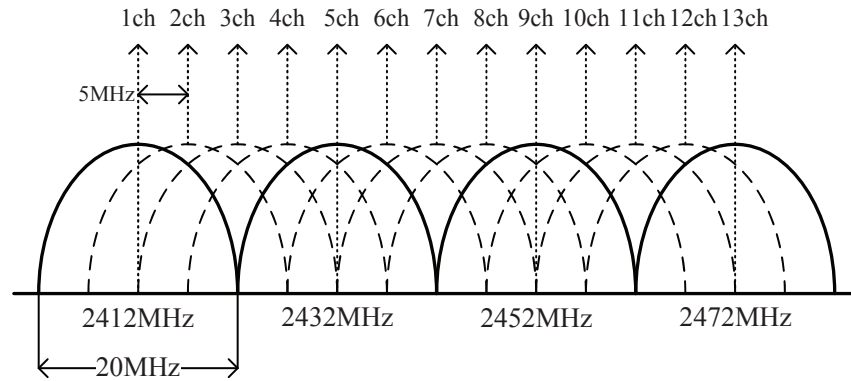


Figure 2.5: WiFi channels in 2.4 GHz band.

In the 5GHz band of the IEEE 802.11n protocol, it is available with 19 uninterrupted channels with a bandwidth of 20MHz and 9 channels with 40MHz bandwidth. For the 80MHz bandwidth, there are four of them. Figure 2.6 shows these WiFi channels for the *IEEE 802.11n* 5GHz band.

2.3 Features of IEEE 802.11n Protocol

The *IEEE 802.11n* protocol has integrated multiple new technologies to enhance its performance. To improve the throughput, the standard incorporates the use of multiple antennas technology, *channel bonding (CB)*, *frame aggregation*, and security improvements. This section outlines these features of the protocol in detail.

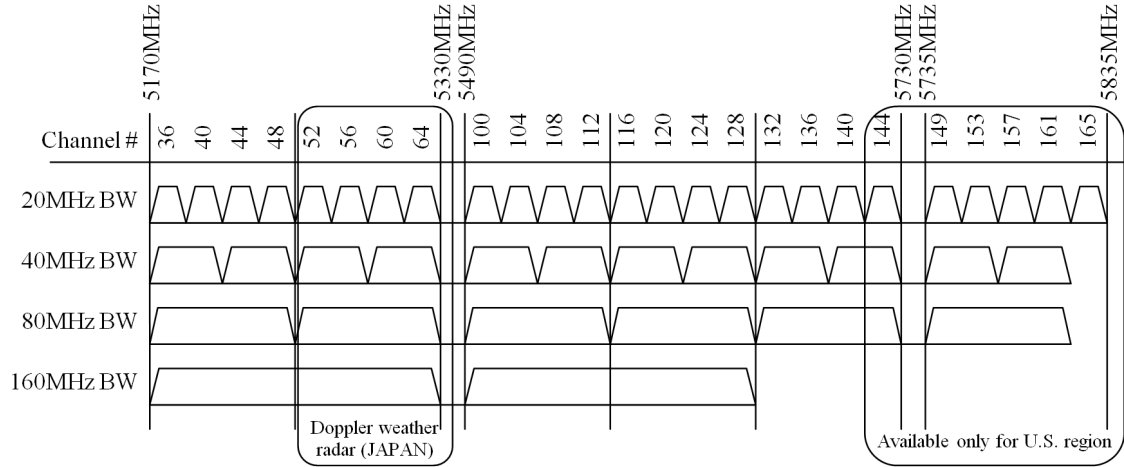


Figure 2.6: WiFi channels in 5 GHz band.

2.3.1 Channel Bonding

The *IEEE 802.11n* protocol has introduced CB as a new feature to improve network capacity. In the CB, two adjacent 20MHz channels are bonded together to form one 40MHz channel [8, 11, 12] as shown in Figure 2.7. However, the use of channel bonding may reduce the number of non-interfered channels for other devices. This is due to the fact that the *IEEE 802.11n* protocol has only two non-interfering bonded channels in the 2.4GHz band. Table 2.4 summarizes the CB for the 13, 20MHz channels at 2.4GHz band [46]. In Table 2.5, different channel bandwidths and spatial streams are described in regards to *IEEE 802.11n* throughput.

Table 2.4: Channel bonding in IEEE 802.11n.

20MHz		40MHz	
center frequency of primary channel	center frequency of secondary channel	bonded channel	center frequency of bonded channel
1	5	1+5	3
2	6	2+6	4
3	7	3+7	5
4	8	4+8	6
5	9	5+9	7
6	10	6+10	8
7	11	7+11	9
8	12	8+12	10
9	13	9+13	11

2.3.2 Partially Overlapping Channels

In *IEEE 802.11n*, at the 2.4GHz band, each channel has a width of 20MHz , and the two adjacent channel bands are separated by 5MHz . As a result, all the adjacent channels are partially overlapped with each other. Therefore, each channel is partially overlapped with a minimum of three neighboring channels.

Table 2.5: Effects of channel bandwidth and spatial stream's selection towards IEEE 802.11n's throughput.

Stream number	Bandwidth	
	20 MHz	40 MHz
one Streams	72.2Mbps	150Mbps
two Streams	144.4Mbps	300Mbps
three Streams	216.7Mbps	450Mbps
four Streams	288.9Mbps	600Mbps

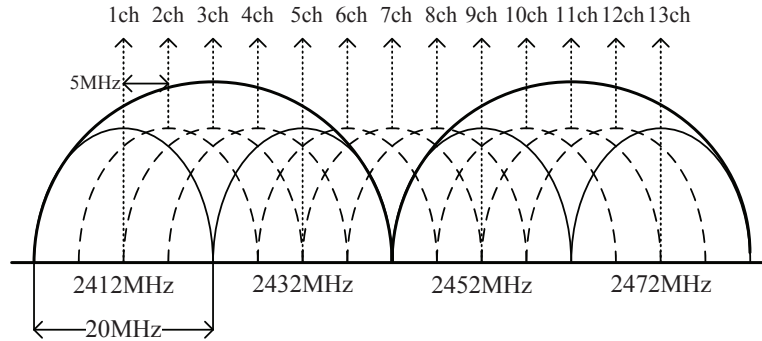


Figure 2.7: IEEE802.11n channel bonding concept.

2.3.3 Multiple Input Multiple Output (MIMO)

The *IEEE 802.11n* standard incorporates MIMO as another feature to significantly enhance throughput. MIMO increases the number of transmitting (T_X) and receiving (R_X) antennas by up to four times, without requiring additional bandwidth or transmission power. By doing so, it overcomes the impact of multi-path and fading, and achieves high data throughput even in limited bandwidth channels. Compared to *single-input single-output (SISO)* single antenna technology, MIMO provides improved performance. The comparison between SISO and 4×4 MIMO as shown in Figure 2.8. When the *space-time block coding (STBC)* is applied in the 4×4 MIMO link, the sender can transmit four copies of the data streams over four antennas to enhance the reliability and the effective range of data transmissions.

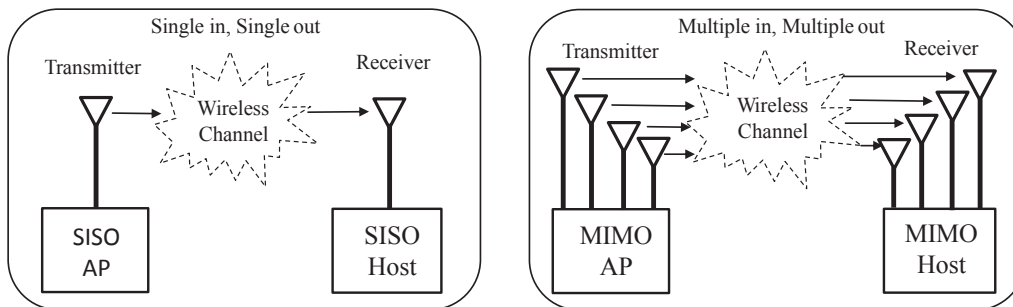


Figure 2.8: Comparison between SISO and 4×4 MIMO technology.

2.3.4 MAC Layer Enhancements

In addition to channel bonding and MIMO, the *IEEE 802.11n* protocol also enhances performance through *frame aggregation* and the appropriate selection of *modulation and coding scheme (MCS)*.

- *Frame Aggregation:*

The *IEEE 802.11n* protocol offers additional performance enhancement through frame aggregation at the MAC layer, in addition to MIMO. Frame aggregation allows multiple frames to be transmitted as a single large frame, with a single preamble and header information, reducing overhead. The protocol introduces two types of aggregation: *Aggregation of MAC service data units (A-MSDUs)* and *aggregation of MAC protocol data units (A-MPDUs)*. *Frame aggregation* involves packing multiple A-MSDUs and A-MPDUs together, averaging them over multiple frames, and reducing overheads, ultimately increasing the user-level data rate [57].

- *Modulation and Coding Scheme:*

The *IEEE 802.11n* standard defines a range of *modulation and coding scheme (MCS)*, which represent different combinations of modulation and error-correction codes. There are 31 MCS modes in *IEEE 802.11n*, which use *Orthogonal Frequency Division Multiplexing (OFDM)* to provide enhanced protection against selective fading. The standard also increases the number of OFDM sub-carriers to 56 (52 usable) in *high throughput (HT)* mode with 20MHz channel width, and 114 (108 usable) in HT mode with 40MHz. Each of these sub-carriers can be modulated using *BPSK*, *QPSK*, *16-QAM*, or *64-QAM*, and *low-density parity-check code (LDPC) forward error correction (FEC)* coding rate of 1/2, 2/3, 3/4, or 5/6 is applied. The selection of coding rates is dependent on the quality of the link between the hosts. A high-quality link can support higher coding rates, allowing for the transmission of more data. Conversely, a link with poor radio conditions will necessitate the use of lower coding rates, resulting in a lower amount of data being transmitted [58].

2.4 IEEE 802.11ac Protocol

This section provides an overview of the *IEEE 802.11ac* protocol, which has been used for measuring throughput, proposing models, and implementing in this thesis. The standard provides faster wireless network speeds than its predecessor, *IEEE 802.11n*, by incorporating wider channel bandwidths, up to 160MHz, and higher order modulation schemes, up to 256-QAM. Additionally, *IEEE 802.11ac* introduces *Multi-User Multiple Input Multiple Output (MU-MIMO)*, which enables multiple devices to transmit and receive data simultaneously. Table 2.6 presents a summary of the main features of the *IEEE 802.11ac* standard [9, 59].

2.5 Features of IEEE 802.11ac Protocol

The performance of the *IEEE 802.11ac* protocol has been significantly improved through the integration of multiple new technologies. To enhance its throughput, the protocol utilizes multiple antennas technology, channel bonding, and beamforming that enhance signal quality and range. Additionally, the protocol supports *Multi-User Multiple Input Multiple Output (MU-MIMO)* which

Table 2.6: IEEE 802.11ac specifications.

Specification	IEEE 802.11ac
Frequency Band	5 GHz
Available Channel	19 ch
Max. Speed	6.93 Gbps
Max. Bandwidth	160MHz
Max. Spatial Streams	8
Subcarrier Modulation Scheme	256 QAM
Release Date	Dec 2013

enables simultaneous communication among multiple devices. Other improvements include enhanced network security and Quality of Service (QoS) [9, 59]. This section provides a detailed explanation of these features.

2.5.1 Channel Bonding

The CB was introduced in the IEEE 802.11n amendment for improving the network performance of WLAN. The IEEE 802.11ac further extends this feature by allowing 80MHz or 160MHz channels with four or eight single channels respectively. These channels are classified into primary and secondary channels. The primary channel is the channel that is used for the majority of data transmission. The secondary channels are used for additional data transmission and to help mitigate interference from other wireless networks. Depending on the width of the channel obtained through channel bonding, there can be primary 20 MHz, secondary 20 MHz, primary 40 MHz, secondary 40 MHz, primary 80 MHz, and secondary 80 MHz channels. The primary 40 MHz and 80 MHz channels include the primary 20 MHz and 40 MHz channels, respectively. Figure 2.9 shows this organization [9, 13].

2.5.2 Non Channel Bonding

In *IEEE 802.11ac*, each channel has a width of 20 MHz. It has a total of 19 channels, which are all separated from each other by at least 20 MHz. This means that there is no overlap between any of the channels, which helps to reduce interference.

2.5.3 Multi-User Multiple Input Multiple Output (MU-MIMO)

The *IEEE 802.11ac* standard defines *MU-MIMO* as a technology that enables multiple stations with multiple antennas to simultaneously transmit and/or receive independent data streams. This means that stations with multiple antennas can transmit multiple data streams to multiple users simultaneously over the same frequency channel. For instance, an AP with four antennas can serve four users with single antennas or two users with two antennas each, by sending up to one data stream per receiving antenna over the same frequency [9, 13]. The comparison between Single User MIMO and Multi User MIMO as shown in Figure 2.10. *Single User MIMO (SU-MIMO)* technology operates by sending or receiving multiple streams of data between just one device at a time, requiring both the transmitting and receiving Wi-Fi radios to support MIMO technology with multiple antennas. In contrast, *Multi-User MIMO (MU-MIMO)* technology enables Wi-Fi to transmit multiple data streams simultaneously to different Wi-Fi devices, instead of just one

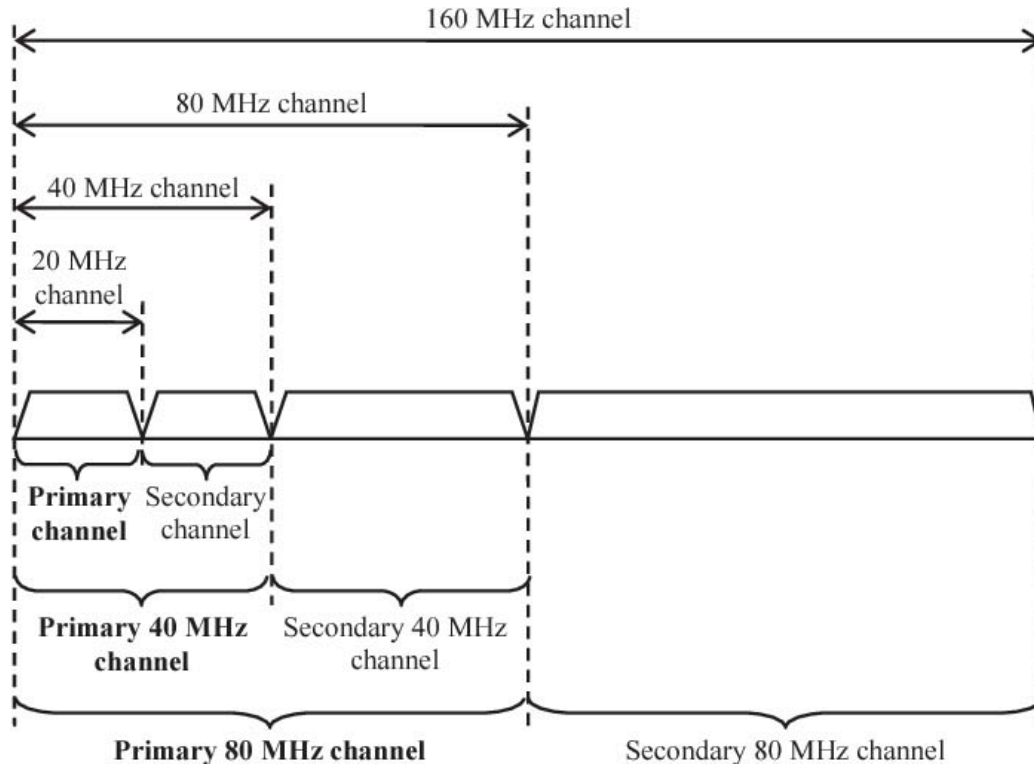


Figure 2.9: Primary and secondary channels in 802.11ac

device at a time as with SU-MIMO. Additionally, MU-MIMO can transmit to Wi-Fi devices with a single antenna, making it more versatile. However, the receiving Wi-Fi devices must also support MU-MIMO technology for this to work.

2.6 Linux Tools for Wireless Networking

The open-source operating system, *Linux* has played a significant role in the development of wireless networks by facilitating the implementation of algorithms, protocols, methods, and devices [60]. This section provides an overview of the tools and software utilized in the thesis for conducting measurements and implementing the *elastic WLAN system*.

2.6.1 ‘arp-scan’ - to Explore Currently Active Devices

The command-line tool *arp-scan* [61] utilizes the *address resolution protocol (ARP)* to identify and fingerprint the IP addresses of hosts on the local area network. This tool is capable of functioning on both IEEE802.11 wireless networks and wired Ethernet networks that use the same data-link protocol. In *Linux*, *arp-scan* can be installed by downloading the source code from [62] or using the following command:

```
$ sudo apt-get install arp-scan
```

The simplest command to scan the network using *arp-scan* is given by:

```
$ arp-scan --interface=wlan0 --localnet
```

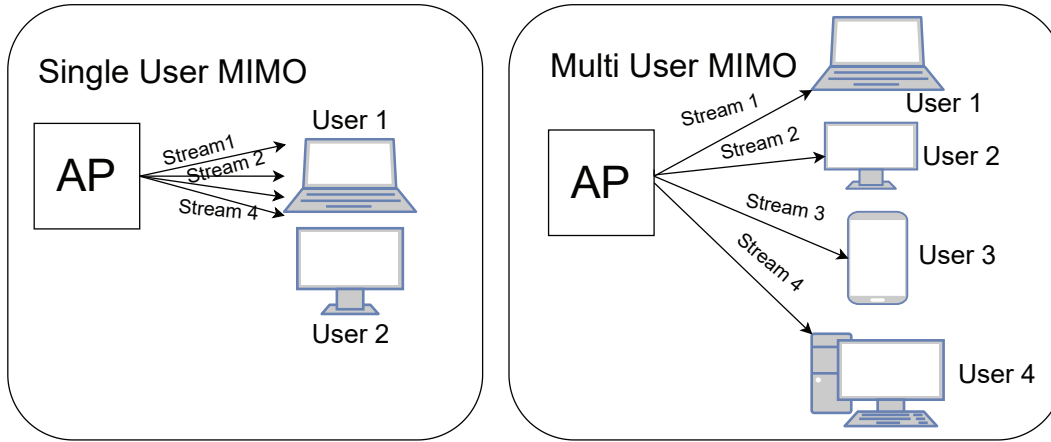


Figure 2.10: Comparison between Single User MIMO and Multi User MIMO

`-interface=eth0` specifies the interface to be used for scanning devices. The `arp-scan` uses `-localnet` to scan all the possible IP addresses in the network that connect to the interface, which is defined by the interface IP address and net mask. Here, the interface `wlan0` is used as an example.

2.6.2 ‘nm-tool’ - to Collect Host Information

`nm-tool` is used to obtain information about the devices in the local area network, including the wireless network [63, 64]. In this thesis, we use `nm-tool` to gather information about the host, such as the currently associated AP, the associable APs, and *signal strength* received from each AP. This tool is installed as a part of the *NetworkManager* package on the *Linux* operating system [65]. The *NetworkManager* package is typically pre-installed on the Ubuntu distribution of *Linux*. It can be installed using the following command manually:

```
$ sudo apt-get install network-manager
```

The simple way to run `nm-tool` is:

```
$ nm-tool
```

2.6.3 ‘hostapd’ - to Make AP-mode Linux-PC

Hostapd is a software that allows a network interface card to function as an AP and authentication server. It supports IEEE 802.11 AP management, along with other IEEE 802.1X protocols and security applications. In *Linux*, *hostapd* can be installed by downloading the source code from [66] or using the following command:

```
$ sudo apt-get install hostapd
```

After installing this tool on a *Linux* PC with WLAN driver support for AP mode, it is possible to set up the Linux PC as an access point using a command-line interface. The *hostapd* can be started or stopped by the following commands:

```
$ sudo /etc/init.d/hostapd start
$ sudo /etc/init.d/hostapd stop
```


2.6.4 ‘ssh’ - to Remotely Execute Command

The *Secure Shell (ssh)* is a network protocol that uses encryption to establish a secure connection between a client and a server, allowing a user to start a remote shell session securely on a distant machine [67, 68]. It comprises two components, namely, the *SSH client* and *SSH server*, and ensures secure communication between them over an insecure network. The open source version of *ssh* is *OpenSSH* [69], it can be installed using the following command [70]:

```
$ sudo apt-get install openssh-server openssh-client
```

Below is an example of how to remotely access the AP over the network using *ssh* command line [67, 68, 71]:

```
$ ssh username@192.168.40.21
username@192.168.40.21's password:
```

Here, 192.168.40.21 represents the IP address of the AP.

2.6.5 ‘nmcli’ - to Change Associated AP

nmcli [72] is a command-line Linux tool to manage, configure, and control the powerful *NetworkManager* package. *nmcli* is pre-included in the *NetworkManager* package. *nmcli* is used to associate a host with an AP through the following command line:

```
$ sudo -s nmcli dev wifi connect NewSSID password PASSWORD
```

The above command connects the host to the AP specified with *NewSSID* using the corresponding password *PASSWORD* of the AP.

2.6.6 ‘iwconfig’ - to Collect Information of Active Network Interface

The *iwconfig* [73] is a *Linux* tool that operates through the command-line interface and enables viewing and modifying the parameters of the active network interface for wireless operations. This tool is typically installed by default in the Ubuntu distribution. It can also be installed manually using the following command:

```
$ sudo apt-get install wireless-tools
```

The following list shows the use of *iwconfig* to display the information of the currently associated AP using the network interface *wlan0*:

```
$ iwconfig wlan0
```

2.6.7 ‘sed-tool’ - to Assign the Channels to the APs

sed is a useful text processing feature of *Linux*. The full form of *sed* is Stream Editor [74]. In this thesis, we used this command for assigning the channels to the APs using the following command:

```
sed -i -e 's/. * channel. * /channel ='NewChannel'/' /etc/hostapd/hostapd.conf
```

Here, ‘s’ represents the substitution command and *NewChannel* does the channel to be assigned in the *hostapd.conf* file of the AP.

2.6.8 ‘iperf’ - to Measure Link Speed

The *iperf* software is used to measure the available bandwidth or throughput on IP networks. It supports both TCP and UDP protocols [75]. It is commonly included as a default installation in the *Ubuntu* distribution. It can also be installed manually using the following command:

```
$ sudo apt-get install iperf
```

In order to measure the TCP throughput between two devices using *iperf*, one device must run the server-mode and the other device must run the client-mode. The packets will be transmitted from the client to the server. The output of *iperf* contains a time-stamped report of the transmitted data amount and the measured throughput. The following list shows the typical use of *iperf* on the server and client side:

```
$ iperf -s //server side  
$ iperf -c 192.168.40.21 //client side
```

Here, 192.168.40.21 represents the IP address of the server. In this thesis, we use *iperf* to measure the throughput between an AP and a host through the *IEEE 802.11n* and *11ac* protocol.

2.7 Summary

This chapter provided an overview of wireless network technologies, highlighting the important features of *IEEE 802.11n* and *IEEE 802.11ac* protocols, as well as *Linux* tools that are used in this study. The next chapter will focus on reviewing previous studies related to this thesis.

Chapter 3

Review of Previous Studies

This chapter provides an overview of our previous studies that are related to the thesis. Firstly, we discuss the elastic WLAN system. Secondly, we review the study of the throughput estimation model in WLAN. Thirdly, we review the parameter optimization tool that is used to optimize the parameter values for the throughput estimation model. Finally, we review the study of the active AP configuration algorithm for the elastic WLAN system.

3.1 Elastic WLAN System

This section provides the overview of the elastic WLAN system. It utilizes the dynamic network configuration optimization by activating or deactivating APs based on traffic demands and network conditions. Thus, it can effectively reduce energy consumption, minimizes interference, and enhances overall throughput performance.

3.1.1 Overview

Nowadays, WLANs have become widely deployed in many places, including educational institutions and public places such as buses, airplanes, and trains. However, unplanned or independently controlled APs can lead to issues of decreased throughput and energy wastage. Then, WLANs may suffer from over-allocation problems due to redundant APs with overlapping coverage areas. Besides, WLANs can become overloaded, resulting in poor performance for connected hosts. Therefore, it is crucial for WLANs to be adaptive and responsive to network traffic demands and conditions by adjusting the number of active APs and the associated hosts. To address this need, we have studied the *elastic WLAN system*, which aims to optimize WLAN performances by dynamically managing APs and enhancing network efficiency.

The motivations for studying the elastic WLAN system are outlined below:

1. Reduction of operational costs and energy consumption:
 - Companies, educational institutions, and offices often deploy a large number of APs to ensure high WLAN performance during peak hours, resulting in APs being active throughout the day. However, during off-peak hours or holidays, only a fraction of these APs are utilized. The elastic WLAN system addresses this issue by dynamically adjusting the number of active APs based on traffic demands, resulting in a reduction in both the number of APs required and energy consumption.

- In developing countries, unstable Internet connections are often experienced due to frequent power supply disruptions. The elastic WLAN system offers a solution by optimizing power usage, leading to improved network performance. By efficiently managing power consumption, the system ensures a more stable and reliable Internet connection, even in areas with unreliable electricity supply.

2. Improving WLAN performance:

- When the existing active APs are insufficient to cover all users, the additional APs will be necessary to meet the required traffic demands and ensure the optimal WLAN performance.
- If the WLAN performance becomes low due to a shortage of *internet service provider (ISP)* connections or power supplies, it will switch to cellular networks using *mobile APs* to maintain the required performance.
- In a dense WLAN, as the number of access points (APs) increases, the interference caused by signal overlaps can lead to a decrease of the throughput. The elastic WLAN system addresses this issue by dynamically adjusting the assigned orthogonal channels of APs. This allows for a reduction in interferences between APs and ultimately improves the performance of the WLAN.

3.1.2 Design and Operational Flow

The *elastic WLAN system* has been designed and implemented to dynamically optimize the network topology and the configuration in accordance with the network conditions. Figure 3.1 depicts an example topology of the system.

The elastic WLAN system implementation adopts the *management server* to manage and control the hosts and the APs by running the *active AP configuration algorithm* [6]. This server has not only administrative access to all devices in the network but also controls the overall network using the following three-step procedure:

1. It explores all network devices and also collects the information that are required for the active AP configuration algorithm.
2. After that, it executes the active AP configuration algorithm through the derived inputs from previous step. The output of the algorithm contains the list of the active APs, the host associations, and the assigned channels of each active APs.
3. Finally, it applies the algorithm output into the network by changing the host associations, activating or deactivating the specified APs, and allocating the channels to the APs.

3.2 Throughput Estimation Model

The link speed or throughput between an access point (AP) and a host can be influenced by several factors, including the modulation and coding scheme, the transmission power, the transmission distance, and the presence of obstacles. The accurate calculation of the theoretical link speed can be a complex task [28, 76]. This section provides an overview of the throughput estimation model in WLANs.

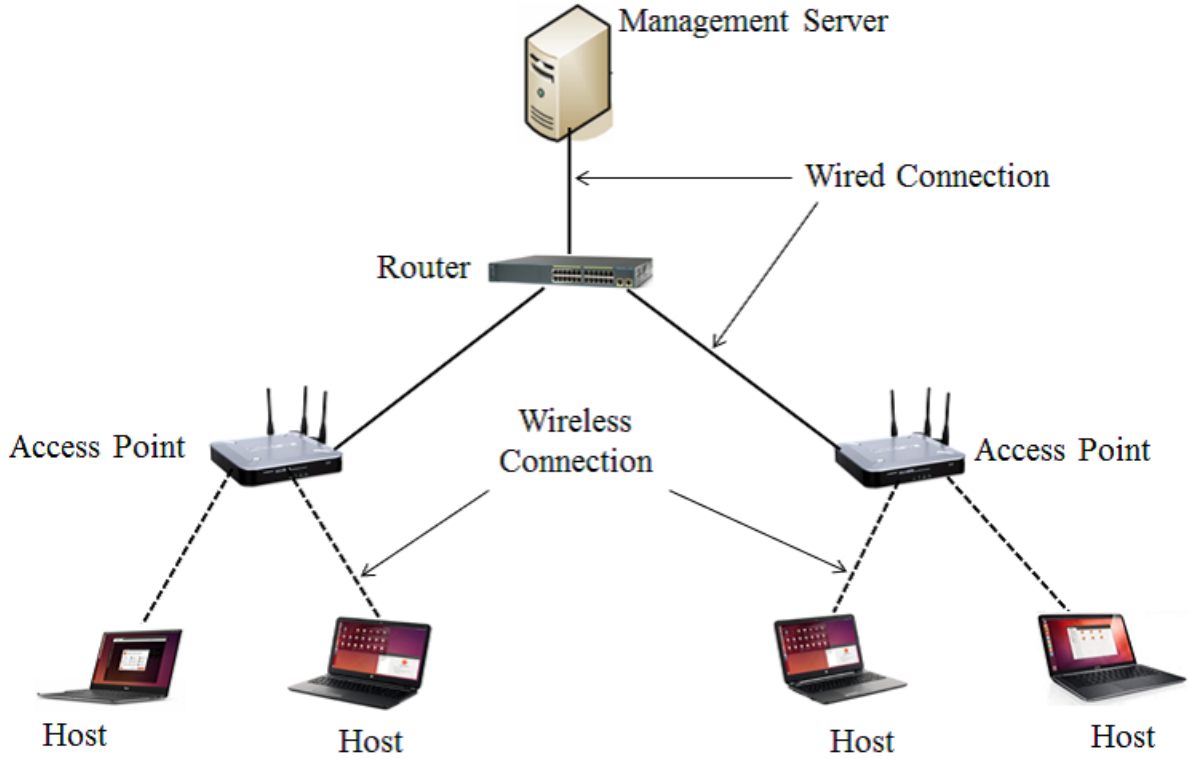


Figure 3.1: Design of elastic WLAN system.

3.2.1 Overview of Model

The throughput estimation model is used to estimate the throughput of a wireless communication link between a source node and a destination node in WLAN. First, it estimates the *received signal strength (RSS)* using *log distance path loss model* at the destination node [76]. Then, the sigmoid function is used to convert the RSS into throughput.

3.2.2 Received Signal Strength Estimation

First, the Euclidean distance d (m) is calculated for each link (AP/host pair) by:

$$d = \sqrt{(AP_x - H_x)^2 + (AP_y - H_y)^2} \quad (3.1)$$

where AP_x , H_x represent the x coordinate of the AP and the host, and AP_y , H_y do the y coordinate, respectively. After that, the *received signal strength (RSS)* P_d (dBm) is estimated at the host by:

$$P_d = P_1 - 10\alpha \log_{10} d - \sum_k n_k W_k \quad (3.2)$$

where P_1 represents RSS from the AP to the host at the 1m distance when there is no obstacle, α represents the path loss exponent, n_k is the number of *type k* obstacles or walls along the route between the AP and the host, and W_k represents the signal attenuation ratio (dBm) for *type k* ($k=1, 2, \dots, 6$) obstacle. It is noted that in a building, there are several types of walls that have different thickness and materials, which makes the different signal attenuation factor.

The value of α depends on the specific environment in which the radio signal is propagating. It represents the amount of loss that the signal will experience as it propagates through the environment. *type_k* obstacles represents the obstacle or wall that has the k-th signal attenuation factor W_k . In this paper, we consider six types of obstacles, namely, the corridor wall for W_1 , the partition wall for W_2 , the intervening wall for W_3 , the glass wall for W_4 , the elevator wall for W_5 , and the door for W_6 [29].

3.2.3 Throughput Estimation from Received Signal Strength

From *received signal strength* P_d , the throughput/link speed is calculated by using the *sigmoid function* [29],

$$tp = \frac{a}{1 + e^{-\left(\frac{(120+P_d)-b}{c}\right)}} \quad (3.3)$$

where a , b , and c are the constant parameters of the sigmoid function that should be optimized by parameter optimization tool [30]. The assumption of the sigmoid function for the throughput estimation is based on our real-world measurement results which clearly reflects the relationship between the RSSs and the estimated throughput. Figure 3.2 demonstrates the sigmoid function curve using $a = 84$, $b = 56.5$ and $c = 6.5$.

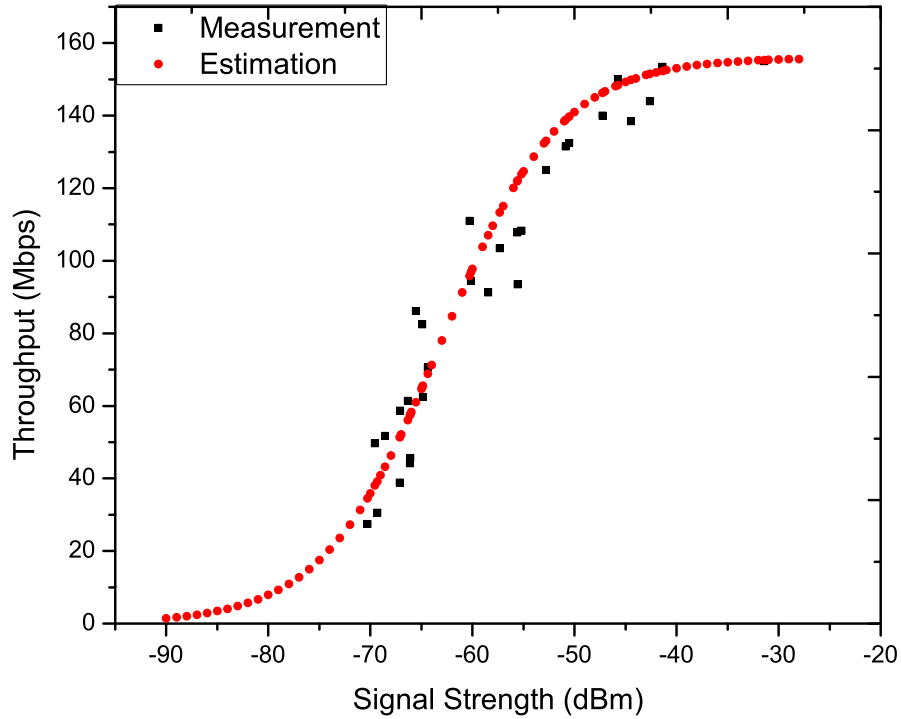


Figure 3.2: Sigmoid function for throughput estimation from signal strength.

3.2.4 Throughput Reduction Factor

When multiple hosts are communicating with an AP simultaneously, the *throughput reduction factor* is introduced in the throughput estimation model to improve the estimation accuracy [77]. It is calculated by:

$$tp \times srf(m) \quad (3.4)$$

where m is the number of hosts communicating with AP, tp represents the estimated single link throughput between AP_i and $host_j$, and $srf(m)$ is the empirically derived contention factor at AP_i among the associated hosts for sending data through the CSMA/CA protocol. $srf(m)$ is calculated by

$$srf(m) = \left(\frac{1}{m + \frac{0.1(m-1)}{4}} \right) \times 1 - (0.1 \times m - 1) \quad (3.5)$$

3.3 Parameter Optimization Tool

This section introduces the parameter optimization tool. It uses measurement results to optimize the parameter values for the throughput estimation model.

3.3.1 Overview

The throughput estimation model has several parameters whose values may determine the estimation accuracy. In this thesis, the parameter values of the model are optimized using the *parameter optimization tool*. This tool has been designed in a general form, making it applicable to various algorithms or logics that require parameter optimization. Typically, the parameter optimization tool and the throughput estimation model are implemented as separate programs. The tool operates the model program as a child process and evaluates the effectiveness of the current parameter values based on the throughput estimation error provided in the output file [30].

3.3.2 Required Files for Tool

A user of the parameter optimization tool needs to prepare the following five files.

3.3.2.1 Parameter Specification File

The *parameter specification file* provides instructions on how to adjust the values of each parameter during the search process. The file name for this specification should be “parameter.csv”. Each line in the file corresponds to a specific parameter and must follow the order of “parameter name”, “initial value”, “lower limit”, “upper limit”, and “change step”.

The following example shows the specifications for three parameters, x , y , and z .

— Example of parameter.csv. —

```
01: x, 50, 0, 100, 5
02: y, -20, -50, 0, 2
03: z, 0.5, 0.0, 1.0, 0.1
```

3.3.2.2 Model Program File

The *model program file* is a code file to run the throughput estimation model, which could be executed through the command line. Any name is possible for this file. In addition, this file must satisfy the following two conditions:

- (1) When the program is executed, it receives the path for the parameter file in the argument and applies the parameter values in the file to the model.
- (2) When the program is completed, it outputs the score as the evaluation value in the text file “result.txt”.

With (1), the model program can read the parameter values that are generated by the tool. With (2), the tool program can read the score that is calculated in the model program. For example, the parameter file for the previous example with three parameters is explored when their initial ones are used.

3.3.2.3 Sample Input Data File

The *sample input data file* contains the input data set to the model program such that the result of the program is evaluated and used to optimize the parameter values by the tool. To improve the accuracy of the obtained parameter values, multimodal sample input data sets should be collected and adopted in the tool.

3.3.2.4 Score Output File

The *score output file* involves the score from the model program to evaluate the current parameter values. The score is given by the difference between measured and estimated throughputs in the throughput estimation model.

3.3.2.5 Script File for Execution

The *script file* describes the sequence of the commands to execute the model program. The file name must be “run.sh”. This file also describes the paths to the input files to the model program. By modifying this script file, the user is able to change the name and the arguments for the model program, and can run multiple programs sequentially to obtain one score. In the following example, line 10 executes the model program, which is coded in Java, with three paths to the input files. When the model program is executed with multiple sample input data files continuously, the array to describe these files should be prepared and the loop procedure should be adopted.


```

01: #!/bin/bash
02: DIR=$(cd $(dirname $0); pwd)
03: PARAMETER_FILE=$1
04: cd "$DIR"
05: #folder for sample input data files
06: INPUT_DIR="./file/3f_ap1/"
07: #measurement data file
08: MEASUREMENT_FILE="./file/evaluate/indoor_3f_ap1.csv"
09: #execute the model program
10: java -jar ThroughputEstimation.jar
   ${INPUT_DIR} ${MEASUREMENT_FILE}
   $PARAMETER_FILE
11: #move \enquote{result.txt} to the folder containing the parameter file
12: mv result.txt 'dirname $PARAMETER_FILE'

```

3.3.3 Processing Flow of Tool

The processing flow of the tool is described as follows:

- (1) The parameter optimization tool (T) generates the initial parameter file by copying the initial values in the parameter specification file.
- (2) T executes the script file using the current parameter file.
 - (2-a) The model program (M) reads one sample input data file.
 - (2-b) M computes the algorithm/logic.
 - (2-c) M writes the score in the score output file.
- (3) T reads the score from the score output file.
- (4) When the termination condition is satisfied, T goes to (5). Otherwise, T goes to (6).
- (5) T changes the parameter file based on the algorithm in the next section, and goes to (2).
- (6) T selects the parameter values with the best score and outputs it.

3.3.3.1 Symbols in Algorithm

The following symbols are used in the parameter optimization algorithm in the tool:

- P : the set of the n parameters for the algorithm/logic in the model program whose values should be optimized.
- p_i : the i th parameter in P ($1 \leq i \leq n$).
- Δp_i : the change step for p_i .
- t_i : the tabu period for p_i in the tabu table.

- $S(P)$: the score of the algorithm/logic using P .
- P_{best} : the best set of the parameters.
- $S(P_{best})$: the score of the algorithm/logic where P_{best} is used.
- L : the log or cache of generated parameter values and their scores.

3.3.4 Parameter Optimization Algorithm

In this section, we describe the parameter optimization algorithm adopted in the *parameter optimization tool* [30].

3.3.4.1 Algorithm Procedure

The algorithm procedure that minimizes the score is described as follows:

- (1) Clear the generated parameter log L .
- (2) Set the initial value in the parameter file for any p_i in P , and set 0 for any tabu period t_i , and set a large value for $S(P_{best})$.
- (3) Generate the neighborhood parameter value sets for P by:
 - (3-a) Randomly select one parameter p_i for $t_i = 0$.
 - (3-b) Calculate parameter values of p_i^- and p_i^+ by:

$$\begin{aligned} p_i^- &= p_i - \Delta p_i, \text{ if } p_i > \text{lower limit,} \\ p_i^+ &= p_i + \Delta p_i, \text{ if } p_i < \text{upper limit.} \end{aligned}$$

- (3-c) Generate the neighborhood parameter value sets P^- and P^+ by replacing p_i by p_i^- or p_i^+ :

$$\begin{aligned} P^- &= \{p_1, p_2, \dots, p_i^-, \dots, p_n\} \\ P^+ &= \{p_1, p_2, \dots, p_i^+, \dots, p_n\} \end{aligned}$$

- (4) When P (P^- , P^+) exists in L , obtain $S(P)$ ($S(P^-)$, $S(P^+)$) from L . Otherwise, execute the model program using P (P^- , P^+) to obtain $S(P)$ ($S(P^-)$, $S(P^+)$), and write P and $S(P)$ (P^- and $S(P^-)$, P^+ and $S(P^+)$) into L .
- (5) Compare $S(P)$, $S(P^-)$, and $S(P^+)$, and select the parameter value set that has the largest one among them.
- (6) Update the tabu period by:
 - (6-a) Decrement t_i by -1 if $t_i > 0$.
 - (6-b) Set the given constant tabu period TB for t_i if $S(P)$ is largest at (5) and p_i is selected at (3-a).

- (7) When $S(P)$ is continuously largest at (5) for the given constant times, go to (8). Otherwise, go to (3).
- (8) When the hill-climbing procedure in (9) is applied to the given constant times HT , terminate the algorithm and output P_{best} . Otherwise, go to (9).
- (9) Apply the hill-climbing procedure:
 - (9-a) If $S(P) < S(P_{best})$, update P_{best} and $S(P_{best})$ by P and $S(P)$ respectively.
 - (9-a) Reset P by P_{best} .
 - (9-c) Randomly select p_i in P , and randomly change the value of p_i within its range.
 - (9-d) Go to (3).

3.4 Active AP Configuration Algorithm

This section presents the review of the *active AP configuration algorithm* used in the elastic WLAN system. The algorithm focuses on optimizing both the number of active access points (APs) and the associations with hosts [6, 28].

3.4.1 Problem Formulation

The active AP configuration problem for this algorithm is formulated as follows:

1. Inputs:

- Number of hosts: H
- Number of APs: $N = N^D + N^V + N^M$ where N^D , N^V , and N^M represent the number of DAPs, VAPs, and MAPs respectively.
- Link speed between AP_i and $host_j$ for $i = 1$ to N , $j = 1$ to H : s_{ij} , where the link speed can be estimated by the model in [29].
- Minimum link speed for association: S
- Number of non-interfered channels: C

2. Outputs:

- Set of active APs
- Set of hosts associated with each active AP
- Channel assigned to each active AP

3. Objectives:

- To minimize E_1 .
- Holding the first objective, to maximize E_2 .
- Holding the two objectives, to minimize E_3 for channel assignments.

Let, E_1 represents the number of active APs (DAPs, VAPs, and MAPs) in the network:

$$E_1 = E_1^D + E_1^V + E_1^M \quad (3.6)$$

where E_1^D represents the number of active DAPs, E_1^V does the number of active VAPs, and E_1^M does the number of active MAPs respectively.

The transmission delay of the j th AP can be defined by:

$$T_j = \sum_{k \in \mathcal{P}_j} \frac{D_k}{s_{jk}} \quad (3.7)$$

where D_k represents the traffic of the k th host, s_{jk} does the link speed between the j th AP to the k th host, and \mathcal{P}_j does the set of the hosts associated with the j th AP. Then, the average throughput TH_{ij} of the i th host associated with the j th AP can be estimated by:

$$TH_{ij} = \frac{D_i}{T_j} = \frac{D_i}{\sum_{k \in \mathcal{P}_j} \frac{D_k}{s_{jk}}} \quad (3.8)$$

Since the real traffic of each host is unpredictable, we assume the identical traffic for every host, which can be represented by the unit traffic for the sake of simplicity. Then, the average host throughput for AP_j is given by:

$$TH_j = \frac{1}{\sum_{k \in \mathcal{P}_j} \frac{1}{s_{jk}}} \quad (3.9)$$

If $TH_j \geq G$, the minimum host throughput constraint is satisfied, where G represents the threshold for this constraint. Then, the second objective function E_2 is defined to maximize the *minimum average host throughput* for the bottleneck, AP which is given by:

$$E_2 = \min_j [TH_j] \quad (3.10)$$

E_3 signifies the total interfered communication time:

$$E_3 = \sum_{i=1}^N [IT_i] = \sum_{i=1}^N \left[\sum_{\substack{k \in I_i \\ c_k = c_i}} T_k \right] \quad (3.11)$$

where IT_i represents the *interfered communication time* for AP_i , T_i does the *total communication time* for AP_i , I_i does the *set of the interfered APs* with AP_i , and c_i does the *assigned channel* to AP_i . They are given by follows:

- T_k is given by the sum of the time that is required to transmit one bit data between AP_k and its each associated host.
- I_i represents the set of the indices of the APs that are interfered with AP_i if they are assigned the same channel.
- c_k signifies the channel assigned to AP_k by the active AP configuration algorithm.
- IT_i is given by the sum of the total communication time for the APs that are interfered with AP_i .

4. Constraints:

- Minimum host throughput constraint: each host in the network must enjoy the given threshold G on average when all the hosts are communicating simultaneously.
- Bandwidth limit constraint: the bandwidth of the wired network to the Internet must be less than or equal to the total available bandwidth of the network B^a .
- Channel assignment constraint: each AP must be assigned one channel between 1 and C .

3.4.2 Algorithm Procedure

The active AP configuration algorithm consists of three phases: the *active AP and associated host selection* phase, the *channel assignment* phase, and the *channel load averaging* phase.

3.4.2.1 Active AP and Associated Host Selection Phase

In this phase, the set of the active APs and their associated hosts are selected. This phase comprises following eight steps:

1. Preprocessing

The link speed for each possible pair of an AP and a host is estimated with the measurement or the throughput estimation model [29]. Then, this step initializes the variables for the following steps:

- (a) For each AP, make a list of hosts that can be associated with this AP, where the throughput of the link between a host and an AP is S or larger, it can be associated. This list is called the *associable host list for APs*.
- (b) Initialize each AP as the *non-active* AP. Initially, only the DAPs are selected as *candidate APs*.

2. Initial Solution Generation

An initial solution to the cost function E_1 is derived using a greedy algorithm [78], which repeats the following procedures:

- (a) Select the AP that can be associated with the maximum number of non-associated hosts.
- (b) Activate this AP and increment E_1 by one.
- (c) Update the number of non-associated hosts in the host list for APs.

3. Host Association Improvement

The cost function E_2 is calculated for the greedy solution using Eq. (3.10). Then, this solution is improved by repeating the following procedure:

- (a) Find the AP that gives the lowest host throughput in Eq. (3.10).
- (b) Select one host randomly from the associated hosts with this AP, and associate it with another active AP that is selected randomly. Then, calculate E_2 .
- (c) Keep the new association only if this new E_2 is higher than the previous E_2 . Otherwise, return to the previous association.

4. AP Selection Optimization

The cost functions E_1 and E_2 are further jointly improved in this step under the constraints mentioned before by the *local search* [79]. This local search repeats the following three procedures:

- (a) If the current solution satisfies the *minimum host throughput constraint*, it seeks to reduce the number of active APs E_1 by deactivating an active AP. In the implementation, it repeats to 1) randomly select an active AP and deactivate it, 2) apply *Host Association Improvement*, and 3) check the feasibility of this deactivation.
- (b) Otherwise, it seeks to improve the *minimum average host throughput* E_2 with the same number of active APs by changing the active AP. In the implementation, it repeats to 1) randomly select a non-active AP and activate it, 2) apply *Host Association Improvement*, and 3) check the possibility of deactivating another active AP.
- (c) If (b). is not achieved, it seeks to satisfy the *minimum host throughput constraint* by increasing the number of active APs while improving the *minimum average host throughput*. In the implementation, it 1) randomly selects a non-active AP and 2) applies *Host Association Improvement*.

5. Link Speed Normalization

The fairness criterion will be applied when the total expected bandwidth exceeds B^a . Generally, the link speed is normalized as:

- (a) Calculate the expected total bandwidth B^e by the summation of the throughputs of all the APs.
- (b) If $B^e > B^a$, adjust each AP-host link speed as:

$$\hat{s}_{ij} = s_{ij} \times \frac{B^a}{B^e} \quad (3.12)$$

where \hat{s}_{ij} is the normalized link speed.

6. Termination Check

The algorithm is terminated when either of the following conditions is satisfied:

- (a) The *minimum host throughput constraint* is satisfied.
- (b) All the APs in the network have been activated.

7. Additional VAP Activation

If all the DAPs become active but the *minimum host throughput constraint* is still not satisfied, VAPs are newly selected as candidate APs. A VAP is slower than a DAP, but faster than a MAP. The locations of hosts are considered as the locations of the candidate VAPs, because user hosts may be used for VAPs. Then, it returns to *AP Selection Optimization* step.

8. Additional MAP Activation

If all the DAPs and VAPs become active but the *minimum host throughput constraint* is still not satisfied, MAPs are newly selected as candidate APs. A MAP is the slowest among the three AP types. The locations of hosts are considered as the locations of the candidate MAPs, because users may have MAPs. Then, it returns to *AP Selection Optimization* step.

3.4.2.2 Channel Assignment Phase

In this phase, the assigned channels to the active APs are selected and it has the following four steps:

1. Preprocessing

The interference and delay conditions of the network are represented by a graph.

- (a) Construct the *interference graph*, $G = (V, E)$, from the APs and the hosts, where the vertex V represents the set of APs and the edge E presents the existence of the interference between two APs. $e(i, j) \in E$ if AP_i is interfered with AP_j in the network.
- (b) Calculate the *communication time* for each AP. The communication time T_i for AP_i is defined as the total time when the AP transmits 1-bit to all the associated hosts. It is given by:

$$T_i = \sum_j \frac{1}{sp_{ij}} \quad (3.13)$$

where sp_{ij} represents the link speed between AP_i and $host_j$.

- (c) Calculate the *neighbor interfered communication time* for each AP. The neighbor interfered communication time NT_i for AP_i is given by:

$$NT_i = \sum_{e(i,k)=1} T_k \quad (3.14)$$

2. Interfered AP Set Generation

The set of APs that are interfering with each other is found for each AP.

- (a) Sort the APs in descending order of NT_i , where the tie-break is resolved by T_i .
- (b) Find the interfered AP set for each AP by repeating the following steps:
 - i. Initialize the interfered AP set by $I_i = \{i\}$ for AP_i .
 - ii. Expand I_i by examining the APs in sorted order in a) whether the AP is interfered with each AP in I_i . If so, include this AP, AP_j , into I_i , i.e. $I_i = I_i \cup \{j\}$.
- (c) Calculate the total interfered communication time AT_i for AP_i , which is given by:

$$AT_i = \sum_{k \in I_i} T_k \quad (3.15)$$

3. Initial Solution Construction

Then, an initial solution is derived with a greedy algorithm.

- (a) Sort the APs in descending order of the total interfered communication time AT_i , where the tie-break is resolved by NT_i .
- (b) Assign a channel c to AP_i such that the interfered communication time IT_i is minimized if it is assigned. IT_i is given by:

$$IT_i = \sum_{\substack{k \in I_i \\ c_k = c_i}} T_k \quad (3.16)$$

where c_k represents the assigned channel to AP_k .

- (c) Repeat 2) until each active AP is assigned to one channel.
- (d) Calculate the cost function E_3 using Eq. (3.11) and save this initial solution as the best solution E_3^{best} .

4. Solution Improvement by Simulated Annealing

Finally, the initial solution is improved by repeating the following *simulated annealing* (SA) procedure with the constant SA temperature T^{SA} at the SA repeating times R^{SA} , where T^{SA} and R^{SA} are given algorithm parameters:

- (a) Randomly select one AP and one new channel for the channel change trial.
- (b) Calculate the interfered communication time IT_i after assigning this new channel by:
- (c) Calculate E_3^{new} using Eq. (3.11) for the new channel assignment, and $\Delta E_3 = E_3^{new} - E_3$.
- (d) If $\Delta E_3 \leq 0$, accept the new channel assignment, and address this new solution as the best one.
- (e) Otherwise, generate a 0-1 random number, $rand$, and if $rand \leq \frac{-\Delta E_3}{T^{SA}}$, then accept the new channel assignment.

3.4.2.3 Channel Load Averaging Phase

Once the channel assignment is completed using a limited number of channels, there is a possibility of imbalanced total loads across different channels assigned to the access points (APs). To address this issue, in this phase, the channel load is equalized by adjusting the associations between APs and hosts. It has four steps as follows:

1. Preprocessing

The AP flag for each AP is initialized with *OFF* to avoid processing the same AP.

2. AP Selection

One AP is selected to move its associated host to a different AP that is assigned a different channel.

- (a) Terminate the procedure if each AP has *ON* AP flag.
- (b) Initialize the host flag by *OFF* for each host.
- (c) Select one AP, say AP_i , that satisfies the two conditions:
 - i. The AP flag is *OFF*.
 - ii. The interfered communication time IT_i is largest among the *OFF* APs.
- (d) Set the AP flag *ON*.

3. Host Selection

Then, one host associated with AP_i is selected for the associated AP movement.

- (a) Select one host, say H_j , that satisfies the four conditions:
 - i. The host flag is *OFF*.
 - ii. The host is associated with AP_i .
 - iii. The host can be associated with another AP assigned a different channel from AP_i , or is located out of the interference range of AP_i .

- iv. The link speed with AP_i is the smallest among the hosts satisfying (a)–(c).
- (b) If one host is selected, set the host flag ON .
- (c) Otherwise, return to *AP Selection* for the new AP selection.

4. Association Change Application

Finally, the new associated AP is selected for H_j .

- (a) Select the AP that has the largest link speed among the APs that are assigned to a different channel from AP_i and can be associated with H_j .
- (b) Calculate the new cost function E_3^{new} with Eq. (3.11) if H_j is associated with this AP.
- (c) If E_3^{new} is equal to or smaller than the previous E_3 , accept the new association, and return to *Host Selection*.
- (d) Otherwise, select another AP that has the next largest link speed, and return to 3).
- (e) If no such AP exists, return to *Host Selection* for the new host selection.

3.5 Summary

This chapter reviewed our previous studies related to this thesis. Firstly, we discussed the elastic WLAN system. Secondly, the throughput estimation model in WLANs was discussed. Thirdly, the parameter optimization tool was reviewed. Finally, this chapter was concluded with the active AP configuration algorithm. In the next chapter, the results of the throughput experiments conducted for interface selections in the dual interface active AP configuration will be presented.

Chapter 4

Experimental Observation for Interface Selection

This chapter presents the experimental and estimation results for the two cases of interface selections in dual interface at the same AP. Firstly, we describe the experimental setup for interface selection. Then, we show the measured and estimated single-link throughput results.

4.1 Introduction

Raspberry Pi 4B is a small, low-power, card-size, and single-board computer. It can be used in a variety of practical systems requiring some computation or networking abilities. It has a built-in wireless network interface (NIC) that supports IEEE 802.11n and 11ac. By using *hostapd* software, it can be used as a *software AP* in WLAN [80].

Archer T4U is a wireless adapter, and is designed to work with the latest wireless standards, including IEEE 802.11ac, to allow faster data transfers and wider ranges. It has a high-performance antenna that is designed to improve the signal strength and reduce interferences. Therefore, we can realize the AP that has two network interfaces with a low cost.

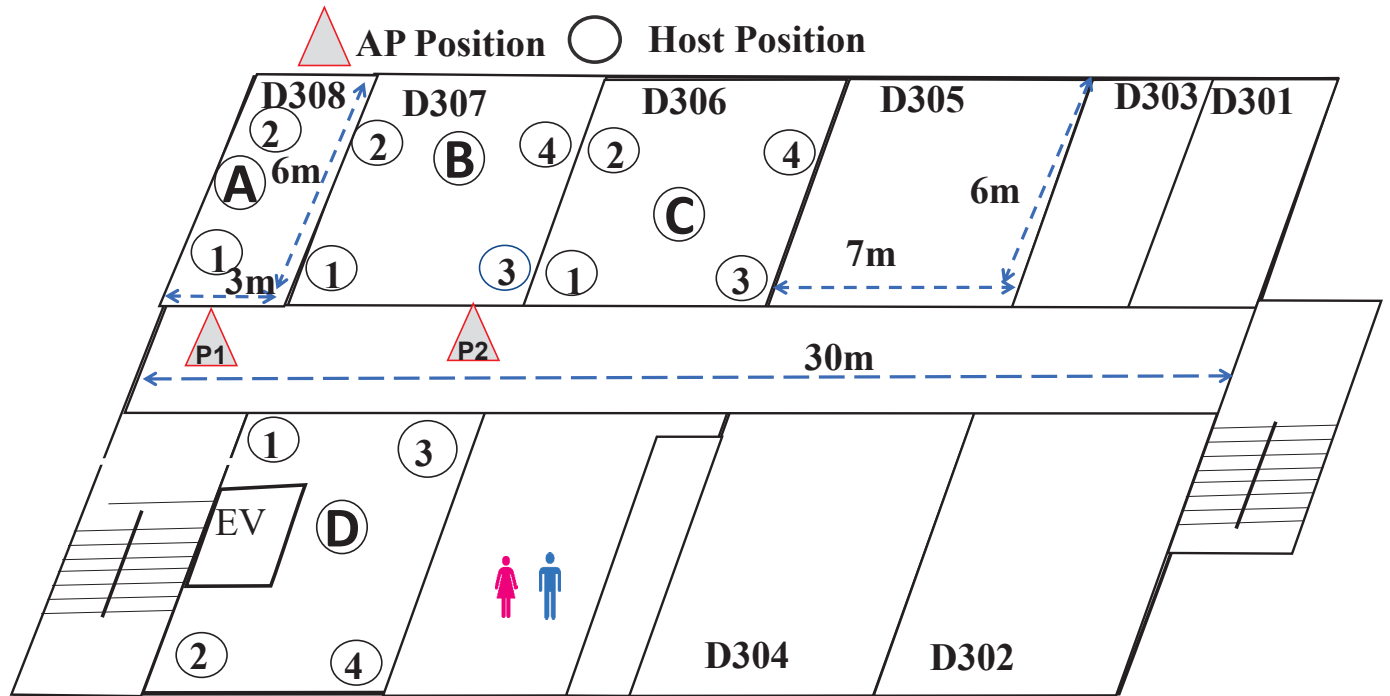
In WLAN, the throughput performance can be different by the protocol selections of the embedded NIC and the external one. We consider the two groups for *Raspberry Pi* and *Archer T4u* adapter for measurements as follows:

Group A: *Raspberry Pi* for 11n at 2.4GHz and external for 11ac at 5GHz.

Group B: *Raspberry Pi* for 11ac at 5GHz and external for 11n at 2.4GHz.

4.2 Experimental Setup

Firstly, we conducted experiments by measuring the *single link throughput* for two network topologies. Figure 4.1 illustrates the experiment field in the 3rd floor of Engineering Building #2 at Okayama University. The host locations are represented by circles, while the AP locations are represented by triangles. Figure 4.2 demonstrates the network topology for the *dual interface* with two hosts. *Linux* laptop PCs are used for the client hosts as well as server. Both interfaces utilized the 20MHz and 40MHz bonded channels. The hardware and software presented in Table 4.1 are taken into account in the measurements. The dual interfaces are configured through the commands in [31].



3rd Floor Engineering Building #2, Okayama University

Figure 4.1: Experiment field #1.

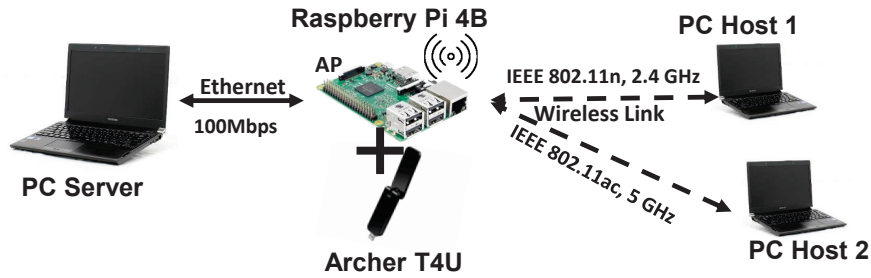


Figure 4.2: Testbed topology for dual interface with two hosts.

4.3 Throughput Results

Figures 4.3 to 4.6 show the measured and estimated single link throughput results for 20MHz and 40MHz between *Group A* and *Group B*, respectively. In each Figure, *11n Mea.* and *11ac Mea.* represent the measured throughput for 11n and 11ac interface, *11n Est.* and *11ac Est.* represent the estimated throughput for 11n and 11ac interface. For both interfaces, the throughput is estimated using Eq. 3.3 by the *parameter optimization tool* in Table 4.2.

Table 4.1: Devices and software specifications.

Access Point	
model	Raspberry Pi 4B
CPU	Broadcom BCM2711 @1.5GHz
RAM	8GB LPDDR4-3200 SDRAM
Operating System	Linux Raspbian
Software	hostapd V 2.10
External NIC	Archer T4U V3.0 AC1300
server PC	
Model	Fujitsu Lifebook S761/C
CPU	Intel Core i5-2520M @2.5GHz
RAM	4GB DDR3 1333MHz
Operating System	Linux Ubuntu 14.04 LTS (kernel 3.13.0-57)
Software	iperf 2.0.5
host PC	
Model	1. Toshiba Dynabook R731/B 2. Toshiba Dynabook R734/K 3. Fujitsu Lifebook S761/C
CPU	1. Intel Core i5-2520M @2.5GHz 2. Intel Core i5-4300M @2.6GHz 3. Intel Core i5-2520M @2.5GHz
RAM	4GB DDR3 1333MHz
Operating System	Linux Ubuntu 14.04 LTS (kernel 3.13.0-57)
Software	iperf 2.0.5

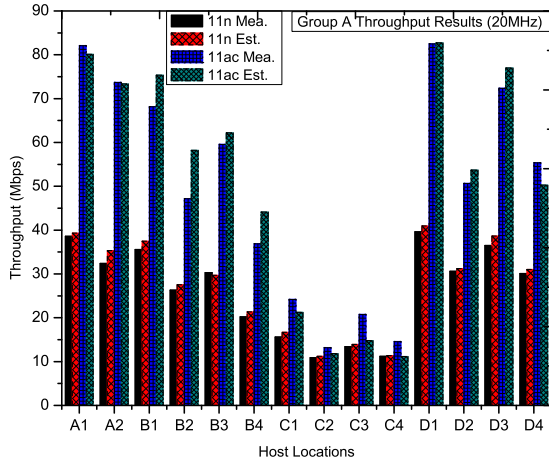
Table 4.2: Parameter optimization results for Engineering Building #2 (Field #1).

Parameter	20MHz				40MHz			
	Group A		Group B		Group A		Group B	
	11n	11ac	11n	11ac	11n	11ac	11n	11ac
P1	-28.1	-27.8	-31.0	-35.0	-28.9	-31.0	-31.1	-36.1
α	2.2	2.4	1.65	2.05	2.20	2.15	1.6	2.18
W1	7.5	7.1	3.5	5.8	7.21	2.1	6.5	4.2
W2	6	8	2.6	9.2	6.9	8.5	3.5	4.1
W3	4	4	3.1	2.8	3.4	3.7	3.5	4.4
W4	2.5	2	1.1	3.5	4.7	1.8	3.5	4.55
W5	2.4	2.2	1	2.5	2.11	7.0	2.5	2.1
W6	2	2.4	1	2.5	2.5	1.5	1.5	1.5
a	42	84	51	46	63.5	133	66	77
b	57	56.5	69	52.3	62	58	70	54.1
c	6.5	6.5	5.28	5.90	6.78	6.30	5.2	5.2

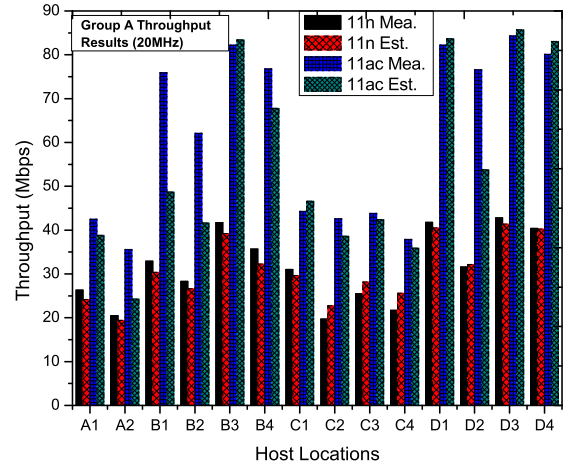
Table 4.3: Parameter optimization results for Graduate School Building (Field #2).

Parameter	Group A			
	20MHz		40MHz	
	11n	11ac	11n	11ac
P1	-27.1	-27	-28.5	-30.5
α	2.2	2.25	1.7	2.0
W1	4.5	2.4	6.5	2.3
W2	3	3.6	4.2	6.4
W3	2	2	3.1	1.8
W4	1.9	1	1.5	4.2
W5	1.8	2	2	4.3
W6	1.2	1.3	2	5.3
a	43.75	85	65	134.5
b	56.8	57	62	58.5
c	7	6.8	6.78	6.25

To obtain more accurate results, we choose 14 different host locations in the experiment field with two different AP positions. From the Figures 4.3 to 4.6, we showed that *Group A* is better than *Group B* for the dual interface, and the measured throughput and the estimated throughput matched well for each host position. In Group A, *Raspberry Pi* is used for 11n, and *Archer T4U* is for 11ac, and in Group B, they are opposite. The performance of Group A is better than that of Group B, because *Archer T4U* uses the more advanced wireless chipset and the higher-quality antenna than *Raspberry Pi*, which can more enhance the throughput performance when *Archer T4U* is used for the faster communication protocol of 11ac at 5GHz. Thus, we should use the built-in wireless network interface card (NIC) of the *Raspberry Pi* device for 11n and the *Archer T4U* wireless NIC adapter for 11ac.

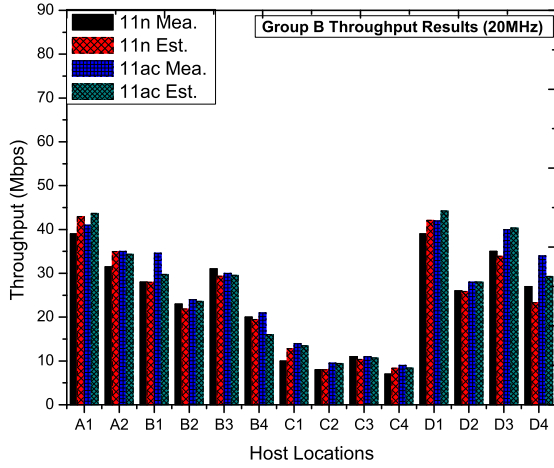


(a) Results for AP1.

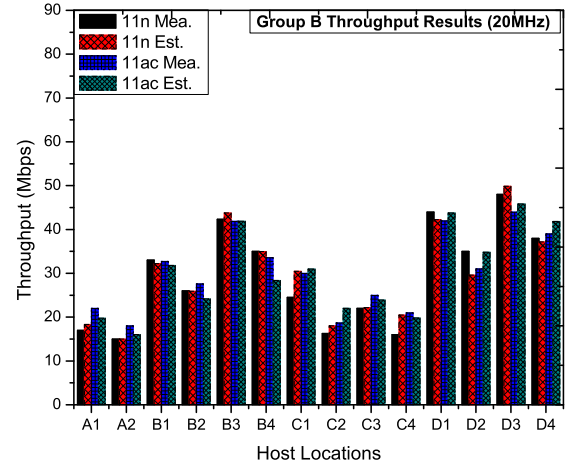


(b) Results for AP2.

Figure 4.3: Group A throughput results for 20MHz.

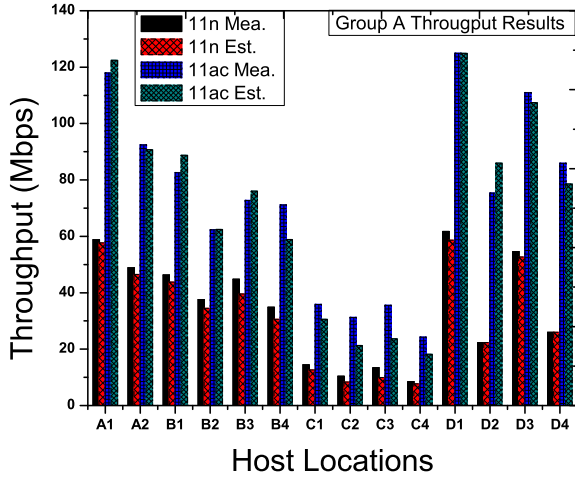


(a) Results for AP1.

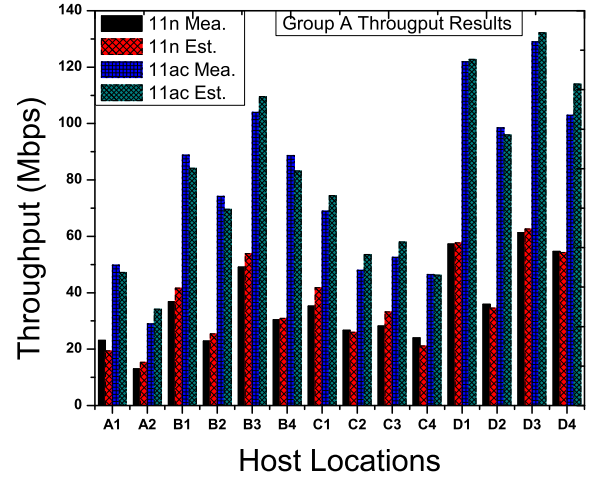


(b) Results for AP2.

Figure 4.4: Group B throughput results for 20MHz.

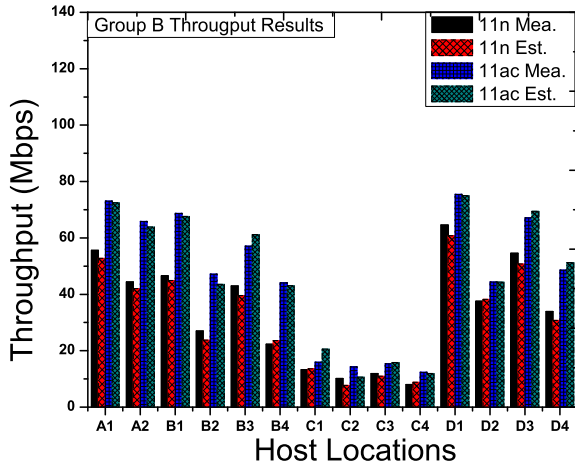


(a) Results for AP1.

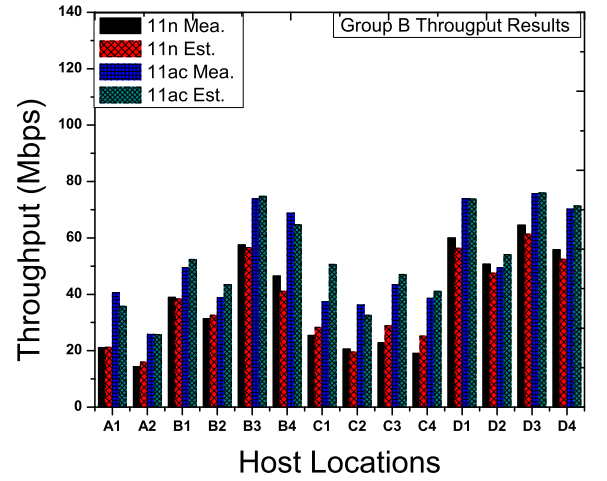


(b) Results for AP2.

Figure 4.5: Group A throughput results for 40MHz.



(a) Results for AP1.



(b) Results for AP2.

Figure 4.6: Group B throughput results for 40MHz.

4.4 Summary

This chapter presented the experimental and estimation results regarding interface selection in a dual interface at the same AP. Firstly, we discussed the experimental setup which is used for interface selection. Then, we showed the measured and estimated results for single-link throughput. In the next chapter, we will present the *active AP configuration algorithm* for the *dual interface* at the same AP.

Chapter 5

Proposal of Active AP Configuration Algorithm for Dual Interface

This chapter presents the extension of the active AP configuration algorithm for the dual interface of one AP. Firstly, we describe the formulation of the extended AP configuration algorithm. Then, we present the procedure.

5.1 Introduction

Wireless local area networks (WLANs) have gained popularity for the Internet access around the worldwide. WLANs are widely deployed in various locations due to high-speed data transfer, flexible coverage ranges, low costs, and easy installation. In WLANs, hosts are connected wirelessly to an access point (AP), providing greater scalability and flexibility compared to wired LANs. As a result, WLAN services are available in offices, schools, and public places, such as hotels, airports, malls, stations, and even trains and airplanes [1, 2, 8].

In a network field, multiple APs are often deployed to support a large number of users. Such situations are called *dense WLAN environments*. In dense WLAN environments, network performances can be degraded by radio signal interferences between nearby APs and hosts. To avoid this performance degradation, the network configuration of WLAN, such as activations/deactivations of APs and associations of hosts with active APs, should be properly optimized, according to traffic demands. If redundant APs exist, they should be deactivated to reduce energy consumptions [16, 17, 19, 28]. Therefore, we have proposed the *active AP configuration algorithm* to dynamically optimize the network configuration by activating or deactivating APs according to traffic demands [6].

The current *active AP configuration algorithm* assumes that any AP has only one *IEEE 802.11n* network interface. Besides, it does not consider the concurrent communications of multiple hosts at an AP at estimating the throughput using the throughput estimation model. As a result, the accurately estimating throughput becomes challenging in scenarios where multiple hosts are communicating simultaneously.

Currently, the simultaneous use of *IEEE 802.11n* and *11ac* interfaces at one AP has become possible with the advancement of devices as discussed in Chapter 4. Therefore, a *dual interface* at an AP using *11n* and *11ac* for *2.4GHz* and *5GHz* frequency bands can be a promising approach to enhance the network performance while reducing interferences. The *dual interface* can reduce the necessary number of active APs to meet the traffic demands, which can further reduce the

interference and enhance the overall throughput.

In this chapter, we propose the extension of the *AP configuration algorithm* to consider *dual interfaces* of 11n and 11ac at the AP in WLAN. In addition to selecting the active APs, their assigned channels, and associated hosts in the previous algorithm, the extended one assigns the network interface to each host. The throughput estimation model for 11ac is used together with the model for 11n to consider the throughput differences of the two protocols.

5.2 Active AP Configuration Algorithm for Dual Interface

The *active AP configuration algorithm* selects the active APs and their host associations that can minimize the number of active APs with dual interfaces under the *minimum host throughput constraint*.

5.2.1 Problem Formulation

The problem formulation for the dual interfaces at the AP is as follows:

1. **Inputs:**

- Number of APs: N where each AP has the dual interfaces, one for 11n and another for 11ac.
- Link speed between AP_i using each interface and $host_j$ for $i = 1$ to N , $j = 1$ to H : tp_{ij} , where the link speed can be estimated using Eq. (3.3) with corresponding parameter values in Table 4.2 and 4.3.
- Number of orthogonal channels (OCs) for each interface: C

2. **Outputs:**

- Set of active APs with dual interfaces.
- Set of hosts associated with each interface of every active AP.
- Channel assigned to each interface of every active AP.

3. **Objectives:**

The average host throughput calculation of AP_j in Eq. (3.9) is modified as follows:

$$TH_j = \frac{1}{\sum_k \frac{1}{tp \times srf(m)}} \quad (5.1)$$

4. **Constraints:**

- Minimum host throughput constraint: every host connected to an AP interface must enjoy the given threshold G on average when all the hosts are communicating simultaneously.
- Channel assignment constraint: each interface of an AP must be assigned a channel.

5.2.2 Algorithm Procedure

The procedure of the extended active AP configuration algorithm for dual interfaces is as follows:

1. *Preprocessing*

The link speed for each possible pair of an AP interface and a host is estimated by the *throughput estimation model* using Eq. (3.3). Besides, the 11n interface of an AP is selected as the initial candidate one for any host.

2. *Initial Solution Generation*

An initial solution with the cost function E_1 is derived by a greedy algorithm [78], which repeats the following procedures:

- (a) Select the AP that can be associated with the maximum number of non-associated hosts.
- (b) Activate this AP and increment E_1 by one.
- (c) Update the number of non-associated hosts in the host list for APs.

3. *Host Association Improvement*

The host association improvement procedure is as follows:

- (a) Change the association of each host to the interface of AP which provides the highest throughput using Eq. (3.4) within the associable AP interfaces. At this stage, compute the cost function E_2 and keep it as the best found cost function E_2^{best} .
- (b) Find the interface of the AP that provides the lowest host throughput using Eq. (3.10). Then, make the *modifiable hosts* list that are associated with the AP interface, which can be connected with another AP interfaces.
- (c) Choose one host at random from the list of *modifiable hosts*. Then, associate this host at random with other associable active AP interface. Compute the new cost function E_2^{new} .
- (d) Replace E_2^{best} by the newly found E_2^{new} , if $E_2^{new} > E_2^{best}$, and retain the new AP-host association. If not, go back to the previous association and select the best cost function E_2^{best} .

4. *AP Selection Optimization*

The *AP Selection Optimization* phase optimizes the selection of the active APs with dual interfaces and the AP-host associations to further minimize both E_1 and E_2 by the local search method [79].

The procedure is described as follows:

- (a) (a) Initialize E_1^{best} and E_2^{best} by the current algorithm output after the *Host Association Improvement* phase.
 - (b) Initialize the *selection flag* for every AP interface by OFF, and
 - (c) Initialize the number of current active AP interfaces P by 0 ($P = 0$).
- Here, the *selection flag* is introduced to avoid continuously processing the same AP interface.
- (b) Repeat the following procedure $18 \times N \times H$ times:

- i. (i) Randomly select the one active 11ac AP interface with the OFF selection flag and deactivate it. If it is found, decrease P by 1.
(ii) If the active 11ac interface is not found for deactivation in (i), randomly select one active 11n AP interface with the OFF selection flag and deactivate it. If it is found, decrease P by 1.
(iii) Set the selection flag ON for this deactivated interface.
- ii. (i) Find a new associable active AP interface for each host that is currently associated with the AP interface that is deactivated in (a), and
(ii) Associate the host with this new interface. If multiple active AP interfaces can be associated with this host, select the one that is associated with the largest number of hosts among them.
- iii. If some host cannot be associated with any new AP interface, repeat the following steps until every host can be associated with an active interface.
(i) Find a non-active 11n interface with the OFF selection flag that can be associated with the host associated with the deactivated interface.
(ii) If there is no such interface, find a non-active 11n interface with the ON selection flag.
(iii) If multiple interfaces are found in both cases, select one interface that provides the maximum link speeds to the host for tie break.
(iv) If one interface is found, increase P by 1, activate this interface, and set the selection flag ON.
- iv. If the total number of the interfaces of the current active APs, which is given by $E_1 \times 2$, is smaller than P , apply the following procedure until it satisfies $E_1 \times 2 = P$:
(i) Randomly select one non-active 11ac interface with the OFF selection flag of the active AP that has the active 11n interface and activate the interface.
(ii) If there is no such 11ac interface in (i), randomly select one 11ac interface with the ON selection flag of the active AP and activate the interface.
(iii) If there is no such 11ac interface in (ii), randomly select one non-active 11n interface with the OFF selection flag of the non-active AP and activate the AP.
(iv) If there is no such 11n interface in (iii), randomly select one non-active 11n interface with the ON selection flag of the non-active AP and activate the AP.
(v) If a non-active interface is selected, activate this interface and increase P by 1.
- v. Apply the following steps:
(i) Set the selection flag to OFF for the newly activated interface of the AP in Step (4(b)iii).
(ii) If the number of the active APs ($=E_1$) is smaller than E_1^{best} , apply the *Host Association Improvement* phase, calculate (E_2), and update E_1^{best} , E_2^{best} .
(iii) Otherwise, if $E_1 = E_1^{best}$, apply the *Host Association Improvement* phase, and calculate E_2 . If E_2 is smaller than E_2^{best} , update E_1^{best} .
- vi. If at least one interface of an active AP has the OFF selection flag, go to Step (4(b)i).
- (c) Apply the following steps to satisfy the average throughput constraint with $18 \times N$ times
 - i. If $1/E_2^{best} \geq G$ (minimum throughput constraint) is satisfied, mark the solution as feasible, and go to Step (4).

- ii. Otherwise, $1/E_2^{best} \geq G$ (minimum throughput constraint) is not satisfied and if $P = 0$, apply the following steps:
 - (i) Set $P = E_1^{best} + 1$.
 - (ii) Randomly select one 11ac interface of the active AP. If not found, randomly select one 11n interface of an AP, and run the *Host Association Improvement* phase. Update E_1^{best} and E_2^{best} .
 - (iii) Go back to Step (4b).
- (d) Output the best found result with E_1^{best} and E_2^{best}

5. *Link Speed Normalization*

The fairness criterion will be adopted when the total expected bandwidth exceeds B^a . Then, the link speed is normalized.

6. *Termination Check*

For every active AP, if either of the two interfaces is not activated, activate the interface, and apply the *Host Association Improvement* phase. Then, if the minimum host throughput constraint is satisfied, terminate the algorithm. Otherwise, go to the *AP Selection Optimization* phase.

5.3 Summary

In this chapter, we presented the extension of the active AP configuration algorithm for the dual interface at one AP. Firstly, we described the formulation of the extended AP configuration algorithm. Then, we presented the procedure of the extended active AP configuration algorithm for the dual interface. In the next chapter, we will discuss the implementation procedure of the extended active AP configuration algorithm on real testbed.

Chapter 6

Testbed Implementation

This chapter presents the testbed implementation of the elastic WLAN system using Raspberry Pi and Linux PCs. Firstly, we describe the testbed implementation and its network topology. Secondly, we review the setup of the dual interface. Thirdly, we present the dual interface AP configuration using Raspberry Pi. Finally, we show the execution flow of dual interfaces in the elastic WLAN system.

6.1 Testbed Implementation

We have chosen the Linux environment as the implementation platform for our elastic WLAN system. Linux is an open-source operating system that offers a wide range of tools, making it ideal for implementing new algorithms, protocols, methods, and devices in wireless networks. It provides numerous open-source tools that are easily configurable and offer flexibility for integrations with other software and devices. In contrast, we found that network configuration and management tools in Windows operating system are less flexible, less configurable, and not open sources [60].

For our implementation, we are using *Ubuntu*, which is a popular Linux distribution for general-purpose users. Specifically, we adopted the *Ubuntu LTS 14.04* version in the *elastic WLAN system testbed* to ensure compatibility and similarity of throughput performance in measurements. In our future studies, we plan to explore implementations of the elastic WLAN system on various platforms and the latest versions of the *Ubuntu* operating system. The device environments and software in Table 4.1 are used for the testbed implementation of the system.

6.2 Testbed Network Topology

Figure 6.1 illustrates the example network topology of the elastic WLAN system testbed for experiments. The testbed included four *Linux laptop PCs* for the hosts, one Linux laptop PC for the server, and *Raspberry Pi* for the AP. The management server controls the APs and the hosts by using the administrative access rights to them. The APs are connected to the server through wired connections. The hosts and the APs are connected through wireless connections.

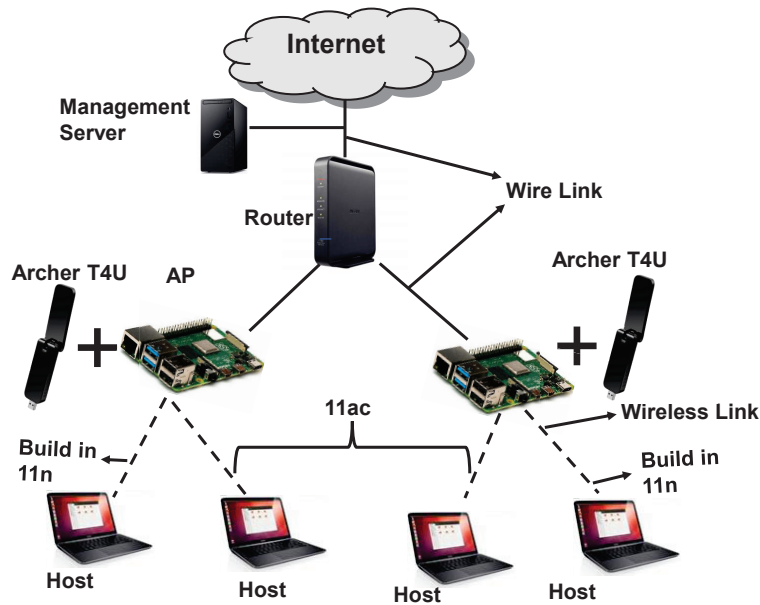


Figure 6.1: Example network topology of elastic WLAN system testbed.

6.3 Dual Interface AP Configuration Using Raspberry Pi

This section explains how to configure the dual interface on Raspberry Pi using *hostapd* daemon [81,82].

1. Install the *hostapd* and *dnsmasq* using the following command [82]:

```
$ sudo apt-get install hostapd
$ sudo apt-get install dnsmasq
```

2. Uncomment and set *DAEMON_CONF* to the absolute path of the *hostapd* configuration file to start *hostapd* during system boot:

```
DAEMON_CONF="/etc/hostapd/hostapd.conf"
DAEMON_CONF="/etc/hostapd/hostapd1.conf"
```

3. Configure a static IP for the *wlan0* and *wlan1* interface by modifying the *dhcpcd* configuration file:

```
$ sudo nano /etc/dhcpcd.conf
```

At the end of the *dhcpcd* file, add the following example content and then save.

```
interface wlan1
static ip_address=192.168.2.1/24
```



```
interface wlan0
static ip_address=192.168.3.1/24
```

4. Configure the DHCP server (*dnsmasq*) for network configuration parameters, such as the IP addresses, for interfaces and services. As for the dhcp rules, rename the default configuration file for backup and edit a new one:

```
$ sudo mv /etc/dnsmasq.conf /etc/dnsmasq.conf.orig
$ sudo nano /etc/dnsmasq.conf
```

Add the following example content in *dnsmasq* file, then save:

```
interface=wlan1
dhcp-range=192.168.2.2,192.168.2.40,255.255.255.0,24h
interface=wlan0
dhcp-range=192.168.3.2,192.168.3.40,255.255.255.0,24h
```

5. To build a bridge between the *wlan0*, *wlan1*, and *eth0* interfaces, install a package using the following command:

```
$ sudo apt-get install bridge-utils
```

Then, add a new bridge (called *br0*) using the following command:

```
$ sudo brctl addbr br0
```

Next, connect the *eth0* interface to bridge using the following command:

```
$ sudo brctl addif br0 eth0
```

6. Setup the *wlan0*, *wlan1* interfaces in the network interface configuration file */etc/network/interfaces*.

Use the following command for the interface file:

```
$ sudo nano /etc/network/interfaces
```

Add the following lines at the end of the file:

```
auto br0
iface br0 inet manual
bridge_ports eth0 wlan0 wlan1
```

7. Modify the configuration file `/etc/hostapd/hostapd.conf` for `wlan0` with the desired SSID and PASSWORD.

A simple example of the configuration file is given below:

```
interface=wlan0
bridge=br0
hw_mode=g
channel=1
ssid=SSID
wpa_passphrase=PASSWORD
```

8. Add the another one configuration file `/etc/hostapd-/hostapd5g.conf` for `wlan1` with the desired SSID and PASSWORD.

A simple example of the configuration file is given below:

```
interface=wlan1
bridge=br0
hw_mode=a
channel=36
ssid=SSID5G
wpa_passphrase=PASSWORD
```

6.4 Execution Flow of Dual Interface in Elastic WLAN

Figure 6.2 shows the execution flow of the elastic WLAN system testbed implementation.

6.4.1 Generation of input for dual interface AP Configuration Algorithm

In this step, the server obtains all the connected devices to the network and generates the input for the dual interface active AP configuration algorithm using the following procedure:

1. The server explores all the connected devices to the network using `arp-scan` [62]. The command is given below:

```
$ sudo arp-scan --interface=eth0 192.168.10.0/24
```

Here, `-interface=eth0` represents the interface and `192.168.10.0/24` is the network IP range to scan. The output includes the available hosts and APs in the network together with their IP and MAC addresses. To identify the hosts and APs in this system using their MAC addresses a simple C program is developed. The server then generates the list of authorized APs and authorized hosts.

2. The following command collects the *receiving signal strength (RSS)* of each host from each AP using `nm-tool` [63, 64]. `ssh` protocol is used to execute the command remotely in each host [67, 68, 71].

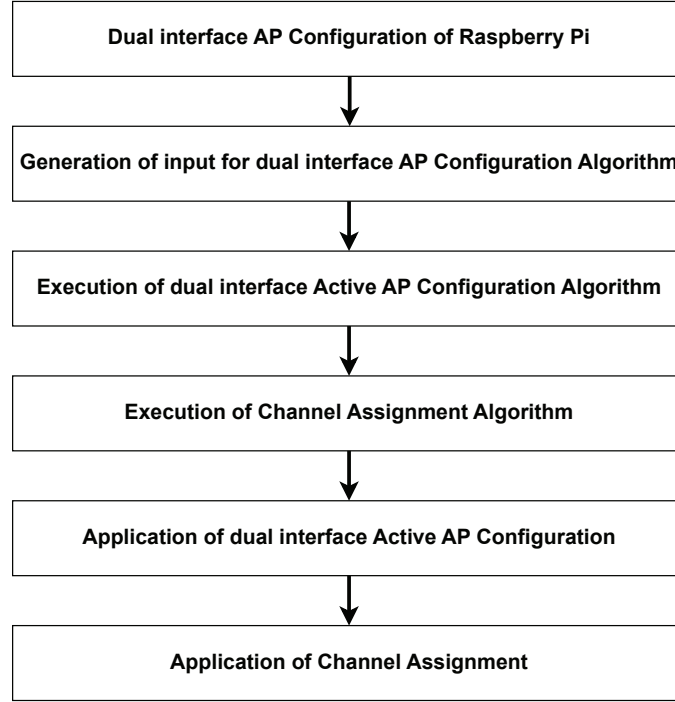


Figure 6.2: Execution flow of elastic WLAN system.

```
$ sudo nm-tool
```

3. After this, the server converts the receiving signal strength to the estimated link speed using the sigmoid function in [29], and generates the input for the dual interface active AP configuration algorithm.

6.4.2 Execution of dual interface Active AP Configuration Algorithm

The dual interface active AP configuration algorithm is executed in this step. The following commands compile the program for the algorithm and execute it respectively. The *minimum host throughput constraint* and the *bandwidth limitation constraint* are specified by the user.

```
$ g++ -o apc DualAPConfigurationAlgorithm.cpp
$ ./apc input.txt min_host_throughput bw_limit
```

Here, *input.txt* presents the input file generated in the previous step, *min_host_throughput* does the minimum host throughput constraint, and *bw_limit* does the bandwidth limitation constraint. After this, the list of active APs with dual interface and their associations with the hosts are obtained.

6.4.3 Execution of Channel Assignment Algorithm

The following commands compile the program for the channel assignment extension and execute it respectively.

```
$ g++ -o ca DualChannelAssignment.cpp
```

```
$ ./ca HostAPassociation.txt num_of_channels
```

HostAPassociation.txt presents the input file to the channel assignment that contains the list of the active APs with the dual interfaces and their associations with the hosts, and *num_of_channels* does the number of available channels for both interfaces.

6.4.4 Application of dual interface Active AP Configuration

The management server applies the output of the two algorithm.

1. The server activates or deactivates the APs in the network according to the algorithm output using the following command.

```
$ sudo /etc/init.d/hostapd start
$ sudo /etc/init.d/hostapd stop
```

2. The following command connects a host to a new AP using *nmcli* [83,84]. *NewSSID* represents the new AP for the host and *PASSWORD* does the security key of the AP. The server modifies the AP-host association according to the algorithm output using this command.

```
$ sudo -s nmcli dev wifi connect NewSSID password PASSWORD
```

6.4.5 Application of Channel Assignment

The server uses the following commands to assign the new channel to the Raspberry Pi AP using *sed* [74]. For this, the server modifies the configuration file */etc/hostapd/hostapd.conf* and */etc/hostapd/hostapd5g.conf* with the channel number.

```
$ sed -i -e 's/. *channel.*/channel=' $NewChannel '/'
/etc/hostapd/ hostapd.conf
$ sed -i -e 's/. *channel.*/channel=' $NewChannel '/'
/etc/hostapd/ hostapd5g.conf
$ sudo /etc/init.d/hostapd restart
```

Here, 's' denotes the substitution command while *NewChannel* represents the channel that will be assigned in the *hostapd.conf* file of the AP. Once the new channel is assigned, the server is restarted to ensure the changes are applied and take effect.

6.5 Summary

In this chapter, we reviewed the testbed implementation details procedure for the dual interface active AP configuration algorithm using Raspberry Pi. In the next chapter, we will present evaluation results through simulations and testbed experiments.

Chapter 7

Evaluation of Active AP Configuration Algorithm for Dual Interface

This chapter presents the evaluation of the proposed dual interface active AP configuration algorithm. Firstly, we evaluate the proposal through simulations. Then, we evaluate the proposal through testbed experiments.

7.1 Evaluation by Simulation

In this section, we evaluate the active AP configuration algorithm for the APs with the dual interfaces through simulations using the *WIMNET* simulator.

7.1.1 WIMNET simulator

The WIMNET simulator is designed to simulate the least functions for wireless communications between hosts and access points (APs). These functions are crucial for estimating data throughputs and delays of packet transmissions in WLAN. Originally, the *WIMNET simulator* [33] has been developed to evaluate a large-scale wireless Internet-access mesh network with a reasonable CPU time on a standard PC. It supports multiple APs that can create extensive WLANs and establish wireless links for communication.

In order to simulate various networking scenarios, the WIMNET simulator requires the selection of several network field parameters. These parameters include the locations of access points (APs) and hosts, their associations, the presence of different types of wall obstacles, and the communication channels. The simulator synchronizes a sequence of functions, such as host movements, communication request arrivals, and wireless link activations, using a global clock known as a time slot. Each time slot represents a fixed unit of time. Within a certain number of time slots, a host or AP can complete one-frame transmission and receive acknowledgments. By adjusting the length of the time slot and the number of time slots for link activation, different transmission rates can be achieved.

7.1.2 Network Environments

In this thesis, we evaluate our approach by taking into account the *indoor network environment*. This environment comprises multiple rooms that serve as offices, laboratories, or classrooms within

a building. It's worth mentioning that WLANs are commonly deployed in such buildings. In such an environment, WLAN users typically utilize personal computers (PCs) positioned at fixed locations where chairs and tables are available. They access the Internet while seated, with their PCs placed on tables. This is due to the larger and heavier nature of PCs compared to smartphones, often requiring the use of both hands. Consequently, WLAN users exhibit lower mobility compared to users of cellular systems. In such cases, the possible host locations can be determined by the fixed positions of desks and chairs.

7.1.3 Simulation Platform

Table 7.1 shows the hardware and software specifications for the simulations.

Table 7.1: Simulation environment.

simulator	WIMNET simulator
CPU	Intel Core i7
memory	8GB
OS	Ubuntu LTS 14.04

To evaluate the performance, we consider the *nearest AP association approach (NAP)* as a simple method for AP-host associations. NAP ensures the fastest link speed between an AP and a host when no other hosts are associated with the AP. Our algorithm is used to select the active APs for this purpose. Table 7.2 provides the summary of the simulation parameters which are used in the WIMNET simulator for the simulations.

Table 7.2: Simulation parameters in WIMNET simulator.

Parameter	Values
packet size	1500 bytes
max. transmission rate	150 Mbit/s
propagation model	log distance path loss model
rate adaptation algorithm	link speed estimation model [29]
carrier sense threshold	-85 dBm
transmission power	19 dBm
collision threshold	10
RTS/CTS	yes

7.1.4 Evaluation of Large Topology for Non-CB Case

First, we evaluate the proposal in a large-size network topology in Figure 7.1 that models the 3rd floor of Engineering Building #2 at Okayama University. 14 APs with dual interfaces and 40 hosts are distributed in the four rooms.

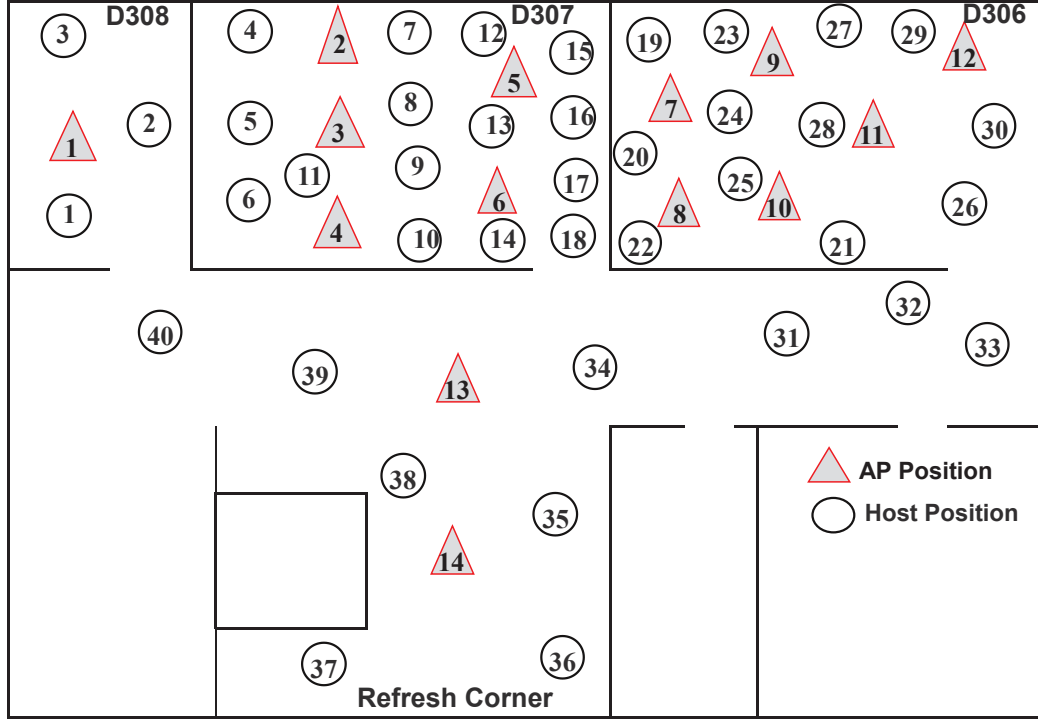


Figure 7.1: Solution for large topology with 40 hosts and 14 APs.

Table 7.3 shows the simulation results using the AP configuration found by the proposal for the dual network interfaces and that by the previous algorithm for the single network interface in [6]. It compares the number of active APs and the overall throughput by the WIMNET simulator when the minimum host throughput constraint is changed from $5Mbps$ to $15Mbps$. The results in this table confirm that the proposal outperforms the previous algorithm. It reduces the number of active APs to satisfy the constraint and increases the overall throughput in the network.

Table 7.3: Performance comparison between proposal and previous for large topology

Min. host throughput constraint G (Mbps)	Total Number of hosts (H)	Proposed algorithm			Algorithm [6]		
		Total number of Active APs out of 14	Ave. min. host throughput (Mbps)	Ave. overall throughput (Mbps)	Total number of Active APs out of 14	Ave. min. host throughput (Mbps)	Ave. overall throughput (Mbps)
5	40	2	5.23	206.11	5	5.17	185.25
10		4	11.71	468.26	10	10.43	410.25
15		6	16.58	667.77	14	14.34	575.3

7.2 Testbed Evaluation for Non-CB Case

This section evaluates the proposal for non-CB case through testbed experiments in Okayama University.

7.2.1 Experimental Setup

In the experiments, *Raspberry Pi 4B* is used for each access point (AP) by running *hostapd* [81]. The embedded wireless NIC is used for the 11n interface, and the *Archer T4U* wireless NIC adapter is used for the 11ac interface. The *Linux* laptop PCs are used for the client hosts and server. For both interfaces, the 20MHz bonded channel is used. The hardware and software presented in Table 4.1 are taken into account in the measurements.

iperf 2.0.5 [75] software is adopted for the throughput measurements as a popular tool for measuring the TCP throughput by sending TCP packets from the server to the host. The TCP traffic at a link automatically saturates by this software. The TCP traffic was generated for each measurement point in this paper using a 477 KB TCP window size and an 8 KB buffer size for four minutes at intervals of 30 s, which was conducted twice on different days. Then, the average value is calculated for each point and is shown in this paper. The packet size of the generated TCP traffic is 1500 bytes. For each point, the average of the measured throughput is computed over four minutes at 30 s intervals on two separate days in order to eliminate the fluctuations in measurements.

7.2.2 Network Fields

The 3rd floor of *Engineering Building #2* and the 2nd floor of *Graduate School Building* at Okayama University were considered as the network fields for simulations and experiments. Figures 7.2 and 7.3 illustrate the layouts of the experiment sites. Table 7.4 shows the devices locations for each topology. The two different network environments were chosen to show the results in different environments.

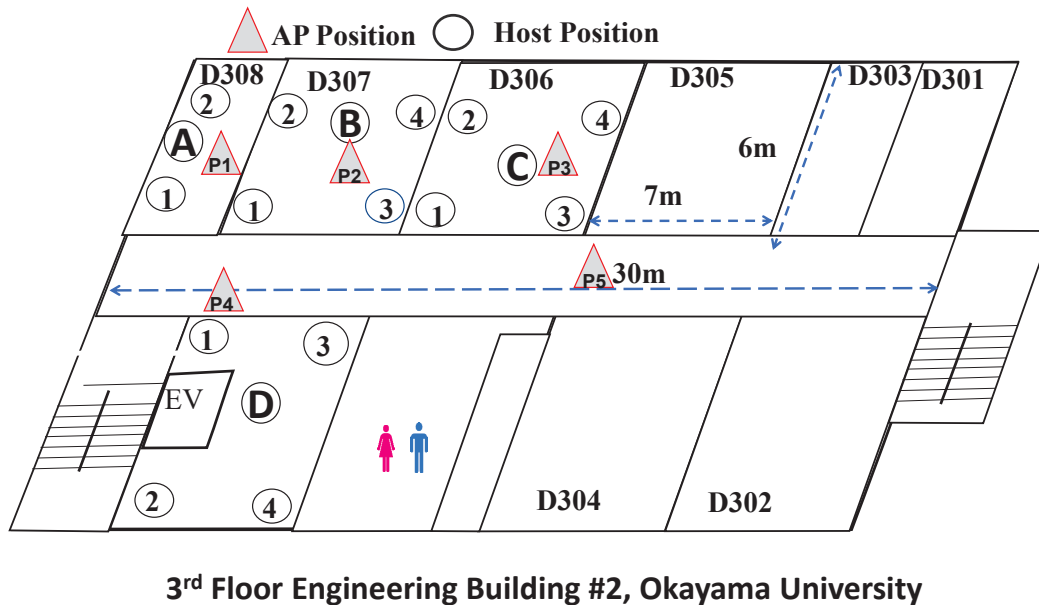


Figure 7.2: Field #1 in Engineering Building #2.

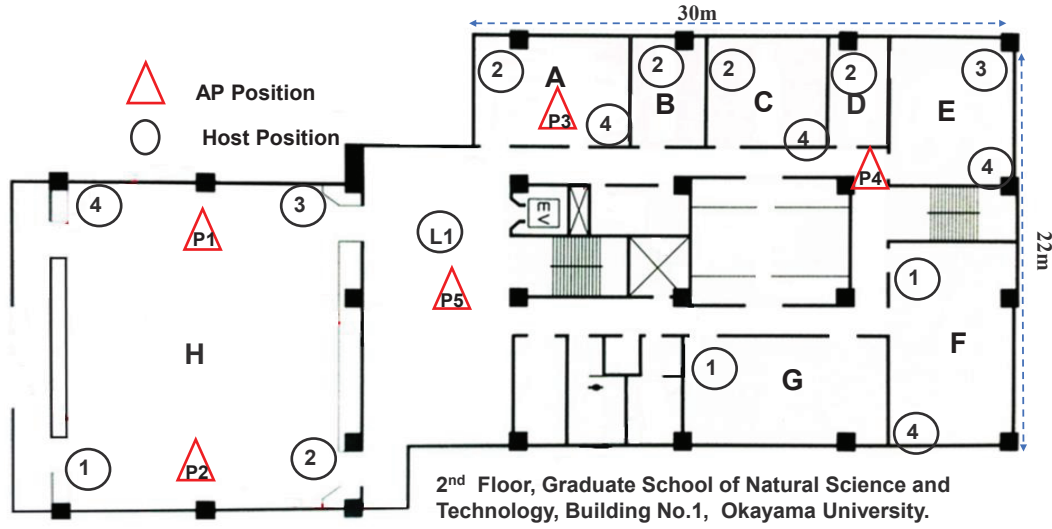


Figure 7.3: Field #2 in Graduate School Building.

Table 7.4: AP and host location for small topology

Exp. Field	Topology	AP position					Host position								
		AP1	AP2	AP3	AP4	AP5	H1	H2	H3	H4	H5	H6	H7	H8	H9
Field #1	1	P1	P2	P3	P4	P5	A1	A2	B2	B4	B3	C3	D4	D1	
	2						front D308	A2	B1	B4	front D305	C3	D4	D2	
	3	P1	P2	P3	P4		A1	A2	B1	B4	C4	C1	front D301	D4	D1
	4						front D308	A2	front D307	B4	C4	front D305	front D301	D4	D2
Field #2	5	P1	P2	P3	P4	P5	H1	H2	H3	H4	L1	A3	B2	D2	E3
	6						front #G	H2	H3	H4	L1	front #A	C2	D2	F1

7.2.3 Results and Discussions

Figure 7.4 shows the overall throughput in the network field of Engineering Building #2. The minimum host throughput constraint is applied with $G = 5Mbps$ and $G = 15Mbps$. For each case, the proposed algorithm provides the higher overall throughput than the previous one [6]. Besides, when the measured throughput and the simulation throughput are compared, they are matched well in any case, which confirms the accuracy of the proposed algorithm.

Figure 7.5 shows the overall throughput in the network field in Graduate School Building. The same minimum host throughput constraints are considered. Again, for each case, the proposal provides the higher overall throughput than the previous [6]. Besides, the measured throughput and the simulation throughput are matched well. Thus, the accuracy and effectiveness of the proposed algorithm are confirmed.

7.3 Testbed Evaluation for CB Case

In this section, we evaluate the proposal for CB case through testbed experiments in Okayama University.

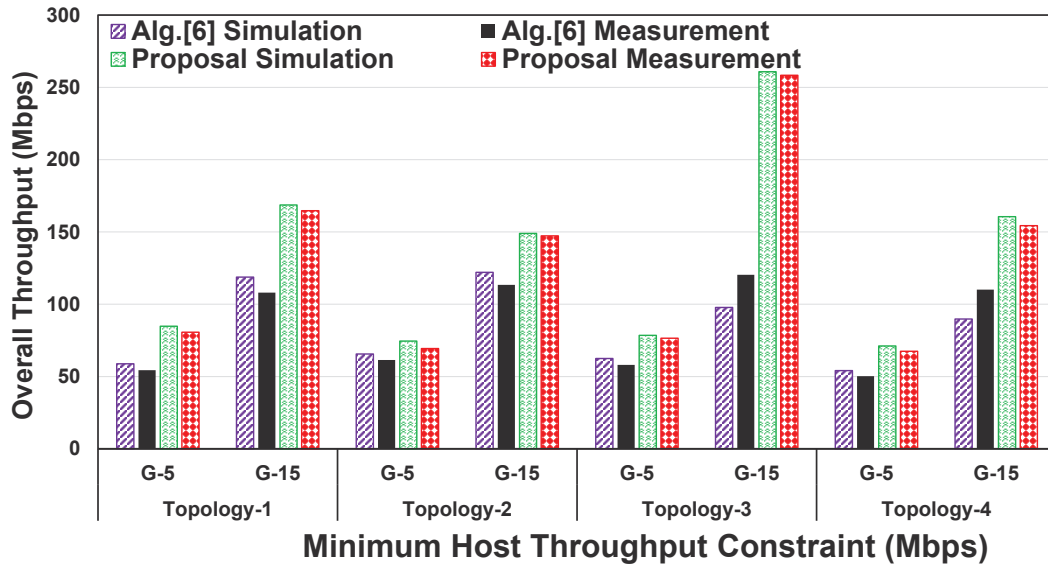


Figure 7.4: Throughput results in Field #1

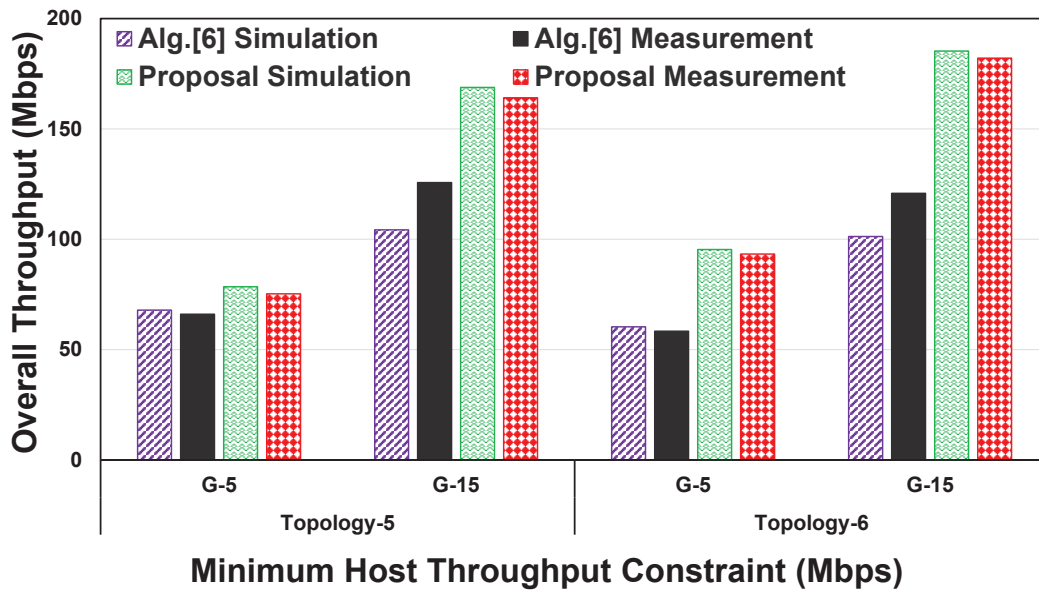


Figure 7.5: Throughput results in Field #2

7.3.1 Experimental Setup

The same experimental setup in 7.2.1 was used for the CB case where both interfaces used the 40MHz bonded channel.

To set the bonded channel for both interfaces, we can modify the /etc/hostapd/hostapd.conf file using the following Linux commands:

```
$ #40MHz for 11n
$ rsn_pairwise=CCMP
$ ht_capab=
[HT40+][SHORT-GI-20][SHORT-GI-40][DSSS_CCK-40][MAX-AMSDU-3839]
```

```
$ #40MHz for 11ac
$ rsn_pairwise=CCMP
$ vht_capab=[MAX-MPDU-3895][SHORT-GI-80][SU-BEAMFORMEE]
```

The [HT40+] needs to set the 20MHz primary channel in the /etc/hostapd/hostapd.conf file. Then, the primary channel will be bonded with the secondary channel to form the 40MHz channel.

7.3.2 Network Fields

The 3rd floor of Engineering Building #2 and the 2nd floor of Graduate School Building at Okayama University were considered as the network fields for simulations and experiments. Figures 7.6 and 7.7 illustrate the network topologies with the AP and host locations in the two fields, respectively. For each topology, we considered five APs with ten hosts. The two different network environments were chosen to show the results in different environments. In both fields, several other WLANs were observed, which may have caused interference in our experiments. In the daytime, a large number of wireless devices, such as smartphones, laptops, and IoT devices, transmit wireless signals. Thus, the daytime situation will make it difficult to measure the performance of the WLAN accurately. The signals from other devices can interfere with the signals being transmitted by the devices under testing, which can result in inaccurate throughput measurements. At night, fewer wireless devices will transmit signals. Thus, the level of interference can be reduced and will make it easier to accurately measure the performance of a WLAN.

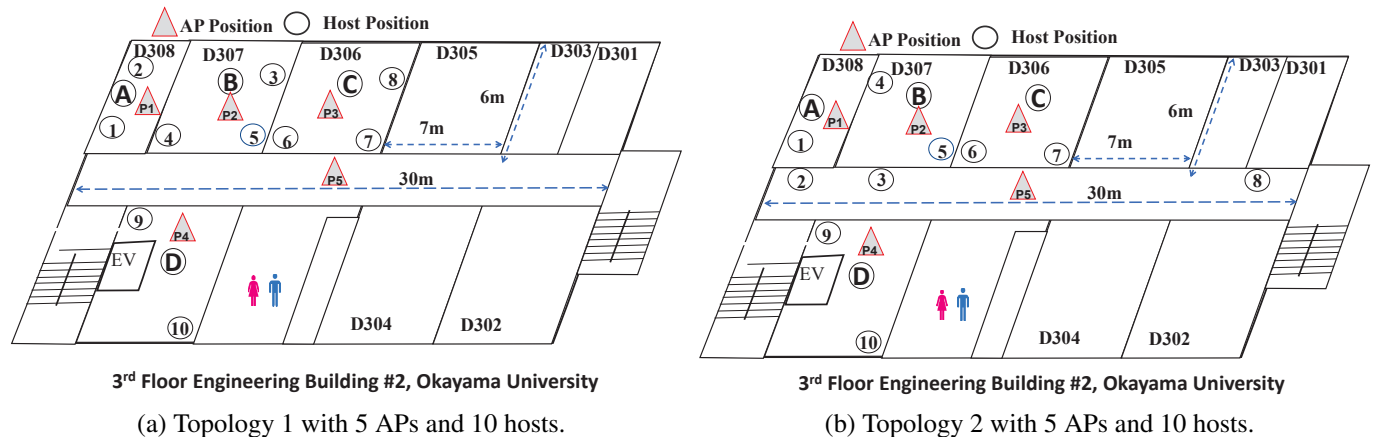


Figure 7.6: Experimental field #1.

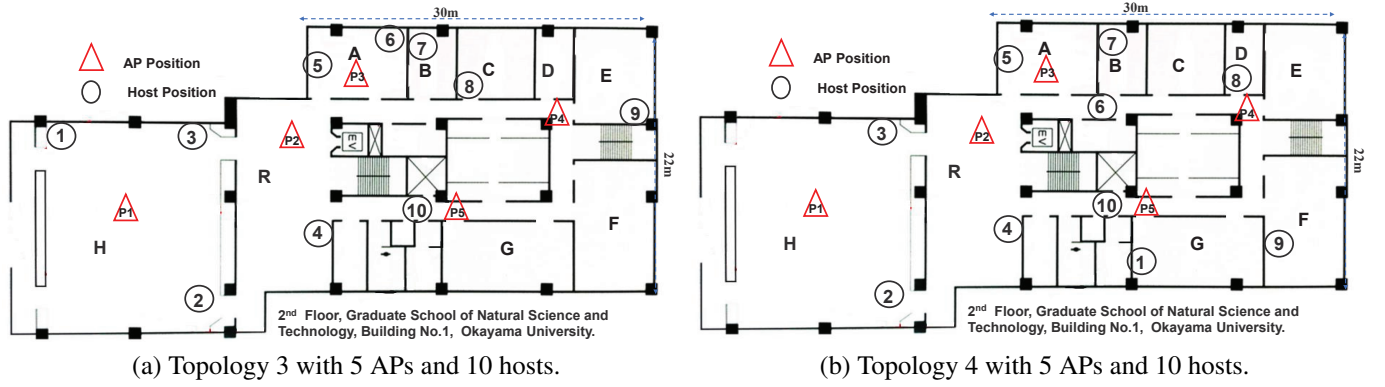


Figure 7.7: Experimental field #2.

7.3.3 Results and Discussions

Table 7.5 shows the simulation and measurement results for the two fields. The *min. host thro. cons. G* (Mbps) represents the threshold value for the *minimum host throughput constraint*. This value is used as the benchmark. All the hosts in the network should achieve it regardless of the relative distances from the associated APs. In our study, the value was selected manually, taking into account the number of hosts, the number of APs, and the overall network capacity of the WLAN. The number of active APs should be increased when this constant is not satisfied. In Table 7.5, the *ave. min. host thro.* (Mbps) represents the average minimum host throughput (Mbps), and the *ave. overall thro.* (Mbps) represents the average overall throughput in the network. The *sim.* represents the simulation results, and *mea.* represents the measurement results.

It is noted that *field #1* in Table 7.5 corresponds to *Engineering Building #2*. The *minimum host throughput constraint* is applied with $G = 10$ Mbps and $G = 25$ Mbps. The throughput results from the simulation are well matched with the measurement results in any topology. This confirms the accuracy of the proposed algorithm, and the use of the dual interface at the AP can minimize the number of active APs.

In addition, *field #2* in Table 7.5 corresponds to *Graduate School Building*. The same *minimum host throughput constraint* is considered in this field. Again, the throughput results by the simulation are well matched with the measurement results in each topology. Thus, the accuracy and effectiveness of the proposed algorithm are confirmed.

Additionally, we evaluated the proposal by comparing the performances between the single 11ac interface case and the dual interface case when the same number of APs are activated. As shown in Table 7.6, both the minimum host throughput and the average overall throughput results were increased by the proposal. The dual interfaces at the AP operated at different frequency bands. It allows one AP to communicate with two hosts simultaneously without interference. In addition, it can reduce interference between neighboring APs in a dense WLAN, because the number of active APs can be reduced. As a result, the performance of the proposal is greater than that of the single 11ac interface case. The measured throughput results are well-matched with the simulated ones in any case. It is confirmed that the proposal provides higher overall throughput than 11ac only when the same number of APs are activated.

Table 7.5: Simulation and measurement results for CB cases.

exp. field	topology	min. host thro. cons. G (Mbps)	single interface						dual interface		
			11n			11ac					
			total active APs	ave. min. host thro. (Mbps)		total active APs	ave. min. host thro. (Mbps)		total active APs	ave. min. host thro. (Mbps)	
				both	sim		mea.	both		sim.	mea.
field #1	1	10	3	13.32	12.56	2	14.78	13.66	1	12.32	11.47
		25	5	26.34	25.02	3	26.64	25.08	2	26.65	25.19
	2	10	3	13.58	12.77	2	16.54	14.84	1	13.09	12.19
		25	5	25.13	24.78	3	26.23	25.09	2	26.89	25.37
field #2	3	10	3	12.93	11.76	2	12.50	11.36	1	12.34	11.53
		25	5	25.01	24.04	4	31.18	29.63	3	37.71	35.83
	4	10	3	13.45	12.66	3	17.39	15.73	2	19.94	18.25
		25	5	23.80	22.85	4	27.71	26.02	3	36.55	34.76

Table 7.6: Comparative analysis between 11ac and dual interface.

exp. field	topology	minimum host throughput constraint G (Mbps)	dual interface				
			total active APs	average minimum host throughput (Mbps)		average overall throughput (Mbps)	
				both	sim.	mea	sim.
field #1	1	20	2	26.65	25.19	267.19	261.93
		30	3	37.88	35.92	235.24	231.06
	2	20	2	26.89	25.37	259.12	255.97
		30	3	36.38	34.02	225.77	220.63
field #2	3	20	2	24.22	22.32	266.34	259.36
		40	4	45.72	42.36	215.75	210.36
	4	20	3	36.55	34.76	238.45	233.93
		40	4	43.67	40.92	222.35	217.21

7.3.4 Comparisons with Previous Studies

To clarify the effectiveness of the proposed extended algorithm, we conducted simulations and experiments using the same network configuration as that for our previous algorithm [31].

Table 7.7 shows the simulation and measurement results for both the single interface and the dual interface under non-CB cases in both experimental fields when the same number of APs was activated as before. The results in Table 7.7 show that the average minimum host throughput and the average overall throughput results become smaller due to the smaller channel bandwidth, compared with the proposal. Figures 7.8–7.11 show the average overall throughput with respect

to the number of active APs for each topology for both CB and non-CB cases. It is noted that the simulated results are well-matched with the measured ones for all the cases. This confirms that the proposal with the CB provides higher throughput than the previous one with the non-CB, regardless of the interference.

Table 7.7: Simulation and measurement results for non-CB cases [31]

exp. field	topology	min. host thro. cons. G (Mbps)	single interface						dual interface		
			11n			11ac					
			total active APs	ave. min. host thro. (Mbps)		total active APs	ave. min. host thro. (Mbps)		total active APs	ave. min. host thro. (Mbps)	
				both	sim		mea.	both		sim.	mea.
field #1	1	5	3	8.20	6.96	2	8.78	7.14	1	8.14	7.82
		15	5	14.05	12.26	3	15.12	13.25	2	16.11	15.16
	2	5	3	8.15	6.93	2	9.21	7.96	1	8.14	7.02
		15	5	14.46	12.98	3	15.65	14.42	2	16.43	14.88
field #2	3	5	3	7.95	6.90	2	9.21	8.11	1	8.01	7.12
		15	5	15.05	13.81	4	17.79	16.08	3	19.95	17.96
	4	5	3	8.34	7.07	3	12.89	11.06	2	13.24	12.10
		15	5	15.97	13.89	4	19.04	17.35	3	21.15	20.67

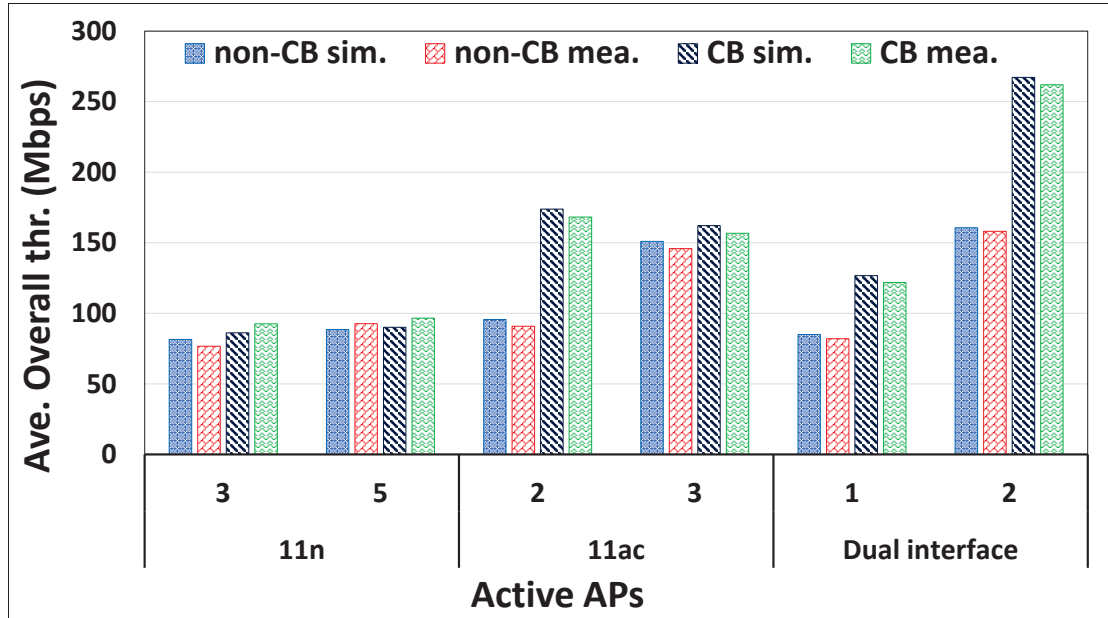


Figure 7.8: Throughput results for Topology 1.

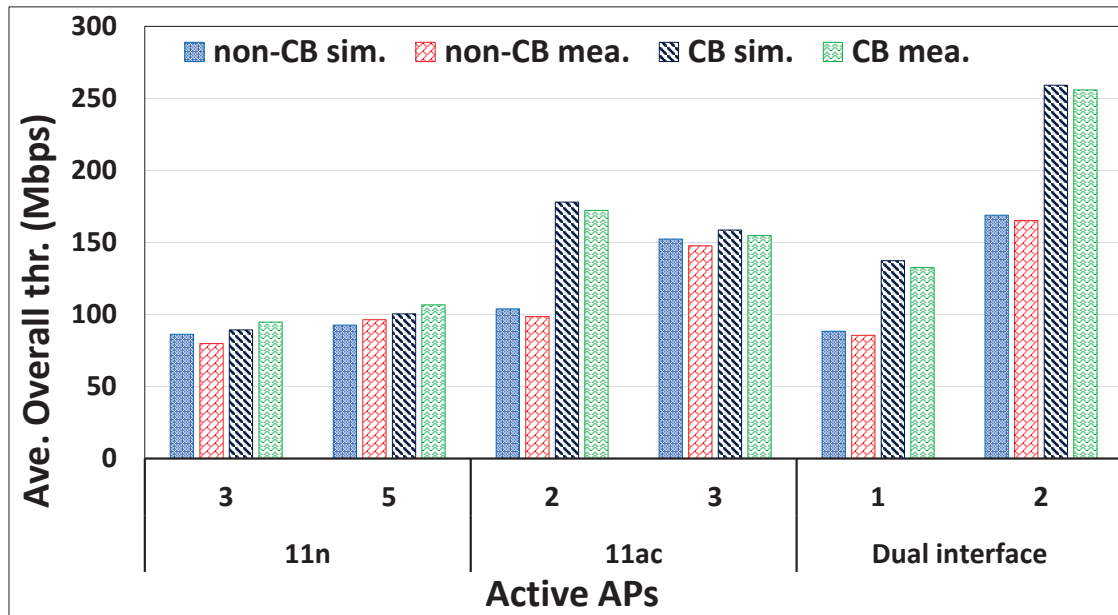


Figure 7.9: Throughput results for Topology 2.

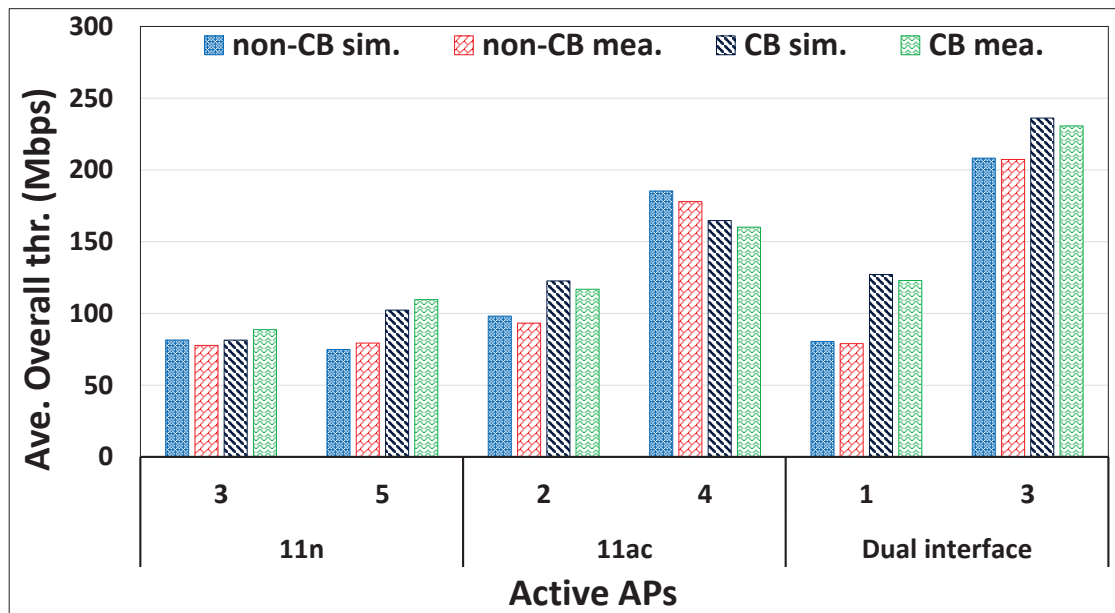


Figure 7.10: Throughput results for Topology 3.

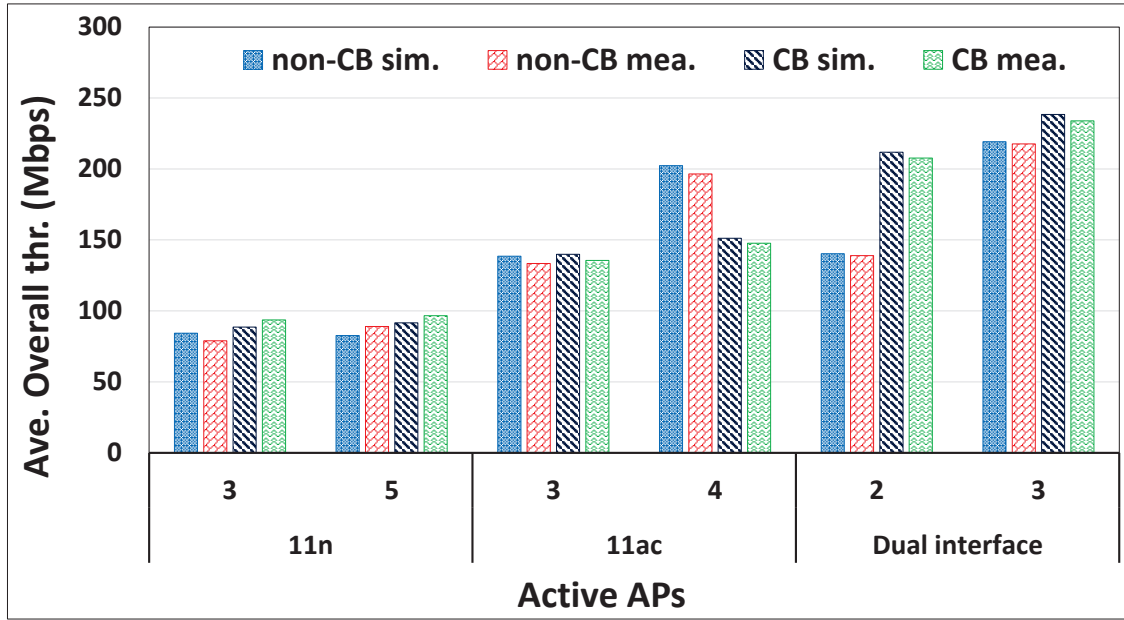


Figure 7.11: Throughput results for Topology4.

To compare the throughput performance of our approach with that of an existing approach, the *band-steering (BSTR)* method using the dual interfaces [85], was considered in experiments. In this method, each host is associated with the AP that provides the highest *received signal strength indicator (RSSI)* among the active APs.

The same number of active APs with dual interfaces and the same number of hosts are used as the ones for the proposal. Figures 7.12 and 7.13 show the average minimum host throughput and the average overall throughput results with respect to the number of active APs in *Engineering Building #2 at Okayama University* as the network field by *BSTR* and by the proposal. The results suggest that the proposal provides higher throughput than *BSTR* in any case. Thus, the effectiveness of the proposal has been confirmed.

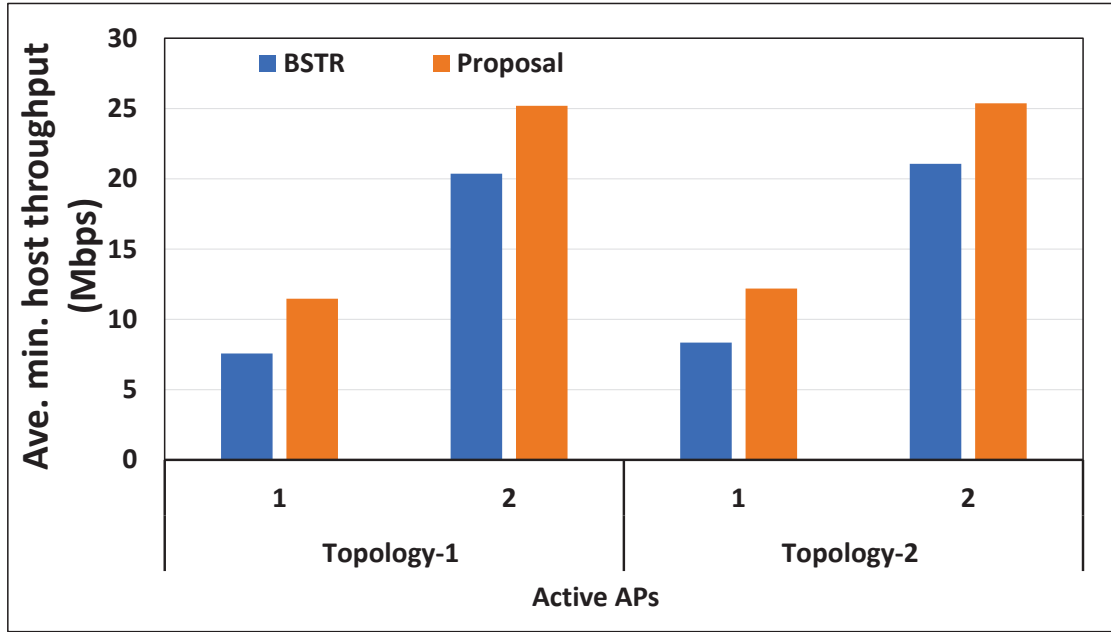


Figure 7.12: Minimum host throughput result.

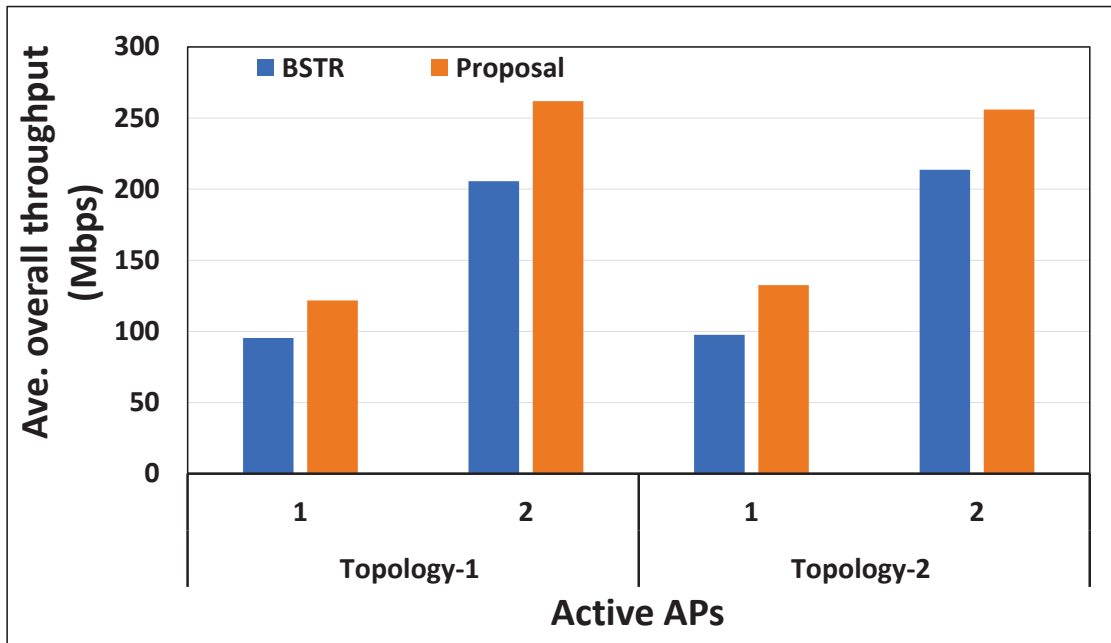


Figure 7.13: Overall throughput result.

7.4 Summary

This chapter presented the evaluation results for the *dual interface active AP configuration algorithm* through WIMNET simulations and testbed experiments. In the next chapter, we will present the related works to this thesis.

Chapter 8

Related Works in Literature

In this chapter, we provide a brief overview of the related works in the literature. A substantial amount of research works have been conducted to improve the performance of WLAN system and to reduce the operational cost.

In [86], Qiu et al. presented a joint optimization method for access point placement, power assignment, channel assignment, and resource unit assignment in dense IEEE 802.11ax WLAN. The objective of the proposal is to reduce the cost regarding the number of access points (APs) while meeting the requirements of specified failure tolerance and a specified two-tier flow. The authors designed a heuristic algorithm based on polynomial time complexity in order to find high-quality solutions and compared the throughput results by the greedy and random solution.

In [87], Kong et al. proposed a method for optimizing AP deployment based on the multi-purpose particle swarm optimisation algorithm (MOPSO). First, the performance of a single AP was evaluated using the random geometry theory to estimate the number of APs that should be deployed in the WLAN, based on the user's service needs. Then, the proposed scheme uses the MOPSO algorithm to obtain the optimal position and optimal transmit power of the APs. Finally, a greedy algorithm is used in order to remove the redundant APs.

In [88], Liu et al. proposed analytical expressions to measure the throughput performance of WLAN while considering the inter-network interference. They adopted the airtime concept to express the simultaneous transmission time and the carrier-sensing duration to understand the carrier-sensing duration for this proposal.

In [89], Tewari and Ghosh addressed the issues of the efficient AP placements and the joint assignment of partially overlapping channels and transmission powers to the APs to maximize the network throughput.

In [90], Islam et al. presented a joint AP association and bandwidth-allocation scheme in order to optimize network throughput and utilization taking into account access points' (APs) capacity and user demand. They adopted the rounding and bandwidth allocation algorithm (RBAA) for this model.

In [91, 92], Zhi et al. and Liu et al. proposed novel AP deployment methods by using a genetic algorithm (GA) and fruit fly optimization algorithm (FOA), respectively. In both proposals, the AP locations and the transmit powers of all the activated APs are optimized while ensuring complete coverage of the desired service area.

In [93], Tian et al. proposed a genetic algorithm with the Cramer–Rao lower bound (GA-CRLB) method for finding optimal AP positioning that maximizes localization accuracy and coverage simultaneously. This method adopted CRLB as an evaluation methodology for evaluating the localization accuracy of the deployment plan, and then the genetic algorithm was used to rapidly

find the optimal AP deployment strategy.

In [94], Wan et al. jointly presented a user association and bandwidth allocation algorithm called the bandwidth-constrained association (BCA) to maximize the total network throughput while satisfying user device bandwidth requirements. They adopted an efficient polynomial time method for producing the best user association.

In [95], Heiba et al. proposed an active AP configuration algorithm that activates the minimum number of APs according to the number of users and their locations on the network. It finds optimal channel assignments to the active APs from the non-interfering channels to minimize the total interference. It also balances the loads among the active APs by simply switching users to the least loaded AP.

In [96], Tang et al. proposed an association control algorithm using the maximum aggregated bandwidth utility (MABU) to optimize the throughput in WLANs. Their solution achieves the load balancing between the APs while considering users' traffic demands.

In [97], Nishat et al. proposed a QoS differentiation scheme called Slickfi for high-speed WLANs. It aims to enhance the real-time application performance while maintaining the high network throughput. It achieves the high performance by isolating real-time traffic and data traffic using two radios in APs managing adjacent-channel interference and allowing the per-frame channel width adaptation to maximize the network throughput.

In [98], Tang et al. proposed a new optimal AP deployment (OAD) algorithm based on fuzzy C-clustering, which provides good performance in the maximum number of WLAN users and balanced the traffic load among the APs.

In [99], Farsi et al. presented a joint and separate optimization planning method for both the AP placement and the channel assignment in WLAN to ensure sufficient capacities of users. They adopted a fuzzy C-clustering algorithm for AP deployments.

In [100], Yan et al. presented an algorithm for AP deployment based on physical distance and channel isolation (DPDCI) in urban environments, which can effectively reduce interference and improve the system's overall throughput. The algorithm optimizes the placement of APs and allocates the channel to APs by adopting a genetic algorithm.

In [101], Abusubaih et al. analyzed the ability of IEEE 802.11n dual-band APs to support heterogeneous client adapters and applications. With testbed experiments, they concluded that dual-band APs can effectively reduce the negative effects of heterogeneity on clients' network adapters.

In [102], Lima et al. divided the WLAN design into two stages. The first stage solves the problem of the AP location using a non-dominated sorting genetic algorithm that aims minimizing the number of APs on the network. The second stage solves the problem of the channel assignment by modifying the degree of saturation algorithm (DSATUR) to take into account the load of each AP.

In [103], Fang and Lin presented an algorithm for the AP setup optimization in an indoor environment. Their algorithm aims choosing a proper set of AP locations so that the signal at a host is maximized and the noise is minimized simultaneously.

In [104], Eisenblatter et al. proposed a joint optimization algorithm for the AP placement and the channel assignment using the integer linear programming model. The objective of the AP placement is to maximize the total throughput by minimizing the interference between the APs.

Most of the approaches aim to enhance network performance by utilizing a single interface at the access point (AP). However, none of these approaches consider the utilization of dual interfaces based on IEEE 802.11n and 11ac standards at the AP. By incorporating dual interfaces, the number of active APs can be minimized in accordance with user demand, leading to improved network performance.

Chapter 9

Conclusion

This thesis presented the study of the *active AP configuration algorithm* for the *IEEE 802.11n/ac wireless local-area network (WLAN)* with *dual network interfaces* of access points (APs). The main goal of this algorithm is to minimize the number of active APs such that every host satisfies the *minimum host throughput constraint* in the network when any AP has the dual network interface with 11n at 2.4GHz and 11ac at 5GHz. By reducing the number of active APs, it aims to reduce the amount of energy consumed by the network.

Firstly, this thesis surveyed the *IEEE 802.11* wireless network technologies related to this study, including the overview of IEEE 802.11 WLAN, IEEE 802.11n and 11ac protocols, their features, and Linux tools for wireless networking.

Secondly, the thesis reviewed our previous studies related to this study, including the *elastic WLAN system*, the *throughput estimation model*, the *parameter optimization tool*, and the *active AP configuration algorithm* for single network interfaces of APs.

Thirdly, the thesis presented experimental observations for the two cases of interface selections in the dual interface at the same AP. It described the experimental setup and showed the throughput results. The results confirmed that the built-in *wireless network interface card (NIC)* of the *Raspberry Pi* device should be used for 11n and the *Archer T4U* wireless NIC adapter be for 11ac.

Fourthly, the thesis proposed the *active AP configuration algorithm* for the *dual interface* at the same AP in WLAN. This algorithm selects the associated AP and its network interface to each host. The *throughput estimation model* for 11ac is used together with the model for 11n. Moreover, the *throughput reduction factor* is incorporated into the *throughput estimation model* to improve the estimation accuracy when multiple hosts are associated with the same AP. Therefore, it can greatly enhance the network performance while reducing the number of active APs in the network for smaller power consumptions and less interferences. The effectiveness of the proposed algorithm was confirmed through simulations using the *WIMNET simulator*.

Finally, the thesis presented the testbed implementation of the proposal in the elastic WLAN system using Raspberry Pi and Linux PCs. At first, it evaluated the proposal when APs are operating under *non-channel bonding (non-CB)*. Due to the wide channel width, the *channel bonding (CB)* channels can provide higher throughputs. In a CB channel, two adjacent channels are combined into one channel. Therefore, secondly, it evaluated the proposal when the APs are operating under CB. The experimental results confirmed the effectiveness of the proposed algorithm.

In future studies, I will further improve the *active AP configuration algorithm* by considering partially overlapping channels (POCs), transmission power optimizations, and the *11ax* protocol. I will also study the extension of the active AP configuration algorithm with heterogeneous dual interfaces in WLAN. Then, I will evaluate the proposals to confirm their effectiveness in various network fields and topologies.

Bibliography

- [1] B. P. Crow, I. Widjaja, J. G. Kim, and P. T. Sakai, "IEEE 802.11 wireless local area networks," *IEEE Commun. Mag.*, vol. 35, no. 9, pp. 116-126, Sep. 1997.
- [2] L. Kassa, M. Davis, J. Cai, and J. Deng. "A new adaptive frame aggregation method for downlink WLAN MU-MIMO channels," *J. Commun.*, vol. 16. no. 8, pp. 311-322, Aug. 2021.
- [3] M. Elkhodr, S. Shahrestani, and H. Cheung, "Emerging wireless technologies in the internet of things: a comparative study," *Int. J. Wireless Mob. Netw.*, vol. 8, no. 5, pp. 67–82, Oct. 2016.
- [4] K. Pahlavan and P. Krishnamurthy, "Evolution and impact of wi-fi technology and applications: a historical perspective," *Int. J. Wireless Info. Netw.*, vol. 28, no. 1, pp. 3-19, Mar. 2021.
- [5] LAN MAN standards committee, " Part11: Wireless LAN medium access control (MAC) and physical Layer (PHY) specifications," *The Institute of Electrical and Electronics Engineering*, Jun. 1997.
- [6] M. S. A. Mamun, N. Funabiki, K. S. Lwin, M. E. Islam, and W.-C. Kao, "A channel assignment extension of active access-point configuration algorithm for elastic WLAN system and its implementation using Raspberry Pi," *Int. J. Netw. Comput.*, vol. 7, no. 2, pp. 248-270, Jul. 2017.
- [7] R. B. M. A. Abdelrahman, A .B. A. Mustafa, and A. A. Osman, "A Comparison between IEEE 802.11a, b, g, n and ac Standards," *IOSR J. of Comp. Eng.*, vol.17, issue 5, ver. III, pp. 26-29, Sep. 2015.
- [8] M. S. Gast, "802.11 n: a survival guide," O'Reilly Media, Inc, 2012.
- [9] M. S. Gast, " 802.11 ac: a survival guide: Wi-Fi at gigabit and beyond," O'Reilly Media, Inc 1st ed.; 2013.
- [10] E. Khorov, A. Kiryanov, A. Lyakhov, and G. Bianchi, "A tutorial on IEEE 802.11 ax high efficiency WLANs," *IEEE Commun. Surveys Tuts.* vol. 21, no. 1, pp. 197–216, Sep. 2019.
- [11] L. Deek, E. Garcia-Villegas, E. Belding, S. J. Lee, and K. Almeroth, "The impact of channel bonding on 802.11 n network management," in *Proc. ACM Conf. emerging Netw. Experi. Technol.*, no. 11, pp. 1-12, Dec. 2011.

- [12] M. Han, S. Khairy, L. X. Cai, and Y. Cheng, "Performance analysis of opportunistic channel bonding in multi-channel WLANs," in Proc. IEEE Global Comm. Conf. (GLOBECOM), pp. 1-6, Dec. 2016.
- [13] Y. Daldoul, D. E. Meddour, and A. Ksentini, "IEEE 802.11 ac: Effect of channel bonding on spectrum utilization in dense environments," in Proc. IEEE Int'l Conf. Comm. (ICC), pp. 1-6, May. 2017.
- [14] H. Gebre-Amlak, M. T. Islam, D. Cummins, M. A. Mansoori, and B.-Y. Choi, "Protocol heterogeneity issues of incremental high-density wi-fi deployment," in Proc. Int. Conf. Wired/Wireless Int.t Comm., pp. 159–170, Jun. 2018.
- [15] D. Gong and Y. Yang, "AP association in 802.11n WLANs with heterogeneous clients," in Proc. IEEE Conf. Comp. Comm. (INFOCOM 12), pp. 1440–1448, Mar. 2012.
- [16] S. L. Kang, G. Y. H. Chen, and J. Rogers, "Wireless LAN access point location planning," in Proc. Inst. Ind.l Eng. Asian Conf., pp. 907-914. Jul. 2013.
- [17] Y. Kim, M. S. Kim, S. Lee, D. Griffith, and N. Golmie, "AP selection algorithm with adaptive CCAT for dense wireless networks," in Proc. IEEE Wire. Comm. Net. Conf. (WCNC), pp. 1-6, Mar. 2017.
- [18] S. C. Roy, N. Funabik., K. I. Munene, H. Briantoro, , M. M. Rahman, F. Akhter, and M. Kuribayashi, "A study of throughput drop estimation model for two-link concurrent communications in 5GHZ IEEE 802.11ac WLAN," IEICE Society Conf., pp. S22–23, Sept. 14-17, 2021. (Online, Japan).
- [19] K. I. Munene, N. Funabiki, M. M. Islam, M. Kuribayashi, M. S. A. Mamun, R. W. Sudibyo, and W.-C. Kao, "An extension of throughput drop estimation model for three-link concurrent communications under partially overlapping channels and channel bonding in IEEE 802.11n WLAN," Adv. Sci. Technol. Eng. Syst. J., vol. 4, no. 4, pp. 94-105, Jul. 2019.
- [20] K. I. Munene, N. Funabiki, R. W. Sudibyo, M. M. Islam, M. Kuribayashi, and W. -C. Kao, "An extension of throughput drop estimation model for three-link concurrent communications under partially overlapping channels and channel bonding in IEEE 802.11n WLAN," IEICE Tech. Rep., vol. 118, no. 371, pp. 89-94, Dec. 2018.
- [21] S. C. Roy, N. Funabik, B. Wu, Y. Sun, E. E. Htet, and M. Kuribayashi, "A throughput drop estimation model with non-channel bonding for two-link concurrent communications in IEEE 802.11ac WLAN," IEICE Society Conf., pp. S7–8, Mar. 7-10, 2023.
- [22] J. P. Romero, J. S. González, R. Agustí, B. Lorenzo, and S. Glisic, "Power-efficient resource allocation in a heterogeneous network with cellular and D2D capabilities," IEEE Trans. Vehicular Tech. vol. 65, no. 11, pp. 9272-9286, Jan. 2016.
- [23] W. K. Soo, T. C. Ling, A. H. Maw, and S. T. Win, "Survey on load-balancing methods in 802.11 infrastructure mode wireless networks for improving quality of service," ACM Comp. Surveys (CSUR), vol. 51, no. 2, pp.1-21, Feb. 2018.
- [24] R. G. Garroppo, G. Nencioni, L. Tavanti, B. Gendron, and M. G. Scutellà, "Optimal Access Point Power Management for Green IEEE 802.11 Networks," Sensors, vol. 21, no. 6, pp. 2076 , Mar. 2021.

- [25] A. Trivedi, J. Gummeson, and P. Shenoy, "Empirical characterization of mobility of multi-device internet users," arXiv preprint arXiv:2003.08512, vol. 1, no. 1, pp. 1-23, May 2020. [Online]. Available: <https://arxiv.org/pdf/2003.08512.pdf>. Accessed 05 May, 2023.
- [26] "Electricity sector in Bangladesh," [Online]. Available: https://en.wikipedia.org/wiki/Electricity_sector_in_Bangladesh. Accessed 05 May, 2023.
- [27] "Electricity to transform rural Myanmar," [Online]. Available: <https://www.worldbank.org/en/news/feature/2015/09/16/electricity-to-transform-rural-myanmar>. Accessed 05 May, 2023.
- [28] M. S. A. Mamun, M. E. Islam, and N. Funabiki, "An active access-point configuration algorithm for elastic wireless local-area network system using heterogeneous devices," *Int. J. Netw. Comput.*, vol. 6, no. 2, pp. 395-419, Jul. 2016.
- [29] K. S. Lwin, N. Funabiki, C. Taniguchi, K. K. Zaw, M. S. A. Mamun, M. Kuribayashi, and W.-C. Kao, "A minimax approach for access point setup optimization in IEEE 802.11 n wireless networks," *Int. J. Netw. Comput.*, vol. 7, no. 2, pp. 187-207, Jul. 2017.
- [30] N. Funabiki, C. Taniguchi, K. S. Lwin, K. K. Zaw, and W. C. Kao, "A parameter optimization tool and its application to throughput estimation model for wireless LAN," in *Proc. Int. Works. Virtual Environ. Net. -Ori. Appli. (VENOA)*, pp. 701-710, Jul. 2017.
- [31] S. C. Roy, N. Funabik, K. I. Munene, M. M. Rahman, and M. Kuribayashi, "An extension of active access-point configuration algorithm to IEEE 802.11n and 11ac dual interfaces in wireless local-area network," *Int. J. Future Comput. Commun.*, vol. 11, no. 2, pp. 18-26, Jun. 2022.
- [32] S. C. Roy, N. Funabik, M. M. Rahman, B. Wu, M. Kuribayashi, and W. -C. Kao "A Study of the Active Access-Point Configuration Algorithm under Channel Bonding to Dual IEEE 802.11n and 11ac Interfaces in an Elastic WLAN System for IoT Applications," *Signals*, vol. 4, no. 2, pp. 274-296, Apr. 2023.
- [33] N. Funabiki, "Wireless mesh networks," *InTech-Open Access Pub.*, Jan. 2011.[Online]. Available: <http://www.intechopen.com/books/wireless-mesh-networks>. Accessed 05 May, 2023.
- [34] Raspberry Pi. [Online]. Available: <http://raspberrypi.org>. Accessed 09 May, 2023.
- [35] Archer T4U. [Online]. Available: <https://www.tp-link.com/jp/home-networking/adapter/archer-t4u/>. Accessed 09 May, 2023.
- [36] M. S. Gast, "802.11 wireless networks - the definitive guide," 2nd ed., O'Reilly, Sebastopol, Apr. 2005.
- [37] S. Banerji, and R. S. Chowdhury, "On IEEE 802.11: wireless LAN technology," *Int. J. Mobi. Net. Commun. Telematics*, vol. 3, no. 4, Aug. 2013.
- [38] S. Banerji, "Upcoming standards in wireless local area networks," *Wirel. Mob. Technol.*, vol. 1, no. 1, Sep. 2013.

- [39] I. Poole, "IEEE 802.11 Wi-Fi standards," [Online]. Available: <https://www.electronics-notes.com/articles/connectivity/wifi-ieee-802-11/what-is-wifi.php>. Accessed 11 May, 2023.
- [40] V. Beal, "What is 802.11 wireless LAN standards?," [Online]. Available: https://www.webopedia.com/TERM/8/802_11.html. Accessed 11 May, 2023.
- [41] C. C. Choon, "A study of active access-point selection algorithm for wireless mesh network under practical conditions," Ph. D. thesis, Grad. Sch. Natural Sci. Tech., Okayama Univ., Japan, Sep. 2015.
- [42] "IEEE 802.11 - Wikipedia," [Online]. Available: http://en.wikipedia.org/wiki/IEEE_802.11. Accessed 11 May, 2023.
- [43] M. E. Islam, "A study of access-point aggregation algorithm for elastic wireless local-area network system and its implementation," Ph.D. thesis, Grad. School of Natural Science and Technology, Okayama Univ., Japan, Mar. 2016.
- [44] S. Sendra, M. Garcia, C. Turro, and J. Loret, "WLAN IEEE 802.11 a/b/g/n indoor coverage and interference performance study," *Int. J. Adv. Netw. Service.*, vol. 4, no. 1, pp. 209-222, 2011.
- [45] M. M. Islam, "A study of dynamic access-point configuration and power minimization in elastic wireless local-area network system," Ph.D. thesis, Grad. Sch. Natural Sci. Tech., Okayama Univ., Japan, Sep. 2019.
- [46] K. I. Munene, "A study of throughput drop estimation models for concurrently communicating links in wireless local-area networks," Ph.D. thesis, Grad. Sch. Natural Sci. Tech., Okayama Univ., Japan, Apr. 2022.
- [47] P. Likhar, R. S. Yadav and K. M. Rao, "Securing IEEE 802.11g WLAN using open VPN and its impact analysis," *Int. J. of Netw. Security and Its Applications (IJNSA)*, Vol. 3, no. 6, pp. 97-113, Nov. 2011.
- [48] A. S. Tanenbaum, and D. J. Wetherall, "Computer networks," 5th ed., Pearson Prentice Hall, 2011.
- [49] J. H. Yeh, J. C. Chen, and C. C. Lee, "WLAN standards," *IEEE Potentials*, vol. 22, no. 4, Oct. 2003.
- [50] IEEE, "IEEE Std 802.11-2012, IEEE standard for information technology," [Online]. Available: <https://standards.ieee.org/findstds/standard/802.11-2012.html>. Accessed 11 May, 2023.
- [51] O. Bejarano, E. W. Knightly, and M. Park, "IEEE 802.11ac: from channelization to multi-user MIMO," *IEEE Comm. Mag.*, vol. 51, no. 10, pp. 84-90, Oct. 2013.
- [52] L. Verma, M. Fakharzadeh, and S. Choi, "Wifi on steroids: 802.11ac and 802.11ad," *IEEE Trans. Wirel. Comm.*, vol. 20, no. 6, pp. 30-35, Dec. 2013.
- [53] B. Bellalta, "IEEE 802.11 ax: High-efficiency WLANs," *IEEE Wirel. Commu.* vol. 23, no. 1, pp. 38-46, Feb. 2016.

- [54] IEEE, “IEEE SA standards board approvals,” [Online]. Available: <https://standards.ieee.org/about/sasb/sba/feb2021/>. Accessed 11 May, 2023.
- [55] IEEE, “IEEE 802.11ax-2021 - IEEE standard for information technology,” [Online]. Available: <https://standards.ieee.org/ieee/802.11ax/7180/>. Accessed 11 May, 2023.
- [56] M. M. Rahman, “A study of throughput control method for concurrently communicating multiple hosts in wireless local-area network,” Ph.D. thesis, Grad. Sch. Natural Sci. Tech., Okayama Univ., Japan, Sep. 2022.
- [57] A. Saif, M. Othman, “A reliable A-MSDU frame aggregation scheme in 802.11n wireless networks,” in Proc. Comp. Scie., vol. 21, Jan. 2013.
- [58] “IEEE 802.11n-2009 - Wikipedia,” [Online]. Available: http://en.wikipedia.org/wiki/IEEE_802.11n-2009. Accessed 11 May, 2023.
- [59] “IEEE 802.11ac-2013 - Wikipedia,” [Online]. Available: https://en.wikipedia.org/wiki/IEEE_802.11ac-2013. Accessed 11 May, 2023.
- [60] M. Vipin, and S. Srikanth, “Analysis of open source drivers for IEEE 802.11 WLANs,” in Proc. Int. Conf. Wireless Comm. Sensor Comput., pp. 1-5, Jan. 2010.
- [61] “arp-scan user guide,” [Online]. Available: <https://linux.die.net/man/1/arp-scan>. Accessed 11 May, 2023.
- [62] “Arp scan,” [Online]. Available: <https://launchpad.net/ubuntu/trusty/+package/arp-scan>. Accessed 11 May, 2023.
- [63] “nm-tool,” [Online]. Available: <http://linux.die.net/man/1/nm-tool>. Accessed 11 May, 2023.
- [64] “View your current network settings with nm-tool,” [Online]. Available: <https://nfolamp.wordpress.com/2010/05/21/view-your-current-network-settings-with-nm-tool/>. Accessed 11 May, 2023.
- [65] “Network manager,” [Online]. Available: <https://help.ubuntu.com/community/NetworkManager>. Accessed 11 May, 2023.
- [66] J. Malinen, “hostapd: IEEE 802.11 AP, IEEE 802.1 X. WPA/WPA2/EAP/RADIUS authenticator,” [Online]. Available: <https://w1.fi/hostapd/>. Accessed 11 May, 2023.
- [67] D. J. Barrett, R. E. Silverman, and R. G. Byrnes, “SSH, the secure shell: the definitive guide,” O’Reilly Media Inc., 2005.
- [68] T. Ylonen and C. Lonvick, “The secure shell (SSH) protocol architecture,” 2006.
- [69] “OpenSSH,” [Online]. Available: <http://www.openssh.com/>. Accessed 11 May, 2023.
- [70] O. Terpollari, “How to install and configure OpenSSH server in Linux,” [Online]. Available: <http://www.tecmint.com/install-openssh-server-in-linux/>. Accessed 11 May, 2023.

- [71] “Run /execute command using SSH,” [Online]. Available: <https://www.cyberciti.biz/faq/unix-linux-execute-command-using-ssh/>. Accessed 11 May, 2023.
- [72] “nmcli - command-line tool for controlling networkmanager,” [Online]. Available: <http://manpages.ubuntu.com/manpages/maverick/man1/nmcli.1.html>. Accessed 11 May, 2023.
- [73] P. Michel, “about iw,” [Online]. Available: <https://manpages.ubuntu.com/manpages/bionic/man8/iwconfig.8.html>. Accessed 11 May, 2023.
- [74] “Learning linux commands: sed,” [Online]. Available: <https://linuxconfig.org/learning-linux-commands-sed>. Accessed 11 May, 2023.
- [75] “Iperf - the TCP, UDP and SCTP network bandwidth measurement tool”, [Online]. Available: <https://iperf.fr/>. Accessed 11 May, 2023.
- [76] D. B. Faria, “Modeling signal attenuation in IEEE 802.11 wireless LANs,” Tech. Report, TR-KP06-0118, Stanford Univ., Jul. 2005.
- [77] T. Lu, N. Funabiki, M. M. Islam, K. I. Munene, and R. W. Sudibyo “An improved throughput estimation model for concurrent communications of multiple hosts in wireless local-area network,” IEEE Hiroshima Section Student Symp., pp. 360-363, Nov.-Dec. 2019.
- [78] L. A. Wolsey, “An analysis of the greedy algorithm for the submodular set covering problem,” *Combinatorica*, vol.2, no. 4, pp. 385-393, Dec. 1982.
- [79] D. P. Williamson and D. B. Shmoys, “The design of approximation algorithms,” Cambridge Univ. Press, Apr. 2011.
- [80] S. K. Debnath, N. Funabiki, K. S. Lwin, M. S. Al Mamun, R. W. Sudibyo, and S. Huda, “ Raspberry Pi configuration for access-point and its throughput measurements in IEEE802.11n wireless networks ,” in Proc. Tech. Rep., Japan, pp. 101-106, Dec. 2016.
- [81] “Hostapd”, [Online]. Available: <https://w1.fi/hostapd/>, (Accessed 24 May, 2023).
- [82] “Setting up a Raspberry Pi as a wireless access point”, [Online]. Available: <https://thepi.io/how-to-use-your-raspberry-pi-as-a-wireless-access-point/>. Accessed 24 May, 2023.
- [83] “nmcli - command-line tool for controlling NetworkManager”, [Online]. Available: <http://manpages.ubuntu.com/manpages/maverick/man1/nmcli.1.html>. Accessed 29 May, 2023.
- [84] “Using the network manager command-line tool, nmcli”, [Online]. Available: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Networking_Guide/sec-Using_the_NetworkManager_Command_Line_Tool_nmcli.html. Accessed 29 May, 2023.
- [85] Qualcomm. “Band-Steering for dual-band Wi-Fi access points”, [Online]. Available: https://www.qualcomm.com/content/dam/qcomm-martech/dm-assets/documents/qc_band_steering_wp.pdf. Accessed 05 Jun. 2023.

- [86] S. Qiu, X. Chu, Y. W. Leung, and J. K. Y. Ng, “Joint access point placement and power-channel-resource-unit assignment for IEEE 802.11 ax-Based dense WiFi network with QoS requirements,” *IEEE Trans. Mob. Comput.*, vol. 22, no. 5, pp. 2771 - 2788, Nov. 2021.
- [87] Z. Kong, D. Wu, X. Jin, S. Cen, and F. Dong, “Improved AP deployment optimization scheme based on multi-objective particle swarm optimization algorithm,” *KSII Trans. Internet Infor. Syst. (TIIS)*, vol. 15, no. 4, pp. 1568–1589, Apr. 2021.
- [88] J. Liu, T. Aoki, Z. Li, T. Pei, Y. J. Choi, K. Nguyen, and H. Sekiya, “Throughput analysis of IEEE 802.11 WLANs with inter-network interference,” *Appl. Sci.* vol. 10, no. 6, p. 2192, Mar. 2020.
- [89] B. P. Tewari and S. C. Ghosh, “Efficient AP placement through power control and partially overlapping channel assignment,” *Wireless Personal Commun.*, vol. 110, no. 1, pp. 223-244, Jan. 2020.
- [90] M. T. Islam and B. Y. Choi, “Jointly maximizing throughput and utilization for dense enterprise WLANs,” in *Proc. IEEE Int. Smart Cities Conf. (ISC2)*, pp. 753–759, Oct. 2019.
- [91] Z. Zhi, J. Wu, X. Meng, M. Yao, Q. Hu, and Z. Tang, “AP deployment optimization in non-uniform service areas: A genetic algorithm approach,” in *Proc. IEEE 90th Vehicular Tech. Con. (VTC2019-Fall)*, pp. 1–5, Sep. 2019.
- [92] P. Liu, X. Meng, J. Wu, M. Yao, and Z. Tang, “AP deployment optimization for WLAN: A fruit fly optimization approach,” in *Proc. IEEE/CIC Int. Conf. Com. China (ICCC)*, pp. 478–483, Aug. 2018.
- [93] Y. Tian, B. Huang, B. Jia, and L. Zhao, “Optimizing wifi AP placement for both localization and coverage,” in *Proc. Int. Conf. Algor. Archi. Parallel Proce.*, pp. 489–503, Nov. 2018.
- [94] X. Wan, X. Guan, Y. Shen, and B. Y. Choi, “Optimal user association in hybrid WLANs under bandwidth constraints,” in *Proc. IEEE Int Smart Cities Conf. (ISC2)*, pp. 1–8, Sep. 2018.
- [95] N. Heiba, M. Alghoniemy, M. Elwekeil, A. Mokhtar, and M. R. M. Rizk, “An adaptive algorithm for channel assignment and load balancing in elastic IEEE 802.11 WLANs,” in *Proc. Int. Comp. Eng. Conf. (ICENCO)*, pp. 343-347, Dec. 2017.
- [96] H. Tang, L. Yang, J. Dong, Z. Ou, Y. Cu, and J. Wu, “Throughput optimization via association control in wireless LANs,” *Mobi. Netw. Appl.*, vo. 21, no. 3, pp. 453-466, Sep. 2016.
- [97] K. Nishat, F. Javed, S. Salman, N. Yaseen, A. Fida, and I. A. Qazi, “Slickfi: A service differentiation scheme for high-speed wlans using dual radio aps,” in *Proc. 12th Int. Conf. Eme. Net. EXper. Tech. (CoNEXT ' 16)*, pp. 177–189, Dec. 2016.
- [98] S. Tang, L. Ma, and Y. Xu, “A novel AP placement algorithm based on user distribution for indoor WLAN system,” *China Commu.* vol. 13, no. 10, pp. 108–118, Oct. 2016.
- [99] A. Farsi, N. Achir, and K. Boussetta, “WLAN planning: separate and joint optimization of both access point placement and channel assignment,” *Ann. Telecommun.*, vol. 70, no. 5, pp. 263-274, Jun. 2015.

- [100] D. Yan, C. Zhang, H. Liao, L. Yang, P. Li, and G. Yang, "AP deployment research based on physical distance and channel isolation," *Abst. Appl. Anal.*, vol. 2014, p. 941547, Jan. 2014.
- [101] M. A. Abusubaih, S. N. Eddin, and A. Khamayseh, "IEEE 802.11n dual band access points for boosting the performance of heterogeneous WiFi networks," in *Proc. 8th ACM workshop Perfor. Monito. Measure. Heterogen. wireless and wired netw.*, pp. 1–4, Nov. 2013.
- [102] M. P. Lima, E. G. Carrano, and R. H. C. Takahashi, "Multiobjective planning of wireless local area networks (WLAN) using genetic algorithms," in *Proc. IEEE Cong. Evol. Comp.*, pp. 1-8, Jun. 2012.
- [103] S. H. Fang and T. N. Lin, "A novel access point placement approach for WLAN-based location systems," in *Proc. Wireless Commun. Netw. Conf.*, pp. 1-4, Apr. 2010.
- [104] A. Eisenblatter, H. F. Geerdes, and I. Siomina, "Integrated access point placement and channel assignment for wireless LANs in an indoor office environment,' in *Proc. Int. Symp. World. Wirel. Mob. Multi. Netw. (WoWMoM)*, pp. 1-10, Jun. 2007.