A Thesis on

Improve Connection Time, Congestion Control and Power Save Mechanism Of 802.11 Wireless Devices with Changes in MAC Layer

Submitted in fulfilment of the requirements for the award of the degree of

Doctor of Philosophy

by

VISHAL BHARGAVA

(2K17/PhD/CO/11)

Under the supervision of

Prof. N. S. RAGHAVA

Department of ECE, DTU, Delhi

Department of Computer Science and Engineering

Delhi Technological University

Delhi, India

2023

Dedicated to

My Wife Shafali and My beloved Parents

S.L. Bhargava

&

Shobha Bhargava

Candidate declaration

I hereby declare that the thesis entitled "Improve Connection Time, Congestion Control And Power Save Mechanism Of 802.11 Wireless Devices With Changes In MAC Layer" submitted to Delhi Technological University, Delhi, in the partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy in the Department of Computer Science, is an original work and has been done by myself under the supervision of Prof. N.S. Raghava (Supervisor), Department of Electronics & Communication Engineering, Delhi Technological University, Delhi, India.

The interpretations presented are based on my study and understanding of the original texts. The work reported here has not been submitted to any other institute for the award of any other degree.

Date:

(Vishal Bhargava)
Roll No. 2K17/PhD/CO/11
Department of CSE
Delhi Technological University,
Delhi-110042, India

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
(Govt. of National Capital Territory of Delhi)
Shahbad Daulatpur, Main Bawana Road,
Delhi-110042, India

Certificate

This is to certify that the work embodied in the synopsis entitled **"Improve Connection Time, Congestion Control And Power Save Mechanism Of 802.11 Wireless Devices With Changes In MAC Layer"** submitted by **Mr. Vishal Bhargava**, Roll No: 2K17/Ph.D/CO/11 as a part time scholar in the Department of Computer Science & Engineering, Delhi Technological University, is an authentic work carried out by him under my guidance. This work is based on the original research and has not been submitted in full or in part for any other diploma or degree of any university to the best of my knowledge and belief.

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

(Prof. N.S. Raghava)
Supervisor
Professor, ECE Department
Delhi Technological University,
Delhi-110042

Acknowledgment

I address my sincere thanks to Almighty God for giving me the inner power to complete my thesis and guide me in every step of my life.

It is an immense pleasure to have the opportunity to express my heartiest gratitude to everyone who helped me throughout this research journey. With immense joy and heartfelt gratitude, I would like extend my indebtedness to my supervisor, Prof. N. S. Raghava (Dept. of Electronics & Communication Engineering), for his invaluable guidance, mentorship, encouragement, and patience. During the research, his motivation and encouragement have made me strive to work harder to achieve my goals. I am deeply humbled and indebted to my supervisor for continually motivating me to persevere and making me believe in myself during the times of hardships. His technical expertise, precise suggestions, kind nature, and detailed, timely discussions are wholeheartedly appreciated.

Also, my sincere thank goes to Delhi Technological University for considering my candidature for this course. I am also very thankful to Prof. Jai Prakash Saini, Vice-Chancellor, Delhi Technological University, Delhi, India, who has been a constant source of enthusiasm. He has always motivated young researchers like me to pursue excellence to achieve higher goals in academics and research. Also, my sincere thanks reciprocate to Dr. Vinod Kumar (HoD, Dept. of Computer Science and Engineering), Prof. Rajni Jindal (Chairperson DRC, Dept. of Computer Science and Engineering) for insightful comments and valuable suggestions. Special thanks to my seniors and colleagues of Delhi Technological University, Delhi, India. My sincere thanks to all the professors, faculty, researchers, and nonteaching staff of the Computer Science Department.

I also wish to take this opportunity to thank all my teachers who have taught me and shaped me into the person I am, aggravated me to be an academician, and have directly indirectly made me capable of succeeding in completing this research work. I am deeply thankful to all my colleagues and friends during my journey as a Ph.D. scholar. The engaging discussions, brainstorming sessions, and collaborative teamwork significantly impacted my growth as an independent researcher.

I would also like to thank my wife who always supported me in all my endeavors and believed in me and encouraged me in all the challenging times. Finally, but most importantly, I would like to express my deepest gratitude to my parents who stood by me like a pillar of strength and always supported me to realize my goals. I will cherish their utmost love and blessings throughout my life.

Date: **Vishal Bhargava**

Place: Delhi, India.

Abstract

802.11 handheld devices have increasingly supported based wireless local area networks (WLAN). With the growth of Wi-Fi over time, performance and quality are facing many challenges. Wi-Fi works on the principle of the IEEE 802.11-based carrier sense multiple access collision avoidance (CSMA/CA) to transmit packets and distributed coordination function (DCF) protocol based on inter-frame spacing used to make the gap between two frames. DCF protocol functioning & its performance depends on the number of participating devices in the environment. With the increasing number of 802.11 devices, congestion increase in the environment.

In the DCF manner, an 802.11 device gets limited time in the environment to finish its activity, with a rapid increase in the number of devices in the 802.11 environments. The critical operation is always connecting a Wi-Fi device with another device. Another device can be an access point or any other Wi-Fi device (ad-hoc or Wi-Fi direct mode).

However, supporting WLAN functionality is hugely energy-consuming since the connectivity must be maintained even when the device is idle. The power management defined in the 802.11 standards does not identify detailed techniques to handle the problem caused by power consumption-affecting factors (PCAFs) since most of these factors have not been specified in the standards.

The Wi-Fi experience of users is influenced by various quality metrics, including throughput, latency, connection, the probability of successfully connecting to Wi-Fi access points, the time taken to set up a Wi-Fi connection, security, power consumption by battery-operated devices, and collision/congestion control. Among these metrics, the probability of successfully connecting to Wi-Fi access points, the time taken to set up a Wi-Fi connection, power consumption, and collision/congestion control is considered the most essential. This thesis aims to investigate how different factors affect the above-discussed metrics in 802.11 WLANs. To simplify the problem, only infrastructure mode is taken into account. A couple of sub-problems are derived from the main problem. The protocol-related practical solution proposed for all the identified sub- problems. The simulators and real hardware have validated the accuracy of the obtainedsolutions according to availability. This research successfully provides a more reliable, effective, optimal, easy to implement/deploy and practical solution for Wi-Fi device problems.

The three main areas of focus are:

Improved connection time: The proposed changes to the MAC layer could potentially improve the connection time of 802.11 wireless devices. This is because the changes could reduce the number of handshakes and other overhead that is required to establish and maintain a connection.

Improved congestion control: The proposed changes to the MAC layer could also potentially improve the congestion control of 802.11 wireless networks. This is because the changes could allow for more efficient use of the environment, which could help to prevent congestion and packet loss.

Improved power save mechanism: The proposed changes to the MAC layer could also potentially improve the power save mechanism of 802.11 wireless devices. This is because the changes could allow devices to sleep for longer periods of time without losing connectivity, which could help to extend the battery life of devices.

In addition to these specific contribution points, the thesis could also make a number of general contributions to the field of wireless networking. For example, the thesis could provide new insights into the design of MAC layers for wireless networks, and it could also help to identify new opportunities for improving the performance of 802.11 wireless networks.

TABLE OF CONTENTS

List of Abbreviations

AC	Access Category
AIFS	Arbitration Inter Frame Space
AP	Access Point
ARP	Address Resolution Protocol
ATIM	Ad hoc Traffic Indication Message
BA	Block Acknowledge
BLE	Bluetooth Low Energy
BT	Bluetooth
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CW	Contention Window
DCF	Distributed Coordination Function
DHCP	Dynamic Host Configuration Protocol
DTIM	Delivery Traffic Indication Message
EDCA	Enhanced Distributed Channel Access
FOTA	Firmware Over The Air
GTK	Group Transient Key
IFS	Inter Frame Space
IoT	Internet of Things
LI	Listen Interval
MAC	Media Access Control
MIMO	Multiple-Input Multiple-Output

NAV	Network Allocation Vector
OBSS	Overlap Basic Service Set
OS	Operating System
PHY	Physical Layer
PM	Power Management
PS	Power Save
PSM	Power Saving Mode
QoS	Quality of Service
RSSI	Received Signal Strength Indicator
SOC	System on Chip
SSID	Service Set Identifier
STA	Station
TIM	Traffic Indication Map
TWT	Target wake time
TXOP	Transmission Opportunity
VHT	Very High Throughput
WLAN	Wireless Local Area Network
WoW	Wake-On-Wlan
WSN	Wireless Sensor Network

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

Wi-Fi (802.11) is the fastest-growing communication medium in the present era. There are billions of Wi-Fi devices in the current market, and they are expected to grow multiple times to the global population. Wi-Fi technology has seen tremendous growth in the last decade. Wi-Fi old generation (802.11a, 802.bg, 802.11n, 802.11ac) to 6th generation Wi-Fi 802.11ax. Now 7th generation of Wi-Fi that is 802.11be, is also knocking on the door.

With the growth, the problem also occurs. Throughput & security-wise, these 802.11 standards [1] are improving significantly. Devices are increasing exponentially, so the environmental noise is also increasing parallelly. This makes congestion in the environment and collision between frames happens. In this manner, when the number of devices in the 802.11 environment increases, each 802.11 device will get limited time to finish its activity. New amendments are happening; However, the Wi-Fi connection procedure is still thesame as the first standard. Wi-Fi connection time still needs lots of improvement. Userswant to roam without any impact on user browsing. The challenge is to backward scan without interrupting the existing connection during roaming. Connection of a Wi-Fi device with another device which can be AP or any other 802.11 device (ad-hoc or peer-to-peer mode), is the most important operation. Many kinds of research worked on theimportance of connection time and the factors that can affect it most [2,3].

With the increased demand for battery-operated and IoT devices intoday's scenario, it is vital to take care of device power consumption. While maintainingconnection with an AP, the 802.11 device should be power-savvy. While looking for power-saving ways in this research, Wi-Fi connection and maintenance also take care of using lesser power-consuming technology. Identifying different use cases where low power technology can help save energy and maintain 802.11 connection is part of the research.

This thesis aims to study an issue with Wi-Fi at the MAC layer that impacts power efficiency and addresses congestion control and connectivity issues in 802.11 WLANs. The scope of research is limited to infrastructure mode, and several sub-problems will be derived from the main problem. Firstly, find and recognize congestion, connection

and power consumption related facets in the 802.11 standard (mainly considered on 802.11n/802.11ac). CSMA/CA, scanning, authentication, association and power-save factors are studied deeply. Secondly, this thesis will evaluate the existing method and introduce protocol through measurements on a testbed. Thirdly, a simulation platform will be developed to explore the problem further. Lastly, a new protocol/algorithm for improving congestion control, fast connection, and power conservation performance in 802.11 WLANs with results will be proposed.

Section 1.1 provide details about the IEEE network. Section 1.2 contains the motivation of the study and discusses the challenges of 802.11 MAC, whereas section 1.3 elaborates upon the working mechanism of Wi-Fi. Section 1.4 discusses and explains the Wi-Fi connection process. Section 1.5 talks about the power management introduction for 802.11. Section 1.6 contains the research objective of the study. Section 1.7 describes the outline of the thesis, and the chapter ends with the list of publications in section 1.8.

1.1 IEEE Network

Before understanding the concept of 802.11 WLAN, it is important to first discuss 802 standard key concepts. Figure 1.1 shows the 802 group components according to the OSI model.

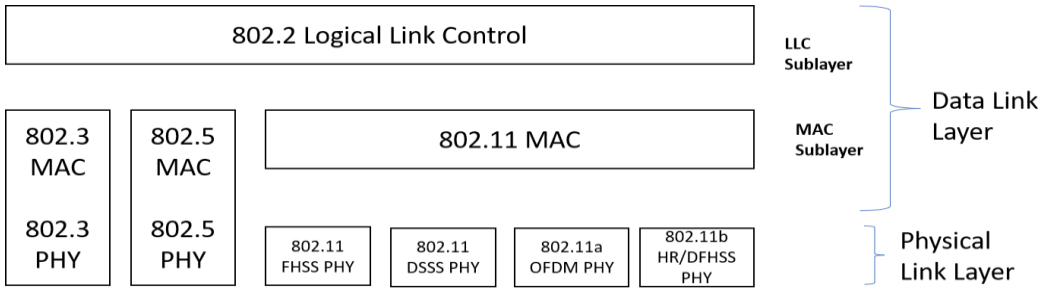

Figure 1.1 Relationship between IEEE 802 family and OSI model

IEEE 802 is a family of networking standards evolved by an organization. This organization is called the Institute of Electrical and Electronics Engineers (IEEE). These standards define the OSI model's physical and data-link layers, which is the commonly adopted model to describing the communication protocols used in networks. These standards provide the foundation for many widely used networking technologies, including Ethernet, Wi-Fi, and Zigbee. The IEEE 802 standards are continuously updated to reflect the ever-evolving networking landscape, and new standards are added

as new technologies are developed.

The IEEE 802 standards are divided into several subfamilies (table 1.1), each specifying a different type of network or technology. Some of the main subfamilies include:

Table 1.1 IEEE 802.11 Family

IEEE 802.3	Ethernet networks
IEEE 802.11	Wireless LANs (Wi-Fi)
IEEE 802.15	wireless personal area networks (WPANs)
IEEE 802.16	wireless metropolitan area networks (WMANs)
IEEE 802.1	LAN/MAN management and internetworking

Two lower layers of the OSI model (physical and data link layer) are the main focus area for IEEE 802.11 specifications. MAC (sub-part of data link layer) and physical (PHY) component is present in all 802.11 networks. PHY is responsible for details of transmission and reception and MAC define a certain rule to access to medium and send the data. The frequency-hopping spread-spectrum (FHSS) physical layer and the direct-sequence spread-spectrum (DSSS) link layer are two physical layers that make up the basic 802.11 specifications. Later iterations of 802.11 introduced more physical layers. A high-rate direct-sequence layer (HR/DSSS) is specified by 802.11b, whose products first appeared on the market in 1999 and account for most of the installed base. The basis of the physical layer is orthogonal frequency division multiplexing is described by 802.11a (OFDM).

1.1.1 802.11 Terminologies and Design

The 802.11 LAN (Local Area Network) standard consists of several components that work together to provide wireless network connectivity. These combined components offer a secure and reliable wireless network for users to access network resources and communicate with each other. Figure 1.2 shows these components and describe one-by-one below:

Figure 1.2 Components of 802.11 LANs

Distribution system

The 802.11 distribution system is the system that distributes wireless signals throughout a building or area. It is made up of central wireless devices which are connected to a wired network and communicate with 802.11 devices such as notebooks, mobile phones, and tablets. These central devices work together to provide seamless wireless coverage and to manage the communication between 802.11 devices and the ethernet network.

Access points

The above discusses the central device, basically an AP. An IEEE 802.11 AP is a networking apparatus that enables 802.11 devices to connect to a wired world. It acts as a flyover between 802.11 devices and the wired infrastructure. An AP typically includes a wireless transmitter, receiver and antenna to communicate with wireless devices and an ethernet interface to talk to the other world. Access points can be set up to work in many wireless modes such as infrastructure mode or ad-hoc mode and provide a range of characteristics such as security, data rates, name of network and QoS. Access points can also be configured to work with different wireless standards, such as 802.11a/b/g/n and 802.11ac/ax. In WLAN, an AP performs two specific tasks: First is to transfer packets to the destination station and the second is toconnect a wired and wireless network.

Wireless medium

Wi-Fi specifies the technology for wireless local area networks (WLANs) formed on 2.4 GHz. 5.0 GHz and 6.0 GHz bands. The wireless medium in 802.11 is the radio

frequency (RF), which is used to transmit and receive frames between wireless machines. This wireless medium is shared in all connected devices of the network, and 802.11 protocol works on carrier sense multiple access with collision avoidance (CSMA/CA) mechanism.

Stations

A station device is a device that can connect to an AP in order to communicate with other devices in the network. This typically includes laptops,smartphones, tablets, and other Wi-Fi supported devices. The station device can connect to the access point wirelessly and communicate with other devices in the network to exchange information such as data, voice, and video.

Basic Service Set

A basic service set (BSS) is a composition of many STAs and a maximum of one access point. In present days, WLAN service is used in addition to existing wired networks, but in future, it can fully replace wired network

1.1.2 The 802.11 MAC

The 802.11 specification is centred around the Media Access Control (MAC) layer, whose responsibility is to manage user data transmitted over the air. This layer is integrated with the physical layer and is responsible for the core framing operations and communication with wired network backbones. The specification also allows for interoperability between different physical layers, regardless of their varying transmission speeds.

1.2 Challenges for the MAC

The traditional wired and wireless network environment differs from each other, which create a challenging situation for network protocol designers. Figure 1.3 covers the 802.11 Media Access Control (MAC) protocol faces challenges that can impact its performance. These challenges include:

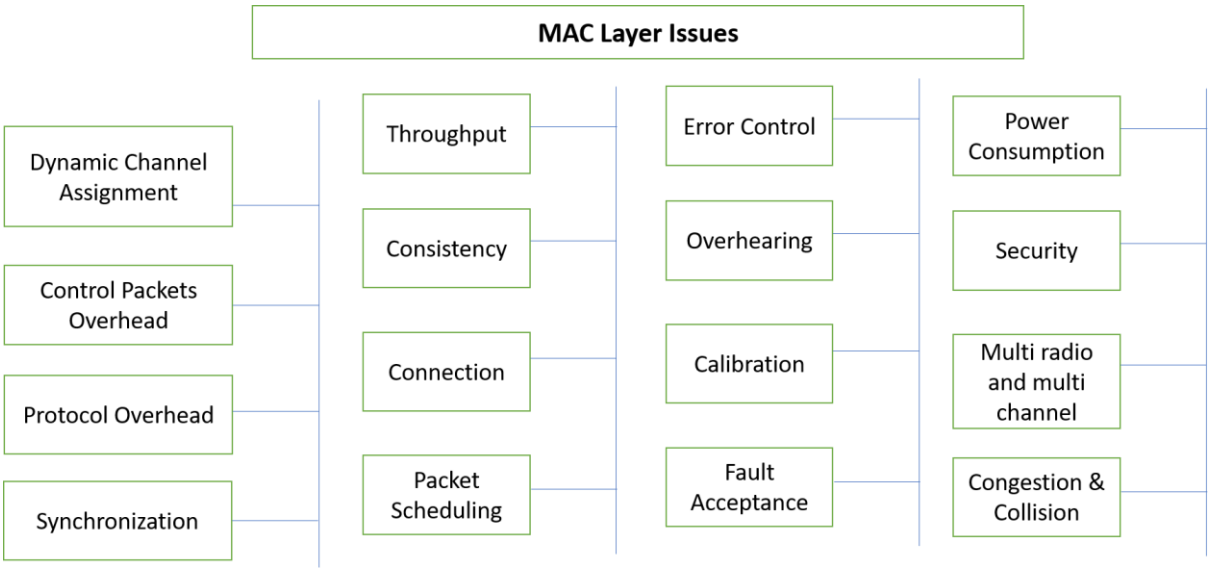

Figure 1.3 MAC Layer Issues

- Interference: Devices using the same frequency range can interfere with the 802.11 network and impact performance.
- Hidden Node Problem: When the coverage areas of multiple access points overlap, the "hidden node problem" can cause collisions and reduce network performance.
- Bandwidth Constraints: The limited amount of available bandwidth can become a bottleneck in congested areas, reducing network performance.
- Security: 802.11 networks can be vulnerable to security threats such as eavesdropping, spoofing, and denial-of-service attacks, making security a major concern.
- Power Management: Battery-operated devices in 802.11 networks may face power consumption issues.
- Connection: The process of connecting to an access point and transitioning from one access point to another can be challenging, particularly in networks with a high density of access points.
- QoS: Ensuring a consistent level of service for all network users can be difficult in 802.11 networks, which operate on a best-effort delivery basis.
-

1.3 Wi-Fi working

To transmit packets, Wi-Fi uses Carrier Sense Multiple Access Collision Avoidance (CSMA/CA) mechanism, and to make a gap between two frames, Wi-Fi uses DCF protocol which is built on an Inter-frame spacing mechanism. 802.11 uses inter-frame spacing for all high-level frames to avoid collision and to device access medium randomly; Contention Window (CW) mechanism introduces into the distributed wireless medium. Figure 1.4 shows Inter-frame spacing and their relationship which provide priority access for a pick transmission of frames necessary for correct network operation. 802.11 defined several types of interframe spacing. DCF based Interframe spacing (IFS) is discussed here:

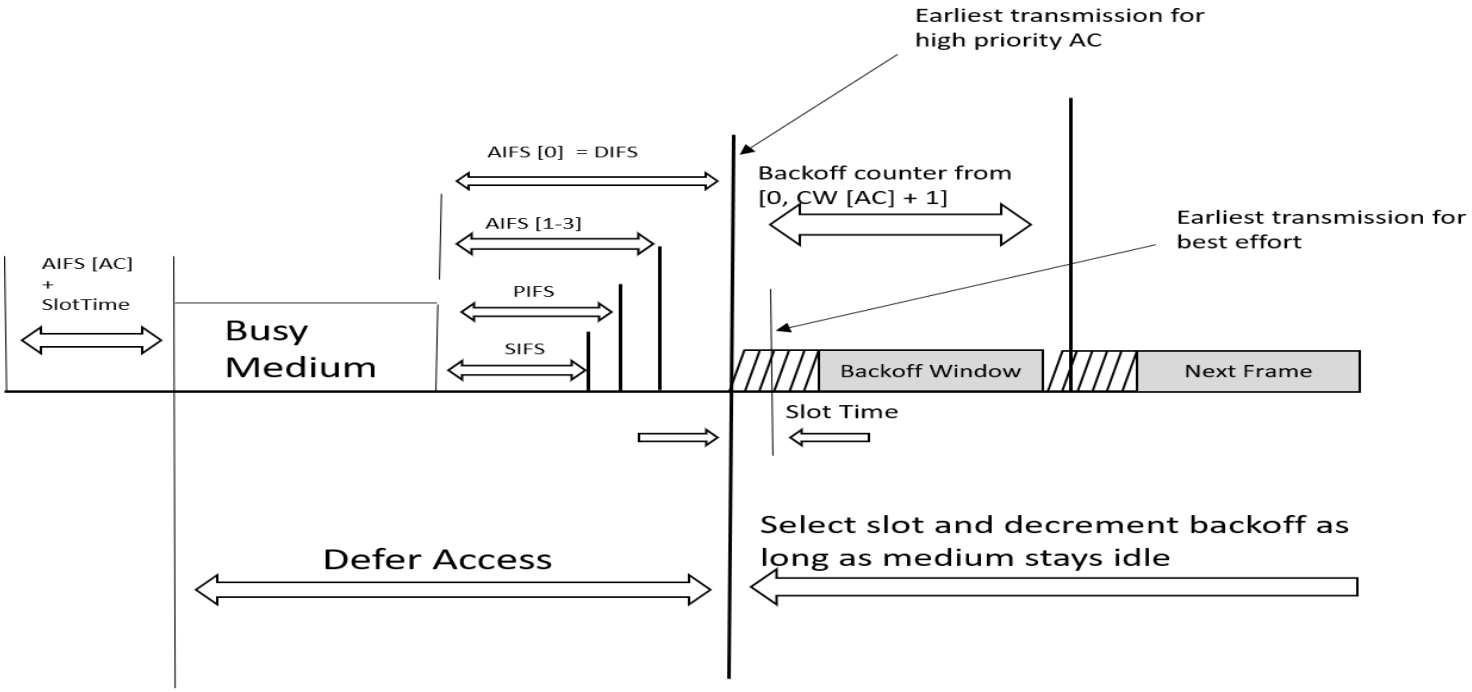

Figure 1.4 IFS Relationship

The Short Inter Frame Spacing (SIFS) worth is employed for control frame (like ACK & CTS) for high priority ack based data; DCF Inter Frame Spacing (DIFS) is employed for non-QoS data frames; Arbitrated Inter Frame Spacing (AIFS) is employed for QoS information frames and is variable supported the WMM Access class (AC) to that the frame is allotted & these values change according to modulation.

Before each frame transmission, Wi-Fi stations wait for a random timer which varies according to the contention window. As we discussed, Wi-Fi works on CSMA/CD. A

different device can also send data at the same time, and this is called an external collision. When an external collision happens, the station does random back-off & increase contention window (Figure 1.5). By increasing the contention window size after each collision, the stations are more likely to wait for a longer period of time before transmitting again. This reduces the probability that two or more stations will transmit at the same time and minimizes the number of collisions.

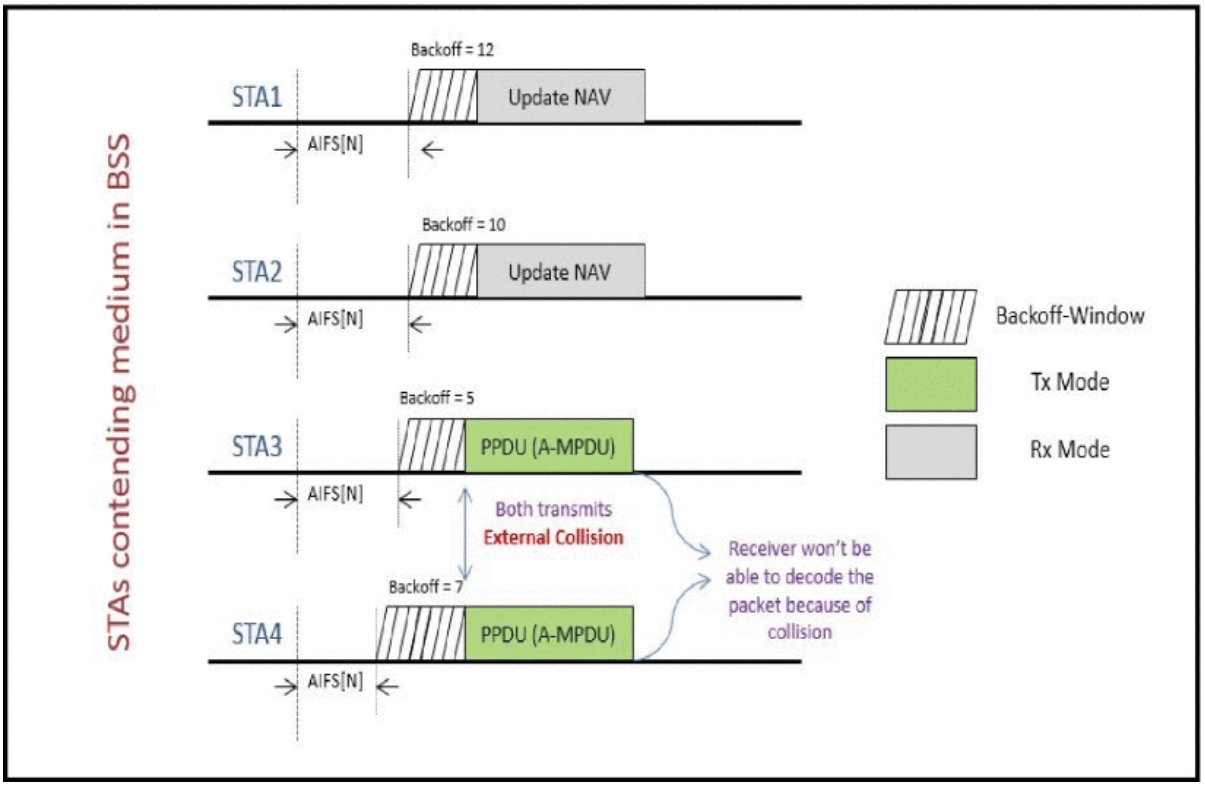

Figure 1.5 External Collision

If a station does not receive an acknowledgment frame after transmitting a packet, it knows that a collision has occurred. In this case, the station will double the size of its contention window and try to transmit the packet again. This process will continue until the contention window reaches its maximum size. This can be known as Truncated Binary Exponential Backoff. The initial tiny contention window size is marked as Contention Window Minimum (CWMin), and thelargest boundary size is marked as Contention Window maximum (CWMax). Once WMM QoS is used, a mathematical advantage provides to frame in the forms of inter-frame spacing and contention window size, which varies according to frame category. This technique of probability-based medium competition introduces an excessivequantity of network overhead to reduce the likelihood of a frame collision.

The 802.11e specification defines a priority scheme for the different types of traffic.

Wi-Fi have four access categories:

- Background data
- Best efforts
- Voice
- Video

Actually, this priority queue creates to prioritize the data frames of every station. These queues are named with the above four access categories according to data type. So, depending on the type of data placed in the respective queue, if it is video data, it will be placed in the 4th queue & categorized as video access categories.

If a device has a certain type of data which will transmit at the same time [4], an internal collision can occur. Figure 1.6 shows same internal collision, suppose several application wants to send data at same time.

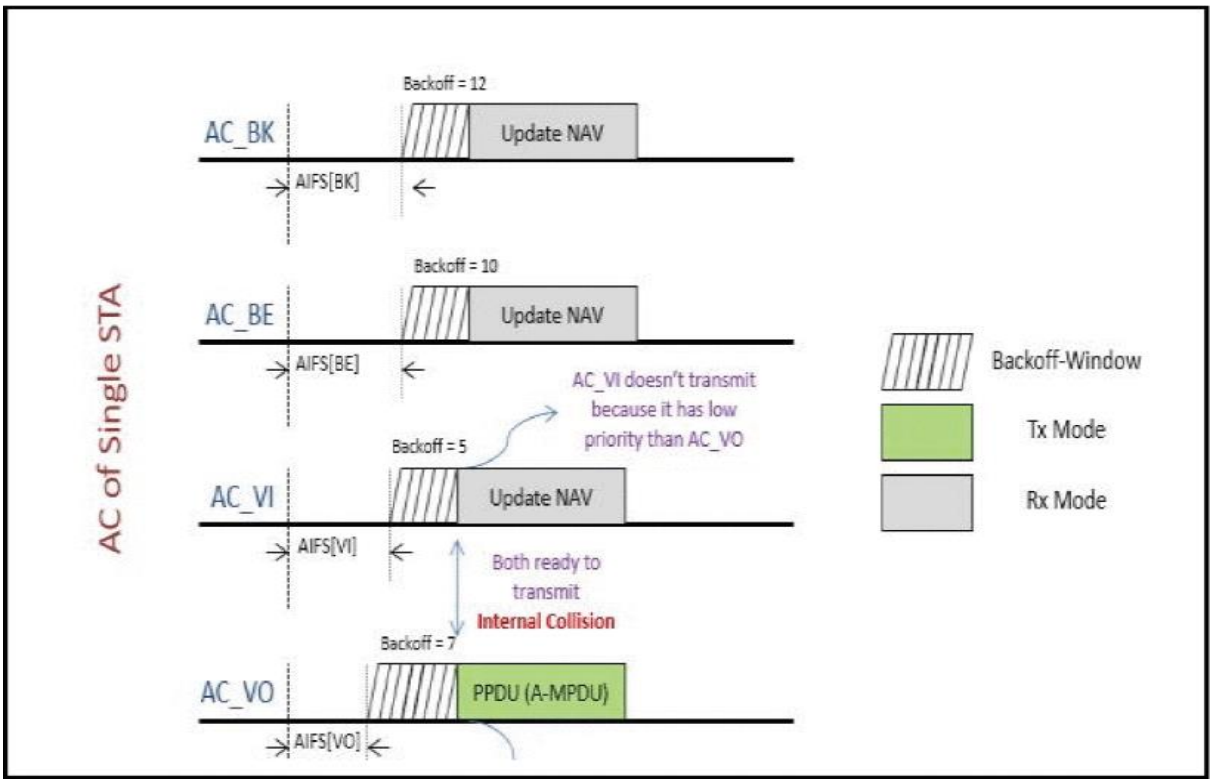

Figure 1.6 Internal Collison

The amended IEEE 802.11 wireless protocol specification suggests using a priority theme that may give up to eight priority categories for Wi-Fi traffic. This amendment was added to IEEE 802.11e Wi-Fi specification to support QoS. Wi-Fi alliance also

provides a certification test to test this type of device which supports 802.11e through the Wi-Fi Multi Media (WMM) certification method.

Some major changes are done in the operation of 802.11 to improve the operation of the initial 802.11 distributed coordination operation, which is currently referred to as EDCA. The major changes are:

- These four priority queues are subject to per station; every queue is treated as a station.
- AIFS, i.e. Arbitrated Inter-Frame Spacing values, are introduced to give priority to one access category over another one, and it is applicable for all information and management frames, and it's enhanced the DCF capabilities.
- Random backoff time is also different for every priority queue, which is drived from CWmin and CWmax values.

The performance of the IEEE 802.11 DCF protocol is proportional to the number of challenging Wi-Fi stations and the size of theCW [5].

First, inter-frame spacing establishes baseline intervals that sure forms of frames are needed to attend to before having the ability to transmit.

The backoff time & IFS calculation is done independently for each queue and parallelly decremented while the system waits for the ideal environment. When two queues attempt to transmit simultaneously, an internal collision occurs; in this case, higher priority will be granted access by MAC (like voice over best effort), and another queue will take this as a physical collision, stop transmission & it will increment its counter of retry, and according to binary exponential backoff increase their CW values. In this manner, a queue work as a self-oriented station. The following Figure 1.7 illustrates these priority queues, When a station (e.g., a Wi-Fi device or access point) receives data frames, it places them into one of these four priority queues based on the specified priorities or the type of service (ToS) field in the frame. The station then uses different contention parameters, such as contention window size AIFS, for each queue. This way, higher-priority queues have shorter backoff times and gain more opportunities to access the channel, thus improving the QoS for time-sensitive traffic.

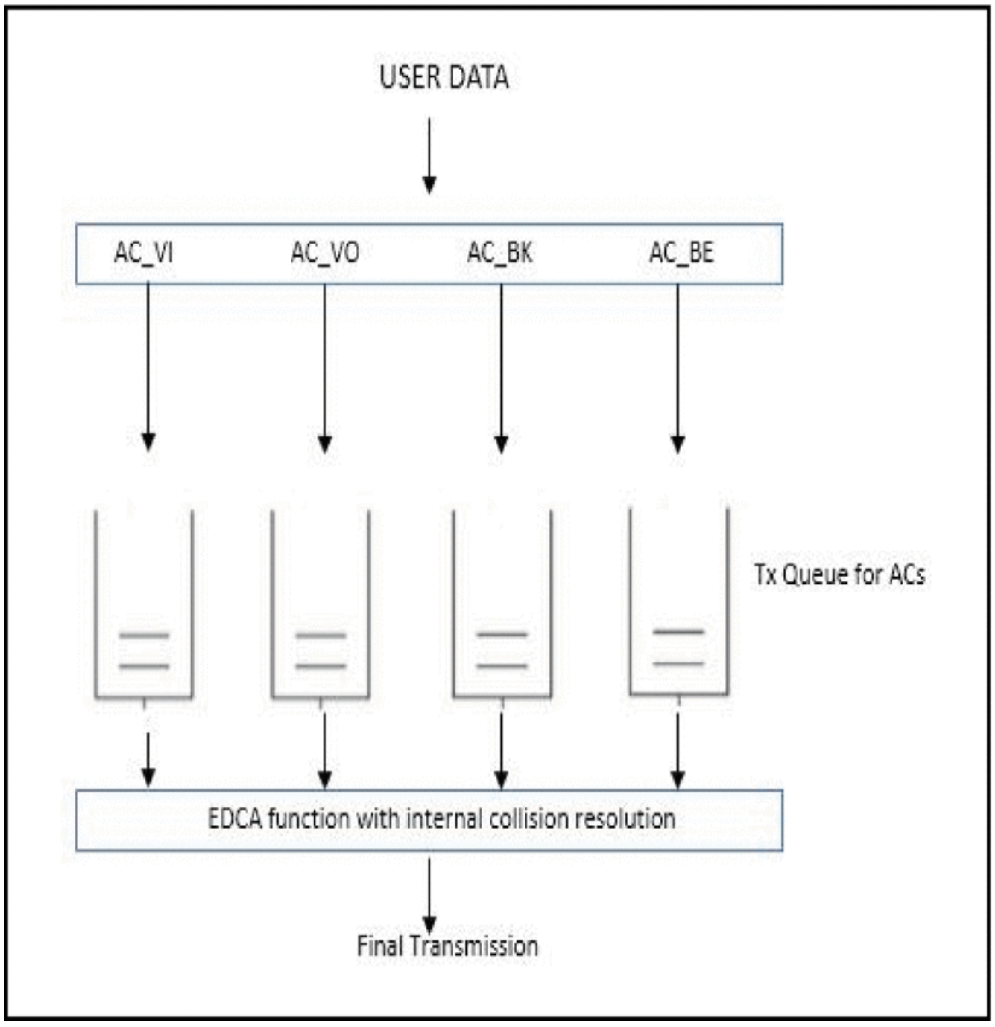

Figure 1.7 Frame Priority Queue

1.4 Wi-Fi Connection Process

Wi-Fi devices connection is a standard process [1] defined by the IEEE 802.11 specification. Any Wi-Fi station device which wants to connect with an AP starts with a scanning operation. Earlier devices have 2.4 GHz band support only, but now most devices also come with 5 GHz band support, and some have an additional 6 GHz band, so scanning work increases for them. To save time, devices perform an active scan compared to a passive scan (performed for DFS channels). More details will cover under section 1.4.1.

Figure 1.8 shows the Wi-Fi connection in open security mode. Scan, Authentication, Association and DHCP process are part of it. When considering WPA/WPA2 security, a four-way handshake is performed after the completion of the association process. In the connection process after the scan DHCP process takes maximum time [3]. Connection sub-phases are scan, authentication/association and DHCP assignment.

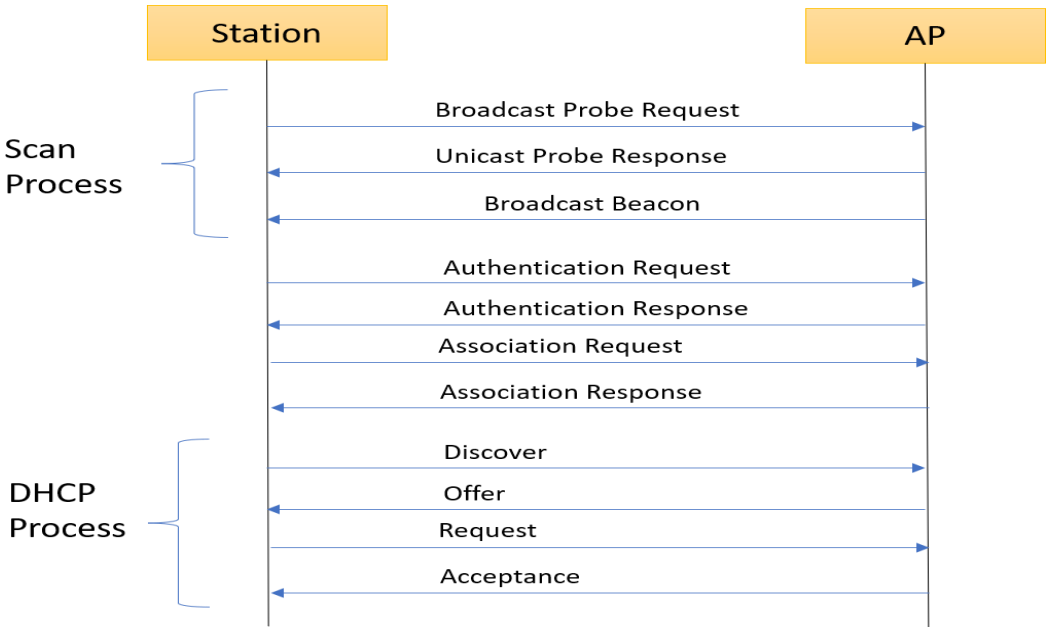

Figure 1.8 Wi-Fi Connection between Station and Access Point.

1.4.1 Scan Phase

The process of locating a desired AP through a station device is called a scan. There are two categories of scans (Figure 1.9): active and passive [1].

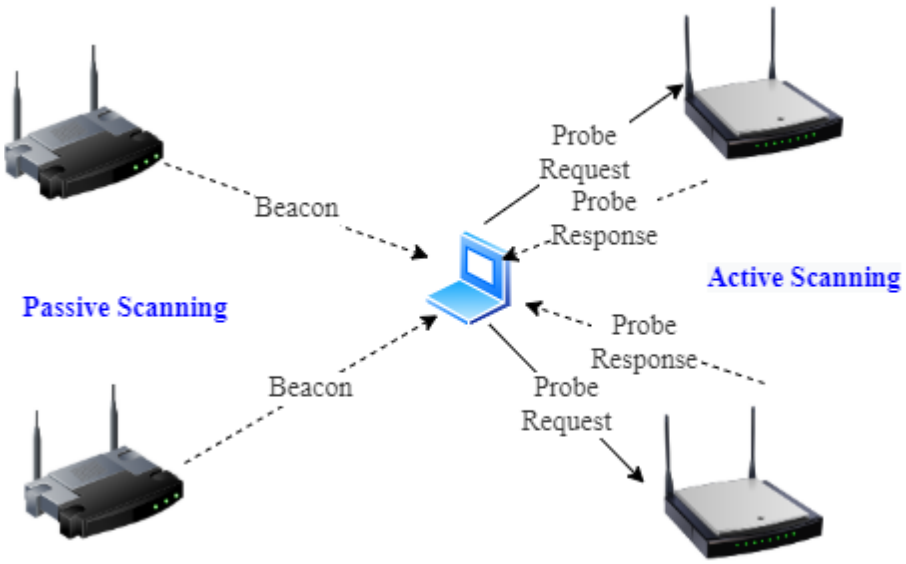

Figure 1.9 Wi-Fi Scanning Methods

Active scan: In active scanning, the station device sends out a Probe Request frame to the AccessPoint, which can either be a unicast or broadcast. The Access Point responds with a Probe Response, which the station uses to make a connection decision.

Passive scan: In passive scanning, the station decides to connect based on the Beacon frame received from the Access Point. Although passive scanning is a power-saving

technique, it can be slowercompared to active scanning [137].

1.4.2 Authentication/Association Phase

After the scan, it is the Wi-Fi station's turn to demonstrate its capabilities to the Access Point, such as its authenticity and supported features. Based on these points, the Access Point decides if the station can connect to it or not. During this phase, a 4-step process of frame exchanges, including an Authentication Request, an Authentication Response, an Association Request, and an Association Response.

1.4.3 DHCP Phase

In scanning and authentication/association phase, AP and Wi-Fi only interact from the viewpoint of the MAC layer. Packet created on MAC layer and transmitted. Upper OSI layer don’t participate. Now, it's an application and IP layer's turn, a station first needs an IP address so the application can use it for communication with other devices.

After a successful association, the station device will interact with the DHCP server which can be inside of the access point or can be a different entity. As an IP address is needed for communication, this phase is also considered in the connection phase, and actual real data transfer after this one.

Every network device has two types of addresses to communicate, so the wireless interface card also has two addresses. The MAC address is a unique identifier assigned to the WNIC by the manufacturer. It is a 12-digit hexadecimal number that is burned into the WNIC's hardware, and The IP address is a logical address that is assigned to the WNIC by a DHCP server. It is a 32-bit number that is used to identify the WNIC on the network. IP address work on layer 3(networking layer) can be considered as layer three address, and MAC address is considered as layer two address.

Dynamic Host Configuration Protocol (DHCP) is used to provide an IP address to a device.

DHCP mechanism is a combination of 4 processes (DORA). Stated below are the message exchanges between the DHCP client and server (Figure 1.10).

- Discover
- Offer

- Request
- Acknowledge

Below diagram shows the exchange of message (Discover, Offer, Request & Acknowledge) between the DHCP client and Server:

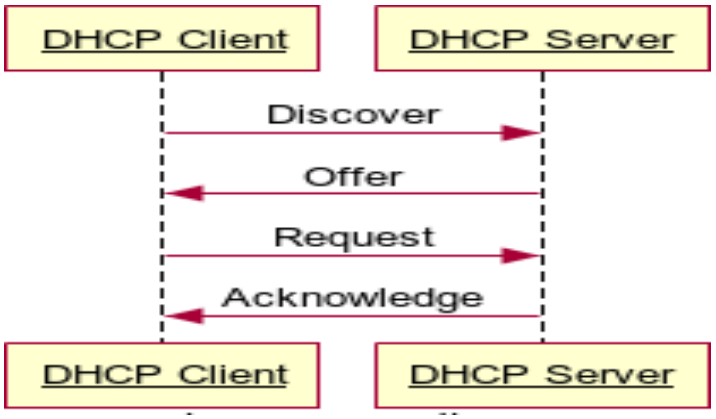

Figure 1.10 Exchange of DHCP Messages between DHCP Client and DHCP Server

DHCP frame exchange between a client and server, here specifically data focused on the data link layer (layer 2) and networking layer (layer 3).

Discover frame or message1

Device which is searching for an IP address sends a DHCP Discover message intended to search appropriate DHCP server. Figure 1.11 shows DHCP discover message from layer 2 perspective and Figure 1.12 from layer 3 perspective. It is broadcast on both layer.

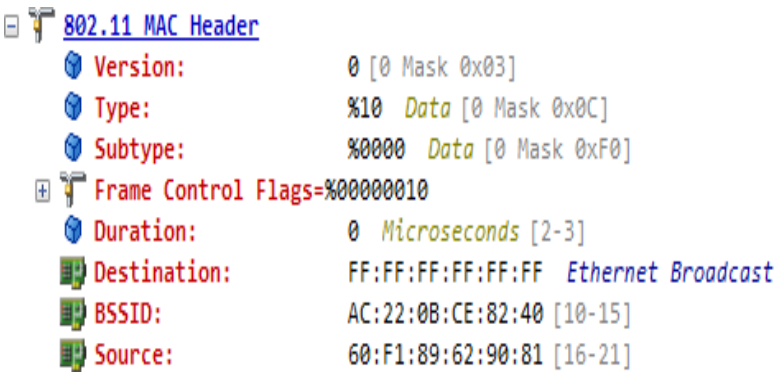

Figure 1.11 Sniffer capture of DHCP Discover (MAC Layer Perspective)

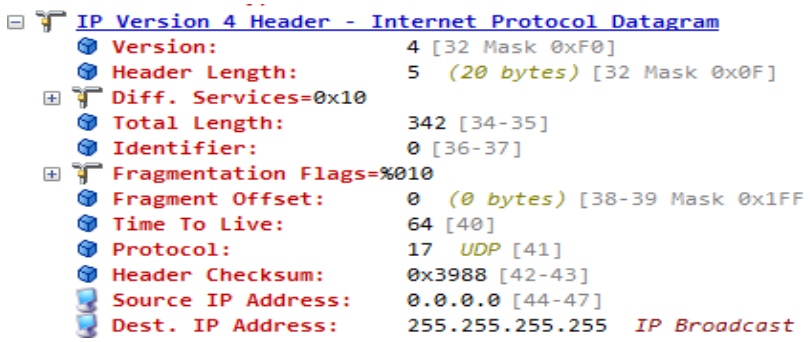

Figure 1.12 Sniffer capture of DHCP Discover (IP Layer Perspective)

Offer frame or message 2

In response to Discover message, the DHCP server sends the DHCP offer message (shown in Figure 1.13).

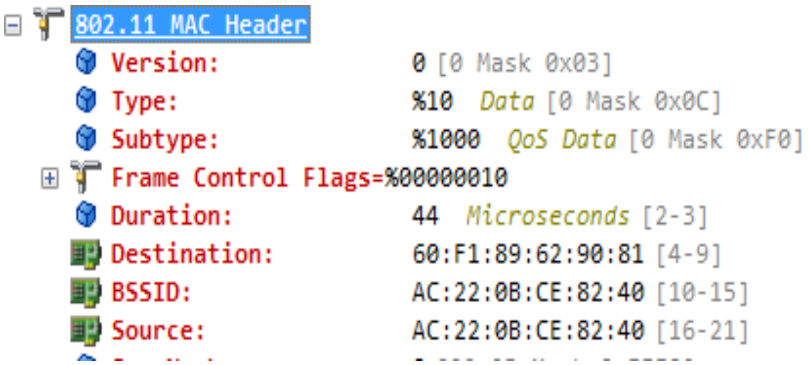

Figure 1.13 Sniffer capture of DHCP Offer(MAC Layer Perspective).

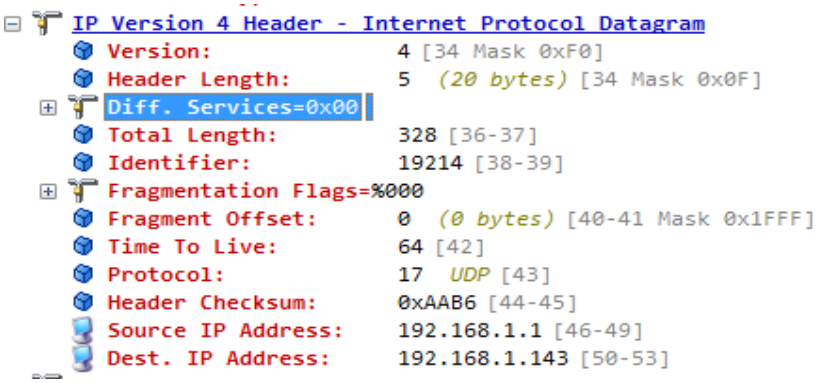

Figure 1.14 Sniffer capture of DHCP Offer (IP Layer Perspective).

Request frame or message 3

After receiving the offer, the DHCP client sends a DHCP Request message to the server, in which the client either accept IP address given by server or it can request the new one. It can be seen in Figure 1.15 that at the mac layer as a unicast frame while broadcast

on the IP layer (Figure 1.16) and in this frame DHCP server provides an IP to the client.

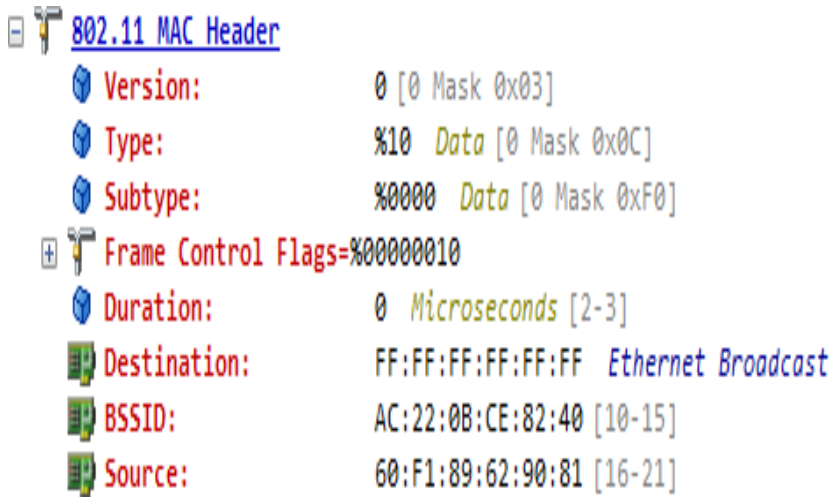

Figure 1.15 Sniffer capture of DHCP Request (MAC Layer Perspective).

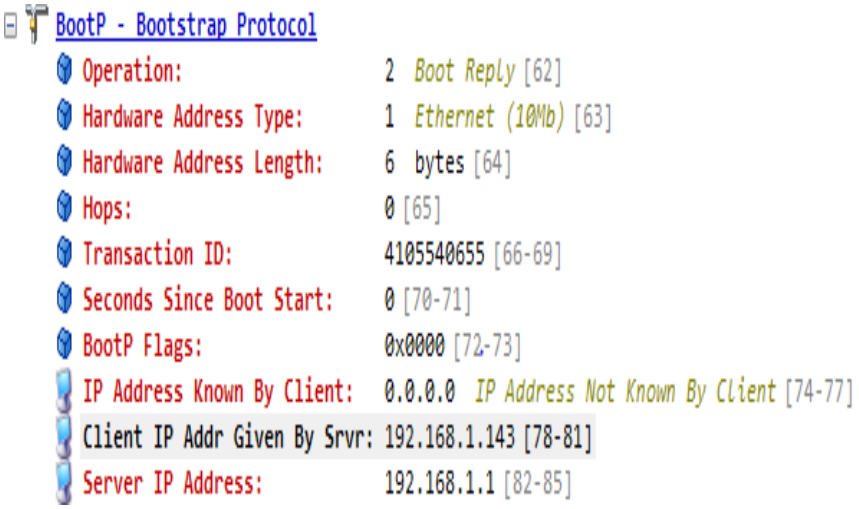

Figure 1.16 Sniffer capture of DHCP Request (IP Layer Perspective).

Acceptance frame or message 4

After the request frame, the server sends an acceptance/acknowledge frame, whether it has accepted the client IP or denied it. The reason code (a 16-bit field in the frame exchange) is also mentioned in case of failure so a client can get information about why the DHCP server has rejected the request. This frame is unicast on both layers. Figure 1.17 cover from MAC layer perspective while Figure 1.18 cover from IP layer perspective.

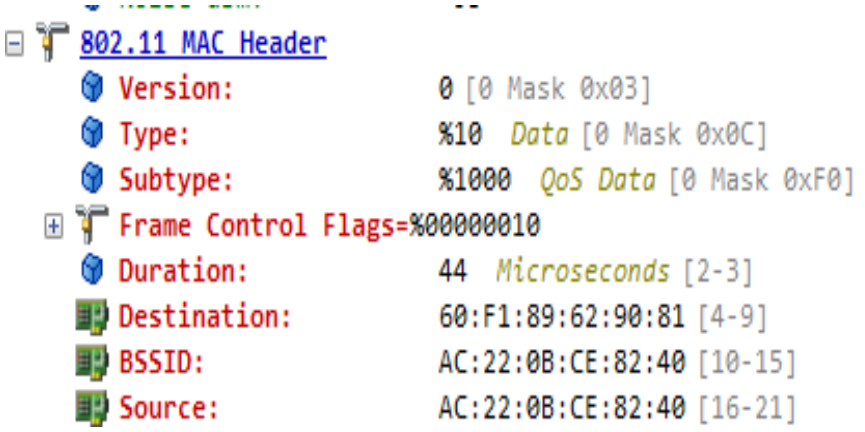

Figure 1.17 Sniffer capture of DHCP Acknowledge (MAC Layer Perspective)

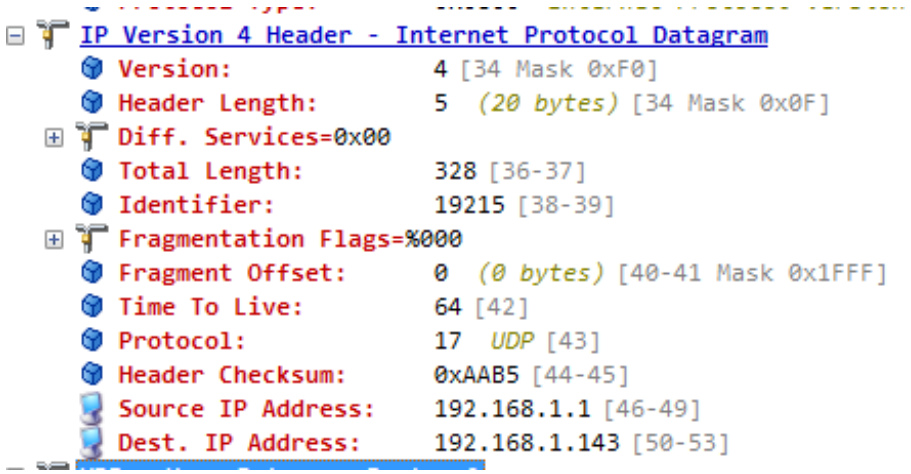

Figure 1.18 Sniffer capture of DHCP Acknowledge (IP Layer Perspective)

Below Figure 1.19 shows complete sniffer capture for a station to access-point Wi-Ficonnection in open security mode:

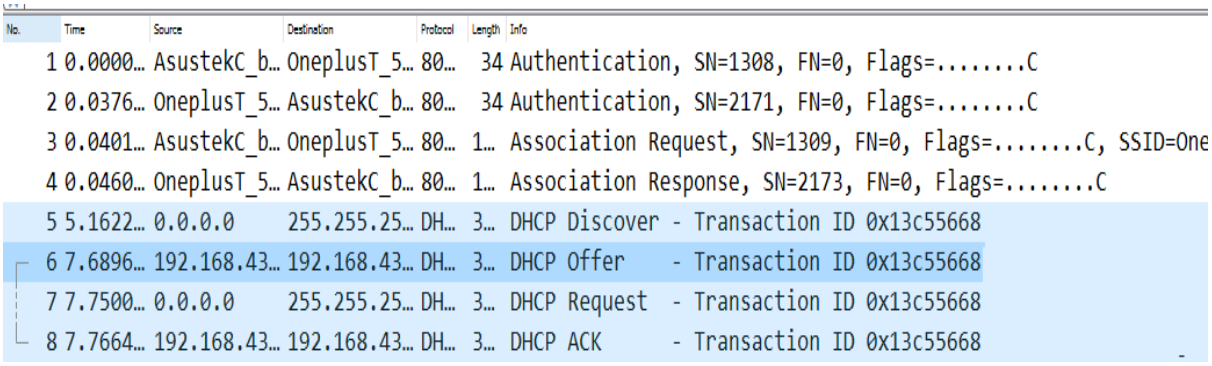

Figure 1.19 Sniffer capture of Wi-Fi Connection in open mode.

1.5 Power management in IEEE 802.11

According to IEEE 802.11 specifications, any device which can connect with Wi-Fi AP can be called an STA device. A Wi-Fi station can be in two modes. The first is the active

mode, where a device is always awake, and AP immediately transmits frames tothe client. The second mode is the PS (Power Save) mode, where a client is in a doze state. In this mode, AP buffers the data destinated for the client. In the power save mechanism, STA radio cannot transmit/receive, and the radio can go into a doze state from an active state. Let's discuss how a Wi-Fi station switches between active to PS mode. In this process, if an STA wants to move into a power save mode, it notifies theaccess point via any data frame, setting the Power Management (PM) bit to 1 (shown in Figure 1.20), and AP start the buffering of data for STA. The PM bit is located in theframe control field of the MAC header in 802.11 frames. The legacy PS-Poll or NULL data frame retrieves data from the AP side. In legacy mode, a control frame, i.e., PS- Poll frame STA sends to recover every single data frame as reflected in Figure 1.22.

How does the STA determine that the AP is storing data for it? The STA utilizes the beacon interval of the access point to know whether the AP is holding any data packet. The AP indicates buffered frames using the Beacon frames' Traffic Indication Map (TIM) element. The LI plays an important role when the device goes into power-save and how much memory buffer is assigned via AP for the station. LI negotiates during the association phase. Figure 1.21 shows LI value sniffer capture from association response frame. Using LI value STA lets AP know how many beacons STAwill go into power-save mode. LI is communicated in units of the beacon interval. Forexample, in (Figure 1.22) association request frame, STA sets listen interval to 1. If STAnever wants to go into power save mode, the LI value is put as 0.

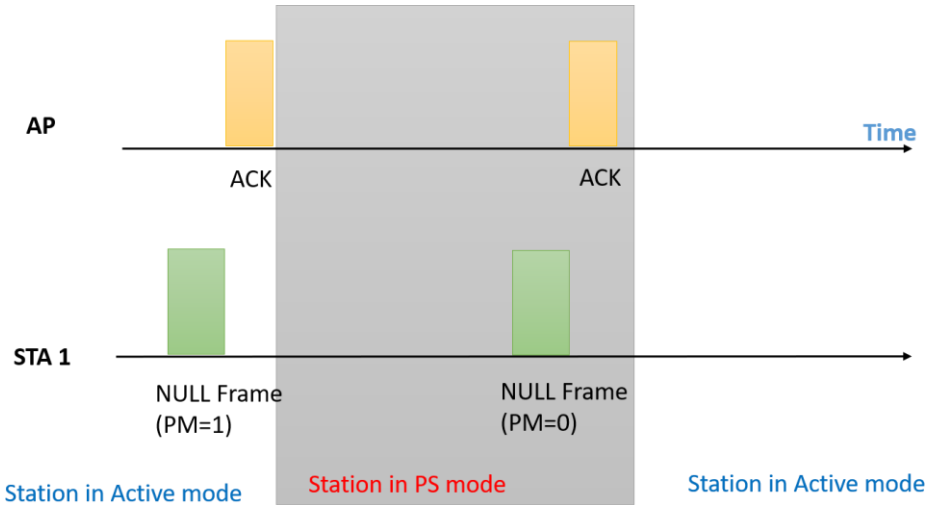

Figure1.20 Power Management Transition

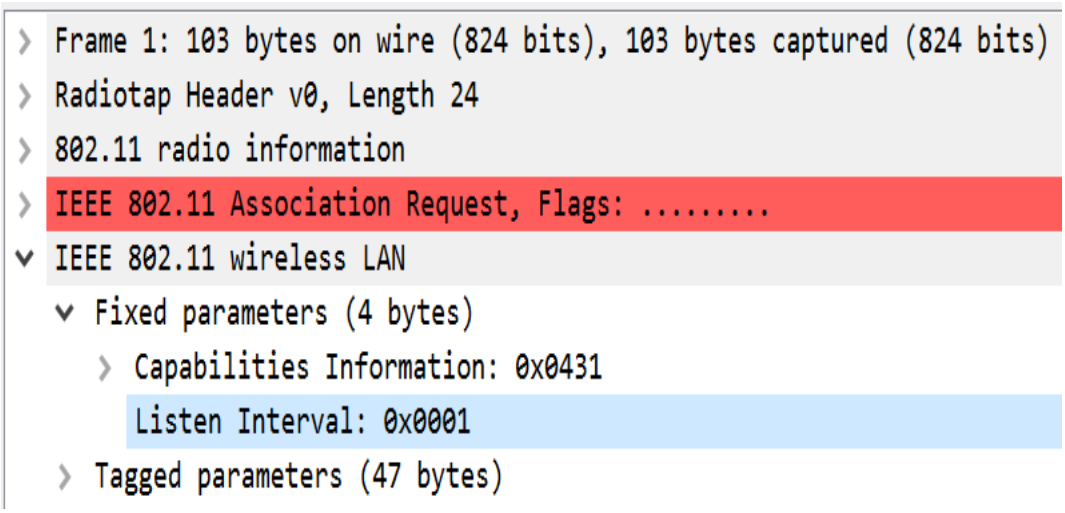

Figure 1.21 "Listen Interval" under the Association Request frame.

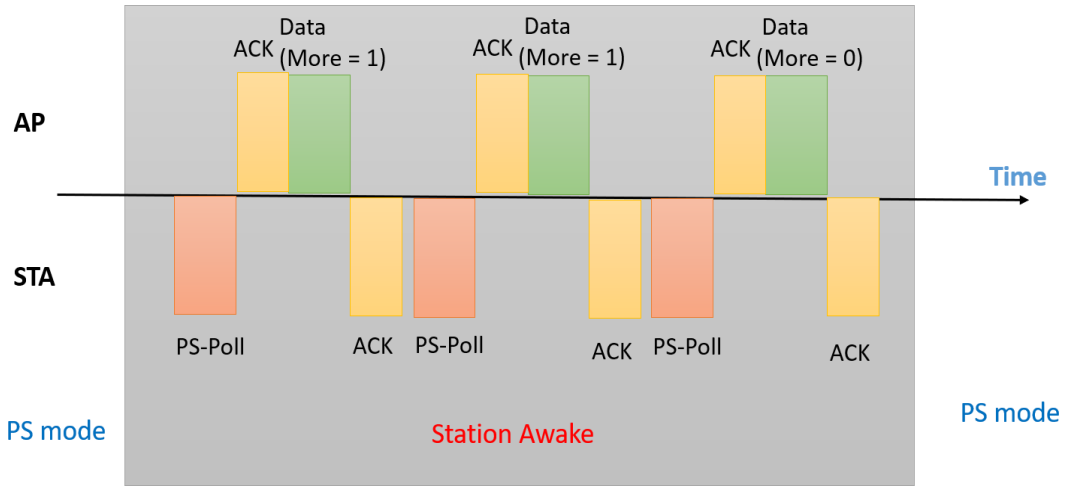

Figure 1.22 Legacy Power Management

An AP may use the Listen Interval information to determine the lifetime of frames it buffers for an STA. But there is no way to change LI during the connection. STA needs to disconnect if it wants to connect with AP with another LI value. AP also defines MAX_LISTEN_INTERVAL, the maximum value of listen-interval that AP supports. The device becomes inactive (in power save mode) before accessing the regular frames from Access Point. The station sleeps for that listen interval, and Access Point accumulates the packets until STA does not wake. The energy also gets wasted when STA awakes in every Delivery Traffic Indication Message (DTIM) Beacon, even when no packet is received. A long LI allows STA to sleep for a long time, which causes more memory on the AP side. This study tries to optimize the AP buffer to overcome this problem and change LI on runtime according to the requirement. In This thesis, we propose a novel approach to managing the power-saving scheme for low-battery station

devices.

Moreover, while looking for power-saving ways in this research, the proposed method will also cover the optimization of AP buffer for STAs operating in power-save mode and keep-alive time management. To connect to an infrastructure WLAN, an STA must initiate the association process by sending an Association Request (AR) to the AP of the WLAN. When STA wants to go into power save mode, it sets its PM bit in its MAC header to let AP know it is no more active. In this case, Access Point will buffer all the information for the client device.

802.11e specification introduced Unscheduled Automatic Power-Save Delivery (UAPSD), which replaced legacy PS-Poll frames with trigger frames (Figure 1.23). When STA live in PS mode, it sends a trigger frame to retrieve all frames from AP and again goes into PS mode. 802.11ac specification introduced Very High Throughput (VHT) Transmission Opportunity (TXOP) power save; in this mechanism, when any STA finds another STA with a TXOP, it will put the radio into low power. 802.11ax (Wi-Fi 6) standard introduced Target wake time (TWT), in which AP manage activity in the network. AP manage the overall network to reduce contention between STAs and STA awake time [6]. 802.11ax supported STA negotiating TWT time with TWT capable AP, and according to that, they can go into PS mode (Figure 1.24). TWT mechanism implemented and revealed through simulation via authors in [7] shows that the TWT reduces the sensor's power consumption. The article [8] discusses TWT's efficiency and problem. TWT and other IEEE 802.11 mechanisms do not set any STAs priority basis on power save. No mechanism tells how AP will take decisions based on STA power requirements. The Presented work in this research takes a more practical approach and gives weight to STA power condition under the power saving mechanism.

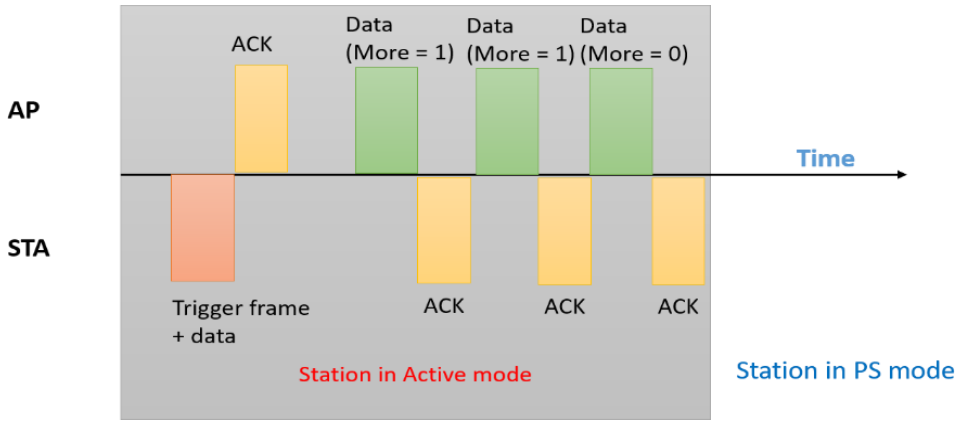

Figure 1.23 802.11e PS Frame Exchange

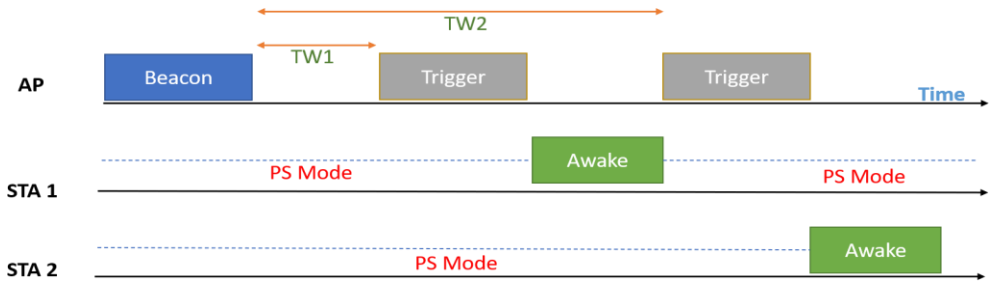

Figure 1.24 802.11ax TWT Power Save Mechanism

1.6 Motivation

The ever-increasing demand for seamless wireless connectivity has driven the proliferation of 802.11 wireless networks, commonly known as Wi-Fi, as the predominant technology for providing internet access in homes, public spaces, and businesses. As the number of connected devices and data-intensive applications continues to surge, optimizing the performance of these wireless networks becomes critical to meet the burgeoning user expectations for reliable, high-speed, and low-latency communications.

Despite the widespread adoption of 802.11 technology, several inherent challenges persist in the current MAC (Media Access Control) layer that hinder the network's overall efficiency, particularly concerning connection time, congestion control, and power save mechanisms. Addressing these challenges has the potential to significantly enhance the user experience and network performance, making it a compelling and pertinent area for research and improvement.

Connection time is the time it takes for a wireless device to connect to a network. The current 802.11 standard specifies a minimum connection time of 100 ms. However, this can be too long for some applications, such as real-time gaming or video streaming.

Congestion control is the process of managing network traffic to prevent congestion. Congestion can occur when too many devices are trying to transmit data at the same time. This can lead to dropped packets, increased latency, and poor performance.

Power save mechanism is the process of reducing the power consumption of a wireless device while it is not actively transmitting or receiving data. This is important for devices with limited battery life, such as laptops, smartphones, and tablets.

The current 802.11 standard does not do a good job of optimizing these three aspects.

This can lead to poor performance and user experience. The proposed thesis will investigate how changes to the MAC layer can be used to improve connection time, congestion control, and power save mechanism in 802.11 wireless devices.

Specifically, the thesis will explore the following research questions:

- How can the connection time of 802.11 wireless devices be improved?
- How can congestion control be improved in 802.11 wireless networks?
- How can power save mechanism be improved in 802.11 wireless devices?

The thesis will use a combination of theoretical analysis, simulation, and experimentation to investigate these research questions. The results of the thesis will be used to develop new protocols and algorithms that can improve the performance of 802.11 wireless devices.

The proposed thesis has the potential to make significant contributions to the field of wireless networking. The results of the thesis could be used to develop new standards for 802.11 or to improve the performance of existing 802.11 devices. The thesis could also have a positive impact on the user experience of wireless devices.

1.7 Research Objectives

All Sub-problem defined above are of different natures, and a step-by-step procedure is used solve them. First, all problems are simulated into the virtual or real environment.A test-bed has constructed to investigate connection problems, power-save and congestion with an experimental study. Then as per the test suit, a set of test steps are defined on the basis of which testing is performed on test-bed. To ensure accurate and valid measurement results, each test case is run multiple times, and the average is calculated. On this basis, a series of observations are concluded. Every problem can becast as a framework/protocol limitation, for example, aiming to maximize station battery usage and AP memory under Wi-Fi use cases. Moreover, we intend to proposea solution without requiring the standard IEEE 802.11 end devices hardware modifications. We suggest software changes on both sides (STA and AP) of the Wi-Fi device. The objectives of the entire study have been classified into four segments:

- The first objective of the study is to enhance 802.11 power consumption

improvement. The study aims to enhance the power consumption of 802.11 by using various low power techniques such as Bluetooth/BLE. The objectives include achieving 802.11 Wi-Fi power save, improving scan power by offloading it to a secondary device, implementing a Tx power feedback mechanism, maintaining a link during power save mode using low power technology, avoiding unnecessary STA awake in every DTIM beacon, offloading Wake on Wireless LAN to low power technology, and implementing an adaptive Listen Interval mechanism based on the battery status of the station device. The implementation of these techniques will help to improve the power efficiency of 802.11 and reduce unnecessary power consumption.

- The second objective focuses on various techniques aimed at improving the connection time of Wi-Fi. The techniques include optimizing scan time through offloading to a secondary device, improving roaming, reducing DHCP time using a cross-layer approach, and making connections faster using low power technology. These techniques can help reduce the time taken to establish a Wi-Fi connection, enabling devices to connect faster and more efficiently. By implementing these techniques, users can enjoy a better user experience when connecting to 802.11 Wi-Fi networks, leading to increased productivity and satisfaction.
- The third objective outlines various congestion control mechanisms that can be implemented in 802.11 Wi-Fi networks. These mechanisms include exchanging information between access points using vendor IE, implementing a Wi-Fi packet power feedback mechanism, selecting channels automatically using a scan mechanism, optimizing contention window in dense environments, and maintaining connections in highly congested environments using low power technology. These techniques aim to reduce network congestion, improve network efficiency and reliability, and maintain connectivity even in challenging environments. By implementing these congestion control mechanisms, 802.11 Wi-Fi networks can provide a better user experience, enabling users to access the network efficiently and reliably, even in highly congested areas.
- The fourth objective is to discuss one practical application and intermittent challenge faced under the research work.

1.8 Outline of the Thesis

Chapter 1 provides an overview of the topic being studied in the thesis. This chapter provides a technical synopsis of 802.11 WLANs, including IEEE 802.11 working architecture, connection procedure and power management specification. In this chapter, a brief introduction to existing Wi-Fi technology and the expected behaviour is described. This introduction sets the stage for understanding the research and problem addressed in this thesis.

Chapter 2 then reviews some of the existing approaches in the performance evaluation of WLANs and discusses several open problems. This chapter covers reviews relevant work and presents research objectives. Literature related to wireless device problems like power saving, connection, and congestion is discussed in the chapter. During research methodology, not only academic researchers' literature papers but IEEE specifications and technical reports are also covered. It is observed that wireless problems are quite familiar to academic researchers. Still, more open-literature data is needed regarding some intermittent practical issues like Wi-Fi device Tx/Rx calibration and firmware updates for the wireless device. Manufacturers of wireless instruments provide technical information in application reports, process-related documents, and technical specifications of different wireless instruments. Therefore, these reports also have been reviewed during the study.

Chapter 3 focuses on the examination of Wi-Fi power savings through an experimental study. The test-bed is used to gather and analyze measurement results, and a WLAN power consumption visualization tool is discussed regarding design, implementation, and testing. Additionally, a new algorithm aimed at improving battery-operated device performance is introduced and implemented within a practical framework.

Chapter 4 discusses 802.11 connection improvement solutions. We start with a scan, which can offload to low-power radio. Later jump to authentication & association steps which can merge with the DHCP layer to reduce 802.11 connection time. Simulation and mathematical time calculation are shown to prove the method's implementation and importance. Later also showed how low-power technology like Bluetooth can help in fast connection and maintain connection in a highly congested environment.

Chapter 5 discusses another subproblem related to congestion control via "improve Collision in Highly Dense Wi-Fi Environment". The simulation demonstrates how the size of the Contention Window impacts the overall Wi-Fi network throughput. It is

shown how optimized CW value can work well in a highly dense environment. And Access point self is not capable enough to do that without understanding the whole environment. So, AP must also talk to each other as they share a common environment. It is described how an AP shared information about the connected station with other AP in OBSS so a collision can reduce in a highly-dense environment. Congestion control using low-power technology and the Tx Power feedback mechanism is also discussed in this chapter.

Chapter 6 discusses intermittent and tricky issues during the problem simulation and their solutions. During work, firmware update problems identify and solved. In this problem, a framework implemented for a software update of embedded devices even on low connectivity for FOTA via on-premise firmware binary creation instead of downloading it from the cloud. The proposed framework is evaluated based on different security aspects and tools. Another problem comes with the exponential growth of Wi-Fi devices, so manufacturing units and assembly lines areheavily overloaded. The most critical feature in the manufacturing line is the calibrationand testing of the Wi-Fi module to provide desired performance. This proposal outlines a method to reduce the time spent on Wi-Fi devices at the assembly line. This can have positive social implications. Assembly-line workers mostly have a high rate of depression and work pressure due to long-time job characteristics. The proposed approach can practically reduce timing at the assembly line, and hence it will boost production and save person-hours for the employee, which can be used for their socialwell-behaviour.

Chapter 7 summarizes the findings of the thesis and outlines potential future work. In the submitted work, some methods are implemented for the first time for 802.11 power save, congestion control and connection improvement. Moreover, the result achieved the practical workability of 802.11 use cases. With protocol implementation, all the corner cases are covered. The simulator and hardware used to develop the methods and changes are software-based; no hardware change is necessary, making it easy to implement. Backward compatibility also achieves during implementation methods, so changes will not break any existing 802.11 mechanisms.

CHAPTER 2
LITERATURE REVIEW

Several researchers work in the Wi-Fi domain, and they are multi-dimensional. In this research, several factors were taken into account when determining which papers to include in the analysis. The main focus was on the issue of Wi-Fi performance. The method of research applied in this study is discussed in this section. When performing a literature review on 802.11 performance, it is important to conduct a thorough literature review in order to establish a solid foundation for your research. In order to do so, we considered the following sources:

- 802.11 Standards from IEEE: Begin by reading the official 802.11 standards, including 802.11a, 802.11b, 802.11g, 802.11n, 802.11ac and 802.11ax, to gain an in-depth understanding of the technical specifications and features of the technology.
- Academic Research Papers and Journal Articles: There is a significant amount of research available on 802.11, including studies on its performance, security, and applications.
- Technical Reports and Whitepapers: Industry organizations, such as the Wi-Fi Alliance, may have technical reports and whitepapers on 802.11 that offer insights into the technology and its implementation.
- Books and Textbooks: Books and textbooks on 802.11 can provide a comprehensive overview of the technology and its applications.

It's essential to critically evaluate the sources used and ensure they are relevant and up-to-date, as the field of wireless networking and 802.11 is rapidly evolving.

In this chapter, a thoroughly extensive literature survey in the field of Wi-Fi systems has been performed. At a broad level, fast operation, power-save and reduce congestion/collision are a great challenge in a Wi-Fi environment, and researchers are working to overcome these challenges. Wi-Fi users are increasing day by day and analyst are working on their behaviour and problems [9]. Section 2.1 discusses Wi-Fi

connection related research work, from scanning, authentication to the DHCP process. Section 2.2 enlists different studies working in different power saving schemes for Wi-Fi devices. Section 2.3 gives an overview of the congestion control and collision avoidance techniques employed in 802.11 systems. Section 2.4 enlists some generic solution discussion and various metrics used in several research studies and Section 2.5 discusses research gaps & limitation. The chapter concludes with the summary in Section 2.6.

2.1 Wi-Fi connection problem discussion

Literature review also covered in Wi-Fi connection sequence. Start from scanning, authentication/association to DHCP process.

Multi-dimensional ways used by researchers to solve Wi-Fi scan problems. In the IoT world, devices become very time savvy; it means IoT devices are designed to be as efficient as possible in their use of time. Even a minor save of timeis a significant achievement. The ability to save time is particularly important for IoT devices because they are often used in applications where latency is critical. For example, in a smart home application, it is important for the IoT devices to be able to respond to commands quickly, so that the user does not have to wait for the device to take action. An 802.11 connection process always starts from the scanning. Castignani et al. [10] provide a comprehensive survey of wireless network scanning techniques in various wireless technologies, including 802.11 WLANs. The authors categorize the scanning techniques into various categories, such as active, passive, and timer-based, and evaluate their performance regarding scan overhead, accuracy, and efficiency. Rishabh et al. [11] suggested a unique access point in a common area which can provide a scan list to the station. The station needs to send a vendor-specific probe request to AP, and in response, the station receives the scan list. By avoiding unnecessary scanning across all channels, the proposed device would provide faster and more reliable connections. The problem is with the approach is station and AP have very different power capabilities; what AP can scan the station will not able to communicate, so the following approach is not suitable in a realistic environment. In another work, Wang [12] used caching of previous connection information where using a guarded action system (GAS) information element scan and connection-related information shared on the particular network element, and all access points talk with that element before creating a new connection. If a piece of old connection information is found on the network element, the same is used via AP to create the latest connection. Unfortunately, the

implementation and deployment part is missing from it, which makes this idea far from originality.The 802.11 scan is a process used by wireless devices to discover and connect to available wireless networks. Sometimes, finding an AP and maintaining an existing connection seems a very tough task, especially in a dense environment [13,14] like shopping malls, railway stations, or a university where so many users work on the samefrequency at the same time. Fast scanning [15] and selective active scanning [16] are discussed to improve Wi-Fi scanning. Unfortunately provided solution also does not fitunder the real environment, and practicality is not addressed in the paper. Active scanning is also used to reduce connection time as shown in several research reports [17]. Partial scanning (perform channel scanning in chunks, not in one go) and special scanning algorithms [18, 19] also works well in the fast handoff between the station and access point.

Literature review shows that some of the problems associated with 802.11 scan include:

1. Long scan time: The scanning process can take a long time to complete, causing delays in network connections and affecting user experience.

2. Battery Drain: The scanning process can consume a lot of power, which can result in rapid battery drain for mobile devices.

3. Interference: The 802.11 scan can interfere with other Wi-Fi networks, leading to network congestion and reduced performance.

4. Security: The 802.11 scan can reveal information about available networks and their security configurations, potentially exposing sensitive information to attackers.

5. Complexity: The 802.11 scan process is complex, requiring a deep understanding of wireless technology and standards, which can be challenging for many users.

In conclusion, the 802.11 scan process is critical for connecting to wireless networks, but it can also cause a variety of problems that can negatively impact user experience, security, and performance.

The next steps in connection are the authentication and association process. The 802.11 authentication and association process can be prone to several problems. Some of these include security vulnerabilities, such as a lack of encryption or the use of weak encryption methods. Another important issue is that the process can be slow, leading to delays in the establishment of network connections. Additionally, the process may be

prone to failures, such as an inability to associate with an access point or failure to complete the authentication process. These problems can result in decreased network performance and user dissatisfaction.

Syahputri and Sriyanto, [20] proposes a fast authentication algorithm to make a quick connection,especially in roaming scenarios. The proposed method allows users to do advanced authentication before moving to other APs. Here the radius server plays a vital role in transferring information to other apps. This approach is very similar to 802.11r [21] fastroaming. In the case of roaming, the authentication frame uses to share 4-way handshake information for a fast connection.

All the above protocols can have some difficulties that need to be considered before implementation. These include:

- Complex setup: Implementing 802.11r or mentioned methods may require substantial modifications to the existing Wi-Fi infrastructure, making it challenging to set up.
- Security risks: If not properly secured, 802.11r can open up the network to security threats like man-in-the-middle attacks.
- Interoperability problems: Not all Wi-Fi devices and infrastructure support 802.11r, which can cause compatibility issues. Interoperability is not addressed in any solution.
- Increased latency: The authentication process can add extra latency that can impact the user experience. For example, Fast roaming can cause a temporary reduction in bandwidth due to the time required to complete the roaming process.

Final step is in Wi-Fi connection is to get IP address from DHCP server. Cross-layer approach is also a working area where one layer of work can offload to another network layer. DHCP offload to the lower layer is a way to improve connection time from seconds to ms. Pre-allocation of DHCP is also performed [22] to improve Wi-Fi energy management via traffic isolation improve connection time. In this article authorfocuses on improving the handover process between 802.11 access points to enable seamless and quick mobility for users with permanent Internet access. The authors propose a new methodology for implementing pre-allocations of DHCP leases, which

helps reduce the delay of DHC P-based IP reconfigurations after a link-layer handover. The pre-allocations are triggered by the normal 802.11 neighborhood scanning performed by the mobile node and are exchanged in link-layer frames. The authors have implemented a prototype and have reduced the DHCP-based IP reconfiguration delay to 18% of the current delay. This work enables mobile nodes to roam frequently with low handover delays, allowing them to explore aggressive roaming policies and find the best QoS. In the way described in the paper, the first time a station joins the network, it needs to send some extra packets, which is overhead and increases latency in the network. With probe request,DHCP allocaion is done via AP with the DHCP server. But it's a waste of resources as99% APs are just used for scan purposes via stations. The available solution is good forroaming purposes, but congestion cases do not consider under implementation.

A novel L3 handoff approach has been introduced by Zúquete and C. Frade [23], where subnet changes are detected by sending a bogus DHCP REQUEST, which causes the DHCP server to send a DHCP NAK. A temporary IP address is selected from ARP requests, and the L3 handoff takes about 190 ms. When the Mobile Node (MN) has visited the subnet before, the SIP session can be updated, and the handoff can be seamless with a delay of about
30 ms. The new approach does not require any infrastructure changes, only modifications to the mobile node, but introduces trade-offs between total handoff delay and duplicate address probability.

2.2 Wi-Fi power saving schemes

There have been many articles welcoming/appreciating the new methods to improve the 802.11 Power Saving Mode (PSM) standard, but power is something which always has scope for improvement. Power consumption is also a challenging area, especially for battery-operated device [24], and offloading of features are performed to overcome this challenge in a small manner. Wi-Fi energy management via traffic isolation [25] is a challenging subject. In the IoT world, devices have become verypower savvy; even a minor loss of power is a significant loss for them. Also, features like FOTA need a high throughput interface, but regular work needs less power consumption. The Wi-Fi performance is much better when the transmit path is shut down [26] and pays attention in the absence of unreserved frames.Energy consumption at the receiving end is one-third less than at the transmitting end. Although traffic sample changes hugely, this will

not be highly efficient for saving energy. Several previous works have investigated techniques to solve the powerconsumption using multiple radios [27] for battery-powered devices. Some research used Wi-Fi with low-power technology, whether BLU-FI or ZigBee attached power save [28]. Receiver design [29, 30] or context-sensitive framework development [31] is also a way to save energy.

Power consumption is also challenging, especially for the battery-operated device [32]. The offloading of features is performed to overcome this challenge in a small manner. Here researcher introduced the Tail Ender scheduling algorithm. More ideas are presented in each condition and real-life scenario for saving more energy. Tail Ender will work specifically on social feeding, e-mail, and net surfing. Using some information, the author found the efficiency of Tail Ender with a flag scheme. As per the survey, the performance of downloading in mobile phones is 60% more than social searching, and in case of net doubts, it was found that more than 50% of downloading outputs. This method only covers active work cases, but idle time power saving is not considered, which is also necessary for battery-operated devices.

In WSNs, the connections between sensor nodes are closely related to their wireless transmission power [33]. Lopez et al [34] revealed power-save use cases and applications in different practicalities. IEEE Wi-Fi standardstry to improve the Wi-Fi power-saving mechanism. With standards, hardware and software are continuously evolving to save power. Furthermore, the 802.11ax standardintroduces a Target Wake Time (TWT) mechanism to reduce power consumption, and later, specific to IoT devices' power saving, IEEE introduced the 802.11ba standard. The Wake-Up Radio (WUR) mode of 802.11ba [35] is a low-power operating mode that only activates the Rx chain and requires special hardware that remains active evenin a dormant state. Both 802.11ba [36] and 802.11ax [37] have limitations in power savings due to the clock drift effect, which negatively affects performance in low-trafficscenarios. Sampleless Wi-Fi [38] introduces new constellation diversity through modifications made to the receiver portion of the Radio Frequency (RF) chain. Hardware changes are costlier, and it is tough to change old devices with the latest hardware.

Omori et al. [39] also uses the RTS/CTS frame with the existing power save mode. Authors claim that STAs can also sleep during the Network Allocation Vector (NAV)

period. However, the presented approach is only applicable for low data traffic and RTS/CTS frame addition, and an extra overhead of frames in the network, which a part of is not considered under the paper when authors conduct performance evaluations with existing methods. Saeed and Kolberg presented a context-aware listen interval in a more software-driven way, using a machine-learning approach [40]. The paper shows the experimental result, revealing that the presented way can save 75% energy compared to the 802.11 standard power-saving mechanisms. However, the presented way is more theoretical in comparison to practical implementation. There is no protocol implementation discussed in the paper.

Tae et al. [41] propose an intelligent reinforcement learning-based power-saving mechanism. The discussed machine learning algorithm dynamically changes the listen intervals of stations, which helps optimize STA's energy consumption. This paper efficiently manages the trade-off between energy consumption and transmission delay. Unfortunately, the article missed the 802.11 implementation phase, AP side changes, and input data consideration. Liu et al. proposed an energy- conserving model [42] for Wi-Fi STAs competing for power save mode and connectedwith the same access point. A Harmonious Power Saving Mechanism (HPSM) handles PSM traffic contention problems, and STA prioritizes according to link resourceconsumption. According to the model, higher priority STA has more chance to deal with AP, and STA does not waste energy to obtain the environment under Carrier SenseMultiple Access/Collision Avoidance (CSMA/CA). Again, the presented model does not cover practical details, and QoS and protocol part are not considered under the implementation phase.

Wu et al. [43] propose a Named Data Networking (NDN)-enabled PSM model forWi-Fi devices, where STA can predict data arrival precisely and go into light or deep sleep mode accordingly. This method is based on a cost function that considers priorityand packet latency and can reduce idle listening time. Simulations show methods effective in improving average power consumption and average transmission delay. Two power modes are discussed here, light and deep sleep modes. However, the paperdoes not reveal much difference between them and which RF chain part is close underwhich mode. One drawback of the method is that STA can only predict unicast packets. Moreover, the implementation phase does not consider broadcast/multicast packets, which are obvious packets under the Wi-Fi domain. Some research used low-power wireless

technology with Wi-Fi, whether BLU-FI or ZigBee attached power save [44]. However, the range of Wi-Fi is different from low-power technology, and these technologies cannot send data at a higher rate. Receiver design [45,46] or context-sensitive framework development [47] is also a way to save energy. Power consumption is also challenging,especially for battery-operated devices [48]. The offloading of features is performed toovercome this challenge in a small manner. Gupta et al. [49] presented two algorithms specially designed for wireless ad hoc networks. One is the Select_Node algorithm to determine how many nodes want to participate in data transmission based on the batterystatus. The second one is the Total_Minimum_Power algorithm, which calculates the total power consumption in a wireless adhoc network and finds the minimum energy required to transmit data from one node to another. These two algorithms help to minimize the consumption of considerable battery power and increase the overall lifetime of the network. Unfortunately, the presented method only applies to the ad-hocmode, not to the infra mode, which is widely used in the Wi-Fi world.

Fonseca et al. [50] presented a resource management framework for 802.11 wireless AP. The results show a significant performance improvement in throughput and an improved working model under interference. The given model does not consider the power save of STAs and memory management for access point memory. The motivation for our work is to introduce a new adaptive power-save feature and a more practical solution to manage fairness in an IoT battery-operated device environment.

2.3 Wi-Fi congestion control and collision avoidance

The relationship between 802.11 congestion control and collision avoidance is an important aspect of the performance of a WLAN based on the 802.11 standard. The MAC protocol, which controls access to the shared wireless channel, plays a crucial role in ensuring efficient networkperformance. Congestion control mechanisms, such as rate adaptation and retransmission, help to regulate network traffic and prevent overloading the channel, while collision avoidance mechanisms, such as carrier sensing and random backoff, prevent multiple devices from transmitting at the same time and colliding on the channel. Both congestion control and collision avoidance mechanisms are closely related and interdependent, as the effectiveness of one mechanism can impact the performance of the other. To achieve optimal network performance, it is important to carefully balance and coordinate the behavior of these mechanisms.

Several paper talks performance of different data transmission to support the QoS (define in IEEE 802.11e MAC protocol [51]) in WLANs and their results show that the how different QoS parameter can improve the throughput & quality of service transmission in the Wi-Fi network [52,53,54,55]. Several new techniques and algorithm proposed to minimize collision rate in the noisy environment, whether it's a new backoff algorithm [56] or accurate delay distribution of DCF functionality [57]. The researcher also proposed new QoS architecture (using 802.11e EDCA) in which adaptive CW used to increase performance in the noisy environment. All the above method consider exiting Wi-Fi protocol & do amendments in it. Inspired by the above, motivation was drawn to propose an addition of Wi-Fi frame which easies to deploy in the vendor's network and increase throughput in the high noise environment.

Yoshiwaka et al. [58] investigated the wake-up success probability based on-demand Wi-Fi wake-up. A high congestion channel is chosen to perform the test. The study demonstrates that a decrease in the number of interfering nodes leads to an improvement in the probability of successful wake-up, whereas an increase in the number of interfering nodes may lead to a decrease in the probability of successful wake-up due to the delay in transmitting wake-up frames.

Deng et al. [59] focuses on improving the performance of the backoff parameters in the IEEE 802.11 standard's collision avoidance mechanism, which results in low channel utilization and high collision probability in congested and error-prone WLAN environments. To tackle this issue, the authors propose a distributed algorithm that allows each station to adjust its contention window size based on the turn-around-time measurement of the channel status. The proposed solution has been evaluated through simulations and has shown to provide significant performance improvement in noisy and unreliable wireless networks. The implementation of this scheme requires major modifications to the IEEE 802.11 networks.

Mishra et al. [60] discuss study of the methods used to handle congestion in 802.11 wirelessnetworks. It evaluates the advancements made at various protocol layers to address theissue and assesses the significance of cross-layer interaction. The study finds that avoidance approaches are more commonly used than mitigation approaches, and that MAC layer improvements have an advantage over transport layer improvements. The parameters used to handle congestion are also summarized, as well as the challenges faced at each protocol layer. The study provides a comprehensive overview

of the techniques used to handle congestion in 802.11 networks and is intended to be helpfulfor researchers in this field.

S. K. Memon et al.[61] provide a review of the QoS guarantee in IEEE 802.11-based Wireless Local Area Networks (WLANs) and focuses on supporting emergency traffic in such networks. The study reveals that the existing IEEE 802.11e EDCA standard is limited and cannot effectively provide a strict QoS guarantee or support emergency traffic in medium to high traffic conditions. The paper concludes that more research is needed in the area ofproviding a strict QoS guarantee and supporting emergency traffic in WLANs. The future work aims to modify the WLAN MAC protocol and reduce the transmission overhead to accommodate a wider variety of emergency nodes over the network.

2.4 Generic solution

Metrics like load balancing, QoS, Throughput, Response time & power consumption are considered by researchers to conduct their study. Researchers use a software-driven approach for wireless solutions [62]. Jose et al. Defined a software-based framework for wireless devices [63]. The proposed solutioncan work on access points, and AP can provide innovative functionalities such as loadbalancing and QoS facility. The problem with implementation is that no real access points software has been taken to reflect the changes discussed in the paper, and backward compatibility is not considered at all.

Microsoft researchers developed a functionality called Blu-Fi [64], which combines the features of Wi-Fi and Bluetooth. Some contact patterns of Bluetooth are used in their device to check whether Wi-Fi is available or not. Bluetoothdevices use a shallow range for an accurate forecast. Blue-Fi reduces the process of learning time for the attributes of linked Wi-Fi.

An intelligent solution Wi-Fox [65] proposed helps reduce STAs traffic asymmetry, increases performance loss due to rate-diversity/fairness, and avoids degradation due to TCP behaviour. It increases Rx throughput by 400-700 %, reducing average response time by 30-40 %. Fast connection and power optimization are challenges in a Wi-Fi environment at a broad level, and researchers are working to overcome these

challenges. A small radio Wi-Fi external device is used [66] to wake up theprimary device, but Wi-Fi still takes more power than Bluetooth. Bluetooth works on an adaptive frequency-hopping concept, so it mitigates the risk of collisions by using spread spectrum techniques.

2.5 Research Gaps and Limitations

According to the studied literature, there are some research gaps that has left behind while using Wi-Fi in current environment like:

- congestion control,
- collision avoidance,
- power save,
- fast Wi-Fi connection,
- power optimization,
- bandwidth utilization,
- security, capacity, data rate and many more.

Some of these above research gaps are the objectives of the PhD thesis and are written below:

- Until now, there is no way present so Access point can share information about own contention window with each other in overlap network, on the basis of which further research will be progressed.
- There is no discussion on the cross-layer approach of application layer, network layer and data link layer in 802.11 architecture. Along with cross layer approach offloading of driver layer work on firmware layer is still undiscovered.
- There is no work that has been done on optimize 802.11 power in case of very small packets. RF chain of 802.11 does works exactly same for every packet, whether it’s big packet or small packet.
- There is no work that has been done on to minimize the total power consumption in bandwidth higher than 20MHz, even some control and management packets only intended to central channel (i.e. 20MHz bandwidth only).

- There is no work that has been done on to manage the total power consumption while reducing the complexity and satisfying the battery constraints of the station device.
- Along with this, the effect of this total power consumption with reduced complexity approach on the different deployment scenarios in the Wi-Fi connection is still undiscovered.
- There is no practical use-cases discussed where low power technology can help to improve Wi-Fi performance. Hence our solution solved these problems in a more well manner. Piyare et al. survey [67] shows different low power wake-up radio, hardware, and networking topologies cover under the paper. But as described, all wake-up radio uses Wi-Fi, and it can improve power a little bit. These all methods can help with Wi-Fi wake-up but can't avoid unnecessary wake-up. Forexample, our proposed method can reduce the unnecessary wake-up due to theDTIM miss case.

2.6 Summary

Chapter 2 of the thesis is a methodical literature review of the research done in the field of Wi-Fi systems. The chapter focuses on the issue of Wi-Fi performance and covers three main attributes including power-save, connection requirement, and congestion avoidance. The research methodology used in the study is discussed in detail. The chapter provides an extensive overview of the Wi-Fi connection related research work, power saving schemes for Wi-Fi devices, congestion control and collision avoidance techniques, metrics used in research studies and various interesting problems. It is observed that the maximum proposed mechanism do not have practical deployment and may not always be computationally efficient for real-time implementation in 802.11 WLANs. The solutions proposed in the existing papers are specific to certain Wi-Fi network configurations and may not be applicable to all Wi-Fi networks. Manysolutions require significant changes to the existing Wi-Fi network architecture and may not be feasible to deployment. Hence needs of practical solutions which are easy to adopt and have backward combability really required. The proposed work in thesis covered the point with a focus on Wi-Fi protocols and every corner case covered. In

next consecutive chapters proposed solution will be presented to overcome all discussed challenges.

CHAPTER 3
POWER CONSUMPTION IMPROVEMENT IN 802.11 WLANS

Wi-Fi researcers are trying hard to extend battery life by optimizing 802.11 power save. The rising number of Wi-Fi and IoT devices demands to reduced Station (STA) device power consumption. IEEE 802.11 Power Save is important because it helps in conserving energy and prolonging battery life in battery-powered devices such as laptops, smartphones, and tablets. When a device is in power save mode, it periodically wakes up to check for new data, but remains in a low-power state for most of the time. This reduces the device's power consumption, making it more energy-efficient.

Additionally, by reducing the amount of time the device spends transmitting and receiving data, 802.11 Power Save helps in reducing network congestion and improving network performance. The 802.11 Power Save mechanism is especially useful in scenarios where there is a large number of battery-powered devices connected to the network, such as in an office building or a conference room.

This chapter takes a more practical approach to solve the above challenges; the presented work aims to reduce embedded Wi-Fi device power consumption and improve access point memory management.

3.1 Introduction

The Listen Interval (LI) plays an important role when the device goes into power-save and how much memory buffer is assigned via AP for the station. However, these parameters are static and do not solve the need of the hour. The LI has been configured during the association phase, and its value remains unchanged until the station does not disconnect. In practice, STA put the LI value as 1, in order to wake up on every beacon, and AP assigned static memory to the connected STA. Xie et al. And Li et al. [68,69] try to change the LI according to the load on AP, unfortunately without prioritizing STA andno practical framework architecture is covered. In our research, more simple and functional designs are preferred to consider IoT device use cases. IoT devices transmit small amounts of data but always strive to maintain a connection to an access point

while consuming minimal power. Using the slightly modifiedframework, we can make it fully compatible with the existing 802.11 standards devices.

This research takes a more practical approach to solve the above challenges; the presented work aims to reduce embedded Wi-Fi device power consumption and with plus it improves access point memory management. Following techniques is used to solve the power-consumption issue:

1. Low power technologies can be used in Wi-Fi power save mode to reduce the power consumption of Wi-Fi devices and extend their battery life. Here are some ways to implement low power technologies in Wi-Fi power save, like Timed wake-up, Wake-on-Wireless-LAN (WoWLAN), connection maintenance. Low power technology like Bluetooth/BLE, Zigbee, Z-Wave, NFC etc is used in devices to communicate with each other wirelessly over short distances. Architecture and other details will elaborate in section 3.2.

2. Adaptive listen-interval: An adaptive listen-interval-based buffer management scheme for the Wi-Fi device has been proposed. In the proposed architecture, dynamic LI is used based on the STA device's battery status and on the basis of the de-fined policy under architecture. Now on the base of dynamic LI, AP allocates buffers for the STA device. For this purpose, in the section, we have introduced 802.11 protocols and policies to accept LI adaption requests. The keep-alive factor is considered under implementation. In IEEE 802.11, the keep-alive factor is a value that determines how often a station in power-save mode (PS mode) sends a NULL data frame to the AP. Proposed LI change, protocol implementation, and activity updates are software-driven, so updates at the existing STA and AP framework are covered under the proposed architecture. We conducted an extensive performance evaluation. The results ofSTA power consumption have been compared with standard 802.11 PSM in different situations. AP buffer memory improvement over conventional static AP memory results is also discussed in the performance evaluation. The method is described in more detail in section 3.3.

3.2 802.11 practical improvements using low power technology

Low power technology like Bluetooth can help 802.11 power save by allowing devices to conserve power while they are idle by utilizing a low power mode. In this mode, the

device's radio is turned off, and the device periodically wakes up to check for incoming traffic. When a Bluetooth radio is working to a Wi-Fi radio, it can use Bluetooth to communicate with nearby devices, reducing the need for Wi-Fi to be active all the time, and thus reducing power consumption. When a Bluetooth radio is working together with a Wi-Fi radio, the Bluetooth radio can use Bluetooth to communicate with nearby devices. This means that the Wi-Fi radio does not need to be active all the time, which can reduce power consumption. Additionally, Bluetooth can be used to coordinate power management between multiple Wi-Fi enabled devices, allowing themto enter power save modes at the same time and further reducing overall power consumption. Bluetooth and 802.11 (Wi-Fi) technologies operate in different domain, so they don't directly affect each other's power usage. Although Bluetooth and Wi-Fi are often seen as competing technologies, they can actually be used together to improve Wi-Fi power savings in certain cases. This sub-chapter will discuss how this is possible.

3.2.1 Architecture of the proposed method

This concept assumes that both Wi-Fi and Bluetooth are presented on the same SOC (system on chip) and both radios cover the range of peer device. The SOC is a centralized hub, and most of the computer components can be present on a single base. It is a combined circuit where many different parts or elements can connect. These components consist of memory, input/output ports, and secondary storage. All chip vendors generally put Wi-Fi BT/BLE on the same interface, as shown in Figure 3.1. On the same SOC, both entities (Wi-Fi and Bluetooth/BLE) can interact and share a common memory. The architecture of a SoC with Wi-Fi and Bluetooth radio present on the same chip typically includes the following components:

CPU: The CPU is the central processing unit of the SoC. It is responsible for executing instructions and coordinating the activities of the other components on the chip.

Memory: The SoC typically includes some amount of memory, such as static random-access memory (SRAM) or dynamic random-access memory (DRAM). This memory is used to store data and instructions that are being processed by the CPU.

Wi-Fi radio: The Wi-Fi radio is responsible for transmitting and receiving Wi-Fi signals. It includes a radio frequency (RF) front-end, a digital baseband, and a modem.

Bluetooth radio: The Bluetooth radio is responsible for transmitting and receiving Bluetooth signals. It includes a radio frequency (RF) front-end, a digital baseband, and a modem.

Other peripherals: The SoC may also include other peripherals, such as a Universal Serial Bus (USB) controller, a graphics processing unit (GPU), or an audio codec.

The concept only covers the infrastructure (centralize) mode, but the same architecture can also be applicable for the Ad-hoc mode. In Ad-hoc mode, Announcement Traffic Indication Message (ATIM) will replace the work of DTIM in infrastructure mode. Instead of AP, another Wi-Fi device that has buffered Unicast or Multicast packets, it is announced via ATIM to other Wi-Fi devices.

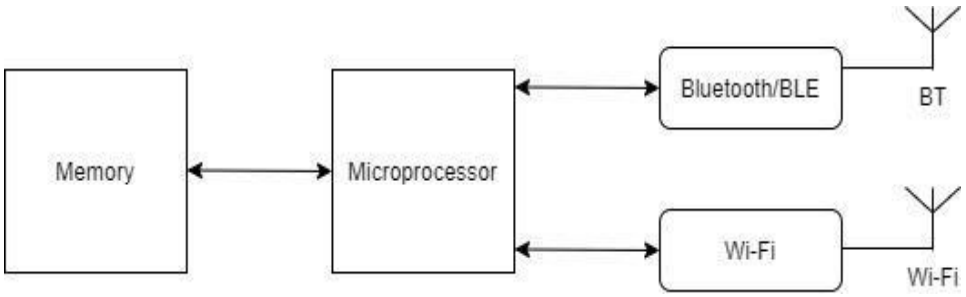

Figure 3.1 Wi-Fi & BT on the same SOC (System on chip)

3.2.1.1 Brief Description

For transferring information between Bluetooth and Wi-Fi, keep Wi-Fi and Bluetooth in the same SOC in IoT devices. Simultaneously make the Wi-Fi and BT connection and exchange the data from any interface. Even BT data exchange can be possible without creating a connection. Chip vendors, like those who manufacture Bluetooth chips or modules, can work on the possibility of exchanging data without a traditional connection setup.

Wi-Fi can go in sleep mode, active mode, or perform the required task, whenever triggered using Bluetooth, as they are connected on a single platform.

Wi-Fi and Bluetooth are involved in both AP and STA on the same SOC, which can help to transmit information between each other (via Software or Hardware). Bluetooth software can update Wi-Fi radio's information and vice versa.

The integration of Wi-Fi and Bluetooth is done to operate with very little energy consumption and decreased cost. The proposed method topology and connection mechanism are shown in Figure 3.2. AP's Bluetooth radio can play the role of master for the STA's Bluetooth (slave) radio. The basic idea of the proposed method is that Wi-Fi data can be delivered using Bluetooth or BLE interface in the following conditions:

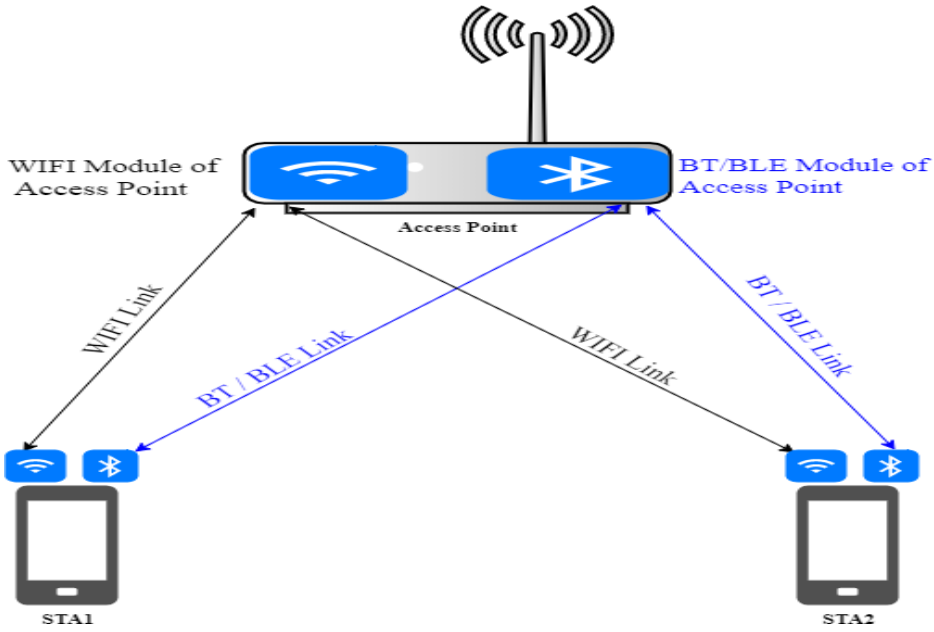

Figure 3.2 Proposed Topology

- The environment is heavy noisy, so the Wi-Fi interface cannot get a chance to transmit its packet.
- The Wi-Fi device is in sleep mode.
- To make a fast connection and maintain it.
- Below two are the Methods that can be used to transmit Wi-Fi packets via Bluetooth radio:
- Wi-Fi packet payload under the Bluetooth packet (Figure 3.3).
- Complete Wi-Fi packets (Header + Payload) sent using Bluetooth radio:
    - Bluetooth physical layer adds its preliminaries.
    - A prefix is attached in the Bluetooth packet, which can tell this packet aims at the Wi-Fi system.

Bluetooth Preamble	Prefix	Wi-Fi Packet

Figure 3.3 Wi-Fi Packet fit into BT Payload

3.2.1.2 Methods used in Mesh Network

All devices are connected with Wi-Fi and Bluetooth interfaces (Figure 3.4). In a mesh

topology, proposed methods can also be implemented.

Mesh network is a term used for many wireless nodes, and all those nodes can communicate with each other in a very vast area. In a wireless mesh network, a single node must connect physically with the internet. Then using that single node, all other wireless nodes can connect in its sector. In this way, those nodes will again share their internet with the nearest nodes. AP position can be dynamic as well; it just needs to maintain Wi-Fi and BT connection with STA's. It can be implementation-dependent. For example, in the EasyMesh [70] network, STA does not care with which Access point it will connect. The proposed architecture easily can fit into Wi-Fi EasyMesh setup.

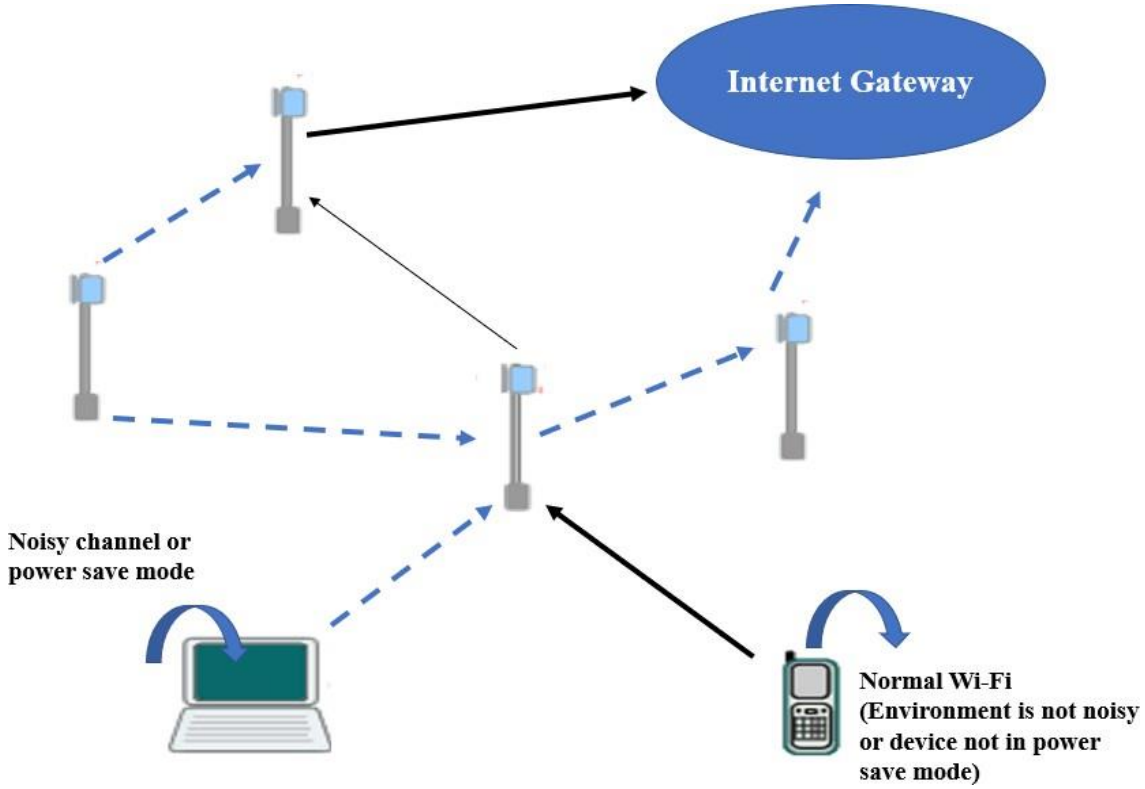

Figure 3.4 Proposed Method Work on Mesh Topology

3.2.2 Station Power Optimization Mechanism

In the present power save algorithms, the STA should awake in every DTIM Beacon to check the packet is present for the STA or not (Figure 3.5). If the packet is not present for the STA, it wastes much energy without any gain.

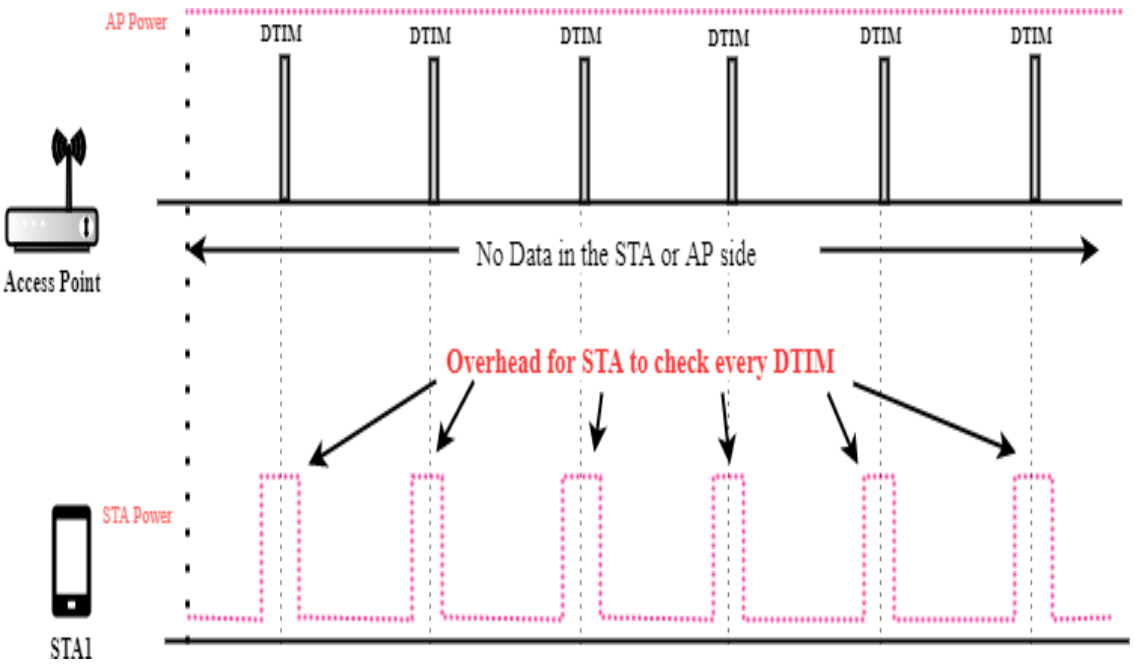

Figure 3.5 Wi-Fi Device Normal Wakeup Pattern

There are so many devices nowadays available which take less power to work.However, Wi-Fi is the device amongst all others, which requires more energy. Bluetooth and ZigBee need less power as compared to Wi-Fi. This technique will follow below mentioned concepts where STA will not be required to wake up on everyDTIM beacon. software-based solutions can be used to enable communication between Wi-Fi and Bluetooth on the same SoC. For example, a software layer can be added on top of the Wi-Fi and Bluetooth drivers to enable communication between the two radios. This can involve translating data between the different protocols used by Wi-Fiand Bluetooth. BT/BLE, a less power-consuming technology, will play a crucial role here. Following things identified using proposed way:

- Now on DTIM Beacon, there is no need for STA to wake up.
- If there are some packets on AP for STA, they will not be directly sent to STA. Instead, it will first trigger a command to the AP BT/BLE interface, waking up the STA.
- BT/BLE interface of AP will now send some selected packets to the STA BT/BLE interface.
- Then, the STA BT/BLE interface will decrypt the code and forward its respective signal to Wi-Fi to wake up the Wi-Fi module. It will save energy

because Wi-Fi will not be required to wake up every time.

- If needed Wi-Fi to wake up, the Wi-Fi segment will transmit a frame with the power save bit reset, and then its current power state can be indicated to AP.

Whenever AP has unicast data for the STA device, it will trigger a BT/BLE interface command to wake up the STA. After this, Wi-Fi STA with Power management bit reset will send a wake-up notification to AP (Figure 3.6).

Whenever there is a broadcast/multicast packets transmission, AP will not wait to wake STAs. It simply transmits a packet after the upcoming DTIM. AP will wake up STA immediately before the DTIM Beacon (Figure 3.7). It will automatically save time, and obviously, it will consume less power.

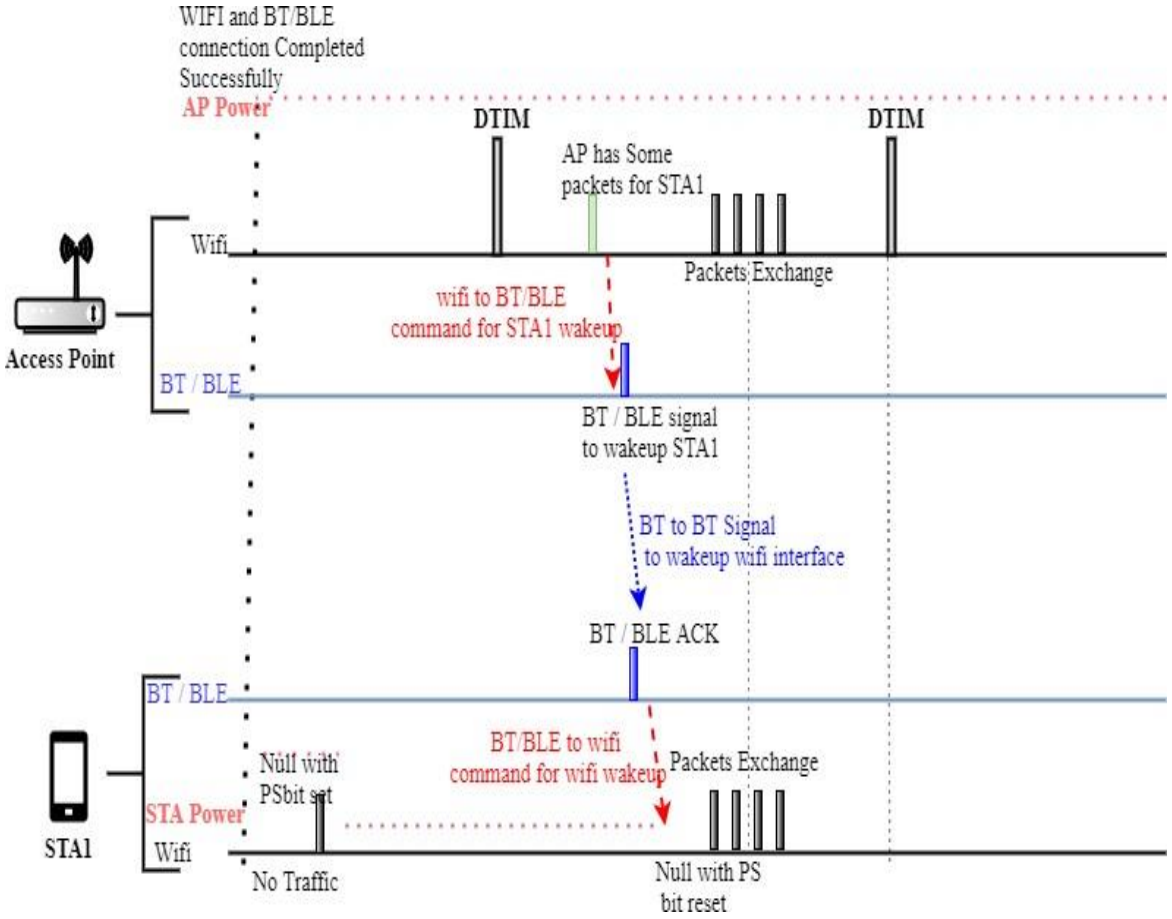

Figure 3.6 Proposed Method in case of Unicast Data Packet

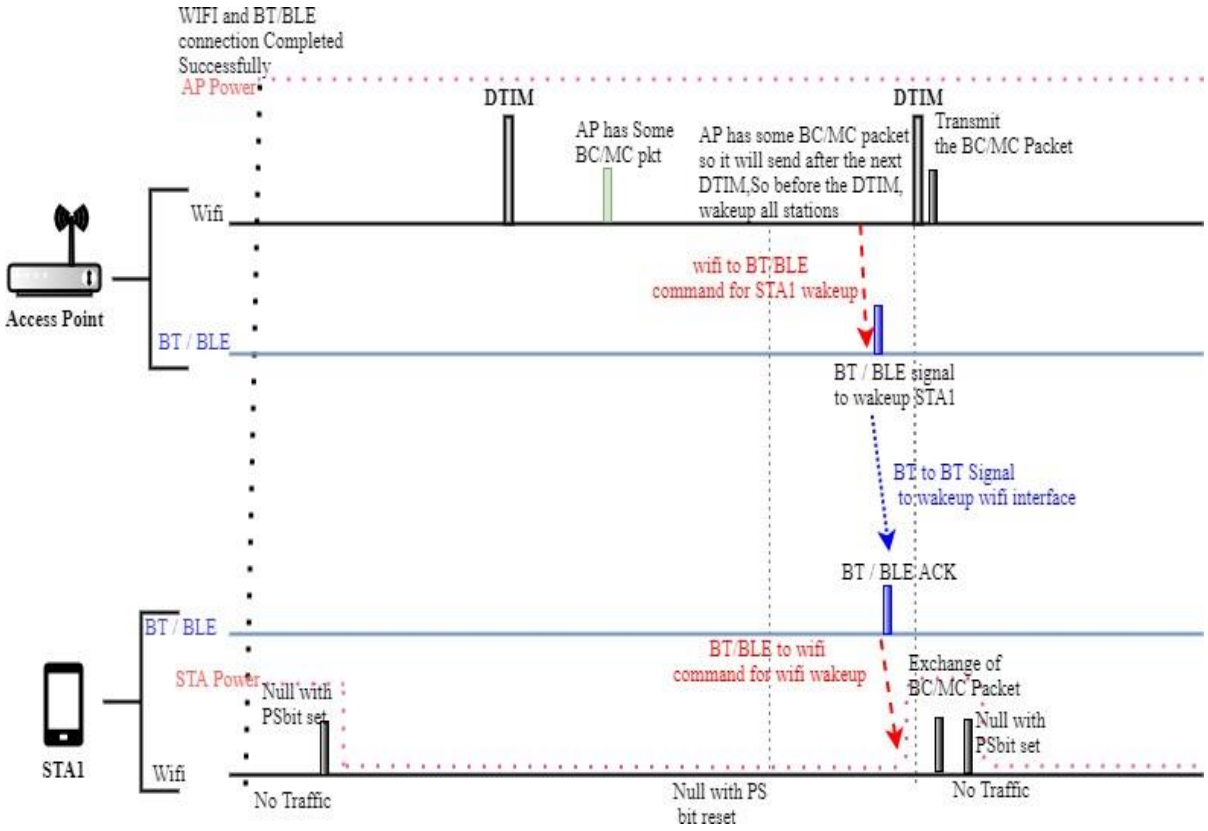

Figure 3.7 Proposed Method in case of Broadcast/Multicast Data Packet

3.2.3 Wake on Wireless LAN improvement

Wake on wireless LAN (WoW) is an operating system (OS) feature in which OS offloads the host side work to the firmware side. It means the Operating system; its processor can go into low power mode, but the Wi-Fi device internal firmware/processor needs to be awake every time to handle some specific packet which the operating system asks to handle. The particular packet can be an ARP request, EAPOL packet, or another particular packet. So, this is a partial energy conservation method.

The Wi-Fi device's internal firmware/processor still needs to awake, and it is a waste of energy. We are proposing solution for this real problem. When Operating System enables the WoW feature at the Station side, it gives all the necessary information to its Wi-Fi entity. The same information Wi-Fi Entity can share with its Bluetooth/BLE entity, and Wi-Fi can go into deep sleep mode; it means it can completely shut down its RF chain.

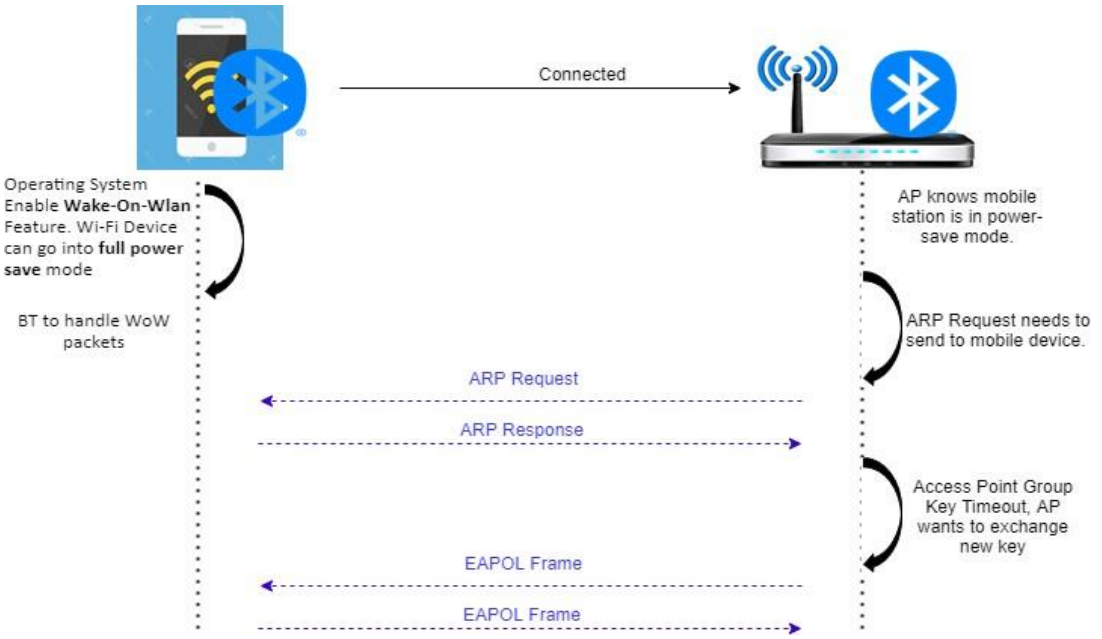

Figure 3.8 Proposed Method in Case of Wake on Wireless Lan Scenario

AP should also know the situation before STA goes into full power save mode. So, when the operating system enables the WoW feature at the client-side, the client Wi-Fi device needs to send a particular packet to AP (Vendor-specific action frame or data frame) so AP would know STA is in WoW mode. After receiving successful acknowledgment for this packet, STA can go into deep sleep mode. If any WoW-related packet comes, it can handle via Bluetooth Entity of AP and STA (Shown in Figure 3.8).

3.2.4 Power consumption results & discussion

This part will evaluate the results in the standard scenario and our proposed model, synthesize the method analyzed. When Wi-Fi wakes up, sleeps, or receives any data, there will be some difference in power consumption. Again, switch on BLE in wake up, sleep mode, and receiving mode for various time intervals. Finally, we will be able to conclude which device must be kept and in which mode so that minimum consumption of power will take place and save more energy and money. We will calculate the power consumption of STA in legacy power save with light downlink traffic. The following acronym will be used to calculate power consumption.

P_{idle}: Idle state power

P_{sleep}: Sleep state power

P_{wake}: Awake state power

T_{DTIM}: beacon DTIM period

$T_{addition}$: extra time before the arrival of a beacon packet

S_{beacon}: size of a beacon frame

S_{SYN}: size of physical synchronization

S_{Header}: size of a MAC header

R_{PHY}: physical link data rate

$$P_{idle} = P_{sleep} + \frac{P_{wake} - P_{sleep}}{T_{DTIM}} * \left(T_{addition} + \frac{S_{beacon} + S_{SYN} + S_{Header}}{R_{PHY}}\right) \quad (1)$$

Here we take the Beacon frame as a reference frame and table 3.1 value taken for the calculation of Pidle [71].

Table 3.1 Default Values for the power calculation in the idle mode.

Parameter	Default Value
S_{beacon}	192 bytes
S_{SYN}	24 bytes
S_{Header}	34 bytes
R_{PHY}	1 Mbps
$T_{addition}$	2 ms
P_{sleep}	3.2 mW
P_{wake}	432 mW

In equation 1, put values from table 3.1:

$$P_{idle} = 3.2 + \left(\frac{959.65}{T_{DTIM}}\right) \quad (2)$$

Energy took by the device to operate d days.

$$W_{idle} = P_{idle} * 24 * d \quad (3)$$

Power calculation in BLE [72]:

The average power consumed by BLE during advertisement/connection:

$$P_{bttx} = 1147.38uA * Vdd \quad (4)$$

The average power consumed by BLE in an idle connected state

$$P_{btidle} = 324.18\,uA * Vdd \quad (5)$$

Now BT will send keep-alive and check for Rx packet.

BT is taking approx T_{bttx} = 200us = 0.2ms for Tx as per reference [72].

The BT checks every TDTIM for the packet, then power consumed by BT:

$$P_{BTA} = \frac{P_{bttx} * T_{bttx} + P_{btidle} * T_{btidle}}{T_{bttx} + T_{btidle}} \quad (6)$$

Suppose BT is checking in every DTIM.

$$T_{DTIM} = T_{bttx} + T_{btidle} \quad (7)$$

And vdd=3V.

$$P_{BTA} = \frac{(0.68 + 0.972 * (T_{DTIM} - 0.2))}{T_{DTIM}} \quad (8)$$

Power calculation in Proposed Method:

BT will take care of the DTIM polling and keep-alive. So, Wi-Fi will act in sleep mode, and BT will operate in active mode.

$$P_{ourMode} = P_{btActivemode} + P_{wifiSleepMode} \quad (9)$$

$$P_{wifiSleepMode} = P_{sleep} = 3.2\,mw \quad (10)$$

$$P_{btActivemode} = P_{BTA} = \frac{0.68 + 0.972 * (T_{DTIM} - 0.2)}{T_{DTIM}} \quad (11)$$

$$W_{ourMode} = P_{ourMode} * 24 * d \quad (12)$$

The Figure 3.9 shows the power difference between standalone Wi-Fi and Combined mode with different DTIM values. For DTIM 50ms and 100ms (from equations 3 &11), the power difference is shown with the number of days between standalone Wi-Fi and Combined mode (Figures 3.10 and 3.11).

Bluetooth can be used to awaken a device from a low power state to initiate a Wi-Fi connection, and to keep the Wi-Fi connection active during periods of inactivity. By doing so, Bluetooth can reduce the frequency of the Wi-Fi connection's power-intensive radio wake-up operations, which can help to improve overall power efficiency.

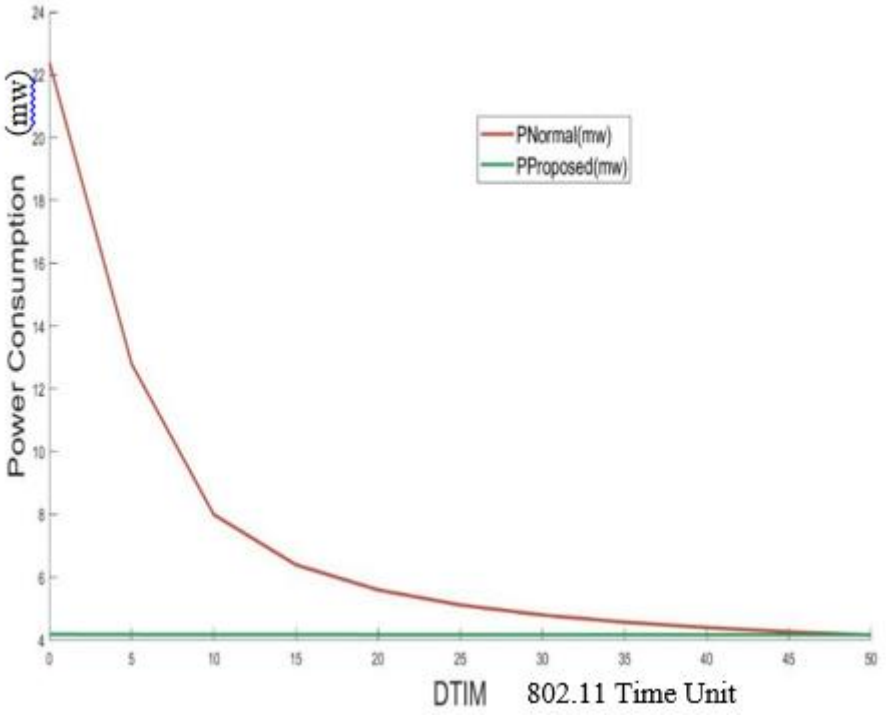

Figure 3.9 Power graph between Normal Scenario and the proposed method (Equation 2 & 9)

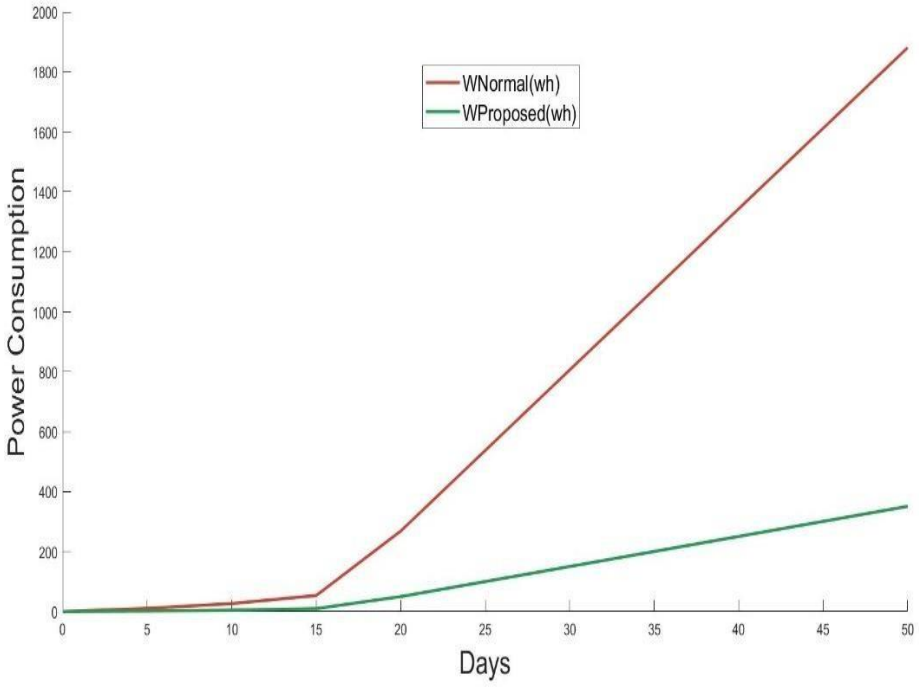

Figure 3.10 Consumption between normal scenario and with proposed method for DTM 50ms

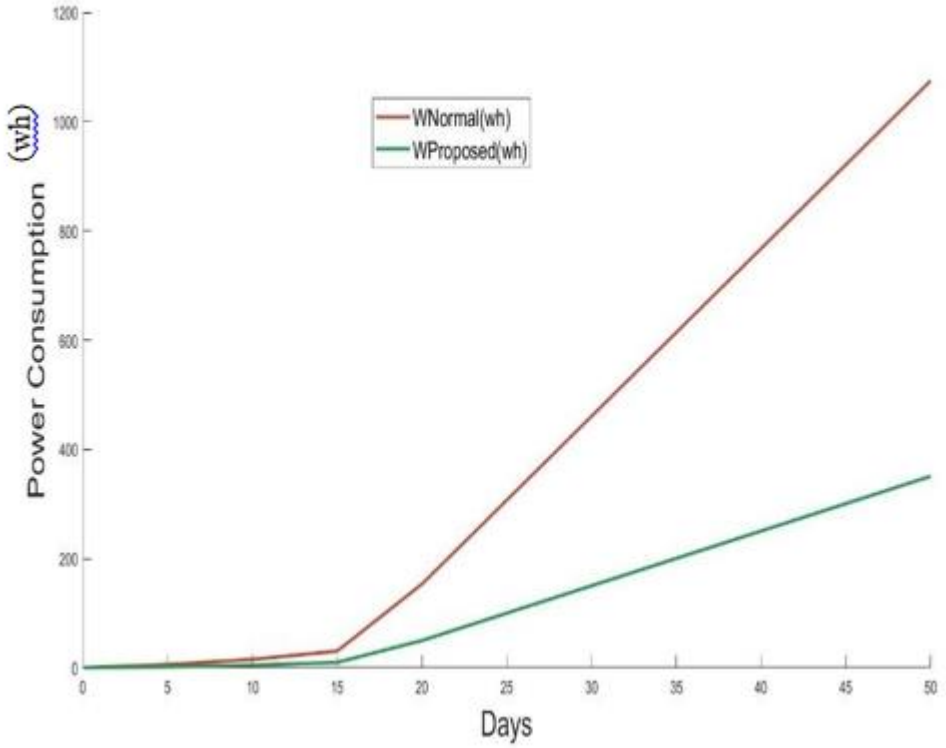

Figure 3.11 Consumption between normal scenario and with proposed method for DTM 100ms

3.3 Adaptive Listen Interval based power-save

Adaptive Listen Interval is a proposed feature in 802.11 power save mode that adjusts the amount of time a device spends in sleep mode to conserve power and listens for incoming Wi-Fi traffic. It does so by analyzing device battery requirement and optimizing the listen interval to reduce power consumption while maintaining a stable connection. This results in improved power efficiency for Wi-Fi devices. The objective is to minimize the device's power consumption while still allowing it to receive data in a timely manner. This results in improved battery life for the device.

3.3.1 Design & Architecture of the proposed framework

The problem can be cast as a framework/protocol limitation aiming to maximize station battery usage and AP memory under Wi-Fi use cases. Moreover, we intend to propose without requiring modification in the standard IEEE 802.11 end devices hardware. We suggest software changes on both sides (STA and AP) of the Wi-Fi device. The first subsection describes access point changes, and the second discusses station side changes. Moreover, these changes are entirely software-driven and don't need any hardware modification. To implement this feature, a user needs to change the Wi-Fi

driver and firmware of the device and deploy it on the device side.

3.3.1.1 AP Framework Description

This framework can classify frames based on the frame type and subtype. Different elements play their role according to the requirement. An AP can have many parts, but we discussed and showed aspects according to our proposed method. This proposal aims to improve battery consumption, especially when the battery dies. It adaptively and dynamically controls AP performance concerning the APs buffer and keep-alive time network characteristics.

Figure 3.12 shows the proposed framework for the access point. The figure only shows the proposed elements. The existing architecture [50] is assumed to be working as it is. With the proposed method, AP also supports legacy STA devices, which means STA can connect to this AP, even does not support dynamic LI. A particular bit is introduced in the beacon to give information about this feature, whether it is supported via AP. Our proposal allowsthe access point to evaluate whether the clients have the right to obtain more buffers. AP framework elements are discussed as follows:

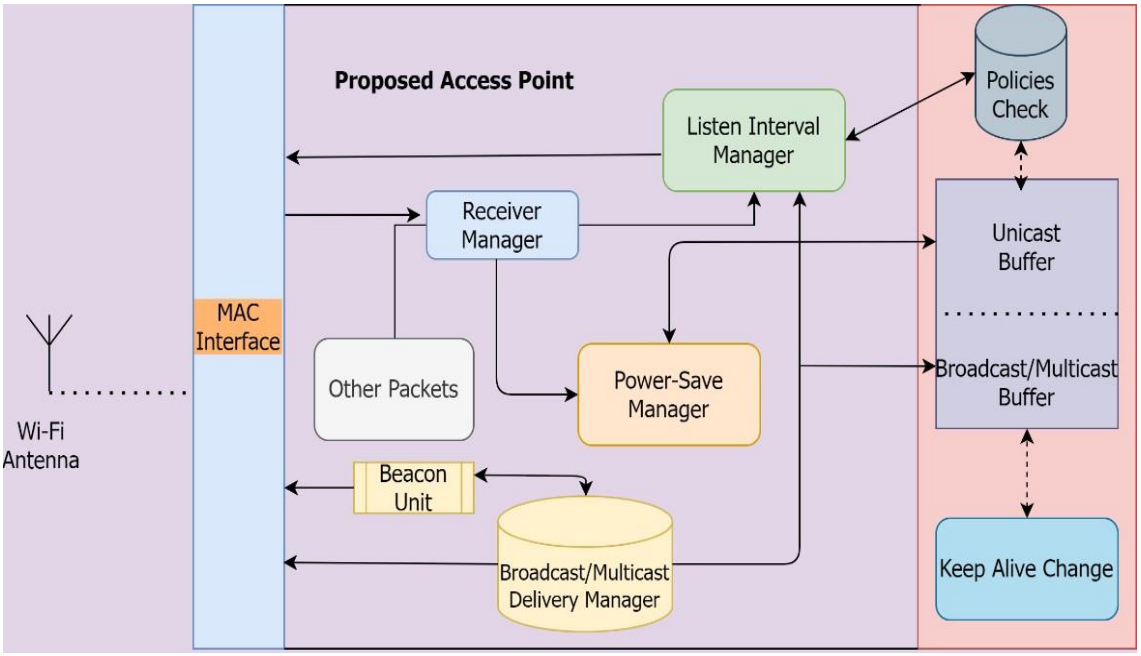

Figure 3.12 Proposed access point Framework

Receiver Manager: The receiver-manager handles all the good packets received via AP. A Receive packet is a good packet when it does not have any receive errors like Integrity check value (ICV) error, CRC, sequence number mismatch, etc. For example, if the frame is with PM=1, it will redirect to the power-save manager,and if the frame is related to changing into LI, it will activate the Listen Interval Manager. Other packets

go on the default path.

Listen Interval Manager: This module decided whether AP should accept a change of LI request from STA. With consideration of policies, accept or reject request decisions are taken.

Listen interval manager consider the following policies in the decision:

1. LI should be like this; all STAs should awake for Broadcast (BC)/ Multicast (MC) frame notification.

2. MAX_LISTEN_INTERVAL should be defined in a way; Transmission Control Protocol (TCP) timeout or upper-layer timeout should not happen.

3. The requested LI should not be greater than MAX_LISTEN_INTERVAL.

4. The memory buffer should be enough to increase LI.

Power-save Manager: This module handles power-save management requests and stores upcoming frames to buffer.

Beacon Unit: Beaconing with supported feature enables performed via this unit.

Broadcast/Multicast Delivery Manager: This module sets DTIM into a beacon to deliver a BC/MC packet to the station. It deals with broadcast/multicast buffers, whether a BC/MC packet needs to buffer or from the buffer when it needs to deliver. Here the Broadcast/Multicast Delivery Manager makes sure at any point, and STA should wake together, so BC/MC packets can provide to them. Dynamically DTIM values change according to LI values of STA and buffer.

3.3.1.2 STA Framework Description

Figure 3.13 shows the proposed changes for the station. The" battery status utility" application is created on the Operating System (OS) level. This utility is amendment on existing station framework [138]. This utility is responsible forreading the battery status, triggering the Wi-Fi connection, and triggering the power save operation, apart from the application layer proposed framework same as the standard Linux OS wireless device framework. Management packet (e.g., Association frame) handled via wpa_supplicant and passed through to the driver. In contrast, data packets (power save packets) go through a standard netlink Socket (SOCK) to the device.

The OS tool is used to read battery status. For example, the upower command-line tool extracts information related to the power source (batteries) on Linux. Example

command:

upower -I $(upower -e | grep BAT) | grep–-color=never -E ”state|to\ full|to\ empty|percentag”

802.11 has three different types of packets: Management, control, and data packets. Association request (a management packet) and other data packets are in the interest of proposed work. In Figure 7, the management path (management packet flow) and data paths(data packet flow) are shown.

Management path: When the connection is triggered, battery status is provided to wpa_supplicant via battery status utility. Messages sent via nl80211 are in the kernel handled in the cfg80211 module and finally reach the device, as shown in Figure 7.

Data path: Power-save related and low battery data received from the utility are passed to the relevant socket, then through the TCP/IP stack, passed to the server, which implements the IP protocol (INTETSRV), and finally reaches the device. The device transfers packets to the intended access point.

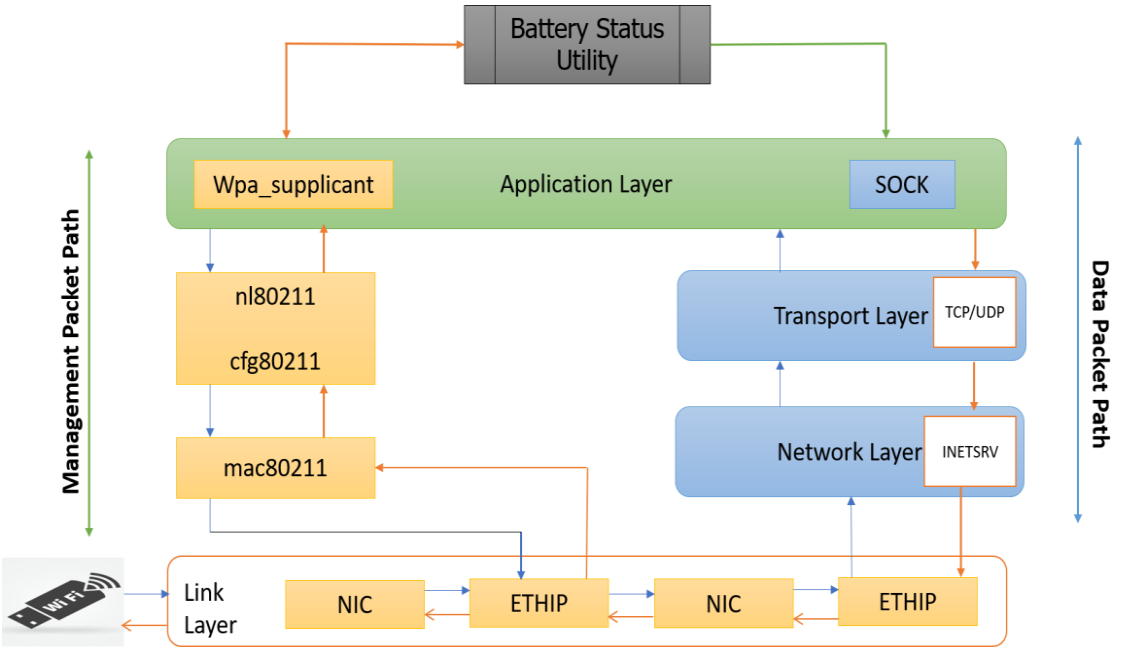

Figure 3.13 Proposed Station Framework

3.3.2 Protocol Implementation

Protocol implementation aims to demonstrate the proposed method’s effectiveness and fit it with the 802.11 standards. Two Atheros ar9271 USB boards are taken to experiment, develop and validate the presented protocols. RF cable is used instead of

open air because it offers several advantages over open air for 802.11 testing. It provides a controlled environment, allows for precise measurements, reduces interference, and is easy to set up. The existing source code [73]needs to be modified according to the proposed protocol. One USB board acts as STA,and the other acts as an AP. The lab setup is shown in Figure 3.14.

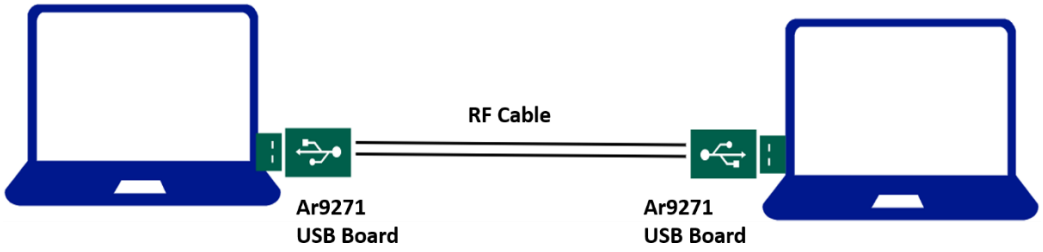

Figure 3.14 Lab Setup

Hostapd is used to configure AP, and wpa_supplicant is used to configure STA. For example, the following command uses to up AP and install hostapd:

```
$ apt-get update

$ apt-get install firmware-atheros

$ apt-get install iw

$ apt-get install hostapd

$ hostapd ~/hostapd.conf //To configure AP

$ apt-get install isc-dhcp-server
```

And station can connect to AP (SSID – LI_AP) with the following command:

```
$ wpa_passphrase LI_R_AP >> /etc/wpa_supplicant/ wlan0_sta.conf
```

In the 802.11 specifications, a vendor-specific element can use for specific information defined for a particular purpose. The vendor element will use here to customize the packet. Mainly four frames have been amended or introduced to implement the proposed feature at STA and AP sides.

First is the Beacon frame amendment on the AP side. A vendor element (Information Element 0xDD) has been added under the beacon frame, showing AP supports the proposed dynamic LI feature.

The sniffer snapshot (Figure 3.15) clearly shows vendor-specific data is 0x1, which means the proposed dynamic LI feature is supported via AP. AP does not support this particular feature if this element is not present or information is 0x0.

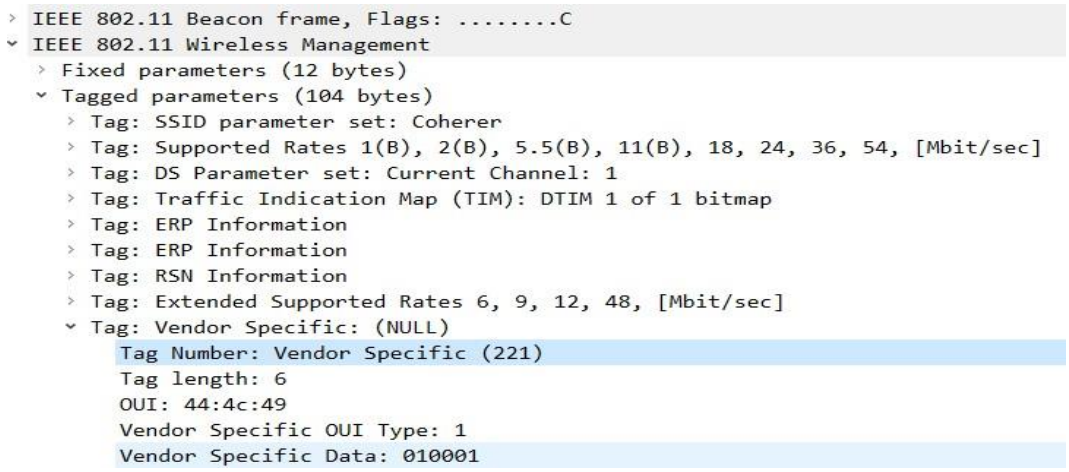

Figure 3.15 Beacon Sniffer with LI feature-enable

The second is Association Request on the STA side. In association request, STA sent LI. If STA never wants to go into power-save mode, value 0 is used by STA. If an STA connects with LI 0, this feature is disabled via AP internally for that STA. Battery-operated devices should not choose 0 as the default value. LI value is expressed in units of beacon intervals. AP uses this LI information in determining the lifetime of frames that it buffers for an STA.

Here a vendor-specific element (ID = 0xDD) is used to send battery status information to AP under the association request frame (Figure 3.16). With this information, STA announced to AP “it supports the proposed dynamic LI feature," and AP can predict STA's future requirements. If the station has enough battery at the time of connection and after connection, “Is STA taking unfair advantage via showing low battery?"

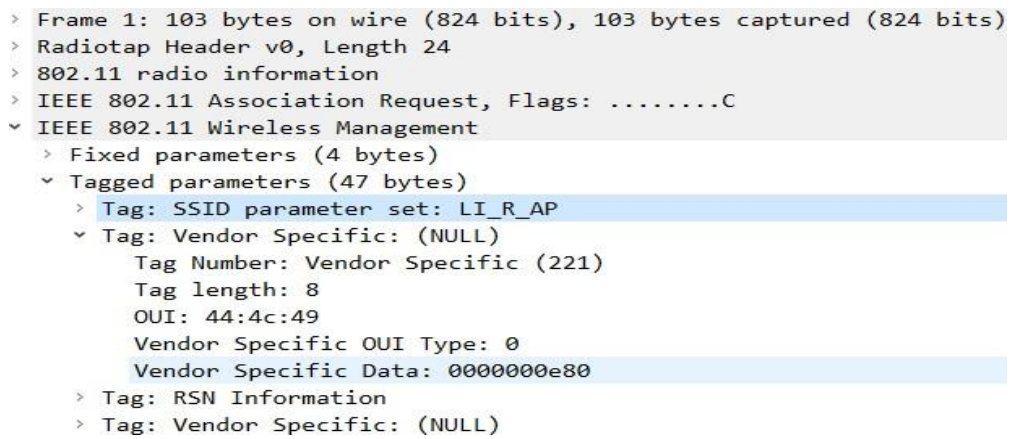

Figure 3.16 Association request sniffer capture with battery status

Formula to send battery status under association request = (total battery * battery percentage)/100

In Figure 10, STA announces it supports the LI feature, and its current battery is 3712 mAh (0x0E80).

The third packet is a new data packet (Figure 3.17) to indicate power save status. An STA can go into sleep mode via a set PM bit into any data packet. Usually, a NULL data frame is used, indicating that power management is enabled to the AP. A particular introduced data packet is used via STA in the proposed mechanism to indicate PSM mode. When STA goes into PSM mode, it constantly updates its battery status to AP. This frame also helps AP to precise its buffer management algorithm.

The fourth is data packets that support dynamic listen interval changes (Figure 11). A particular data packet is sent via STA to change LI according to its battery status. As shown in Figure 3.17, the battery status utility continuously monitors the battery's health. This packet is sent via STA to change the LI value when it's going down from the threshold value (discussed in the algorithm implementation section). It depends on AP whether it accepts or rejects the change LI request. In rejection, AP can suggest a new value. If it agrees to the LI change request, it doesn't need to send the last two frames (Figure 3.18). These changes have been done on both sides (STA & AP). Rejection can be based on the policies defined under section 3.3.1.1.

```
> Frame 108: 104 bytes on wire (832 bits), 104 bytes captured (832 bits)
> Radiotap Header v0, Length 24
> 802.11 radio information
v IEEE 802.11 Data, Flags: .p.P...TC
    Type/Subtype: Data (0x0020)
  v Frame Control Field: 0x0851
      .... ..00 = Version: 0
      .... 10.. = Type: Data frame (2)
      0000 .... = Subtype: 0
    v Flags: 0x51
        .... ..01 = DS status: Frame from STA to DS via an AP (To DS: 1 From DS: 0) (0x1)
        .... .0.. = More Fragments: This is the last fragment
        .... 0... = Retry: Frame is not being retransmitted
        ...1 .... = PWR MGT: STA will go to sleep
        ..0. .... = More Data: No data buffered
        .1.. .... = Protected flag: Data is protected
        0... .... = +HTC/Order flag: Not strictly ordered
    .000 0000 0010 1100 = Duration: 44 microseconds
```

Figure 3.17 Data packet with PM enables sniffer capture with battery status

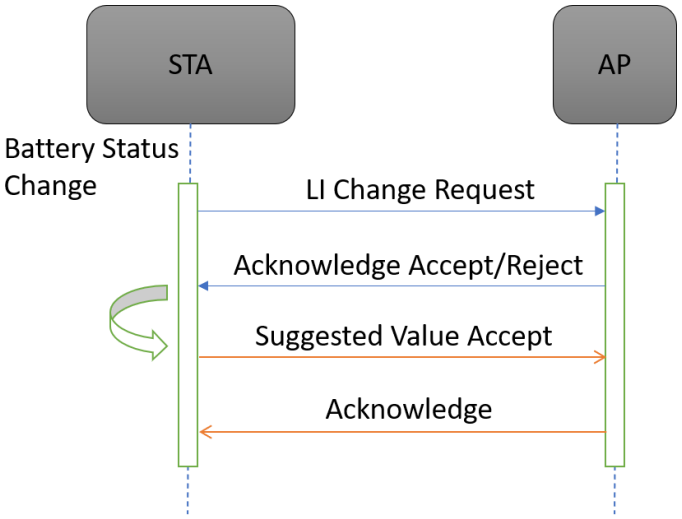

Figure 3.18 LI Change Sequence Diagram

STA can change the LI value again (to the previous value) using the same packet if the STA device charges and the battery level increases.

3.3.3 Algorithm Implementation

This section presents the data structure and mechanism used to implement the STA & AP discussed functionality. The data delivery flow chart is shown in Figure 3.19. The framework of our proposed solution for STA is shown in Algorithm 1 (dynamic battery changes algorithm), which contains several procedures to implement functionality discussed in the framework and protocol implementation phase

As the system's central unit, AP stores the packets for stations that have gone into power save mode. Previously, i.e., buffer allocation for all stations was also fixed when the listen interval and the absolute power save mechanism were static. It is a waste of memory. But in the proposed method, since the listen interval can be dynamic, AP needs to adapt to change (Algorithm 2) and change the buffer limit allocated for each station. The proposed approach uses listen interval, maximum data rates supported viaSTA, and the Number of Special Streams (NSS) to define how much maximum buffershould allocate to the station. Previously, static allocation methods like storing buffer data in arrays could be used. However, now such practices would be expensive, and rather than these, we suggest holding the buffered data using a heap data structure withlinked lists. One of the reasons for adopting this methodology is that now the AP doesnot know how much the listen interval will be, and it can change according to the needsof the STA. Hence, a linked list is a better choice as, unlike arrays, the size of a linkedlist is not predefined, allowing the linked list to increase or decrease in size as the program runs.

Another reason for using the heap data structure is that the AP uses queues to send packets in the FIFO mechanism. Now, to support the Wi-Fi Multimedia (WMM) standard and QoS, queues should also need to store the data as per the priorityof the packets. The heap is a data structure used to store such queues according to theirpriority in the form of binary trees. The algorithm uses max heap data structure, whichmeans the root node must be the greatest among the keys present at all of its children. The exact property must be recursively true for all sub-trees in that Binary Tree. Hence, whenever AP needs to dequeue the buffered packet, it will send the root node buffered data, which will have the highest priority. Of course, the ageing technique is used to overcome the starvation of the deque frame.

WMM prioritizes traffic according to four Access Categories (AC): voice (AC_VO), video (AC_VI), best-effort (AC_BE), and background (AC_BK). Following this and in the order of packet received, the Rx queue will assign weights to each buffered packet, and this weight will be further used to decide the packet's position in the queue.

For different access category packets, different weights will be assigned in the order as (MAX)AC_VO → AC_VI → AC_BE → AC_BK(MIN). For same-category packets, weights will be assigned as per the FIFO mechanism. So the packet first received will have more weight than the packet received afterwards.

A new packet is attached at the tail first per algorithm and moves the packet up the order per the access category. Once reach the same access category packet or the root, stop iteration and enqueue the packet.

The discussed algorithm also ensures data delivery even when the beacon misses via STA (shown in Figure 3.19). If STA missed the beacon due to noise, congestion, or some reason, STA still woke for the next beacon, and AP put delivery status into the next beacon.

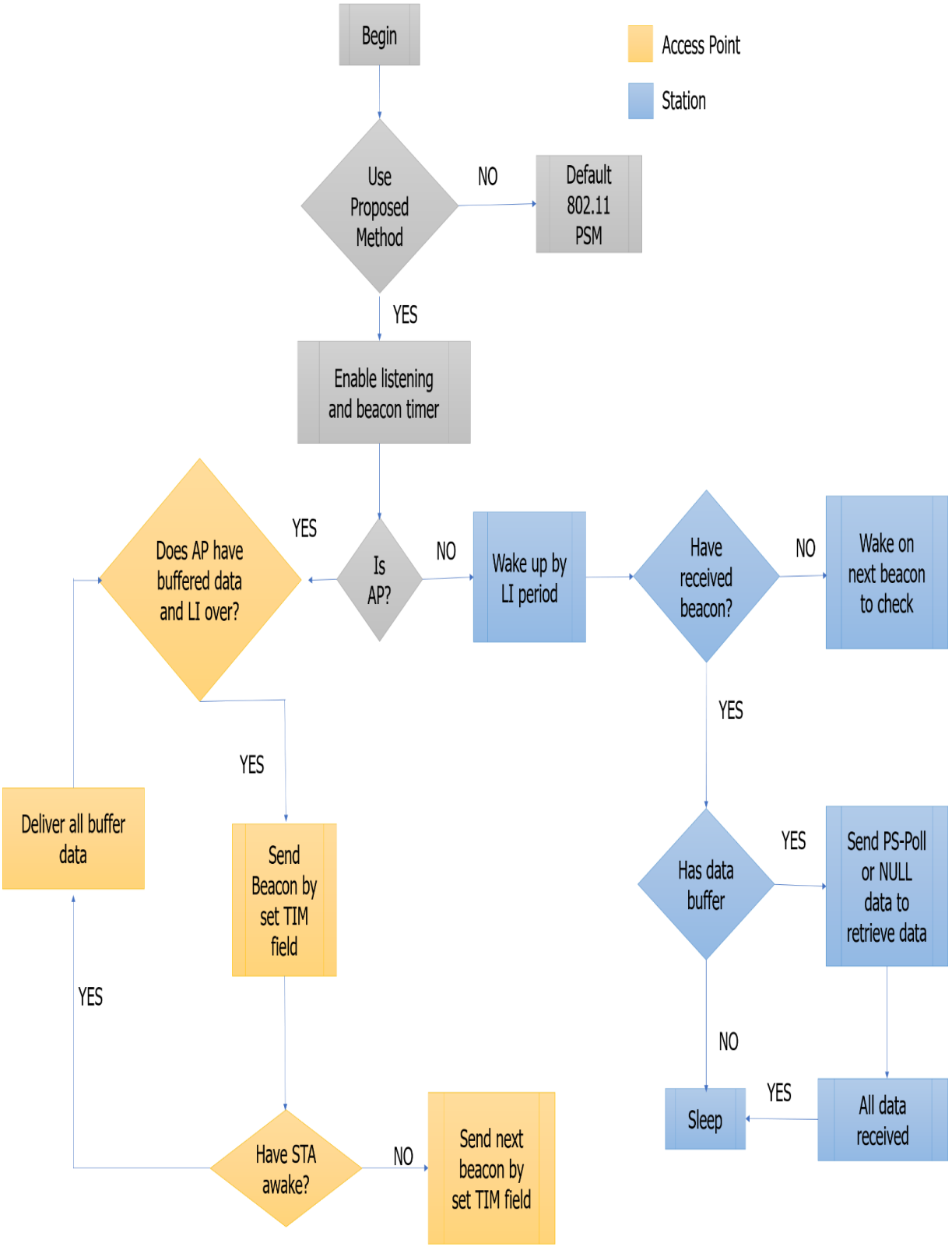

Figure 3.19 Data Delivery Flow Chart

Algorithm . STA Dynamic Battery Change

```
Input:          batteryStatus //current status of battery in percent
                powerState //current power state, battery or charging
                defaultlisteninterval //default listen interval
                gMaxLI //max listen interval supported by AP
Initialization of  //assign variable to default values or with zero.
variables:      newlistenInterval ←0  //variable to store new listen interval
Output:         newlistenInterval //Latest LI value
1               if powerState == DeviceBatteryOn then
2                   if(BatteryStatus < 5%) then
3                     newlistenInterval ← z
4                   else if (BatteryStatus < 10%) then
5                     newlistenInterval ←y
6                   else if (BatteryStatus < 15%) then
7                     newlistenInterval ←x
8                   else
9                     newlistenInterval←defaultlisteninterval
10                  end if
11              else //powerState == DeviceCharging
12                  newlistenInterval←defaultlisteninterval
13              end if
14
15              if newlistenInterval > gMaxLI then
16                  newlistenInterval←gMaxLI
17              end if
18
19              updatelistenintervaltoAP(newlistenInterval)//send a frame to update
20              the new listeninterval to AP
21
22              return newlistenInterval
End Algorithm
```

Algorithm . AP Dynamic Buffer Change

```
Input:          gMaxLI //max listen interval supported by AP
                ReqLIbySTA //requested listen interval by STA
                defaultSTABackupBuffSize //default STA buffer size
Initialization of  //assign variable to default values or with zero.
variables:      STABackupBuffSize ←0  //variable to store allocated buffer
Output:         STABackupBuffSize //Allocated buffer size to station
1               if ReqLIbySTA > gMaxLI then
2                   STABackupBuffSize←0
3                   return 0 //reject the ReqLIbySTA to STA.
```

```
4                       end if
5
6                       if ReqLIbySTA >=z  then
7                               STABackupBuffSize ←(defaultSTABackupBuffSize*z)
8                       else if ReqLIbySTA >=y  then
9                               STABackupBuffSize ←(defaultSTABackupBuffSize*y)
10                      else if ReqLIbySTA >=x  then
11                              STABackupBuffSize ←(defaultSTABackupBuffSize*x)
12                      else
13                              STABackupBuffSize←defaultSTABackupBuffSize
14                      end if
15
16                      return STABackupBuffSize
End Algorithm
```

3.3.4 Performance Evaluation

This section reports the power consumption results with the time of the proposed and conventional methods. The proposed listen adapter adaption decision can be taken via OS, depending on the type of application/requirement. For example, if some background traffic is running, OS can choose a high LI value for more power-save, and if any video/voice traffic is running, OS can go for a low LI value. There can be a difference between YouTube videos and zoom meetings which OS can differentiate easily. In the IoT device environment, mostly STA transfer packets. STA is well aware of its data requirements to manage LI accordingly. Here our goal is to show that "battery status" can use to choose an adaptive LI value. Synthesis of the papers analyzed and provides a summary in other subsections.

3.3.4.1. Power consumption compare

First, we measured the power consumption concerning the changes in the LI. The same hardware on an Ubuntu-based Linux machine is used for conducting the simulation. Two-mode has been chosen to perform the test, one is normal web browsing, and the other is an idle test.

A script-based web browsing is performed to perform the first test. Data interchange will be uniform for all the web-browsing tests. The script performs browsing for the combination of voice, video, normal web-browsing, iperf [74] TCP throughput and (File Transfer Protocol) FTP. There is one main machine connected to the Ethernet switch with an AP. Iperf, a host FTP server, and a video server are running on this

machine. User Datagram Protocol (UDP) and TCP are packets transferred from STA to AP and vice versa. The FTP server receives the files uploaded via the station, and the file is the size of 10Gb. Iperf is used to run TCP/UDP traffic, so the worst-case scenario and maximum throughput can test. The video server used to run 4K videos on STA. This traffic pattern represents daily use cases like watching videos, email checking, file upload/download and web browsing on the wireless medium.

The results presented in Figure 3.20 and Figure 3.21 show a higher LI value can save more power in every scenario. The test duration for the web-based test is 60 minutes, and for the idle test, it is 30 minutes. Voltage is constant so power consumption shown in mA unit. The test is performed with LI values of 1,2,4 & 8.

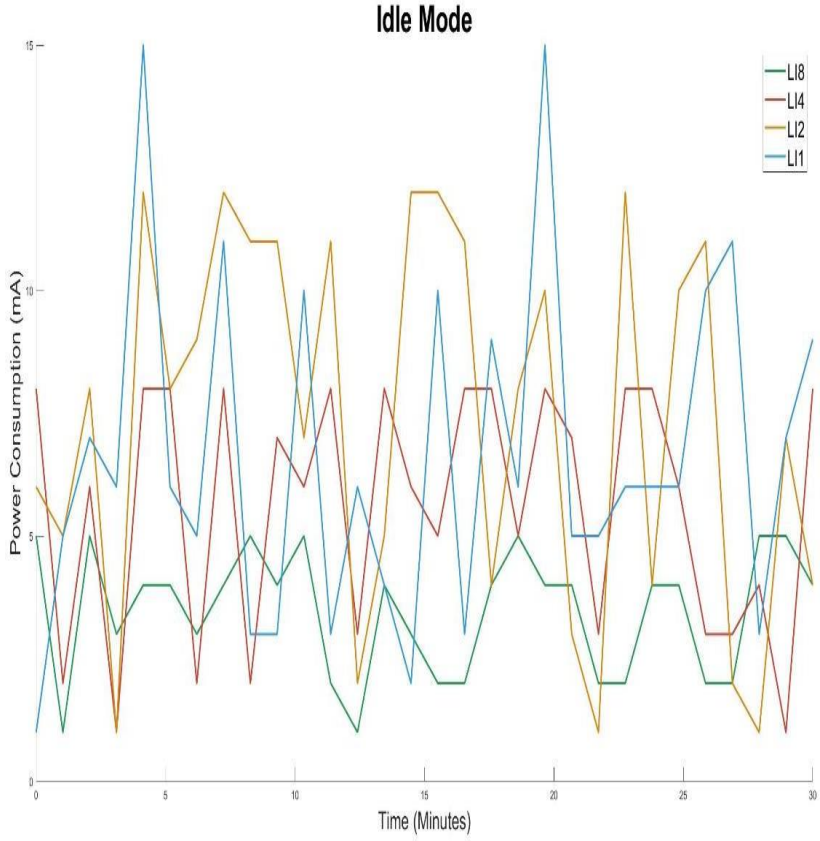

Figure 3.20 Station Power Consumption in idle mode

An Asus AP is taken to connect the station device (configuration shown in table 3.2). Wi-Fi power consumption reading is taken with millimetres, and the same is used to plot the graph (shown in Figure 3.20 and Figure 3.21).

Table 3.2 Test Access Point Configuration

Frequency	Mode	Channel	SSID	Bandwidth
5.0 GHz	11ac	36	Test-LI	80 MHz

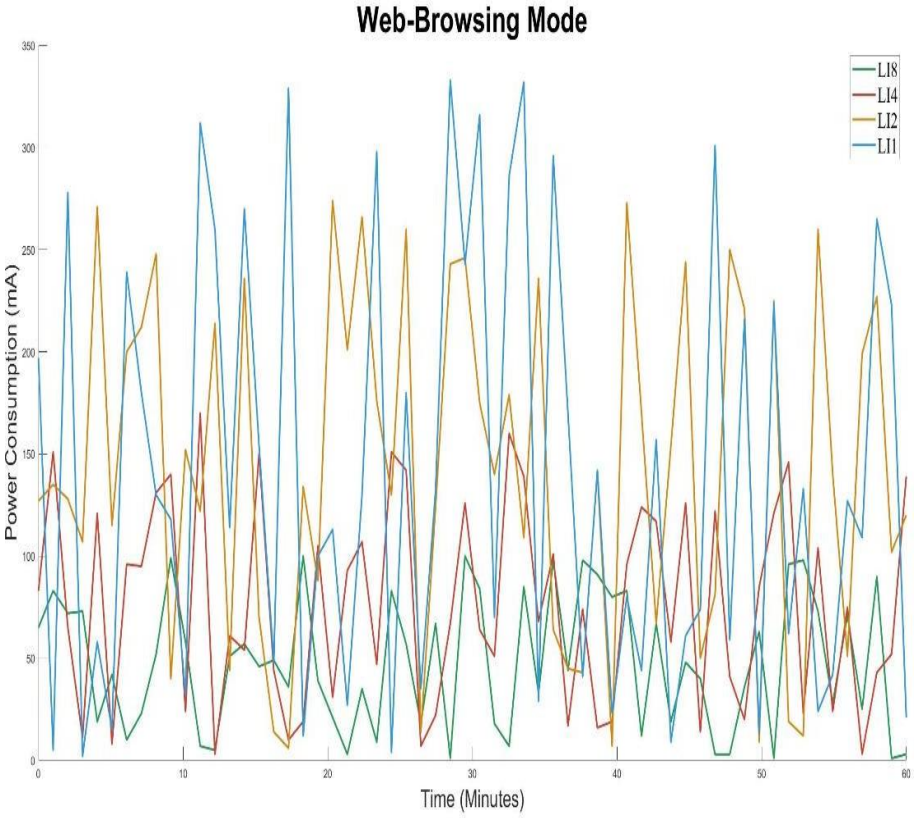

Figure 3.21 Station Power Consumption in web-browsing mode

The system is put into the idle state to perform the idle test, and there is no intentional traffic running on it, and every time, a Wi-Fi connection is to be maintained by STA. There is no such disconnection happening during the test.

3.3.4.2. AP buffer compare

Conventionally, static allocation is done for the STA buffer, which takes 100%of memory. The proposed access point buffer management algorithm (using system simulation) runs for eight station devices to demonstrate the efficiency of the proposed access point framework. The results are shown in the graph (Figure 3.22).

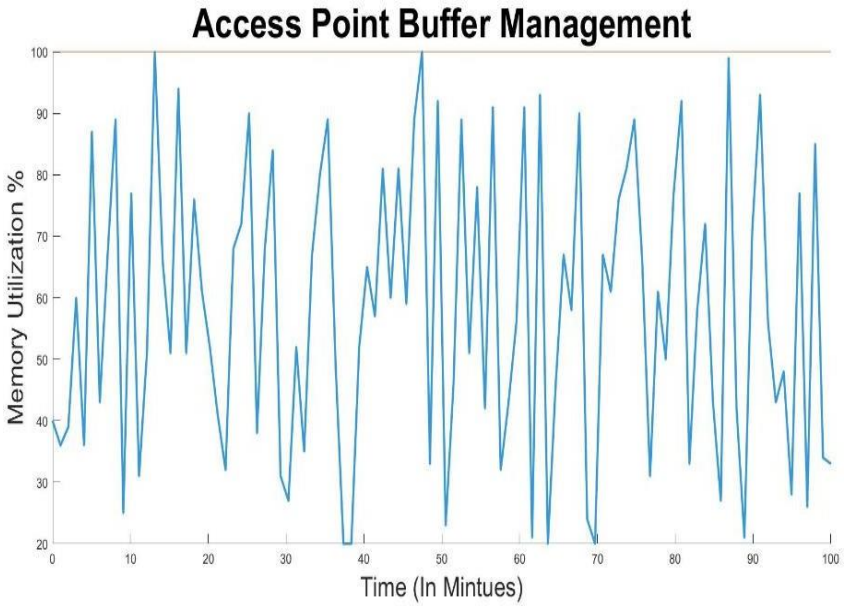

Figure 3.22 Access Point buffer memory utilization with time

At the start of the tests, all software-based 8-station connected to the access point and random Wi-Fi data and sleep tests are performed one hundred times to simulate the test. All the wireless entities present in the model strictly followed CSMA/CA concept and proposed policies. A memory manager unit calculates memory usage during tests, and LI changes accordingly.

3.4 Summary

In this chapter, we have discussed how low power technology can help to improve Wi-Fi power-save by considering some unique factors that remain unaddressed by the existing multi-radio mechanism. We have provided a solution to improve the performance of 802.11 PSM using a combo solution. Our invention focuses on communicating the Wi-Fi buffered packets (unicast/multicast/broadcast) status to STA using the active BT/BLE connection between BT/BLE peers present at the station & access point. The AP communicates buffered packet status to its corresponding active BT/BLE device, further communicating to STA's BT peer. The STA's BT peer will further notify its Wi-Fi entity based on the received message. Based on the message, Wi-Fi STA will take the appropriate decision to exit power save or not. The proposed architecture shows how Wi-Fi chip vendors can apply it in Mesh topology, how it will work, and whether a unique Wi-Fi frame is needed. Finally, mathematical power consumption results show that the proposed mechanism can reduce power significantly,

especially in the IoT devices era. The theoretical analysis and the experimental power results shown and the presented work are practical.

In another study, we have shown how the proposed “Adaptive LI” method can help stations to choose LI values based on the available battery and save reasonable power consumption. The performance results show that battery-operated devices can run for a long time with a long LI value, and AP can also optimize the buffer memory. Moreover, the framework achieved the practical workability of 802.11 use cases. The proposed protocol implements an adaptive LI mechanism between the station and AP with consideration of the 802.11 layers. Furthermore, no hardware change is necessary, making it easy to implement. The discussed method is new, and to our knowledge, no wireless instrument (available in the market) allows changing dynamic STA’s LI and AP buffers. Two Atheros ar9271 USB boards are taken to demonstrate a physical experiment, and proposed protocol development & validation are discussed in this chapter.

CHAPTER 4
REDUCE 802.11 CONNECTION TIME

The need for reducing the connection time in 802.11 networks stems from the increasing demand for fast and reliable wireless connectivity. A longer connection time can result in a slower user experience, causing frustration and decreased productivity. By reducing the connection time, 802.11 networks can provide faster and more responsive wireless connections. This can be especially important in environments where multiple devices are competing for limited network resources, such as incrowded public areas or busy offices. Reducing the connection time can also help to improve overall network performance by freeing up resources for other devices to use. By reducing the amount of time spent establishing a connection, 802.11 networks can provide a more efficient and user-friendly experience.

4.1 Introduction

In this chapter, we describe 3-way to reduce connection time. First one is with the use of cross-layer approach via offloading DHCP work to the MAC layer. DHCP offload to MAC layer is a feature that allows the DHCP client to be moved from the main processor of a device to the MAC layer of the device's wireless network adapter.This offload reduces the load on the main processor, improving performance and efficiency. Second one is Scan improvement, which refers to the optimization of the process of searching for available wireless networks. This can be done through optimizevia offload it to a low power device. Last one is use of low power technology. Low power technology is used to reduce the power consumption of wireless devices, as mentioned in chapter 3.2. Same architecture and principle mentioned under section 3.2.1 can apply to improve connection mechanism in dense and noisy environment.

This chapter focus on by combining these three techniques, the performance and efficiency of 802.11 connections can be significantly improved, leading to better overall user experience.

4.2 Offloading and Merging of DHCP layer to MAC layer

802.11 management frames and DHCP are distinct network protocols that have different functions. 802.11 management frames are used to manage the wireless network, while DHCP (Dynamic Host Configuration Protocol) is used to assign IP addresses to devices on a network. They operate at different layers of the networking stack. 802.11 operates at the physical and data link layers, whereas DHCP operates at the above layers.

Basic idea is to offload DHCP frame at MAC layer and combining 802.11 management with DHCP frames would potentially reduce the overhead associated with sending separate frames, it has a significant impact on the overall time required to complete the authentication, association, and IP address acquisition processes. Moreover, it would reduce introduce additional complexity, which may make it easier to troubleshoot and maintain the network over time.

To perform the test, we have used our own simulator system on windows operating system. We compile our driver and application by Visual Studio 2017 and use WinDDK to build the driver [75]. To perform our test, we developed a simple command line application and a windows WDM driver [76]. Here application acts as both DHCP Client & Server application for the station and access point respectively and this application talk with a simple WDM driver [77] using IOCTL. WDM driver works as the driver itself and as a virtual firmware and hardware device for a Wi-Fi device.

Here we assume access point has an embedded DHCP server. First, we tried to offload DHCP feature, like ARP or EAPOL offload (GTK rekey) performed in case of WoWLan (Wake on Wireless LAN) [78] feature. Offloading means instead of host/driver layer, the feature is performed by firmware layer. The host layer and firmware layer are two of the main layers in a wireless card. The host layer is the software layer that resides on the host computer. It is responsible for communicating with the firmware layer in the wireless card. The firmware layer is the software layer that is embedded in the wireless card. It is responsible for handling the low-level details of wireless communication, such as encoding and decoding data, and managing the wireless card's hardware.

The host layer and firmware layer work together to provide a high-level interface for wireless communication. The host layer provides a set of APIs that allow the host computer to send and receive data over the wireless network. The firmware layer

implements these APIs and handles the low-level details of wireless communication. Offloading is primarily used to reduce the power consumption of devices. It also saves some amount of time by reducing the driver-device latency [79]. Later, we merge a DHCP frame with connection frames.

DHCP mainly IP layer procedure, and it is time-consuming also. If we talk from 802.11 perspectives, DHCP related frames are data frames, while connection frame like authentication & association is management frames. In our work, we have offloaded the DHCP feature and we have combined this feature with MAC layer connection frame via adding DORA [80] support in authentication/association frame. We have already discussed "connection and DHCP both have 4 frames respectively". So, we have done one to one mapping to create new frames which are shown in Table 4.1.

Table 4.1 Proposed Frames

Proposed Frame	**Direction**	**Combination**
Authentication Request	station to AP	Authentication Request + DHCP Discover
Authentication Response	AP to station	Authentication Response + DHCP Offer
Association Request	station to AP	Association Request + DHCP Request
Association Response	AP to station	Association Response + DHCP Acceptance

We have moved all the things to the MAC layer and at the MAC layer, Authentication Request is a unicast frame to AP, while DHCP discover is a multicast frame. Here we are assuming that access point has embedded DHCP server, so discover frame can be unicast as well. Apart from discover frame, all other frames are unicast at mac layer, so it does not provide any problem.

The proposed Auth request frame have special IE (Information Element) present, which shows whether the device supports static IP or a dynamic IP and DHCP related (layer 3) information. If static set-in station's auth request IE, AP's (inbuilt DHCP server) does not need to provide any IP to the station. Same in other connection frames: Auth response, Assoc request & Assoc response frame also have special IE which consists of DHCP related information.

Figure 4.1 shows the general device model about the layers a Wi-Fi device consist. Here firmware runs inside the device hardware, so we assume both are at the same layer.

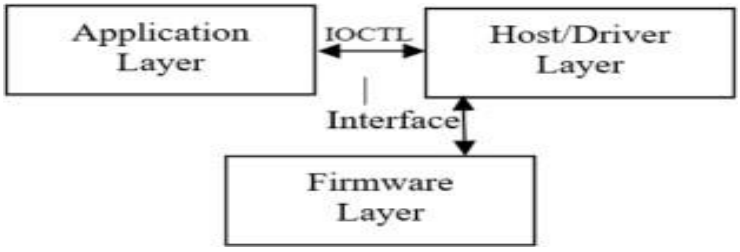

Figure 4.1 General Device Model

To simulate the test, we have created two layers at user space and kernel space. Application runs at user space and driver & virtual firmware functionality runs at kernel space. The application works as a DHCP Client and DHCP server for station & access-point unit respectively. Application talks with the driver using IOCTL and a fixed interface delay gap are given between Host & Firmware function at the driver side. In a real device, an interface exists via which device is connected to the system, this interface can be USB, PCIe, SDIO or anything else, so an interface delay added between driver and firmware function.

The Figure 4.2 shows our simulator model to perform the test. And below steps performed via every entity participate in this test:

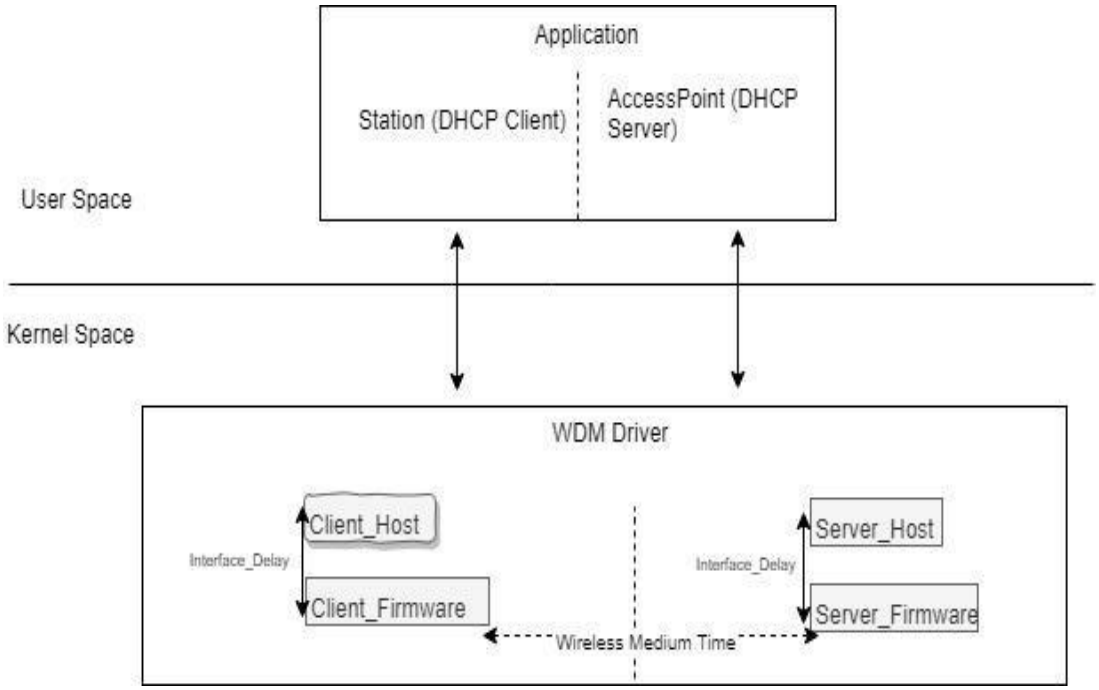

Figure 4.2 Simulator Device Model in our test

1. Create connection command comes to application.
2. The application sends create connection OID to device driver with DHCP or Static IP support.
3. OID is handled by Client_Host function and it bypasses details to Client_Firmware function.
4. Now Client_Firmware sends proposed Auth Request frame to Server_Firmware function.
5. Server_Firmware process auth request frame and response back with auth response frame.
6. Client_Firmware receive Auth response frame and after successful processing, it sends back association request frame to server_firmware.
7. Finally, the association response frame is reverted back by Server_Firmware.
8. Respectively, Client_Firmware and Server_Firmware sync information with its respective application part via Client_Host&Server_Host bypass function.

In case of default behavior, DHCP works at the application layer and connection procedure works at the host or in some cases, at firmware side.

In our model, we have moved all the new frames to the firmware layer for fast execution. The application just provides required information to firmware via the host, so the new frame can be created at firmware side and send to access point as shown in Figure 4.3.

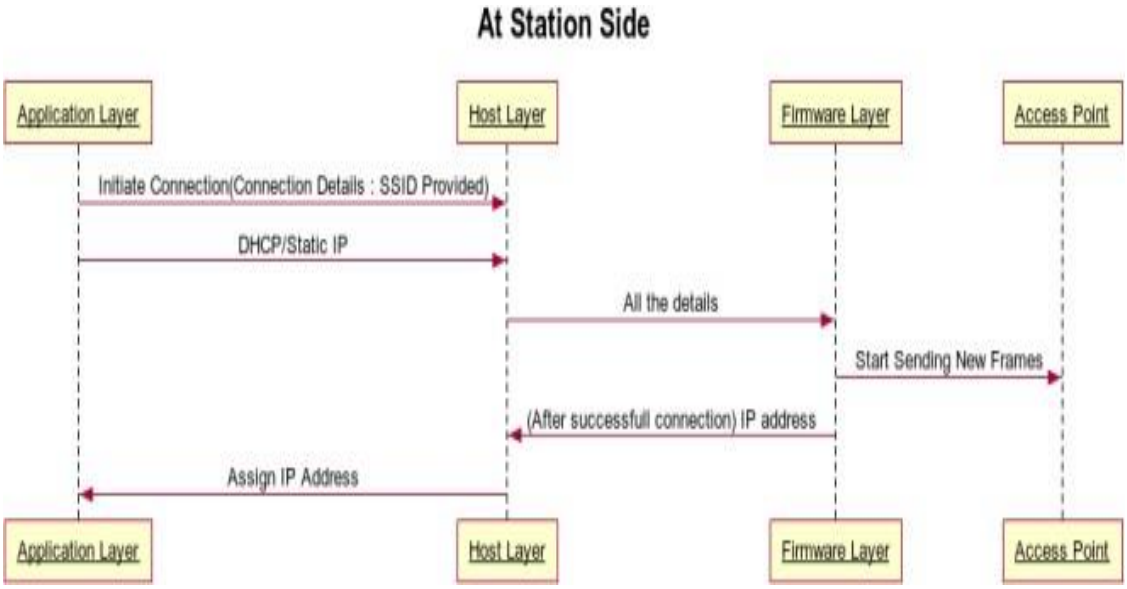

Figure 4.3 Station Side Implementation

Same at Access Point side, DHCP server & authentication works at firmware layer. Below is the packet hex dump generated by our application. The color rule is followed to denote packet internally. Here Yellow color shows MAC Header, Green color shows connection Specific frame data (Auth or Association frame) and Pink color denotes DHCP Relative frame data. The default is the FCS field. In Auth Request, the red color is vendor command 0xDD followed by 0x0A, shown station support dynamic IP configuration instead of Static IP configuration.

For this test:

- AP MAC Address:00:11:22:33:44:55
- Station MAC Address: 00:00:66:77:88:99

A. Auth Request:

B0 00 3C 00 00 11 22 33 44 55 00 00 66 77 88 99 00 11 22 33 44 55 D0 0E00 00 01 00 00 00DD 0ADD 01 00 00 00 00 00 00 00 00 00 00 00 00 15 93 DE D9

B. Auth Response:

B0 00 32 00 00 00 66 77 88 99 00 11 22 33 44 55 00 11 22 33 44 55 E0 9400 00 02 00 00 00DD 02 00 00 00 00 C0 A8 2B 2E C0 A8 2B BE A7 73 EB 4B

C. Assoc Request:

00 00 3C 00 00 11 22 33 44 55 00 00 66 77 88 99 00 11 22 33 44 55 E0 0E11 01 0A

00 00 09 56 49 53 48 41 4C 5F 35 47 01 08 8C 12 98 24 B0 48 60 6CDD 03 00 00 00 00 00 00 00 00 00 00 00 00 32 04 C0 A8 2B 2E 4C 21 96 CE

D. Assoc Response:

10 00 32 00 00 00 66 77 88 99 00 11 22 33 44 55 00 11 22 33 44 55 F0 9411 01 00 00 02 C0 01 08 8C 12 98 24 B0 48 60 6CDD 04 00 00 00 00 C0 A8 2B 2E C0 A8 2B BE 35 DE 27 67

After connection data, DHCP data start from 0xDD (Denotes vendor specific command). Here is the DHCP information inside packets which is exchanged between client and server.

- Message Type - 1 (Discover)
  IP Address Known By Client:0.0.0.0
  Client IP Addr Given By Srvr:0.0.0.0
  Server IP Address:0.0.0.0
- Message Type - 2 (Offer)
  IP Address Known By Client:0.0.0.0
  Client IP Addr Given By Srvr:192.168.43.46
  Server IP Address:192.168.43.179
- Message Type - 3 (Request)
  IP Address Known By Client:0.0.0.0
  Client IP Addr Given By Srvr:0.0.0.0
  Server IP Address:0.0.0.0
  Requested IP Address
  Option Code=50
  Option Length=4
  Address=192.168.43.46
- Message Type - 4 (Ack)
  IP Address Known By Client:0.0.0.0
  Client IP Addr Given By Srvr:192.168.43.46
  Server IP Address:192.168.43.179

It simply provides a reduction of 4 frames in the connection process. We have also performed a timing comparison between a normal connection procedure and our proposed procedure. Here is the timing variable used in the calculation.

- T_{MODE} : User Space to Kernel mode delay time
- $T_{INTERFACE}$: Interface Delay
- T_{TRANS}: Wireless Medium Time (Packet air travel time)

First, we will calculate DIFS , between two separate transmissions:

$T_{SIFS} = 10\mu s$ (1)

$T_{SLOT} = 20\mu s$ (2)

$T_{DIFS} = T_{SIFS} + 2 \times T_{SLOT} = 10\mu s + 2 \times 20\mu s = 50\mu s$ (3)

Coming on to the packet, it firstly consists of PHY header further consisting of PLCP preamble (144bits) and header (48bits). Here we have DSSS mode and assumed the packet transmission rate as 1Mbps. Therefore, time to transmit PHY header will be:

$T_{PHY} = (144 \text{ bits})/(1 \text{ Mbps}) + (48 \text{ bits})/(1 \text{ Mbps})$ (4)

$= 192\mu s$

Next up will be the MAC Header which is 24bytes (192bits) which will also transfer at 1 Mbps. Therefore

$T_{MAC} = (192 \text{ bits})/(1 \text{ Mbps})$ (5)

$= 192\mu s$

Now payload will vary according to the packet, so now we will calculate FCS (Frame Check Sequence) which is 4 bytes (32 bits) long

$T_{FCS} = (32 \text{ bits})/(1 \text{ Mbps})$ (6)

$= 32\mu s$

Also after each packet we have ACK (Acknowledgement frame) sent by the receiver. The MAC header of ACK frame is 10 bytes (80bits) long which will take 80µs to transmit. Therefore ACK transmission time will be:

Using equations (4) and (6)

$T_{ACK} = T_{PHY} + 80\mu s + T_{FCS}$ (7)

$= 304\mu s$

As discussed, payload transmission time will be our variable and total time for complete transmission of any packet would be:

$T_{TRANS} = T_{PHY} + T_{MAC} + T_{PAYLOAD} + T_{ACK}$ (8)

Now we are ready to calculate the total time for each packet present in normal and our proposed model. Let us first see the present model. T_{N_TOTAL} is total transmission time in normal scenario (Table 4.2).

Now the proposed model: In our proposed model we have additional bytes to replace DHCP 4 packets exchange. These additional bytes will take additional time and is to be added in the total transmission time of each packet (Table 4.3). T_{P_TOTAL} is total transmission time in proposed scenario.

To calculate the time between user mode & kernel mode we have used Dbgview.exe [81] and using print we got it.

Table 4.2 Transmission time in normal scenario

Payload	**Length (bytes)**	**$T_{Payload}$ (μs)**	**T_{Trans} (μs)**
Authentication Request	48	48	736
Authentication Response	48	48	736
Association Request	200	200	888
Association Response	128	128	816
DHCP Discover	2400	2400	3088
DHCP Offer	2700	2700	3408
DHCP Request	2400	2400	3088
DHCP ACK	2700	2700	3408
T_{N_TOTAL}			**16,168**

Table 4.3 Transmission time in proposed scenario.

Payload	**Length (bytes)**	**Additional Length (bytes)**	**$T_{Payload}$ (μs)**	**T_{Trans} (μs)**
Authentication Request	48	112	112	848
Authentication Response	48	112	112	848
Association Request	200	160	160	1048
Association Response	128	112	112	928
T_{P_TOTAL}				**3,672**

For interface delay calculation assumption is interface equivalent to USB2.0, and practically USB 2.0 speed is around 40 megabytes per second (MBps) [82]. According to this, 112 byte takes around 2.46 μs.

Here is the value:

- $T_{INTERFACE}$ – 2.46
- T_{MODE} – 93 μs

Table 4.4 Default behavior steps

S. No.	Step Detail	Time
1	Connection Request	T_{MODE}
2	Auth Request via station driver	$T_{INTERFACE} + T_{TRANS}$
3	Auth Response via AP driver	$T_{INTERFACE} + T_{TRANS}$
4	Assoc Request via station driver	$T_{INTERFACE} + T_{TRANS}$
5	Assoc Response via AP driver	$T_{INTERFACE} + T_{TRANS}$
6	DHCP Discover from application (Client side)	$T_{MODE} + T_{INTERFACE} + T_{TRANS}$
7	DHCP Offer from Application (Server side)	$T_{MODE} + T_{INTERFACE} + T_{TRANS}$
8	DHCP Request from application (Client side)	$T_{MODE} + T_{INTERFACE} + T_{TRANS}$
9	DHCP Acceptance from Application (Server side)	$T_{MODE} + T_{INTERFACE} + T_{TRANS}$

Total time in all 9 steps = $8* T_{INTERFACE} + 5* T_{MODE} + T_{P\_TOTAL}$ (9)

Table 4.5 Proposed behavior steps

S. No.	Step Detail	Time
1	Connection Request	$T_{MODE} + T_{INTERFACE}$
2	Auth Request via station driver	T_{TRANS}
3	Auth Response via AP driver	T_{TRANS}
4	Assoc Request via station driver	T_{TRANS}
5	Assoc Response via AP driver	T_{TRANS}
6	DHCP Discover from application (Client side)	$T_{INTERFACE} + T_{MODE}$

Total time in all 6 steps = $2* T_{INTERFACE} + 2* T_{MODE} + T_{N\_TOTAL}$(10)

Total time in default behavior from equation 9

= 8*2.46+5*93+16168 = 18463.24 ms

Total time in proposed behavior from equation 10

=2*2.46+2*93+3672 = 4315.56 ms

It is clearly visible that the proposed model saves connection time in the form of – reduced interface & IOCTL delay and change from 8 frame transmission to 4 frame transmission and it is around 1/4rd of default behavior.

4.3 Connection Improvement using low power technology

Same architecture as covered in chapter 3.2 is use to solve connection problem as well. The connection of Wi-Fi is a necessary process. In typical scenarios, everyone will notice the time required for a Wi-Fi connection and the possibility of the success rate of connection. Scanning, key exchange, DHCP negotiation, and Authentication make the Wi-Fi process longer. DHCP process, if happening repeatedly, can cause some good loss of time from the 802.11 STA connection point of view [83,84].

The first connection step is scanning for the desired AP (Access Point) via an STA device. When STA finds its AP, it will initiate the Authentication process. If STA does not support static IP address (mostly time it does not support), DHCP procedure can offload to Bluetooth/BLE interface. Wi-Fi MAC address and other DHCP desired data given to the BT entity. So, after scanning simultaneously, both interfaces will play their role, the device's Wi-Fi radio transmits authenticated packets, and the Bluetooth radio will send DHCP packets to AP.

Usually, BT/BLE exchange of packets can also be possible without link manager connection. However, there should always be radio and baseband link between STA's and AP's BT peers. Once STA Wi-Fi aims to get linked with AP after the scan, it transmits the Auth request. Simultaneously, DHCP settlement also gets started between AP's BT entity and STA's BT entity. So, four packets, i.e., Authentication Request, Authentication Response, Association Request, and Association Response, are taken care of by the Wi-Fi entity and Four DHCP packets, Discover, Offer, Request, and Acceptance (DORA), will be taken care of by BT/BLE entity.

After completion of the DHCP settlement, the BT entity on the STA end shares the DHCP result (IP address or failure reason) with the Wi-Fi entity of STA. If a Wi-Fi connection failed for some reason (Authentication or Association failure), and DHCP

got a pass, Wi-Fi Entity can discard the BT message. In Figure 4.4, the black arrow signifies data through a Wi-Fi radio, and blue dashed arrows show data through Bluetooth/BLE radio. The same color convention will follow in other upcoming figures.

Another problem solved is keep link during power save mode. The connection must be alive with AP in this case, where the WiFi STA device is in power save mode. For a particular time when the device gets into sleep mode, there will be no transmission to AP. There is a particular interval for which AP expects to receive data, after which it imagines that the device is not alive anymore so that it may disconnect in this case. So, to keep this link, it is required that STA should transmit some active packets to AP at regular intervals. To send that keep-alive packet STA device needs to wake up for some time.

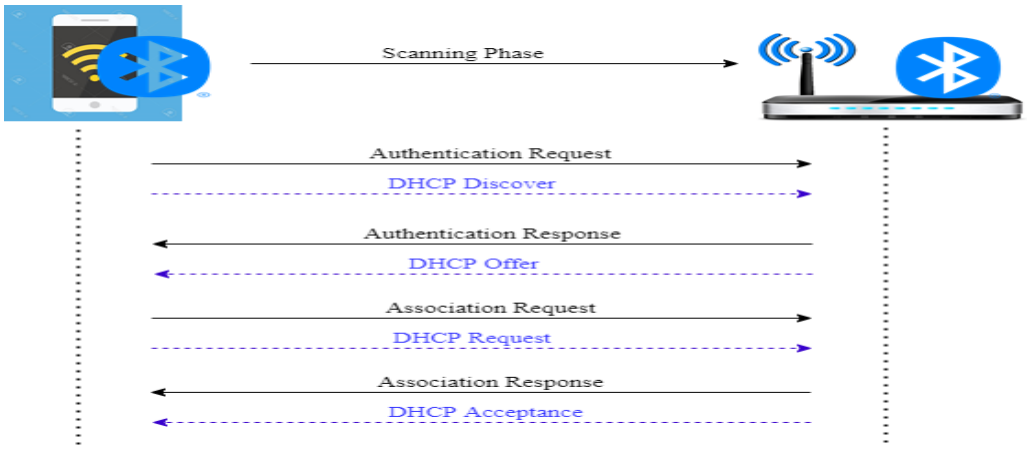

Figure 4.4 BT Entity performing DHCP exchange

STA WiFi instructs to BT entity to continuously confirm activeness to AP, and then STA's BT entity will start transmitting data to AP. Figure 4.5 shows how a keep-alive packet is sent through BT/BLE interface when the WiFi device is in power save mode.

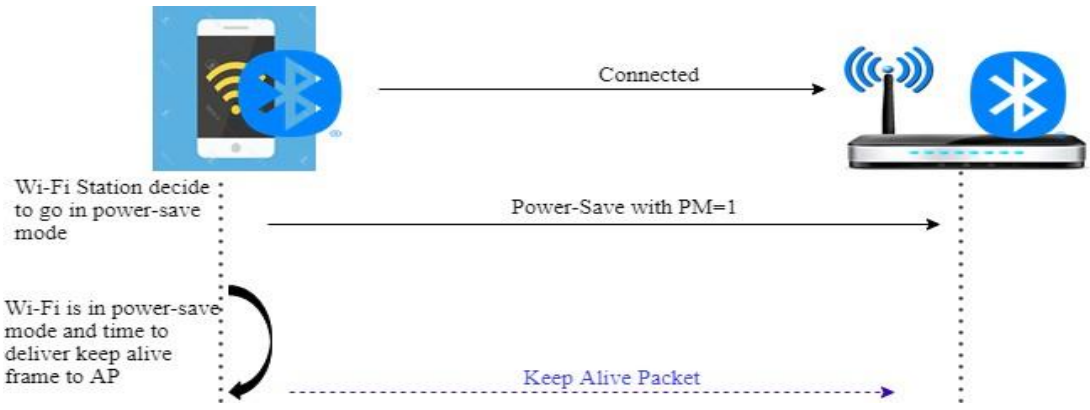

Figure 4.5 Keep Alive Via BT Entity(In Wi-Fi Power Save)

Some other possible way where Bluetooth can help to solve Wi-Fi connection problem.

- Bluetooth can be used to improve Wi-Fi connections by creating a mesh network. In a mesh network, multiple devices act as nodes, extending the range of the Wi-Fi signal and providing more coverage in a given area. This can help to eliminate dead spots, where the Wi-Fi signal is weak or non-existent, and improve overall network performance and reliability.
- Another way that Bluetooth can be used to improve Wi-Fi connections is through the use of a Bluetooth Low Energy (BLE) beacon. BLE beacons can be used to transmit information about the location of Wi-Fi access points, making it easier for devices to connect to the strongest and most reliable Wi-Fi network. This can also help to reduce the time required for devices to connect to Wi-Fi, as they can quickly and easily determine the best network to join.

4.4 Scan Improvement

In start of any Wi-Fi connection scan is the crucial phase, the station always needs to be ready with the latest result. That’s why STA performs background scan even current connection is working fine. The contribution of this sub-chapter is to propose an offloaded low power scanning method to create a fast connection.

4.4.1 System Architecture

The proposed methodology towards this direction is the work on a new amendment to the Wi-Fi scanning process, which introduces a scan-oriented Radio. This radio is an additional interface with extremely low power consumption that is used to perform active and passive on a requirement basis. In contrast, the device's primary radio is to perform another task or switch off. This work describes the IEEE 802.11 scanning mechanism and protocol via secondary radio, discusses its work model, investigates software and hardware dependencies, evaluates different test cases and how much throughput improves via the proposed method.

The proposed System architecture is shown in Figure 4.6. Scan devices have an additional Wi-Fi radio with an extremely low power consumption processor. Main Device has a high clock application processor which performs the main task, and scan

request comes from the operating system or application to the device. Instead of being taken care of via the main device, it offloads to the secondary device.

Scan device has small memory, which saves scan results. Until that time, the main device is in a switched-off state. Results are shared in common memory, which can be accessed via the main device.

The device is operating on channel 36 (5180 MHz). So, the secondary device will scan all channels from the channel list except the operating channel of the main device. In this case, channel 36 will not scan via the secondary device. As the main device already operates on channel 36, it already receives the beacons from other access points (running on the same channel), and it can fill the scan list for channel 36. This way, both devices will not interfere with each other.

In case any disconnection happens, the main device can directly read the scan list, and it can connect to the desired AP. Scan device can serve the purpose of background scanning. When scanning is not required, the scan device can be completely switched off.

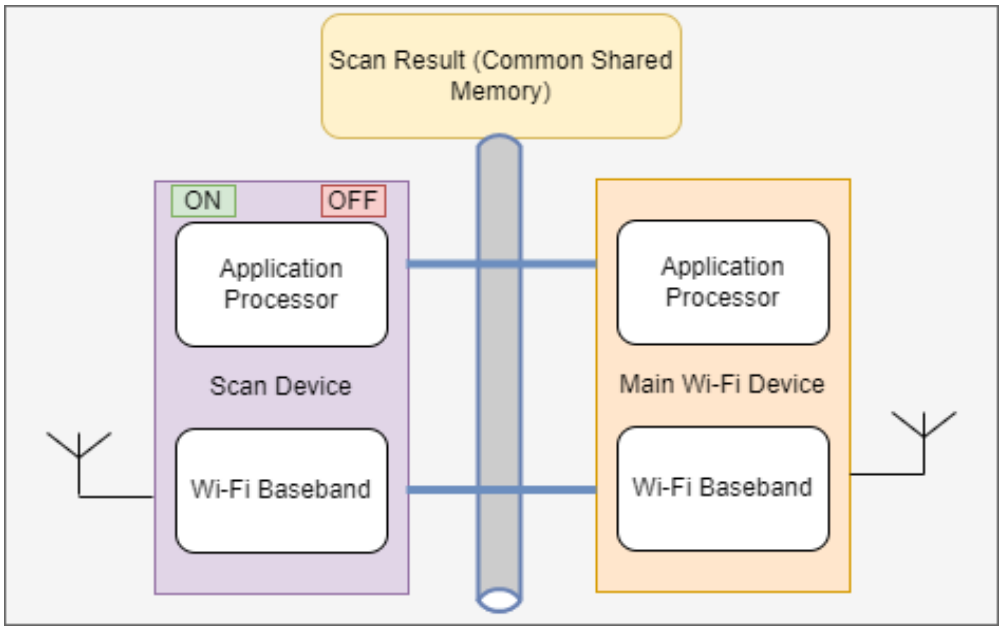

Figure 4.6 Proposed system architecture

4.4.2 Simulation

Existing simulator MATLAB, NS (network simulator) doesn't provide this kind of facility, and no hardware is available to demonstrate this. Own windows device driver-based simulator developed to perform the test. NDIS Miniport [85] driver handles

IOCTL (input/output control) from the application layer. In the driver, two different elements are created. One works as a Main Wi-Fi device, and it talks to the application layer. Other elements work as scan devices, and the Main WI-Fi device only interacts with this element (Figure 4.7).

An application creation which performs an initial and background scan initiates a Wi-Fi connection and transfers some random data.

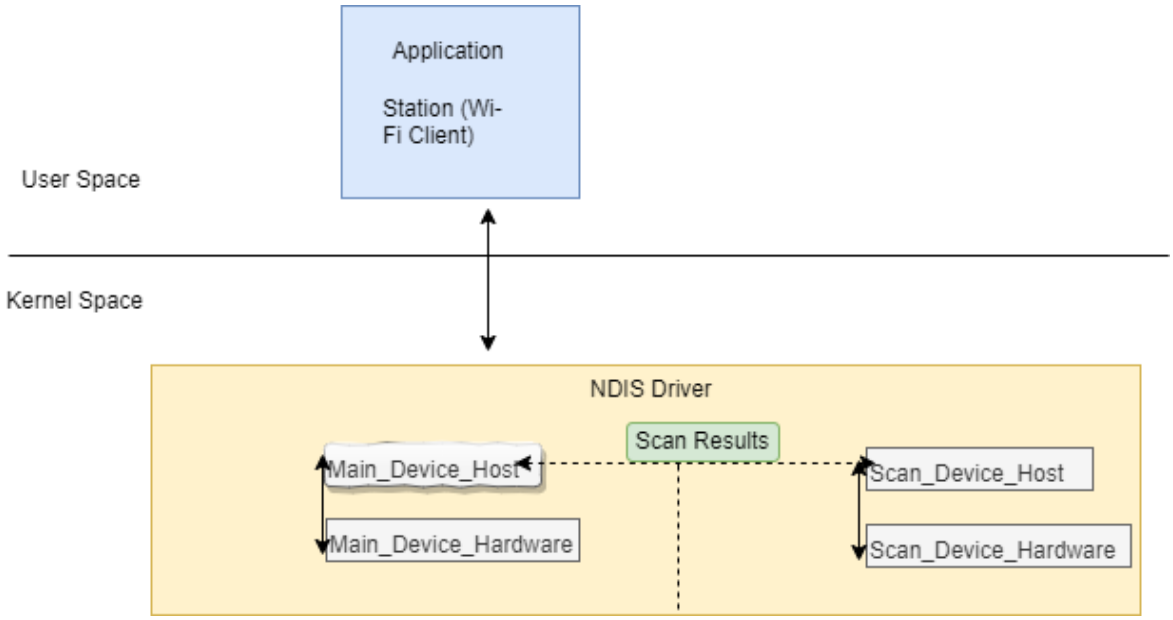

Figure 4.7 Simulator Device Model

Application sent "OID_802_11_BSSID_LIST_SCAN" for scan purposes and other OIDs for a different purpose. The Scan Offload Sequence is shown in Figure 4.9. To perform simulation, application when the main device got the scan OID (Object identification),it first sends radio on request to scan device, and after that scan, request sent to scan device. The scan device performs the scanning, and the scan result sends to the main device, which communicates to the application and the main device sends a radio off command to the scan device. The radio on/Off command simulates the power on/off ofthe secondary device.

Here identify some user scenarios problems where the proposed method can deliver the best performance and optimize the network behaviour:

High Throughput Run: Some high throughput is running on the device, for example, a user is watching a high definition of video. If any scan request comes to the Wi-Fi

device, it can degrade the user experience. Proposed behaviour saves from the above degradation.

We took Dlink DWA-X1850 [86] windows dongle and ran throughput with Netgear 802.11ax AP (Configuration shown in table 4.6) and a scan given every 100 ms via script.

Table 4.6 AP Configuration

Frequency	Mode	Channel	SSID	Bandwidth
2.4 GHz	11n	6	TestNgr	20 MHz
5.0 GHz	11ac	36	TestNgr	80 MHz

It clearly shows how background scans impact throughput, and without scan (offload to another device), throughput can be much better, as shown in Figure 4.8.

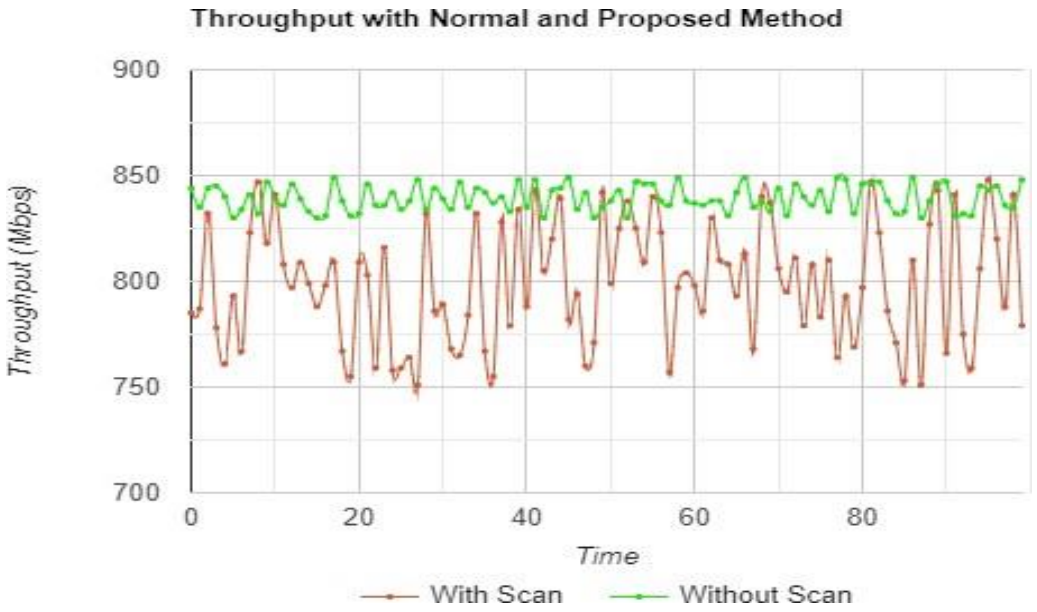

Figure 4.8 Scan impact on throughput

Roaming improvement: The operating system (OS) periodically monitors the RSSI (Received Signal Strength Indicator). If RSSI goes below the defined threshold, the OS asks the Wi-Fi device to roam on another AP. It generally happens in shopping malls and railway stations where many Access points are installed with the same SSID

(Service Set Identifier) and security. So, if RSSI is going down, the application can send a scan request, which can go to the scan device. It can help in roaming performance improvement.

Power-save: The main device took more power than compared to scan device. In the case of 1st device, after offload scan to a secondary device, 1st device can go on low power to save power.

Auto-Channel Selection: The proposed way is beneficial for access point auto channel without impacting throughput on the network. Scan device identifies an operating channel that minimizes interference from other devices without interrupting other tasks. By constantly monitoring the wireless environment and dynamically adjusting the channel in use, autochannel selection can help to prevent congestion, improve network performance, and enhance the overall user experience. For example, if a particular channel becomes congested, the AP can switch to a different channel with less congestion, thereby reducing the risk of network slowdowns and disruptions.

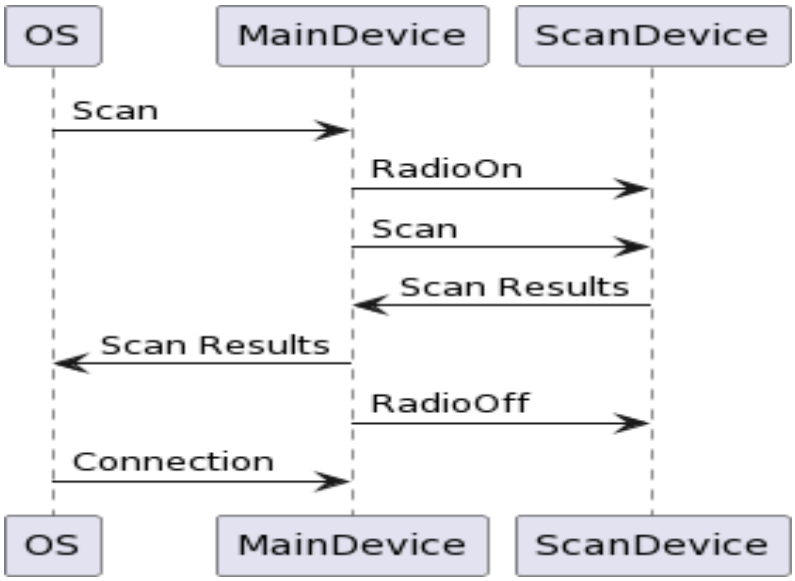

Figure 4.9 Scan offload Sequence

4.5 Summary

In this chapter, we have proposed three optimal ways to solve Wi-Fi connection problem.

1. In this chapter, through simulation and results, a new way is proposed for a fast Wi-Fi connection; DHCP Things moved from application to driver/firmware level and

the number of frames reduced. The proposed solution shown is only for IPV4 but the same can be applied for IPV6 as well. Currently, the IEEE 802.11 specification does not define any of this type of combination frame as described in this section. It is shown that the cross-layer and merging of the frame can significantly improve the Wi-Fi environment. In case of power saving of device, DHCP lease offloading can also be performed using the approach presented in the chapter.

2. In this chapter, we have discussed how low power technology can help to improve and maintain a Wi-Fi connection by multi-radio mechanism. The chapter specifically identified test cases in which low-power technology can help create fast connection, and maintain the connection in a noisy environment.

3. We have shown how scan offload to the secondary device can help to improve 802.11 scanning performance by considering some unique factors that remain unaddressed by the existing scan mechanism. The paper specifically identified test cases in which scan offload can help to improve user behaviour, reduce power consumption, and maintain the connection in a roaming environment. Mechanism and advantages discussed.

CHAPTER 5
CONGESTION AND COLLISION AVOIDANCE

5.1 Introduction

Wi-Fi congestion control is a mechanism used to manage the flow of data in a wireless network and prevent overloading of the network. There are several approaches to Wi-Fi congestion control, including contention-based, low-power technology, and transmission power feedback mechanisms.

Contention-based congestion control involves the use of carrier sensing, which allows devices to detect when the channel is in use, and collision avoidance, which prevents multiple devices from transmitting at the same time. This approach uses the principle of random backoff to resolve collisions, where a device that senses a busy channel waits for a random period of time before trying to transmit again.

Low-power technology like Bluetooth usage is another approach to Wi-Fi congestion control, which involves reducing the power consumption of devices in the network. Bluetooth technology also provides low-power modes, which can be used to conserve power and reduce the amount of interference in the Wi-Fi network. In addition, Bluetooth technology can be used for proximity detection, allowing Wi-Fi devices to automatically adjust their transmission rate or power based on their distance from other devices in the network. Overall, Wi-Fi congestion control using Bluetooth technology offers a unique approach to managing the flow of data in wireless networks, combining the advantages of both Wi-Fi and Bluetooth technologies to achieve optimal network performance.

Transmission power feedback mechanisms are another approach to Wi-Fi congestion control that involve monitoring the transmission power of devices in the network and adjusting it to maintain optimal network performance. This can be done through the use of algorithms that analyze the data rate, error rate, and signal strength of the transmission and adjust the transmission power accordingly.

In wireless networking, 802.11 autochannel selection refers to a mechanism for automatically selecting the best channel for a wireless AP to use. One of

the key benefits of auto channel selection is the ability to control congestion and improve network performance.

Overall, the discussed congestion control mechanisms should be effective for the current application and network requirements. By using a combination of these approaches, network operators can ensure efficient and reliable communication while minimizing congestion and interference.

5.2 Contention Window based collision avoidance

Contention window-based congestion control is a technique used in 802.11 networks to manage network traffic and avoid network overload. It works by having each node in the network maintain a range of random numbers, known as the contention window, which is used to determine the backoff time for that node when attempting to access the shared wireless channel.

When a node wants to transmit data, it waits a random amount of time within its contention window before attempting to access the channel. If the channel is already in use, the node increases the size of its contention window and waits for a longer period of time before trying again. The size of the contention window is adjusted based on network conditions to maintain efficient use of the channel.

For instance, when the network is congested, the contention window size is increased to reduce the number of collisions and prevent network overload. On the other hand, when the network is not congested, the contention window size is reduced to increase the utilization of the channel and improve network performance.

This mechanism is typically implemented using an exponential backoff algorithm, where the size of the contention window is doubled after each collision until it reaches a maximum value. This approach enables the network to quickly respond to changes in network conditions and dynamically adjust the level of congestion control.

In conclusion, contention window based congestion control is a simple and effective solution for managing network traffic in 802.11 networks. By using a random backoff mechanism and dynamically adjusting the size of the contention window, this technique ensures that network resources are used efficiently and network overload is prevented.

5.2.1 Simulation and Results

In this section, we show our simulator setup and simulation results. We used MATLAB WLAN toolbox to simulate the results. We assume all the STA's in the BSS have frames to transmit all the time and we have taken following AIFSN value for each access category (Table 5.1):

Table 5.1 Access Category AIFSN data

Access Category	AIFSN Value
Background data	9
Best effort data	6
Video data	3
Voice data	2

We used below formula to calculate collision rate (Figure 5.1):Collision rate = external collision / total transmission

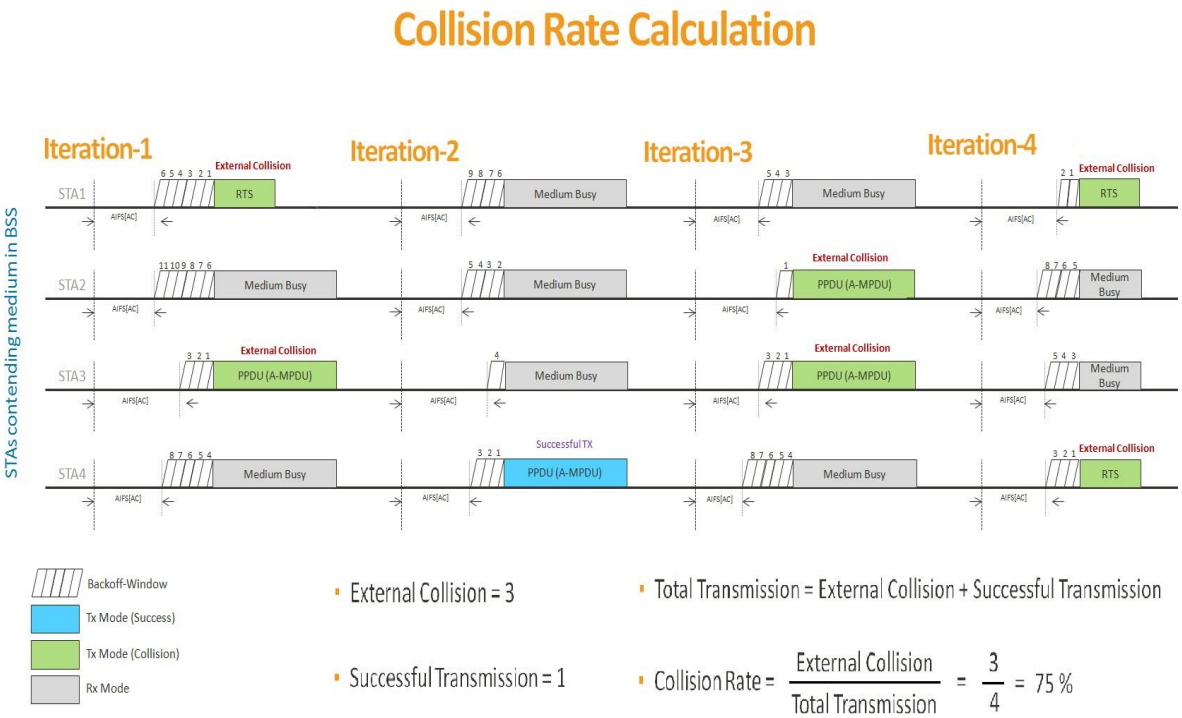

Figure 5.1 Collision Rate

Collision Rate Vs Contention Window [CWmin, CWmax] Relation:

In a Wi-Fi network that employs the CSMA/CA protocol, the contention window determines the duration of time that a station will wait before transmitting a frame after detecting that the channel is idle. As the contention window increases, the collision rate initially decreases due to stations waiting longer before attempting to transmit, which reduces the chances of collisions.

We have simulated Collision rate according to a number of station & with different contention windows on the basis of access categories. Therefore, there is a balancing act between the size of the contention window and the collision rate in Wi-Fi networks. An appropriately chosen contention window size can optimize network performance by minimizing collisions and maximizing channel utilization. Table 5.2 derived from multiple tests using simulation and below graph (Figure 5.2) showing the relationship between them.

Table 5.2 CW With DIFFERENT ACCESS CATEGORY

Cwmin, Cwmax	**AC_BK**	**AC_BE**	**AC_VI**	**AC_VO**
Index 1	15,1023	15,1023	7,15	3,7
Index 2	15,1023	15,1023	7,31	7,7
Index 3	31,1023	31,1023	15,31	7,15
Index 4	31,1023	31,1023	15,63	15,15
Index 5	63,1023	63,1023	15,63	15,31
Index 6	63,1023	63,1023	31,127	15,31
Index 7	63,1023	63,1023	31,127	31,63
Index 8	127,1023	127,1023	63,255	31,63

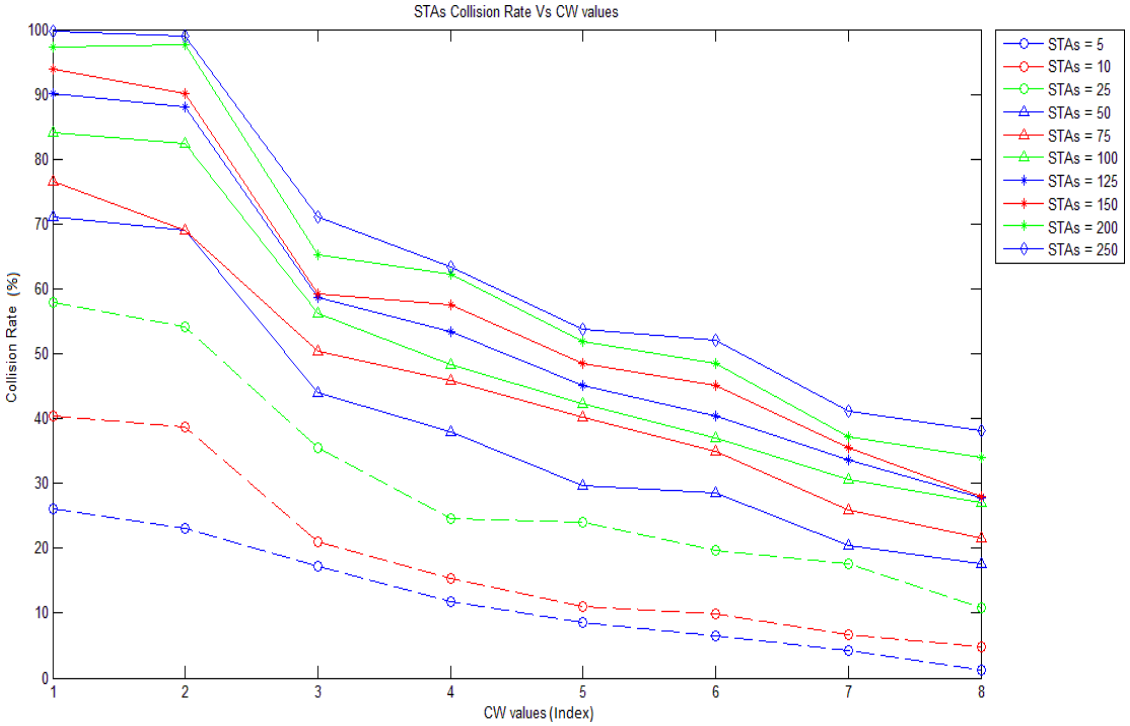

Figure 5.2 STAs Collision Rate vs CW values

Following conclusion derives from the above test:

- As the number of STAs associated with the BSS increases, the collision rate of the BSS decreases, as [CWmin, CWmax] value increases.
- Thus, the AP should increase the [CWmin, CWmax] values to avoid a high collision rate in the highly congested environment.
- As per the Simulation results, the [CWmin, CWmax] values to have collision rate less than 40 % is shown in table 5.3.

Table 5.3 CW FOR DIFFERENT ACCESS CATEGORY WITH THE NUMBER OF STATIONS

Number of Stations	**Cwmin, Cwmax**			
	AC_BK	**AC_BE**	**AC_VI**	**AC_VO**
5	15,1023	15,1023	15,1023	3,7
10	31,1023	31,1023	31,1023	7,15

25	31,1023	31,1023	31,1023	15,15
50	63,1023	63,1023	63,1023	15,31
75	63,1023	63,1023	63,1023	31,63
100	63,1023	63,1023	63,1023	31,63
125	127,1023	127,1023	63,255	31,63
150	127,1023	127,1023	63,255	31,63
200	127,1023	127,1023	63,255	31,63
250	127,1023	127,1023	63,255	31,63

This feedback outlines the relationship between the 802.11 collision rate and the selected contention window. It notes that more stations attempting to transmit at the same time can result in increased collisions, and that the most accurate indicator for selecting the appropriate contention window size is collision rate estimation and the number of stations linked to the access point. However, the number of STAs connected to the access point may not always reflect actual traffic contentions with other STAs, and may not reflect OBSS traffic contention. RTS retry statistics may also be used to estimate the collision rate, as well as adjusting EDCA parameters when the access point announces new settings in the Beacon after connection.

5.2.2 Proposed Work

Figure 5.3 Access Point Sharing information with each other

In an 802.11 wireless network, an AP is responsible for providing wireless connectivity to the clients or stations. The beacon is a management frame that is periodically transmitted by the AP to announce its presence and

provide information about the network. The AP communicates information about the connected stations through information elements, which are fields in a packet that carry specific information (Figure 5.3). The beacon frame can include various information elements that can be used to share information between APs in the same environment. We want Access Point share information about a number of connected stations and active stations to another AP, so we have added an IE (Information Element) in the beacon of the access point, which shows currently connected the station to AP. Current IEEE 802.11 specification, does not have such type of IE. Using this information and above calculation APs can control congestion in the environment.

5.3 Congestion Control using low power technology

Same architecture as covered in chapter 3.2 is use to solve congestion problem as well. Bluetooth technology is a wireless communication standard that can also be used for Wi-Fi congestion control in wireless networks. Bluetooth technology operates in the 2.4 GHz frequency band, which is also used by Wi-Fi, and uses a technique called frequency hopping to reduce interference and increase reliability. Some congestion related sub-problem can solved using proposed approach.

5.3.1 Maintain connection in a highly congested environment

Maintaining an already created connection will become more critical with the increasing demand. WiFi devices at Client-end have less power to transmit than the access point. RSSI value impacts keep-alive time [87]. An alive packet is sent to AP when the client device has no significant transmission between device and AP to prolong connection without fail with the access point.

When there is high traffic in the environment, or it gets very noisy due to other WiFi devices or other noise, it will become difficult for the WiFi client device to send keep-alive information to AP. If any packet does not deliver to AP via STA device in a particular time frame, AP will think that the device is not alive or switched to another AP. As a result, it breaks the established connection with that specific station. So, it is linked to failure cause of congestion [88].

The WiFi STA device will need to send keep-alive information to the Access point to establish a connection. Suppose there is high noise in the network and WiFi STA cannot deliver it to AP. In that case, the WiFi STA entity asks to transmit a packet with the

help of BT radio to AP's BT entity, which notifies AP's connection maintainer that particular STA is alive. Figure 5.4 shows how a keep-alive packet is sent through BT/BLE interface. As discussed in architecture, both radios share the same shared memory. This keep-alive time maintained (Software or Hardware based) via WiFi entity can update via Bluetooth entity.

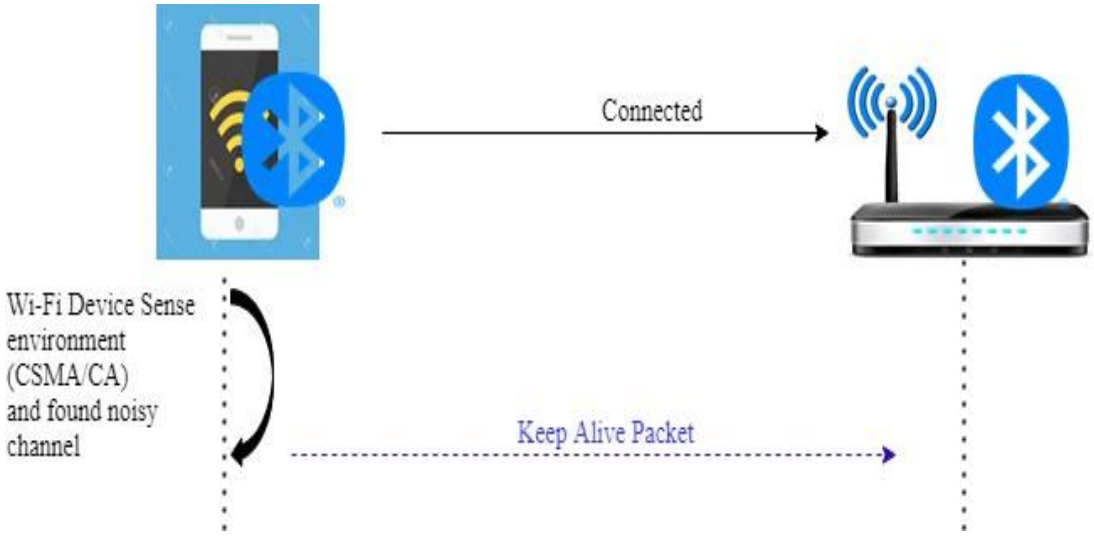

Figure 5.4 Keep Alive Via BT Entity (In congested environment)

5.3.2 Keep link during power save mode

The connection must be alive with AP in this case, where the WiFi STA device is in power save mode. For a particular time when the device gets into sleep mode, there will be no transmission to AP.

There is a particular interval for which AP expects to receive data, after which it imagines that the device is not alive anymore so that it may disconnect in this case. So, to keep this link, it is required that STA should transmit some active packets to AP at regular intervals. To send that keep-alive packet STA device needs to wake up for some time.

STA WiFi instructs to BT entity to continuously confirm activeness to AP, and then STA's BT entity will start transmitting data to AP. Figure 5.5 shows how a keep-alive packet is sent through BT/BLE interface when the WiFi device is in power save mode.

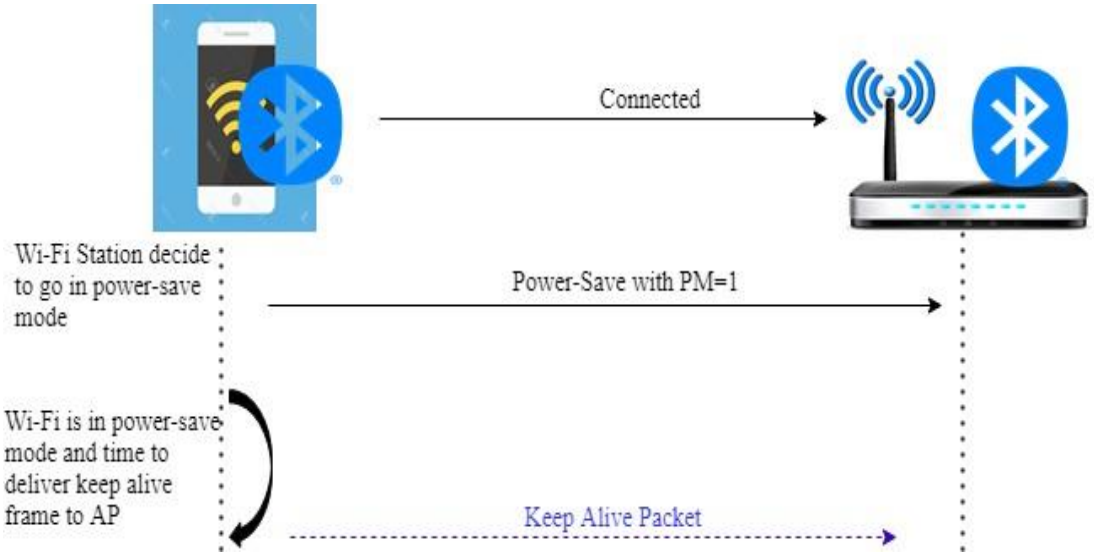

Figure 5.5 Keep Alive Via BT Entity (In WiFi Power Save)

Wi-Fi congestion control using Bluetooth refers to a solution where Bluetooth is utilized to manage congestion on a Wi-Fi network. This can be achieved through various techniques, including offloading data from the Wi-Fi network to the Bluetooth network, and using Bluetooth to prioritize and allocate bandwidth to different devices on the Wi-Fi network.

One of the ways that Bluetooth can be used for Wi-Fi congestion control is through the use of Bluetooth Low Energy (BLE) beacons. BLE beacons can be used to transmit information about the current state of the Wi-Fi network, such as its available bandwidth, to devices connected to the network. This information can then be used by the devices to determine how much data they should transmit over the Wi-Fi network, reducing the likelihood of congestion.

Another way that Bluetooth can be used to manage Wi-Fi congestion is by enabling devices to form ad hoc networks using Bluetooth. This allows devices to communicate with each other directly, without having to rely on the Wi-Fi network. This can be particularly useful in situations where the Wi-Fi network is congested, as it provides an alternative means for devices to communicate, reducing the load on the Wi-Fi network.

In conclusion, the use of Bluetooth for Wi-Fi congestion control provides an innovative and effective way to manage congestion on Wi-Fi networks, making it a valuable tool for organizations and users looking to optimize their Wi-Fi networks and improve the overall user experience.

5.4 Tx Power Feedback Mechanism

In 802.11 networks, if other devices provide feedback about the transmit power of a sender device, this information can be used to reduce congestion. The feedback from other devices provides information about the impact of the sender's transmission on the network, which can be used to adjust the transmit power of the sender in real-time.

For example, if the feedback indicates that the sender's transmission is causing interference with other devices in the network, the sender can reduce its transmit power, which will reduce the range of its transmission and reduce the likelihood of interference with other devices. This can help to reduce the number of collisions in the network and improve network performance.

In addition, by receiving feedback from other devices, the sender can dynamically adjust its transmit power to account for changes in network conditions. This can help to ensure that the network operates efficiently, even in dynamic and rapidly changing environments.

In summary, feedback from other devices in 802.11 networks can be used to reduce congestion by providing information about the impact of a sender's transmission on the network. This information can be used to adjust the transmit power of the sender in real-time, which helps to reduce interference and improve network performance. To provide this information we have proposed a protocol mechanism which can use by other Wi-Fi device to feedback sender device about transmit power.

Proposed Protocol:

In the proposed system, we assume DUT will send N number of packets to the instrument device, and in response, the instrument device will send a specific Wi-Fi packet that can transmit power and other details. BA (Block ACK) frame can also be used to acknowledge transmit power details. But there is a limitation, BA (Block ACK) frame was introduced in the 802.11n standard while the Wi-Fi device still needs to support the legacy standard (802.11 a/b/g). Because of this limitation, a customized Wi-Fi data frame is proposed instead of using an 802.11 standard BA (Block ACK) frame. In 802.11 standard data frame type, subtype – 0x0D (Binary 1101) is a future reserved value. We use this subtype value 0x0D to define the proposed frame (Figure 5.6), called the "Tx details ACK frame."

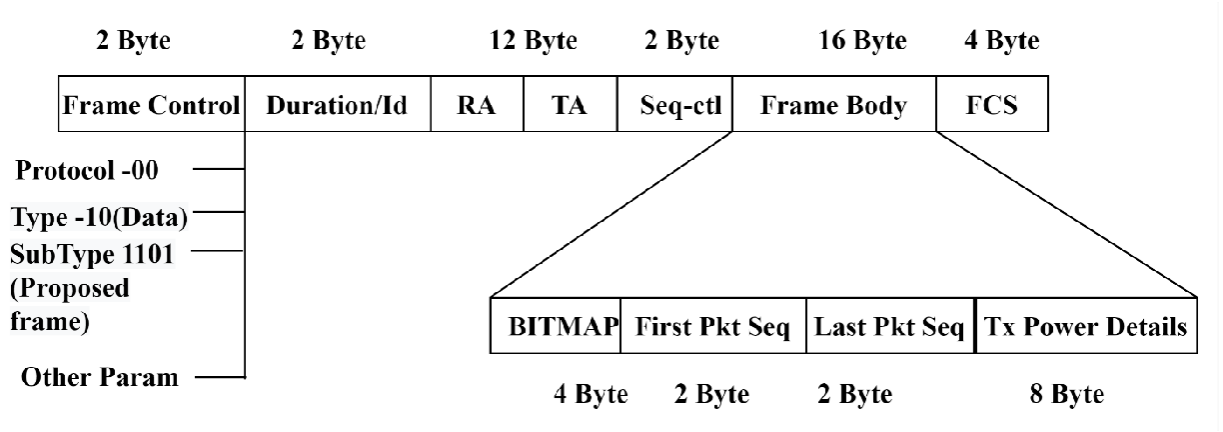

Figure 5.6 Proposed Tx details ACK frame.

The Figure 5.6 shows the proposed Tx details ACK frame (MAC header). The 802.11 MAC (Media Access Control) header frame control is a critical component of the 802.11 protocol that is used to control access to the wireless network. It is a 2-byte field in the 802.11 MAC header that provides information about the type of frame being transmitted, the transmission mode, and other important parameters. In frame control, subtype value "1101" tells "it is the special frame for DUT." It will consist of a 16-byte frame body and include below fields:

- Bitmap field gives the count of acknowledgment of packets. Its size is 4 bytes, so up to 128 packets can be acknowledged as every bit represents one packet. Hence, the count of set bit equals the number of acknowledged packets, and the sequence of set bit numbers denote the sequence number of acknowledged packets.
- The first packet sequence contains the starting sequence number of the received packet.
- The last packet sequence has the last sequence number.
- Tx power details include Tx power-related data. It is the variable field; it can increase according to Tx power parameters.

For example, DUT sends six packets (Sequence number 0-5) to the testing instrument, and seq-3 and seq-4 are not received correctly. The average of received packets' transmit power is 16.1 dBm. Following will indicate in Tx details ACK frame body (Figure 5.7).

- Bitmap – 00000000 00000000 00000000 00110011
- First Pkt Seq – 0
- Last Pkt Seq – 5
- Tx Power Details - 00010000 00000000 11001100 (Only 3 bytes shown, 1st byte contains an absolute number, and 2 bytes contain after decimal data, other 5 bytes can used for others Tx power params).

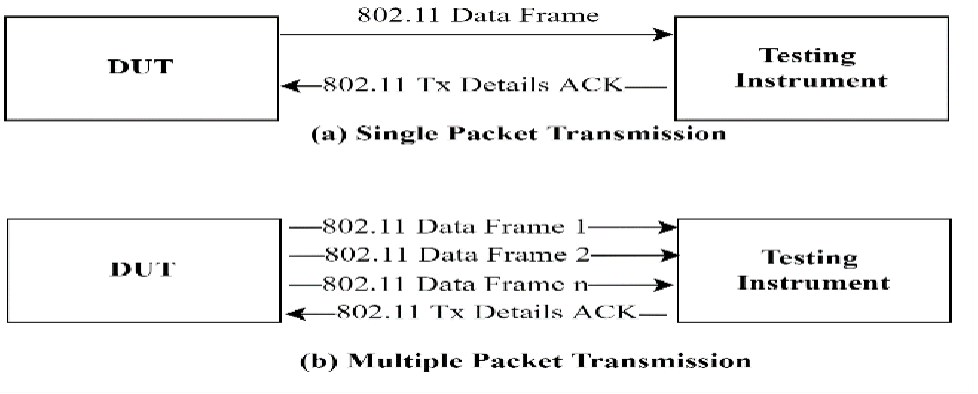

Figure 5.7 Packet transmission and ACK with transmit power parameters (a) Single packet transmission (b) Multiple packet transmission

The goal of using optimal transmit power is to minimize the number of conflicts in the graph. This can be modeled as an optimization problem where the objective function is the number of conflicts, and the transmit power of each node is a decision variable. The optimal transmit power for each node can be found by solving this optimization problem subject to constraints such as maximum transmit power and minimum signal-to-noise ratio.

Mathematically, the problem can be formulated as follows:

minimize C = ∑_i∑_j p_i * p_j * f(d_ij), where C is the number of conflicts, p_i and p_j are the transmit powers of nodes i and j, d_ij is the distance between nodes i and j, and f(d_ij) is a function that represents the conflict relationship between nodes i and j (1 if they interfere, 0 otherwise).

subject to: p_min <= p_i <= p_max, for all i, where p_min and p_max are the minimum and maximum allowed transmit powers.

This optimization problem can be solved using techniques such as gradient descent, conjugate gradient, or simulated annealing, to find the optimal transmit powers that minimize the number of conflicts.

Suppose a sender is transmitting data to a receiver and there are two other devices (A and B) in the same network that are also receiving the signal. The sender uses TPC and receives feedback from devices A and B indicating their transmit power.

If device A is closer to the sender and reports a high transmit power, the sender can reduce its transmit power so as to not interfere with device A. Similarly, if device B is farther away and reports a weaker transmit power, the sender can increase its transmit power to maintain the desired signal quality for device B.

By using TPC, the sender can dynamically adjust its transmit power to reduce interference and optimize network performance, leading to reduced congestion in the network.

In 802.11 using above protocol can solve several problems including:

- Hidden node problem: By reducing the transmission power, the range of each node's transmission is limited, reducing the likelihood of two nodes interfering with each other, thereby reducing the hidden node problem.
- Interference: By controlling the transmission power, it is possible to reduce the amount of interference between nodes in the network, improving overall network performance.
- Power consumption: By reducing the transmission power of each node, the power consumption of the nodes can be reduced, leading to increased battery life for devices that rely on batteries.
- Overlapping coverage areas: By controlling the transmission power, it is possible to avoid overlapping coverage areas, reducing the number of collisions and improving network performance.
- Channel utilization: By controlling the transmission power, it is possible to increase the utilization of the available channel bandwidth, leading to improved network performance.

- Signal-to-noise ratio: By controlling the transmission power, it is possible to maintain a minimum signal-to-noise ratio, ensuring that transmissions are received accurately.

5.5 Summary

In summary, Wi-Fi congestion control is an important aspect of wireless network management that helps prevent network overload and ensure reliable and efficient data transmission. The various approaches to Wi-Fi congestion control, including contention-based, low-power technology, offload scan auto channel selection (Chapter 4.4.2) and transmission power feedback mechanisms, each have their own advantages and trade-offs, and the choice of approach depends on the specific requirements of the network.

CHAPTER 6
APPLICATIONS AND INTERMITTENT ISSUE

6.1 Introduction

Highlighting and considering the potential applications of the research work canprovide a deeper understanding of how the results of the research can be applied in real-world scenarios. This can help to increase the impact and relevance of the research workand provide practical solutions to real-world problems.

Similarly, It is also important to acknowledge any challenges or limitations encountered during the research process. This can provide valuable insights into the difficulties faced, and help to build credibility by demonstrating transparency about the research process. This can also help to identify areas for future research. If the research work is focused on the reliability or performance of a system, then intermittent issues may be a critical component to consider.

"This chapter covers one important application and one intermittent issue that is solved during the research work. The proposed mechanism, 'Tx Power Feedback Mechanism', is discussed in chapter 5.4 and can be applied in the 802.11 assembly line. It is further discussed in chapter 6.2. Additionally, the intermittent issue of firmware over-the-air updates for wireless IoT devices is discussed in section 6.3."

6.2 802.11 Wireless devices assembly line process improvement

Improving the assembly line process for 802.11 wireless devices can increase efficiency, reduce production time and cost, and improve product quality. This subchapter discusses all aspects of the assembly line and explains how our proposed method can help improve the assembly line process for 802.11 wireless devices.

6.2.1 Purpose

With this exponential growth of devices, the assembly lines and manufacturing units are heavily overloaded. More and more People want a working device in the market as soon as possible. The most critical feature in the manufacturing line is calibration and testing of the Wi-Fi module to provide desired performance. The purpose of this study

is to propose an approach to save a significant amount of time for Wi-Fi devices at the assembly line. The method describes how wireless testing equipment can provide more help in the calibration and testing of the wireless device. Use of Wi-Fi Capability at testing instrument side and use of proposed 802.11 Tx details ACK (Acknowledge) frame in the response of DUT's (Device under test) data frame can help test Tx and Rx simultaneously. MAC address calibration is also discussed using the wireless capability approach. Use of Tx Power feedback mechanism under chapter 5.4 is used to solve Wi-Fi calibration process problem under assembly line. The sub-chapter describes wireless device calibration and testing results from environments where very little open-literature data is available, especially organizations (device developers) who want all this information to be secret. The proposed method is novel since it proposes a new 802.11 protocol and uses the testing instrument's signal generation capability to overcome challenges. It is easy to adapt to an assembly line. The technology proposed in this chapter refers to Wi-Fi devices, although it could be intuitive for other wireless technologies such as Bluetooth, UWB (Ultra-wideband), Zigbee, etc.

6.2.2 Assembly line testing introduction

At the assembly line, quality assurance is performed to check whether the device is ready to go on the market or not. Quality parameters can be receiver (Rx) sensitivity, transmit error, transmit power, or other writable data (MAC address or other parameters). Calibration is the act of writing correct values on a device under test (DUT) with a reference standard of a known value. The calibration algorithm dependson the vendor, and it can vary in terms of effectiveness, impact, and complexity.

When a new device comes to the assembly line, first, the transmit and receive test is performed. If both tests are successful, MAC (media access control) address writing is performed on the device [89].

6.2.2.1 Tx calibration test introduction

Limited and controlled transmit power is the critical requirement for wireless transmitting devices for regulatory purposes, and it also needs to co-exist with other wireless technologies [90]. At the manufacturing time, the manufacturer calibrates the transmit power at the factory [91].

Typically, Wi-Fi device transmitters contain power amplifiers to control output power [92]. It uses power written in a specific control register. Calibration of transmit power means mapping the value of the power control register with the actual power value of the device. The transmit power is measured in dBm by testing equipment like NI (National Instrument), Agilent, and Litepoint Wi-Fi tester.

A power table has a mapping of frequencies/channels with power, which can be used by the DUT (Device under test) to determine the power control value used to transmit the signal at a specific power level. Apart from channels, different modes (11a, 11b, 11n, or 11ac), different bandwidths (20 MHz, 40 MHz, and 80 MHz), and different antennas can also be part of this table, which impact transmit power level [93]. These values are written into read-only memory (EEPROM) on the device after the completion of transmitter calibration, which is used by the devices in a usable environment to generate accurate power levels. Tx calibration can be done using carrier wave (CW) tones (only carrier without modulation) or with the continuously modulated signal [94].

6.2.2.2 Rx sensitivity test introduction

Receiver (Rx) sensitivity term means minimum achievable RSSI (received signal strength indication) to obtain a threshold PER (packet error rate). The IEEE 802.11 standard [95] specifies a minimum Rx sensitivity that all 802.11 devices must achieve. For example, achieved Rx sensitivity -87 dBm means the device can decode and understand this power level signal. The receiver's sensitivity is tested by sending different numbers, and power levels of packets from other devices to DUT and checking how sensitive DUT is towards that.

6.2.2.3 MAC address writing introduction

When a device is in the production line, MAC address writing is a part of the calibration process. Customer needs MAC address over-writing during production [89]. In the final stage, after a device has achieved its wireless certificate and all Tx/Rx tests pass, the MAC address is written into ROM memory.

ML models have broad applicability to Wi-Fi topics. Table 6.1 concludes the corresponding Wi-Fi-related feature problem, solved via ML models. A survey paper on Wi-Fi with machine learning [96] covers almost all areas of Wi-Fi. But it does not

cover the assembly line process because Tx calibration and Rx sensitivity check are part of the RF (Radio frequency) chain. RF chain is a cascade of electronic componentsand sub-units that may include amplifiers, filters, mixers, attenuators and detectors [97].Due to the RF chain's diverse hardware characteristics (such as Tx power, Rx Sensitivity) [96] and faultiness of any electronic component, running learning models will be very computationally expensive. Lack of data is also a reason for not consideringthe machine learning model for the proposed method.

Table 6.1 Existing surveys concerning Wi-Fi related problems and ML models

Main Scope	**Address Feature**
Large-scale network monitoring [98]	Wi-Fi analytics
Quality indicators accounting for user satisfaction [99]	Wi-Fi quality indicators
Indoor localization [100,101,102,103,104]	Application-oriented
Human activity detection [105]	
Intrusion detection [106,107]	Wi-Fi security
Coexistence mechanisms [108,109]	Coexistence of 5G and Wi-Fi
Deep learning research in wireless networking [110]	Wi-Fi networks signal processing

Our contributions are fivefold: Understand the Wi-Fi device calibration process, reduce calibration & test time under assembly line, propose an 802.11 protocol to deliver Tx Power as return value, simulation of convolution & proposed method, discussion of other applications with the proposed protocol and perform a comprehensive practical evaluation of our proposed method. We now describe these contributions in detail.

A. Wi-Fi device calibration process: An in-depth review of academic and organization's white papers done in the related field. We provide readers with an overview of how convolution Wi-Fi device calibration has been done and what the area can focus on particular Wi-Fi device problems. Simulation of convolution and proposed method performed better to understand the natural environment of Wi-Fi devices assembly line.

B. Reduce calibration & test time: Wireless technologies are solving a significant problem of assembly lines [26], but wireless devices also need help under assembly lines so that wireless devices can come early to market. IEEE specification does not provide any specific protocol which can give transmit power and related parameters in return. In this work, a protocol mechanism introduces to perform the same. The proposed method takes the help of the 802.11 protocol mechanism in the assembly line as much as possible and shows MAC address writing, and other calibrations can also perform wirelessly. To save the assembly line time, perform Tx calibration and Rx sensitivity test in a single combined test.

C. Assessment of the experiment: The proposed method experiment is performed in the simulation environment with real hardware. With the help of experimentation, it is shown how much time the proposed method can save during the Wi-Fi assembly line process. All major and minor Wi-Fi details are covered under the experiment considering all environment scenarios like operating system behaviour, setup complexity etc.

D. Identify other applications where the proposed protocol advantage can inherit [111]. A different section 6.2.4.4 discusses it in detail. Additionally, we have shown how the proposed method reduced complexity and dependency on the setup side.

E. Future research directions are provided about improving the Wi-Fi devices assembly line process to provide readers with an analysis of what remains to be done in the field.

6.2.3 Methods & Materials

This work covers calibration and testing aspects of the wireless device under the assembly line and maps the actual research field in the manufacturing environment. To evaluate the actual state of wireless device manufacturing and identify the improvements point, the conventional method [112] is simulated in the lab. Simulation has been conducted for the following steps: Tx power calibration, Rx sensitivity test, and MAC address calibration. Sequential execution of the steps described aboveensures reproducibility and scalability of the study and the objectivity of the results.

Below are the steps performed at the manufacturing line with the fresh (uncalibrated) device, and the sequence of steps is shown in Figure 2:

- Calibration
- Verification
- Go/ NoGo

6.2.3.1 Components of Wi-Fi assembly line calibration environment

Manufacturing site testing and calibrating Wi-Fi RF is a significant part. Different components and software/tools are required to operate it. Below various components under assembly line calibration discussed (Figure 6.1) :

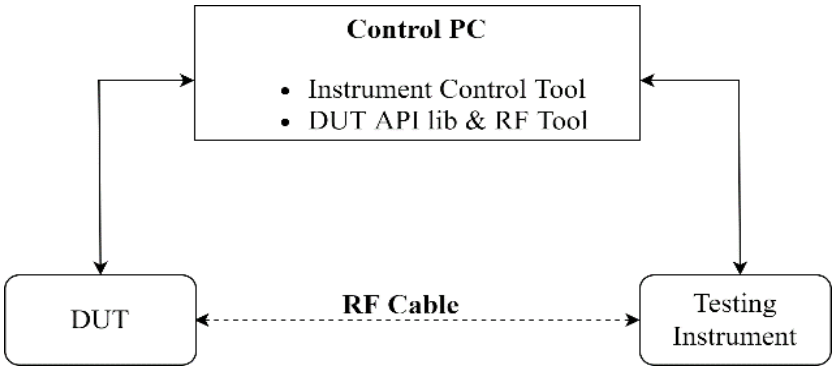

Figure 6.1 Assembly line calibration environment components

DUT - A device under test (DUT) is the final product that is going through the testing process. Functional testing and calibration were done on it.

Test Equipment - To verify DUT's Tx power, Rx sensitivity test equipment is used, which includes RF signal, spectrum analyzer, and the same type of signal generator. It measures Tx power very accurately, precisely and provides interface software to control its behavior.

Control PC - A system that controls DUT as well test equipment. All the software runs on this machine (Table 6.3), whether it is a tool to configure, test RF of DUT, and send/receive any command to the testing instrument.

In most cases, the testing instrument is connected to the system by ethernet, as ethernet is faster than other interfaces. DUT and testing instruments are connected by RF cable, which gives the feel of a wireless environment and saves the environment from external

noise. The open environment can be prone to external noise, lots of Wi-Fi devices & same frequency device can operate in the same environment, so RF cable provides more help in this condition. The use of RF cable makes it necessary to calculate path loss between the transmitter and receiver antenna elements in the line of sight path.

6.2.3.2 Wi-Fi chip configuration and calibration

Wi-Fi chip RF and other configuration information and calibration data are written into EEPROM memory, and it can be written a limited number of times. Chip vendor provides a guideline about which offset of EEPROM consists of what type of information. The table 6.2 shows examples for different configurations written in an EEPROM memory (according to offset).

Table 6.2 EEPROM configuration example

Name	Offset	Details
Device Product Id	0x00	Product id of the device
Device Vendor Id	0x02	Vendor id of the device
MAC address	0x10	Wi-Fi MAC address
TX0_POWER_2.4	0xAA	2.4G Hz frequency power value
TX0_POWER_DELTA_2.4	0xAB	2.4G Hz frequency power delta value
TX0_POWER_5.0	0xCD	5G Hz frequency power value
TX0_POWER_DELTA_5.0	0xCE	5G Hz frequency power delta value

6.2.3.3 Conventional method simulation

To simulate the conventional test, an azure sphere (MT3620) IOT board is taken as DUT and IQXEL Litepoint device as testing instruments [113]. MT3620 board is connected through a USB interface with a window machine, and the IQXEL device is connected through ethernet with the same machine (Figure 6.2). We must run a transmit test and the received test on the DUT. SCPI (Standard Commands for Programmable Instruments) are used to command the testing instrument. Microsoft's Rftool [114] utility is used to control the azure sphere board. RF cable & connector path loss is calculated using IQXEL's VSG & VSA module, and it is taken into account during testing. Before connecting DTU, one end of the RF cable connects to VSA & the other

end connects to VSG. Equipment transfer & receiving the same signal and power difference result in path loss.

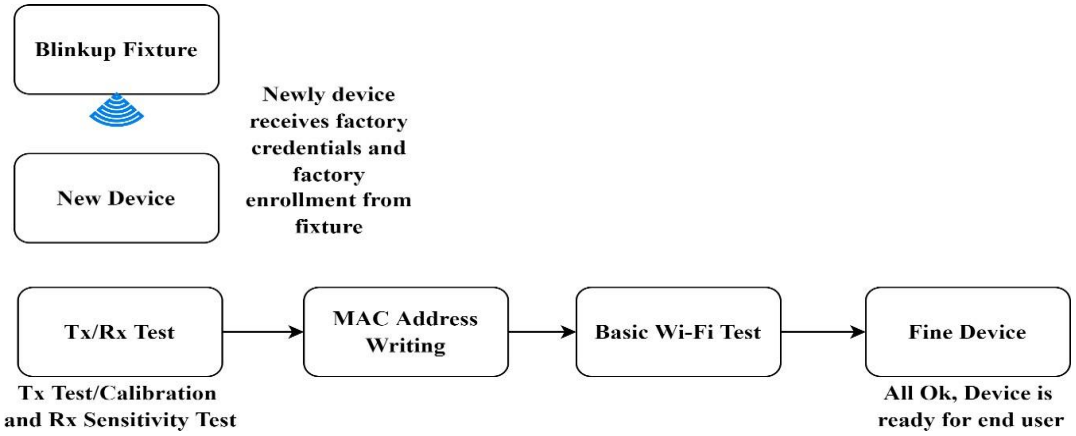

Figure 6.2 Assembly line Wi-Fi device calibration and test

Table 6.3 Software/Tools Used under Assembly line calibration environment

Software/Tools	Provider	Uses
DUT API lib & RF tool	DUT Chip Vendor	Users can program RF settings, e.g., select channel (frequency), antenna configuration. Perform Tx and Rx operation, and Write final settings in e-fuse (EEPROM), which gives the device optimal performance. Help to verify that the radio configuration like transmit power, MAC address, country code, and other settings are correctly written into ROM or not. Example - radiofrequency tool (rftool) [114]
Instrument control tool	Instrument developer	Control instrument behavior. Enable VSA, VSG and perform operation. Example – Iqfact+ [115]

6.2.3.3.1 Tx Test

Figure 6.3 shows the simulated setup for the Tx test. The first step is to select a channel on which DUT transmits, so channel 11 is chosen. rftool specific command used on the

DUT side and SCPI command on the test instrument side. A signal analyzer is enabled on the testing instrument for transmitting tests.

A developed tool is used to send commands to DUT and testing instruments.

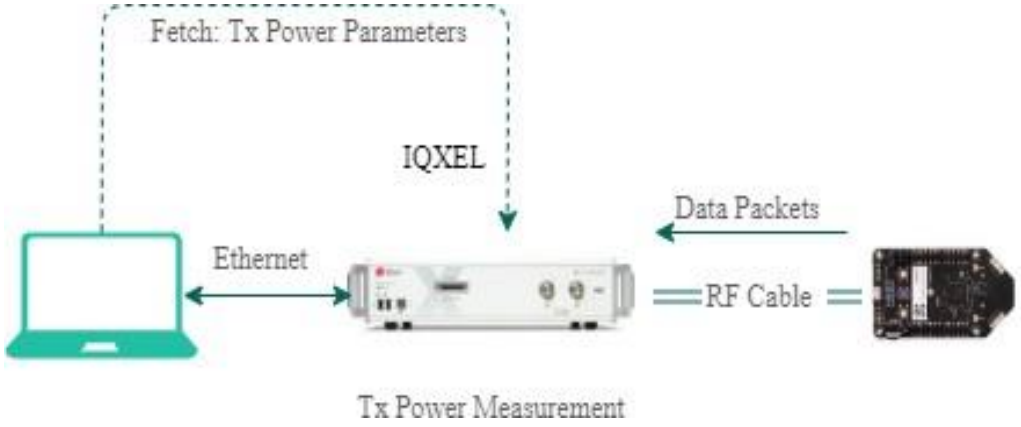

Figure 6.3 Tx power measurement simulation setup

Simulated RF tool commands and SCPI commands are shown below. In this example, the WRITE_DUT macro sends the command to DUT on the command-line interface and the WRITE_TIU macro uses it to send the command to the testing instrument via TCP socket.

APIs to test Tx test:

```
WRITE_DUT antenna 0

WRITE_DUT transmit frame -BSS 11:22:33:44:55:66

WRITE_DUT transmit frame -SOURCE 66:55:44:33:22:11

WRITE_DUT transmit frame -  Destination 11:22:33:44:55:66

WRITE_DUT transmit frame -FrameControl 8841

WRITE_DUT transmit frame -Duration 2000

WRITE_DUT channel 11

WRITE_DUT tx start

// SET VSA FOR CAPTURE

WRITE_TIU VSA1;WRITE_TIU VSA1;FREQ;LOOF 0

WRITE_TIU VSA1;TRIG:SOUR VID

WRITE_TIU VSA1;FREQ %5%0000000

WRITE_TIU VSA1;RLEV:AUTO:TIME 0.05

WRITE_TIU VSA1;RLEV:AUTOs

WRITE_TIU VSA1;SRAT 160000000

WRITE_TIU VSA1;TRIG:LEV -25

WRITE_TIU VSA1;TRIG:OFFS -0.00001

WRITE_TIU VSA1;CAPT:TIME 0.0500
```

```
WRITE_TIU VSA1;INIT

WRITE_TIU 2.4GHz Pathloss 2 2

WRITE_TIU *WAI

//WLAN ANALYSIS AND FETCH FOR TX POWER

WRITE_TIU WIFI;CLE:ALL

WRITE_TIU WIFI;CONF:STAN OFDM

WRITE_TIU WIFI;CONF:OFDM:TRAC:PHAS ON

WRITE_TIU I WIFI;CONF: OFDM:CEST LTF

WRITE_TIU WIFI;CONF:OFDM:TRAC:SCL ON

WRITE_TIU WIFI;CALC:POW 0.3

WRITE_TIU WIFI;CALC:SEGM1:TXQ 0,3

WRITE_TIU I WIFI; FETC:SEGM1:OFDM:SFL: SIGN1:AVER:MARG?

//WLAN ANALYSIS AND FETCH FOR TX SPECTRUM

WRITE_TIU WIFI;CLE:ALL

WRITE_TIU I WIFI;CONF:STAN OFDM

WRITE_TIU WIFI;CONF:OFDM TRAC: PHAS ON

WRITE_TIU WIFI;CONF:OFDM:CEST LTF

WRITE_TIU WIFI;CONF:OFDM:TRAC:SCL ON

WRITE_TIU WIFI; CALC SEGHI:TXQ 0 3 WRITE ICE1 FOR PC

WRITE_TIU WIEL FETCI SECONOMS51 SIGNI SAVER:MARG?

WRE TOP ECHO 1

WRITE_TIU WINE TETO SEGML OTOM:SELI:SICNI AVER:MARGLOTON?

READ TO ECHO 1

WLAN ANALISIS AND FETCH FOR TX SPECTRUM

WRITE ICP 1 WEEK.CLEAR ALL

WRITE_TIU HIFI. CONF STAN OFDH WRITE_TOPILTE ODNESDHET SHAS

WRITE TCP 1 WIFI.CONF OF CHEST LIE

WRITE_TIU

WRITE_TIU WIFI:CALic: SPEC S

WRITE ICD 1 PO

WRITE_TIU LEICESPECIANER MARGO READ ICE ECHO 1

WRITE_TIU TO SPEC AVER ARG FOR? DEAD TOP ECHO

WRITE_DUT tx stop

WRITE_DUT config write MAC Address 00:11:22:33:44:55

WRITE_DUT config write data 0x5B 0x02
```

Rftool also provides EEPROM (E-fuse) offset writing. When the target value is not achieved via DUT, delta offset [116] is used to plus/minus transmit power value. Suppose 13dBm value written to offset and when DUT transmit signal actual power

calculated at instrument side is 12dBm, so +1 will write into delta offset to match correct transmit value written at Tx power offset. Above one is a response-based final calibration method performed for wireless devices. To calibrate Tx power, the "WRITE_DUT config write data 0x5B 0x02" API is used to write delta offset for 2G high channel (Channel 11). After performing this operation, we can see one dBm higher Tx power as output. According to the guideline delta unit is 0.5dB per step, so write value 0x02 gives 1db positive increment at power level.

Here is the output data from the testing instrument device (also can see on the IQXEL webpage interface).

- Peak power
- Avg power (all)
- EVM
- Amplitude Imb
- Phase Imb
- Data rate
- Frequency error
- Number of packets
- CRC status
- RMS phase error

6.2.3.3.2 Rx Sensitivity Test

The earlier discussed setup can also run the Rx test on the device side (Figure 6.6 and 6.7). First configured testing instrument as a signal generator and enabled Rx at DUT side. PER (packet error rate) is checked in this phase. Below APIs used to simulate Rx test:

APIs to test Rx test:

```
WRITE_DUT antenna 0
WRITE_DUT transmit
frame -BSS
11:22:33:44:55:66
WRITE_DUT transmit
frame -SOURCE
66:55:44:33:22:11
WRITE_DUT transmit
frame - Destination
11:22:33:44:55:66
WRITE_DUT transmit
frame -FrameControl
8841 WRITE_DUT
transmit frame -
Duration 2000
```

Statistics checked under Rx sensitivity:

- Total packets received
- FCS errors (FCS error is frame corrupted in PHY level)
- MAC mismatch (MAC mismatch is good Wi-Fi frame not addressed to device)
- Good packets: 0
- Average RSSI (SOC)
- Average RSSI (ANT)

The receiver minimum sensitivity test ensures a device under test (DUT) receives data with a defined maximum packet error rate (PER) of 10% at a defined minimum signal power. In Figure 6.4 & 6.5 threshold value would be 10%.

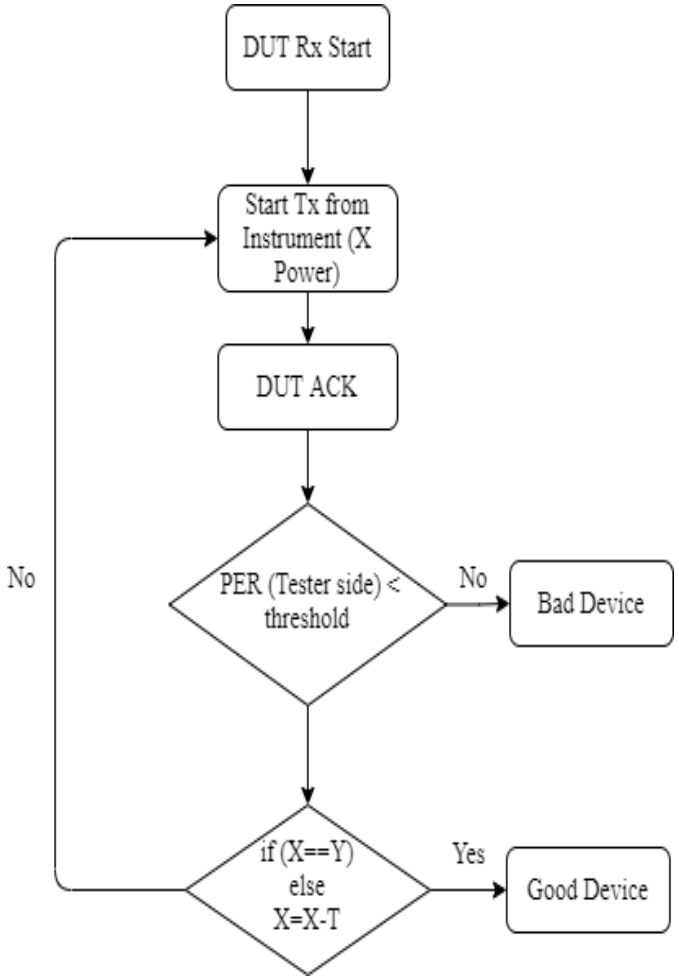

Figure 6.4 Rx test steps flow chart (ACK based)

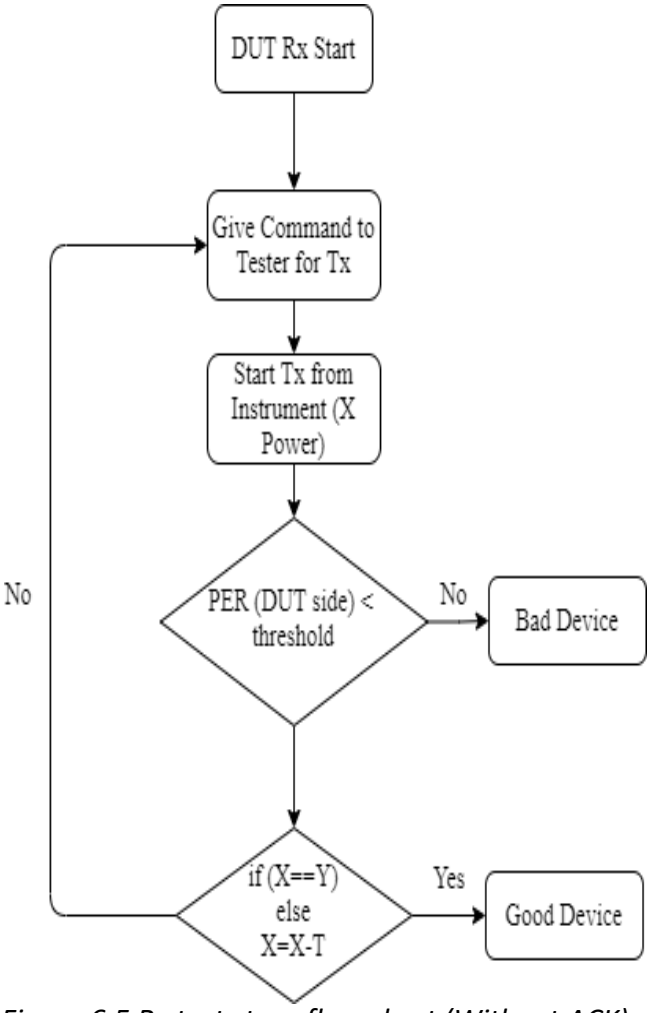

Figure 6.5 Rx test steps flow chart (Without ACK)

Two methods (a*cknowledge* and without a*cknowledge* based) used via vendors to check Rx sensitivity:

ACK (Acknowledge) based

Instead of getting the count of received packets at the DUT side, transmit packet acknowledgment count happened on the instrument side [117]. Firstly, the instrument device sends a hundred packets with X transmit power and checks how many packets

are received at the device side from the count of acknowledgment received. If the packet error rate is less than the threshold, it reduces Tx power via some dBm until it achieves the power level threshold. The same steps are repeated until the Rx PER test get fail/pass.

Without ACK based

Some vendors provide the functionality to disable MAC level ACK at the DUT side.

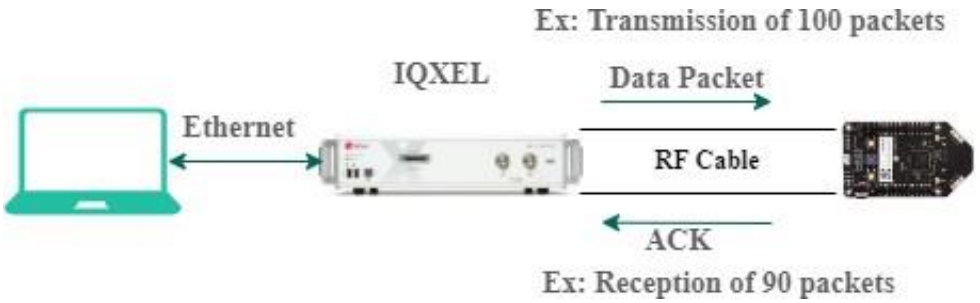

Figure 6.6 Rx sensitivity measurement (Ack based) simulation setup

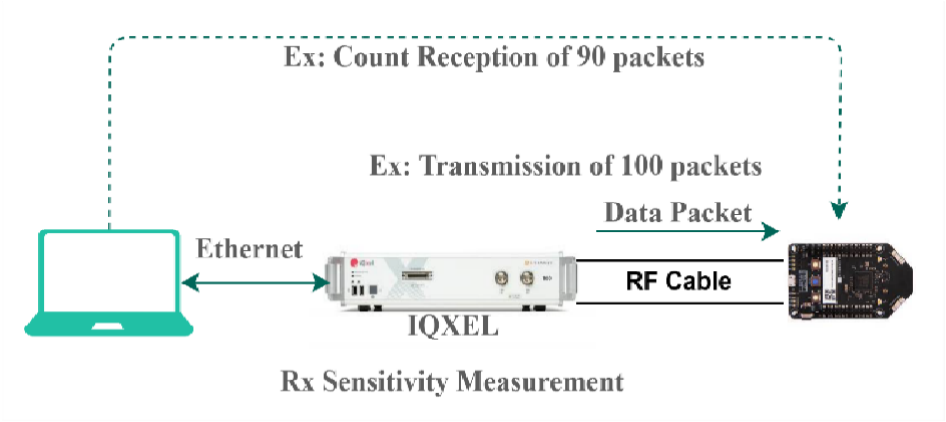

Figure 6.7 Rx sensitivity measurement (Without Ack based) simulation setup

MAC address writing

After Tx/Rx test is finished, MAC address writing is performed (Setup is shown in Figure 6.8). According to guideline [116] API used to write a new MAC address on hardware NIC is “WRITE_DUT config write MAC Address 00:11:22:33:44:55”

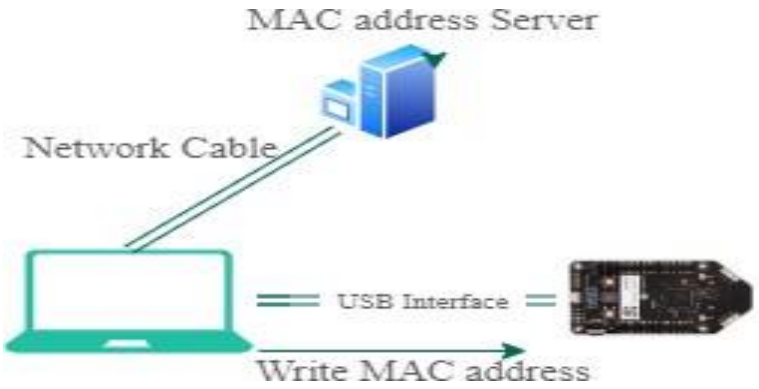

Figure 6.8 MAC address writing simulation setup

Steps to follow:

- The application asks the MAC server to generate a new MAC address.
- Command from PC to DUT
- DUT write MAC address

6.2.4 Proposed Method

This section describes calibration and test method amendment techniques and system-based simulation model as proof of concept. A specific 802.11 data frame is introduced for the calibration process, and this section describes the process of other amendments. Using simulation, uses and understanding of the proposed protocol and techniques will be shown in this section. The first subsection describes protocol and test method amendment. The proposed method is new, and no testing instrument allows changing its functionality. Hence system-based simulation is used and described in subsection 6.2.4.2. Another approach of achieving the same results using two Wi-Fi boards is to perform a physical experiment discussed in another subsection 6.2.4.3.

6.2.4.1 Calibration & test methods amendment

6.2.4.1.1 Tx power calibration

In the proposed system, we assume DUT will send N number of packets to the instrument device, and in response, the instrument device will send a specific Wi-Fi packet that can transmit power and other details. "Tx details ACK frame" mentioned under section 5.4 Tx Power Feedback Mechanism is use to perform Tx power calibration.

In transmit power test and calibration phase, network concept is not tested, hence drop of any packet will not make any impact. To achieve simplicity, in every slot, the first seq can be 0, and the last seq number can be the size of the chunk minus one. Suppose 100 packets will be sent in the slot, sequence number would be 0 to 99, and the next slot can start again from sequence number 0 (no need to sliding window). According to the requirement, the protocol can change accordingly. Tx details acknowledgment can be for a single packet, or it can be for N packets. When a Tx details ACK is received for the N packets, it contains the average transmit power of N packets (Figure 5.7b). In the case of Tx details, ACK received for a single packet has transmitted power only for that single packet (Figure 5.7a). When the instrument device receives the packets, it will calculate Tx power. Instead of sending it to the upper layer, it will respond to DUT in the Tx details ACK packet, and Tx power calibration can perform (Figure 6.9).

6.2.4.1.2 Rx Sensitivity Test

Instead of running the Rx test separately, the Tx power test can also check PER. Tx details acknowledge in the customized data packet. To run the Rx sensitivity test, every Tx packet needs to acknowledge. For example, if for 100 transmit packets,95 acknowledged packets received, PER would be 95%. If PER is less than the threshold, DUT can consider it with good Rx sensitivity and send it for the next step.

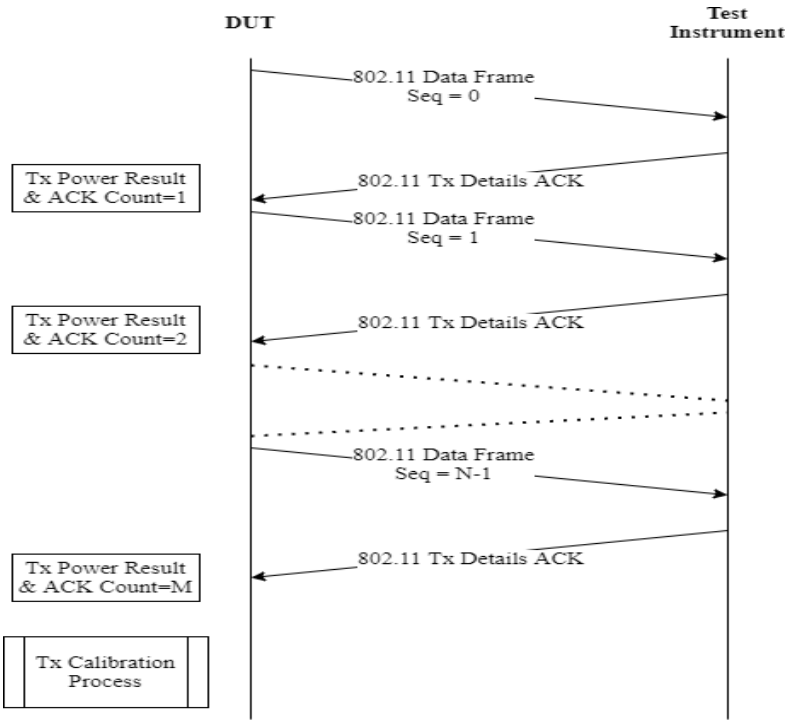

Figure 6.9 Tx/Rx test perform simultaneously.

6.2.4.1.3 MAC address calibration

A specific vendor command is used to send the server's request to generate the unique MAC address to write MAC addresses. When the device receives the MAC address in response from the server, it writes to the read-only memory section.

In the production line (shown in Figure 6.10), a device sends the request for a new MAC address using a particular packet on the wireless medium, and it's received by the device which has a MAC address server running on it. MAC address server produces a new MAC address and sends MAC address in response to the received packet. The device receives a MAC address and runs tests. If it passes all the tests, the received MAC address is written into EEPROM.

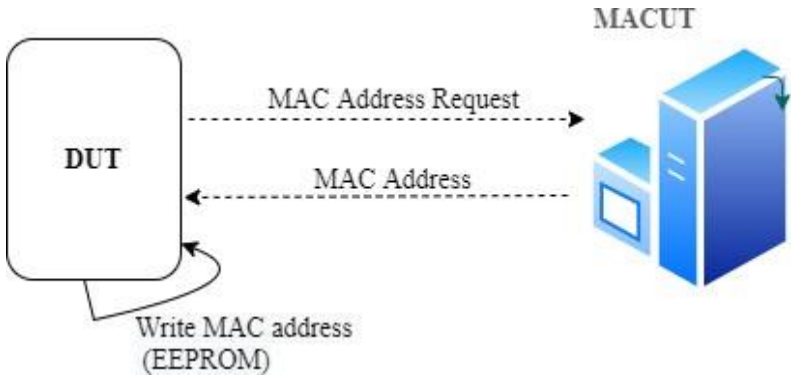

Figure 6.10 MAC Address Writing

6.2.4.1.4 Other calibration

Like MAC address, some other parameters, e.g., vendor ID, class ID, and serial number are also written wirelessly to save time. It depends on what the chip provider wants to write inside the ROM memory.

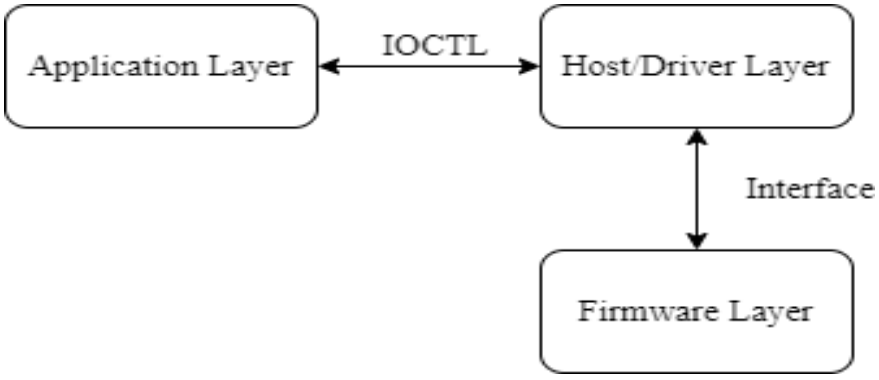

Figure 6.11 General device model

6.2.4.2 Simulation Model

To simulate the assembly line calibration test and method discussed in the last subsection, we developed a simulation tool on the Win10 OS (windows 10 operating system). To develop a user-mode application and kernel-mode device driver, visual studio 2015 is used with WinDDK [75]. WinDDK is mainly used to build the kernel driver, which is developed on the WDM framework [76]. A developed standard simulator tool can act DUT and testing instrument.

Figure 6.11 describes the general model of any hardware device, and it can be Wi-Fi, Bluetooth, or any other device. Firmware is embedded software that runs in the hardware device. In the Figure 6.12, the firmware layer can be taken as a combined unit of hardware and firmware.

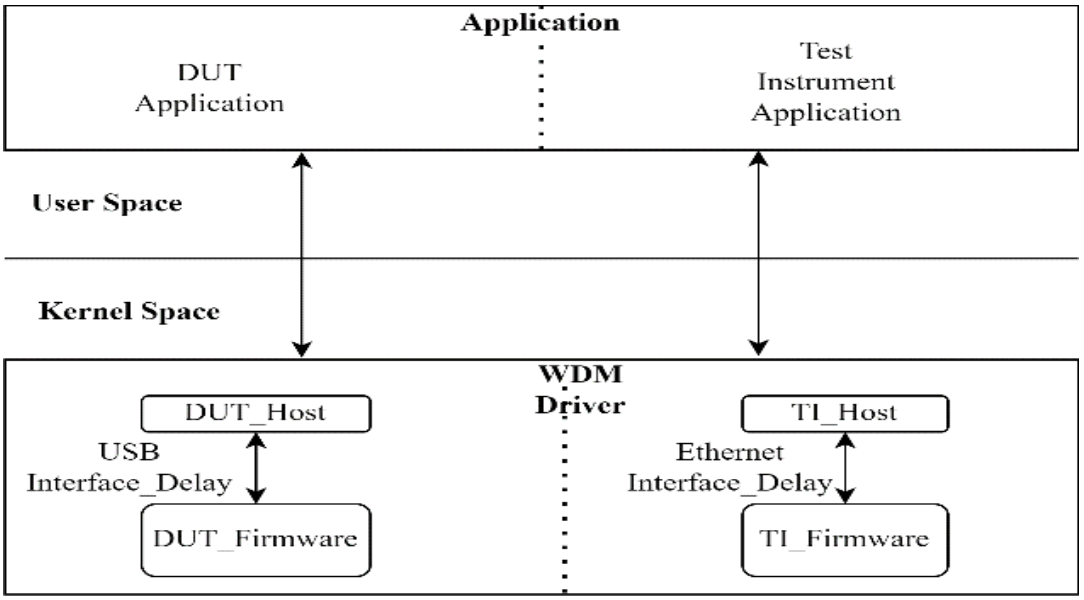

Figure 6.12 Simulator Device Model in our test

We have created two modules at every layer (user space and kernel space) to simulate the test, as shown in Figure 6.12. The application has a DUT, Testing Instrument module that runs at user space and driver, firmware/hardware functionality at kernel space for both modules (DUT and Test Instrument).

The user-space application uses IOCTL (input/output control) to interact with the kernel space. IOCTL [77] request lands to the host module of the device driver. A fixed interface delay gap is given between the host and firmware/hardware modules at the kernel-space driver to simulate the interface between the system and hardware. The

device is connected to the system via a hardware interface in a real scenario. This interface can be USB, I2C, SPI, PCIe, SDIO, etc. For DUT, USB (MT3620 interface) assumes, and it is Ethernet (IQXEL) interface for test instruments.

Using this simulator model, commands are given to DUT and to test instrument application via command line. There is no specific requirement for payload, so random payload is generated at DUT_Firmware and after receiving data (802.11 packets), the TI_Firmware layer responds with Tx Power details (proposed frame).

Below is the packet hex dump generated by our application. The color in the hex-dump and text color specifies the type and value.

Following data transfer from DUT_Firmware

MAC header + Payload (1000 byte) + FCS

08 00 00 00 00 11 22 33 44 55 66 77 88 99 00 99 00 11 22 33 44 55 D0 0E00 AA AA AA ……..

Packet type – 02 (Data), Subtype – 0x0

Payload – Random 1000 bytes data

And in response, the following data received

MAC header + Payload (16 byte) + FCS

8D 01 00 00 00 66 77 88 99 00 99..... 00 00 00 01 00 01 00 01 00 00 06 02 01 10 00 01 00 00 00 00

Packet type – 02 (Data), Subtype – 0xD)

Payload

Bitmap – 01 (Only 1 packet ack received).

First & last packet sequence – 01

Freq error – 0.0

Peak Power – 6.21

Avg Power – 16.01

6.2.4.3 Physical experiment support

Two MT7612 USB boards is taken to experiment and validate the presented method. The existing source code [118] needs to modify according to the proposed protocol. One USB board acts as DUT, and the other acts as a testing instrument. Lab setup is shown in Figure 6.13.

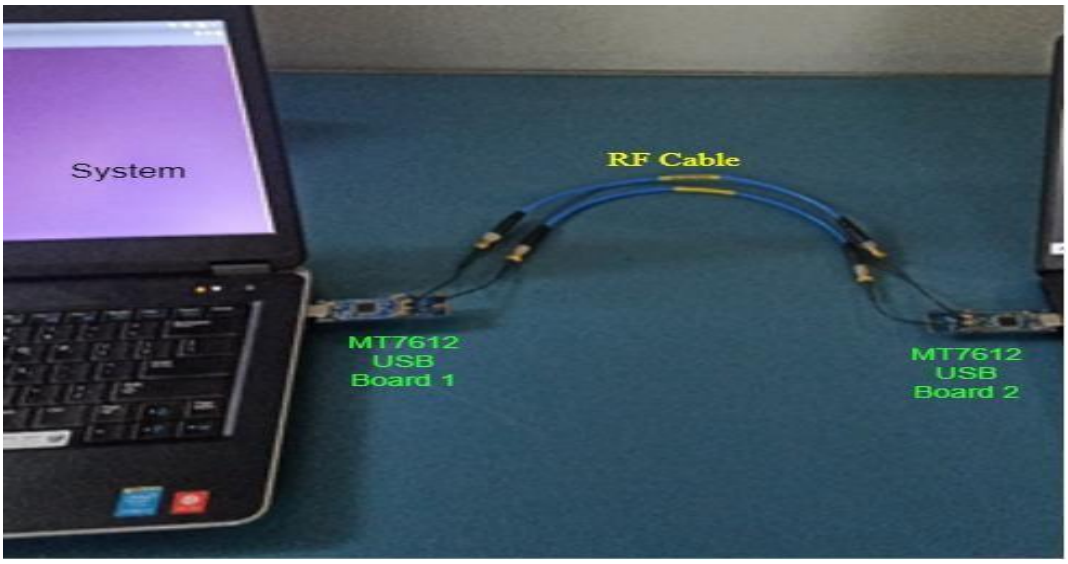

Figure 6.13 Physical experiment (in the lab)

Captured sniffer log (Figure 6.14) shows the implementation of the proposed protocol in the physical world. DUT (192.168.1.3) transmits the 802.11 data frame, and the testing instrument (192.168.1.4) sends "Tx details acknowledge frame" as a response with Tx power details. Set transmit power and get Rx data function APIs are provided in code respectively to set Tx power at the testing instrument side and get the count of the packet at the DUT side. Rx sensitivity is examined with the help of the above APIs, and throughput is also run between both devices using this protocol. APIs are available to select the different channel (frequency), select fixed data rate, and choose different bandwidths, which are used to test multiple combinations with the proposed frame.

A reserved field is used in the proposed frame, and it is not part of the 802.11 specifications. Hence, Wireshark cannot parse acknowledged frame and it just displays an unrecognized frame.

Physical validation does not provide any way to calibrate the chip (as it is already calibrated, market available chip). We have run the Tx/Rx test 1000 times with the proposed and conventional methods with different data rates and on the separate

antenna. Tx packet size from 1st board to 2nd board is taken 1069 byte (1000 byte payload + 69-byte header), 45 byte MAC level ACK size, and 368 bytes proposed Tx details ACK frame from board 2 to board 1 has taken to run the test. With the help of a sniffer, the time has been calculated, and shown in Table 6.4.

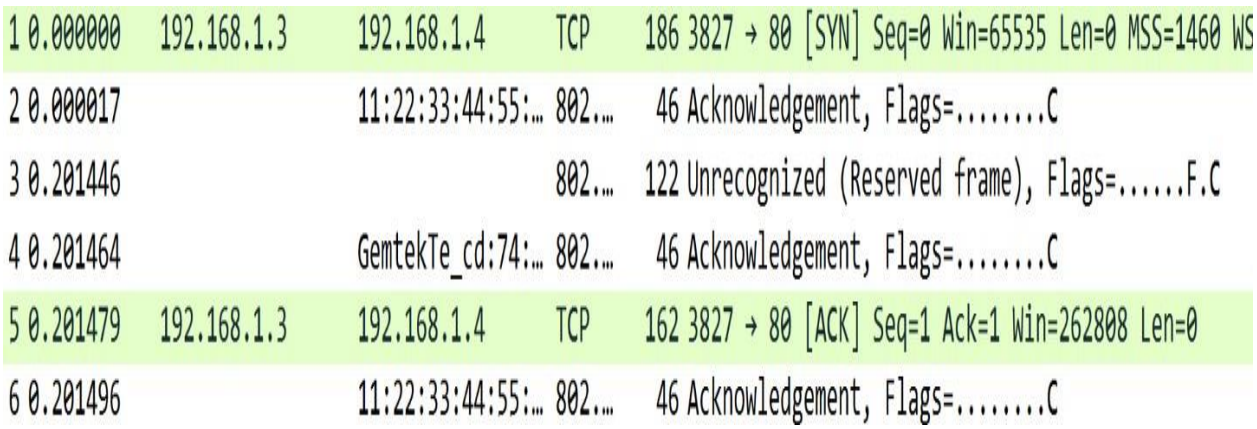

Figure 6.14 Sniffer log

Table 6.4 Tx/Rx test with different condition

Data Rate	Antenna	Channel	Conventional method (in ms)			Proposed Solution (in ms)			Improvement (%)
			Min	Max	Median	Min	Max	Median	
MCS7-B20	Both Antenna	CH10	48.92564	59.23764	54.50382	45.32366	54.28538	49.82376	9.39%
OFDM 54M	Antenna 1	CH1	236.36105	281.73022	258.00856	227.22345	268.54048	241.72501	6.73%
CCK 11M	Antenna 2	CH7	1182.68965	1274.34544	1238.23984	1112.54545	1194.29887	1149.30982	7.73%

6.2.4.4 Other Applications with the proposed approach

Some other use cases can improve with the proposed method/protocol. All Wi-Fi Tx/Rx testing specifications, WLAN measurements [119, 120, 121], protocols/working models are studied to identify mentioned use-cases (shown in Figure 6.17).

Maximum Input Level

During Rx sensitivity, a minimum input sensitivity check is already discussed. The maximum input level is also verified for Rx capability measurement with minimum sensitivity via some vendors. A high Tx power signal transmitted via testing instrument and checked receiver works appropriately after that or not. As shown in Figure 6.15 Testing instrument will send a packet from the highest power which DUT support (assume X is 16 dBm), and it will increase the power till some decided threshold (for example, +4db, Test Count would be 8) and DUT reply power with proposed ACK frame. If DUT can respond to higher power packets, it means the DUT receiver can tolerate X + threshold power.

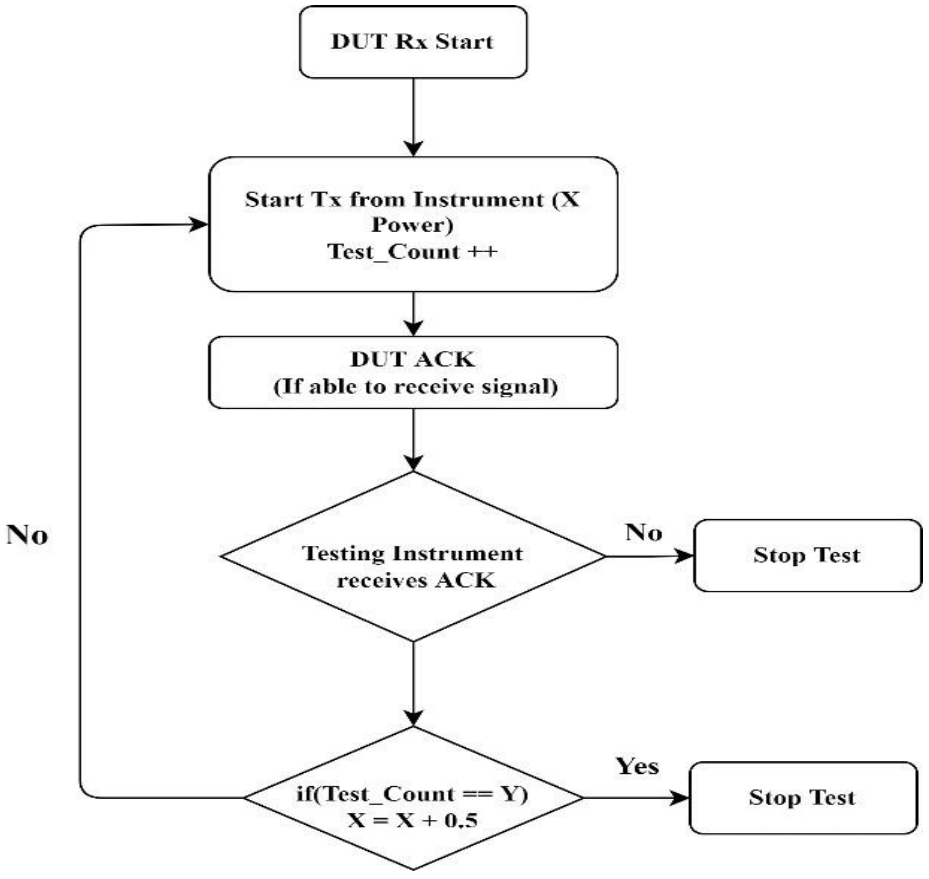

Figure 6.15 Maximum Input Level Rx Test

Adjacent/Nonadjacent Channel Rejection (ACR/NCR)

Another Wi-Fi signal can be present on adjacent or non-adjacent channels in a dense or noisy environment. In this condition, the DUT transmitter and receiver need to get a minimum signal where another signal is present on the adjacent channel. Using the

proposed protocol testing instrument will send two signals of different tx power (as shown in Figure 6.16), and DUT will acknowledge accordingly. The ACK frame response will check which one is correctly received via DUT and how much power degradation happens due to the adjacent channel. For example, in Figure 6.16, the Testing instrument VSG1 sends a packet on 10MHz bandwidth and VSG2 transmit on another 10MHz, and DUT supports 20 MHz bandwidth.

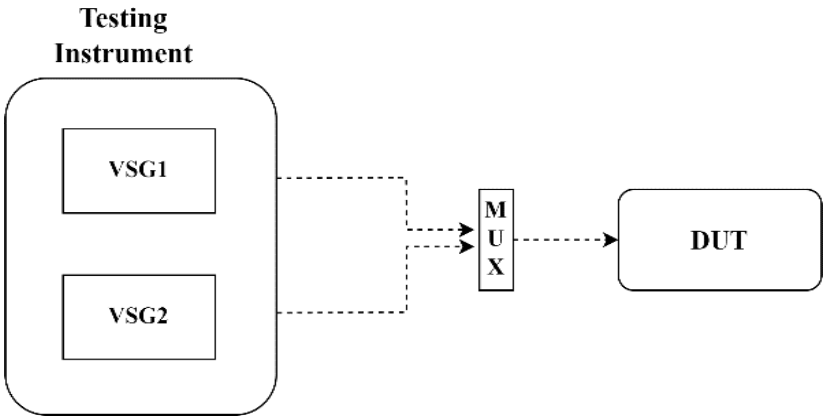

Figure 6.16 Test block diagram of ACR measurement

Roaming

The proposed method can improve the Roaming scenario. STAs can choose which access point they want to connect to when they roam to another one. STA need to send the proposed Tx packet frame, and it will get Tx power in response, so it can decide based on acknowledgement (which AP gives good power in response).

Battery life

Power-saving management in Wi-Fi devices where battery power is a concern is still in its infancy. On runtime, devices can change their power level by taking responses from peer devices using the proposed protocol and can increase battery life. This helps to reduce the amount of power used by the devices and minimize interference between them, leading to more efficient network usage and decreased energy consumption.

Tx Power Regulation Check

Wi-Fi radio bands' power limits are regulated by different countries/organizations [122]. As an example, the European Union sets the power limit to 20 dBm (100 mW) for OFDM and 18 dBm (63 mW) for CCK on the 2.4 GHz band [123]. This regulation

check also can perform via the proposed method at the assembly line using the proposed protocol. The Tx Power Feedback mechanism enables the regulation of the transmission power of Wi-Fi devices. This helps to reduce the amount of power used by the devices and minimize interference between them, leading to more efficient network usage and decreased energy consumption.

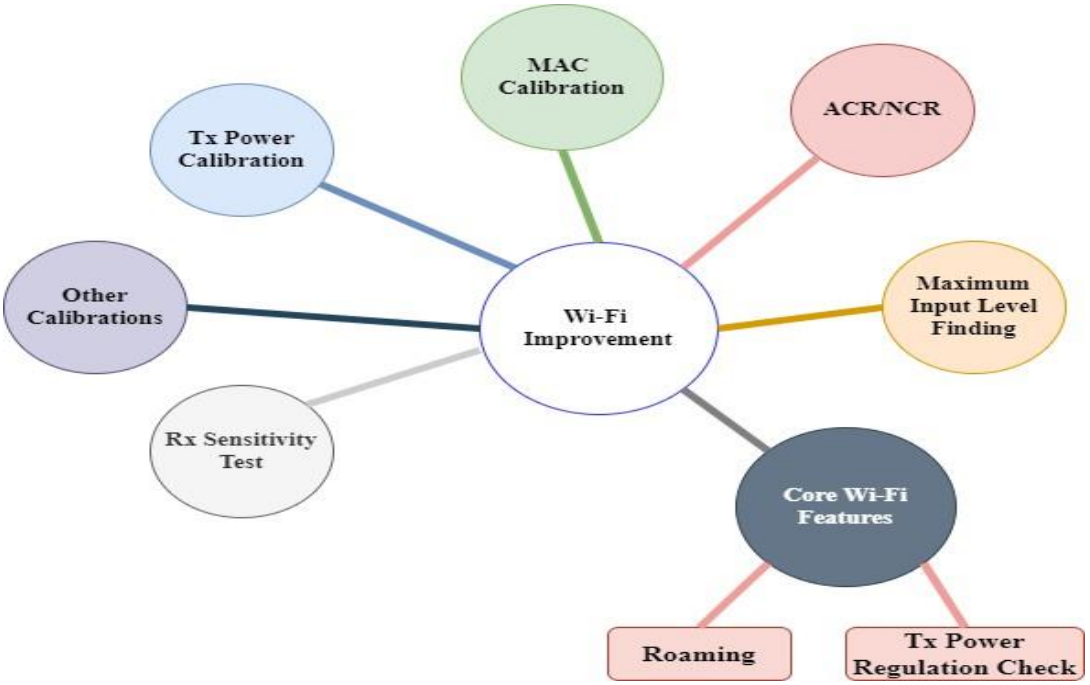

Figure 6.17 Wi-Fi Improvement applications with proposed approach

6.2.5 Results & Discussion

This section reports the results of the execution time of the proposed and conventional methods. Synthesis of the papers analyzed and provides a summary in other subsections.

6.2.5.1 Execution timing compare

We have performed a timing comparison between a standard Tx/Rx test, calibration procedure, and our proposed method. Here is the timing variable used in the calculation.

T_{MODE}: Time is taken via IOCTL (user-space to kernel-space data transfer time)

T_{USB_IFACE}: Time is taken via USB interface to transfer data

T_{WL_TRANS}: Time is taken via packet to transfer in the wireless medium time

T_{ENT_IFACE}: Time is taken via ethernet interface to transfer data

Context switch from user mode to kernel mode is an overhead process for any machine and it requires considerable CPU time (CPU time, also depends on CPU speed, system load and system configuration). Hence we also cover this parameter in out calculation. Context switch time varies from µs to seconds.

A freshly installed Win10 OS machine (intel i5 processor, 4 Gb RAM) is taken to run the test. A minimal print statement is added in IOCTL at the application side and a print statement at the DriverEntry and MJWRITE function at the driver side. Using Dbgview.exe [81] and timestamp print, TMODE is calculated at our machine. The average of 100 samples has been taken to calculate this value.

To calculate the T_{USB_IFACE} interface delay, the assumption is interfaced equivalent to USB 2.0. Wi-Fi devices prioritize USB 2.0 over USB 3.0 even when USB3.0 is much faster than USB2.0. The reason behind, USB 3.0 also runs on a 2.4 GHz frequency which is the same frequency used via Wi-Fi devices, and USB 3.0 interfere with Wi-Fi operation. Practically, USB 2.0 speed is approx. 40 megabytes per second (Mbps) [124]. We have measured a command time from driver to USB on azure sphere board via probe firmware print, and 400 samples have been taken to average the value.

Here is the measured average value:

- T_{USB_IFACE} – 3.12 µs
- T_{MODE} – 93 µs

The same method is used to calculate T_{ENT_IFACE}. It is approx 94.24 µs (average of 400 samples) from tool to instrument device firmware (T_{MODE} + T_{ENT_IFACE}).

802.11 have a standard interframe spacing value. SIFS and slot time is different according to the physical layer. DIFS value is calculated using SIFS and slot time.

$$T_{SIFS} = 10\mu s \quad (1)$$

$$T_{SLOT} = 20\mu s \quad (2)$$

$$T_{DIFS} = T_{SIFS} + 2 \times T_{SLOT} \quad (3)$$
$$= 10\mu s + 2 \times 20\mu s = 50\mu s$$

802.11 packet consists of PHY header, MAC header, and payload with checksum. PHY header further consists of PLCP preamble (144bits) and header (48bits). We have used HR/DSSS mode and assumed that the packet transmission rate is 11Mbps. Therefore, the time to transmit the PHY header will be:

$$T_{PHY} = (144 \text{ bits})/(11 \text{ Mbps}) + (48 \text{ bits})/(11 \text{ Mbps}) \quad (4)$$
$$= 17.45\mu s$$

Next is MAC Header, which is 24bytes (192bits). It will also transfer at 11 Mbps speed. Therefore, transmit MAC header time is

$$T_{MAC} = (192 \text{ bits})/(11 \text{ Mbps}) \quad (5)$$
$$= 17.45\mu s$$

FCS (Frame check sequence) size is 4 bytes (32 bits) long, hence FCS travel time

$$T_{FCS} = (32 \text{ bits})/(11 \text{ Mbps}) \quad (6)$$
$$= 2.90\mu s$$

To transfer any packet, we are assuming there is no random backoff wait time. The device uses DIFS time to transfer a packet. The payload size would be 1000 bytes, which is sufficient to check to transmit power details. Time taken via packet to transfer in wireless medium would be:

$$T_{WL\_TRANS} = T_{PHY} + T_{MAC} + T_{PAYLOAD} + T_{DIFS} \quad (7)$$
$$= 17.45 + 17.45 + (1000*8)/11 + 50$$
$$= 812.17$$

Tx Test Time Calculation

Now we are ready to calculate the total time for each packet present in normal and our proposed model. $T_{N\_TxTOTAL}$ is the total transmission time in a conventional scenario, and $T_{P\_TxTOTAL}$ is in the proposed scheme. Table 6.5 covers all the steps for the traditional technique, and table covers the proposed method.

Table 6.5 Transmission time in the normal scenario

S.No.	Step Detail	Time
1	Command from PC to DUT	$T_{MODE} + T_{USB\_IFACE}$
2	DUT Tx	T_{WL_TRANS}
3	Get Tx power command from PC to instrument	$T_{MODE} + T_{ENT\_IFACE}$
4	Tx power data from the instrument to PC	$T_{MODE} + T_{ENT\_IFACE}$
5	Tx power data from PC to device	$T_{MODE} + T_{USB\_IFACE}$

$$T_{N\_TxTOTAL} = 4 * T_{MODE} + T_{WL\_TRANS} + T_{ENT\_IFACE} + 2 * T_{USB\_IFACE} \quad (8)$$
$$= 4*93 + 812.17 + 2*1.24 + 2*3.12 = 1{,}192.89\mu s$$

Switching from user mode to kernel mode count = 4

$$T_{POWACK} = T_{PHY} + T_{MAC} + 160/11 + T_{FCS} \quad (9)$$
$$= 52.34\mu s$$

Table 6.6 Transmission time in the proposed scenario (TP_TxTOTAL)

S.No.	Step Detail	Time
1	Command from PC to DUT	$T_{MODE} + T_{USB\_IFACE}$
2	packet DUT Tx	T_{WL_TRANS}
3	Testing instrument Tx details Ack	T_{POWACK}

$$T_{P\_TxTOTAL} = T_{MODE} + T_{WL\_TRANS} + T_{DUT\_INTERFACE} \quad (10)$$
$$= 93 + 812.17 + 52.34 = 957.51us$$

Switching from user mode to kernel mode count = 1

The proposed approach saves Tx test time and calibration time. After Tx power calculation is received from the application to the device, the device decides the need to write Tx power in EEPROM memory. While in the proposed scenario, Tx power is directly received to the device, so we save lots of time in calibration. T_{N_TxCal} is the extra

time during calibration of Tx power during the standard scenario, which can save in described calibration method:

$$T_{N\_TxCal} = T_{MODE} + T_{USB\_IFACE} \quad (11)$$
$$= 93 + 3.12 = 96.12\mu s$$

Rx PER Test Time Calculation

Table 6.7 covers all the steps for the Rx test (without ACK), while table 6.8 covers the same test with ACK.

Table 6.7 Rx Test Normal Scenario (Without ACK)

S.No.	Step Detail	Time
1	Command from PC to DUT to enable Rx	T_{MODE} + T_{USB_IFACE}
2	Give the command to start Tx from the instrument side	T_{MODE} + T_{ENT_IFACE}
3	Instrument Tx	T_{WL_TRANS}
4	Read Rx information data from PC to device	T_{MODE} + T_{USB_IFACE}

$$T_{N\_RxTOTAL1} = 3 * T_{MODE} + T_{TRANS} + 2 * T_{USB\_IFACE} + T_{ENT\_INTERFACE} \quad (12)$$
$$= 3 * 93 + 812.17 + 2*3.12 + 1.24 = 1098.65\mu s$$

After each packet, the receiver sends the ACK (Acknowledgement frame). The MAC header of the ACK frame is 10 bytes (80bits) long.

Therefore, ACK transmission time will be:

Using equations (4) and (6)

$$T_{ACK} = T_{PHY} + 80/11 + T_{FCS} \quad (13)$$
$$= 17.45 + 7.27 + 2.90 = 27.62\ \mu s$$

Table 6.8 Rx Test Normal Scenario (ACK based

S.No.	Step Detail	Time
1	Command from PC to DUT to enable Rx	$T_{MODE} + T_{USB\_IFACE}$
2	Give the command to Start Tx from the instrument side	$T_{MODE} + T_{ENT\_IFACE}$
3	Instrument Tx	T_{WL_TRANS}
4	DUT ACK	T_{ACK}
5	Read Rx information data from PC to Instrument Side	$T_{MODE} + T_{ENT\_IFACE}$

$$T_{N\_RxTOTAL2} = 3 * T_{MODE} + T_{ACK} + T_{WL\_TRANS} + 2 * T_{ENT\_IFACE} + T_{USB\_IFACE} \quad (14)$$

$$= 3 * 93 + 27.62 + 812.17 + 2*1.24 + 3.12 = 1124.39\mu s$$

Switching from user mode to kernel mode count = 3

6.2.5.2 Summary

As discussed, our proposed solution can perform Tx/Rx test simultaneously. In this work, timing is only shown for one mode and one packet transmission/receive. Table 6.9 shows the timing difference between the conventional and proposed method. In the assembly line, each Wi-Fi supported 802.11 modes (b,g,n,ac & ax), different bandwidth, different rates and multiple antennas [125] tested for millions of devices, so at the large scale proposed model can save a significant time range (from hours to the number of days).

Table 6.9 Time comparison in conventional & proposed scenario

Method	Tx/Rx test time	Calibration Time
Conventional approach	TN_TxTOTAL + TN_RxTOTAL1 = 1,192.89µs + 1098.65µs **= 2291.54 µs**	**96.12us**
	TN_TxTOTAL + TN_RxTOTAL2 = 1,192.89µs + 1124.65µs **= 2317.54 µs**	
Proposed Approach	**957.51us**	**0**

The context switch is undoubtfully an overhead for a machine (save current stack/registers, copy data from user space to kernel space and restore different state). It is the costliest operation on an operating system. The proposed method also significantly reduces context switching from user space to kernel space, saving the processor time considerably.

6.3 FOTA update for wireless device

These days, IoT (Internet of Things) use cases are spread across all the verticals of industries/organizations. The applications founded by IoT are available in almost all domains. IoT applications are widely used in many industries like agriculture, manufacturing, biomedical, and many more. The number of IoT devices is increasing exponentially, especially in remote areas. The software development process of IoT devices is not a one-time process. These IoT devices deployed in industries need to update regularly, even if a device is running on small software that needs to be updated to get the newly added feature or bug fixes.

The new Computing paradigm used for IoT software updates is "Using more computing to move fewer data." This chapter discusses implementing a framework for a software update of embedded wireless devices even on low connectivity for FOTA via on-premise firmware binary creation instead of downloading it from the cloud. The proposed framework is evaluated based on different security aspects andtools. A high-level security structure is presented under the proposed framework. Results show that the proposed scheme performs better than other FOTA methods in data exchange on network and cloud/device side processing power.

6.3.1 Design and Architecture

In this section, the architecture of the proposed methodology is described and discussed. This section discusses virtualization, FOTA method amendment techniques, and different software/hardware components under the proposed architecture. The first subsection describes various elements to develop the suggested FOTA method. The proposed method is new, and no existing gateway allows to change its functionality. Hence system overview and FOTA flow are described in section 3.2. Gateway architecture and data communication between different operating systems are discussed in consecutive subsections.

6.3.1.1 Components

The following components are considered to develop the proposed FOTA method based on all available standards and practices.

- A machine runs three operating systems, acting as Gateway (Section 3.2).
- An Android device, acting as a Virtual Device.
- A Linux machine [Ubuntu 18.04] is used as a Webserver.
- Wi-Fi protocol to transfer data between gateway and nodes [non-IP devices].
- The CoAP transport stack.
- The LwM2M-based IoT device management webserver.
- AES 256-bit encryption engine.
- Apache webserver

A system with a lightweight hypervisor [126] is selected for this prototype, but this is applicable for any other available operating system and device.

6.3.1.2 System Overview

The proposed FOTA method and system are discussed in Figure 6.18. The following vital players play an important role under the IoT framework:

Device: The device is connected to the gateway using a Wi-Fi link. Device MAC address is used as device_id, Android device's Manufacturer as vendor_id, and model-name as class_id. When a device connects to the gateway, data in plain-text format is sent to the gateway. Following device register on the webserver with the help of the gateway.

Gateway: System OS can behave as a Wi-Fi hotspot [127]. When a device connects to the gateway for the first time, the gateway receives registration information. On Gateway (Core OS), a CoAP client also runs, using standard LwM2M resources to register devices with the Web-Server. Git is used for code versioning software, and a basic firmware code is already deployed at the gateway.

Webserver: The apache webserver is used to host our web application. From here, users can control the IoT devices. Users can view registered devices and can trigger firmware updates from here. A CoAP server runs on the Web-Server, and code sync to the gateway also happens here.

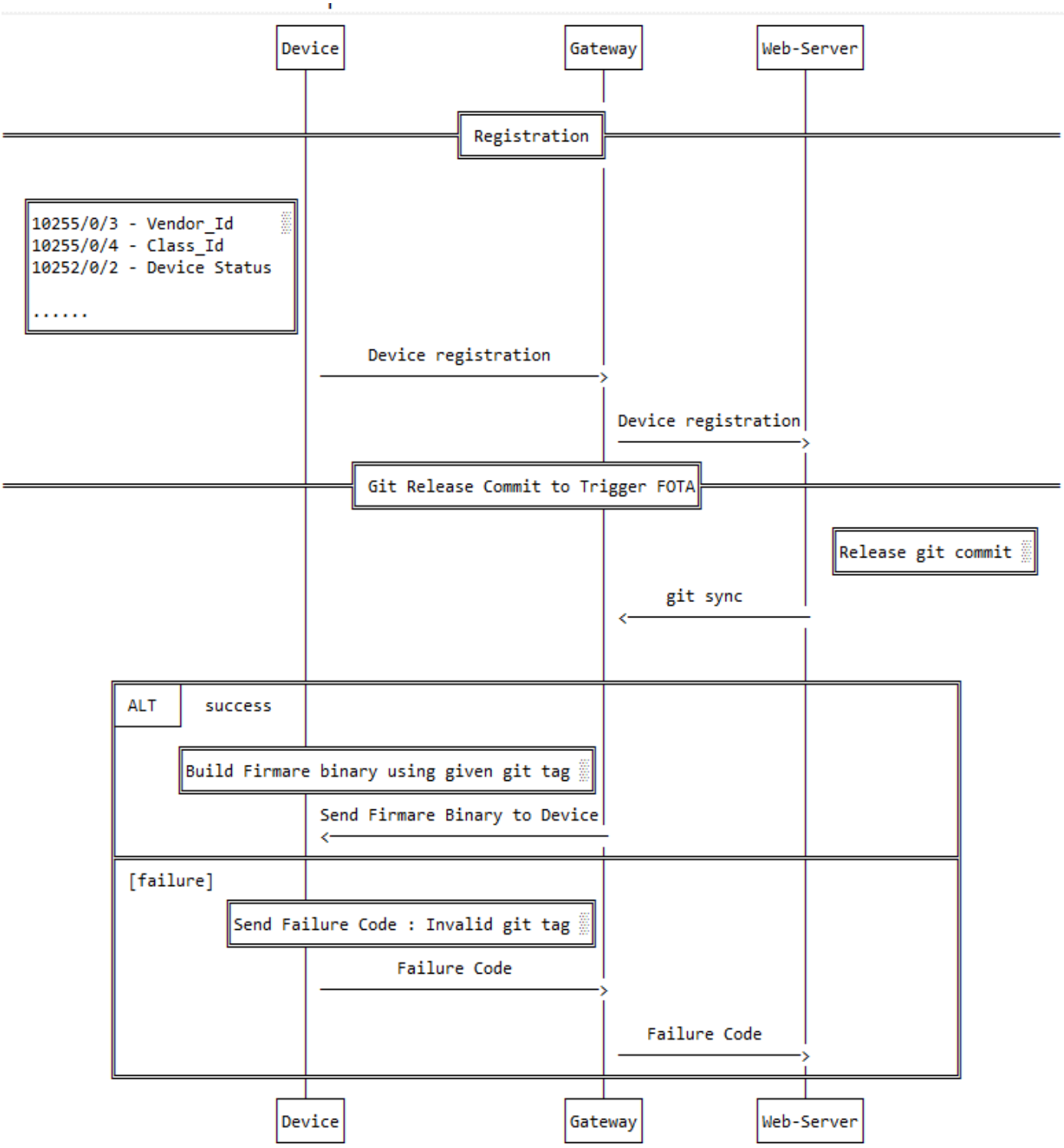

Figure 6.18 Proposed FOTA Method

6.3.1.3 Gateway Architecture

The proposed architecture consists of 3 operating systems, which run with the help of Hyper-V hypervisor [128] on x64 architecture machines (Figure 6.19). Figure 6.20 shows the different components of the respective operating system.

Different OS is meant to provide great modularity for various components. The multi-OS system also makes sense about the security, which will allow one OS to deal with hardware security while another app is also running on the top or alongside. Here are the details of all OS:

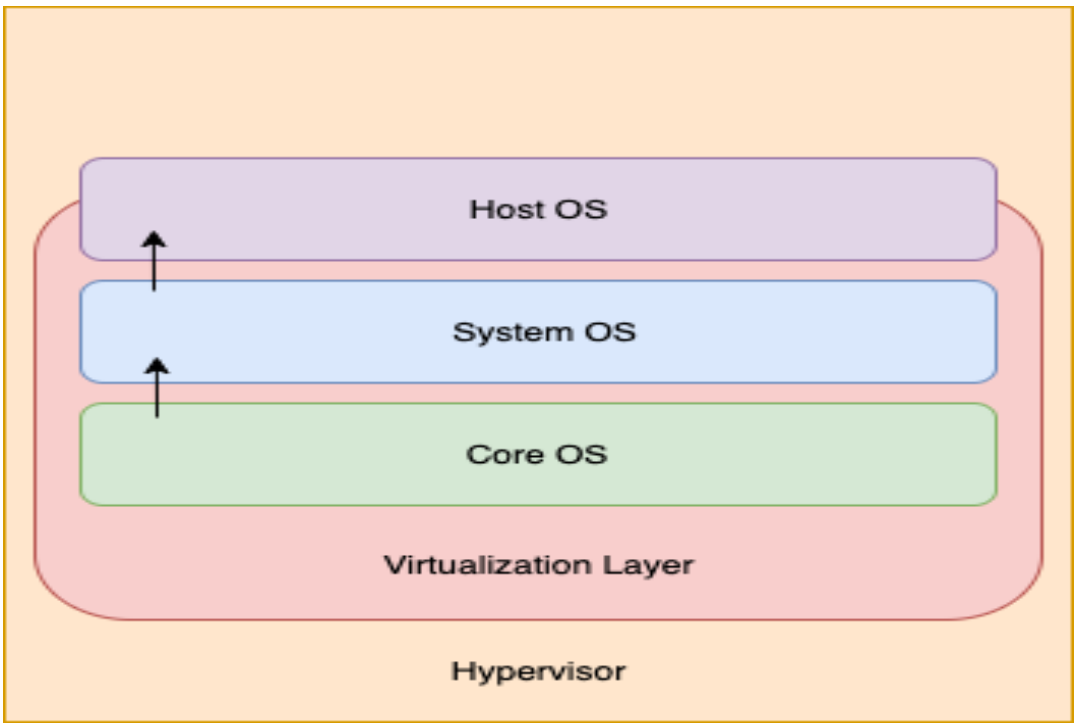

Figure 6.19 Hyper-V hypervisor architecture

The first one, 'Core OS,' is a FreeRTOS based operating system with gateway hardware control. This core OS handles all security responsibilities and hardware control task execution. It does not execute any application.

The second one, 'System OS,' is Ubuntu light operating system, which runs the LWM2M server, version control system and some specific system services. This OS has predefined applications installed, and the user doesn't have permission to install any other application. This OS does not connect with the device management cloud, and it only has internet connectivity with a version control (git) server. The services running on this operating system are responsible for fetching the latest code. This operating

system also contains the necessary toolchains required to build the code. It builds the binary and shares it with the Host OS.

The third one, 'Host OS,' is also an Ubuntu operating system. Gateway default and standard applications run on it. It is also responsible for transferring the firmware to the connected device.

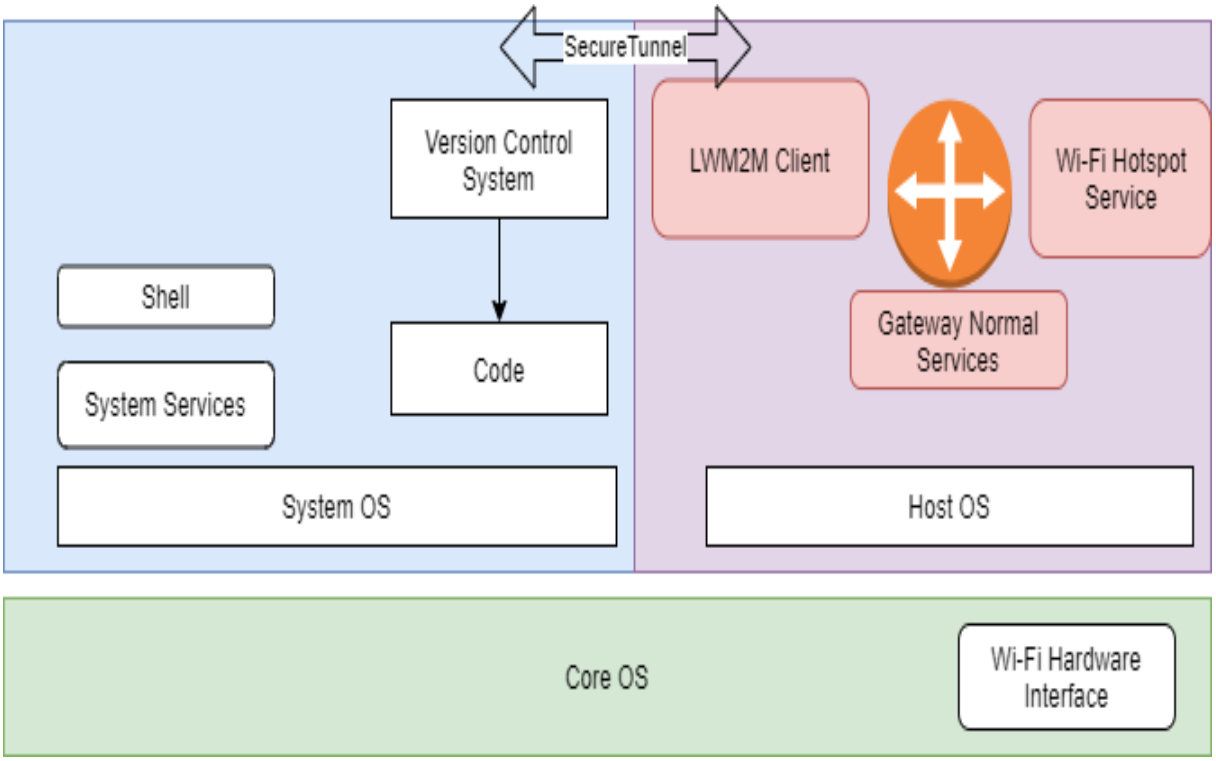

Figure 6.20 Gateway Architecture and Different OS responsibilities

6.3.1.4 Data Communication

Code Checkout on Gateway: On Gateway (System OS), a git hook is used to check out the code. If any code is checked in, it will notify the gateway and pull the code on the gateway. So, the gateway always has the latest firmware code-base on board.

LWM2M Bootstrap: Authentication and provision of LWM2M client and registration with LWM2M server done using this method. Leshan [129] is used to implement it in our setup, which is an open-source OMA lightweight machine-to-machine implementation (server and client).

Register Device: The device sends its information to Gateway (Information is encrypted using AES) to register the device, and with this information, the device gets

registered to Web- Server. Following CoAP message transferred from Gateway to Web-Server:

- Uri-Path: rd
- Status: COAP_STATUS_OK
- Code: COAP_MSG_CODE_REQUEST_POST
- Type: COAP_MSG_TYPE_CONFIRMABLE
- Id: 1178
- Token: 02:00:00:00
- Payload: following data sent (shown in Table 6.10)

Table 6.10 Values of LWM2M Objects and Resources [130]

Object (Object Value)	**Resource (Resource Value)**	**Value**
Device (3)	Current Time (13) Device Type (17) Serial Number (2) Model Number (1) Manufacturer (0)	Current time value Mobile MAC address Mobile Model Mobile Manufacturer
Device Metadata (10255)	Vendor (3) Protocol Supported (0)	Mobile Manufacturer 1
Status of code build (10301)	Version (1) State (2) Result (3)	1 0 0

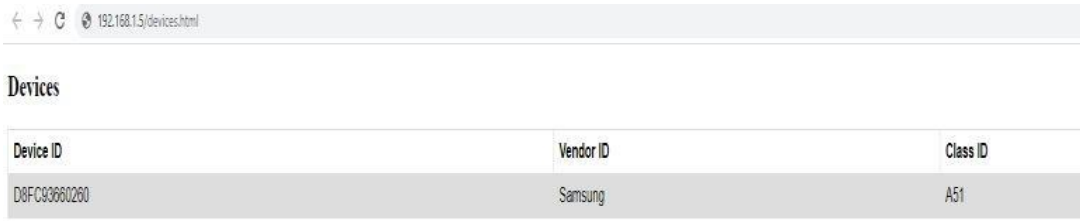

Figure 6.21 Registered Device shown at Web-Server UI

A reserved 10301 object is used to provide the status of code built to the webserver.

LWM2M <object/object_value/resource> with default values. The webserver side can see the device registered successfully (Shown in Figure 6.21).

Code Sync: Now application on system OS parses the git message; if it is a release candidate (Figure 6.22), it will trigger the Firmware binary build. The best part of this proposed architecture is "there is no Manifest handling happens at the gateway," and mostly, IoT device management uses an extra certificate to parse and check the validity of manifest, which is not required in this case.

System OS provides final status to the host OS via calling the secure access API. If everything is fine, firmware binary builds using git commit-id received. If the build system has success/error, some custom codes (Table 6.11) are defined for binary update status (10301/0/3), which lets the webserver know what status during a build.

Table 6.11 Example of Custom Code

Custom Code Value	Meaning
0x400	Success
0x401	General Error
0x402	Access Error
0x403	Missing Parameter
0x404	Internal Error
0x405	Compile Error
0x406	Linker Error
0x407	Other Errors

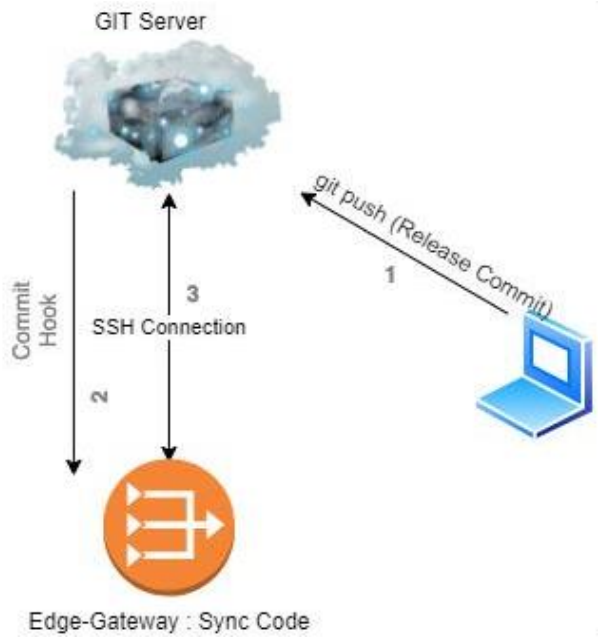

Figure 6.22 Git code-sync with commit-hook

6.3.2 Security aspects of the proposed method

This section covers the security aspects of our proposed model and how it tackles code-breach or build and binary compromises. As the whole system is based on a version-control model like Git or SVN, entire code-base available on the System OS, which will act as a build-machine. Host OS will behave as the gateway for OTA updates for other endpoint devices in the vicinity. The updates will be deployed using Wi-Fi, and one needs to consider that the security stakes will be high if the build-machine gets compromised somehow. The propagated attack will be much higher as it can/will affect all the endpoint devices if the build machine is accessed via an untrusted element. So to prevent these sorts of scenarios, system need to ensure the integrity and trust ability of the machine. The section discusses various offerings that provide significant benefits for IoT device security and device health monitoring systems and decide the best fit for the proposed IoT framework. Following solutions are available:

- Encrypted Filesystem
- Secure-Storage Module
- TPM [Trusted Platform Module]
- File-based encryption strategy

Let's start with encrypted file systems; almost all the filesystems available for IoT devices support some encryption to ensure the integrity and validity of data. The proposed method has added one more security layer using a developed filter driver, which provides AES-level encryption. Encryption Key stored in a particular module called TPM (Trusted Platform Module).

The entire code-base is deployed on the build machine in the proposed method. An OS-based partitioning scheme is used that is separate from the rest of the system and is locked with a cryptographic key that can be stored/generated to/from a cryptographic module like TPM [131]. TPM can be both software-based or hardware-based, but most commonly, it'll be hardware-based as Software TPMs (sTPM) are only intended for development-use and do not provide any great benefits compared to their counterparts. TPM, in this case, will act as a Root-of-Trust for access to our encrypted partition. This work is intended for research and development only, so software-based TPM is used here.

Secure-storage modules are considered as well, but they're ubiquitous. They do not provide a standard specification like TPMs, primarily based on TPM 2.0 specifications by the Trusted Computing Group (TCG). Also, file-based encryption strategies can be incorporated, but this can be complex and might not work on par with most code-bases. If they have many files and transitive dependencies, resulting in inferior build times and might fail at some step, resulting in file corruption and data loss. Using file-based encryption also has a significant drawback as it has different keys for the n-number of files, which can be unlocked independently. In this case, the system can compromise, and some attackers can get access to some of the keys and perform serious modifications. It can open a path to access endpoint devices via reverse-engineering, the code-base, or a backdoor to the system binary. This practice can affect a massive number of devices and can leave them in an inconsistent state. Quite possible the devices may not be able to receive OTA updates and are left out in the wild. To our conclusion, combining an Encrypted file system and a TPM-based solution yields excellent results. It provides the level of security needed for keeping IoT devices secure while having a Root-of-Trust (ROT) that is part of the system itself and is provided by the OEM.

Most TPMs come with three Root-of-Trust (out-of-the-box) and can hold a few more. What can do here is to add an extra layer of security by having different ROTs for a separate process to ensure that the filesystem, build-process, or binary is not compromised in some manner. It can be an improvement over the current approach discussed here.

The proposed method focuses on leveraging the encryption on the filesystem partition and securing the keys' exchange through TPM.

Figure 6.23 illustrates the high-level structure of how the proposed framework security model will work. In the diagram, starting from the top, the system has applications for (build, test, deploy) aspects. There will be an intermediate "Filter Driver," which will play a key role in accessing the data from the disk (a.k.a File-System). It'll be the job of the access provider to pipe data to the application which is asking for it. The applications will not have any access to the filesystem itself. The filesystem will be locked using an encryption scheme, and the encryption key for the disk will be burned down to the core of the TPM module. Only the host OS will be able to decrypt the shared file using a decrypt key stored under TPM.

On System OS, only version control applications allow access to the file system. While on Host OS, only verified applications can access the filesystem. A specific symbolic link is used to access the filter driver. Whenever an application needs access to the filesystem, it will get the driver handle. A request is sent to the IO manager to verify the symbolic link. The I/O manager sends a token as a grant/deny response to an access-provider utility, which the access-provider will parse to identify if the operation is permitted or not. If the grant token is issued, the access provider will request the TPM to issue the encryption key to unlock the filesystem. It'll be the responsibility of the access provider to manage the lock/unlock state of the disk while the operation is active.

If the deny token is issued, it'll simply reject the application request and kill the application, and it will also halt all the operations and send status to the cloud. Resume operations depend on cloud-side device management software. As the framework is leveraging full-disk encryption, it keeps our code-base and binary secure from tampering with external factors while on the gateway.

In this architecture, system OS can only access the git server. Another internet activity is prohibited on this OS, and git uses several protocols. Here only SSH (secure shell) is permitted. To access git, a session key is used, which is also stored under TPM.

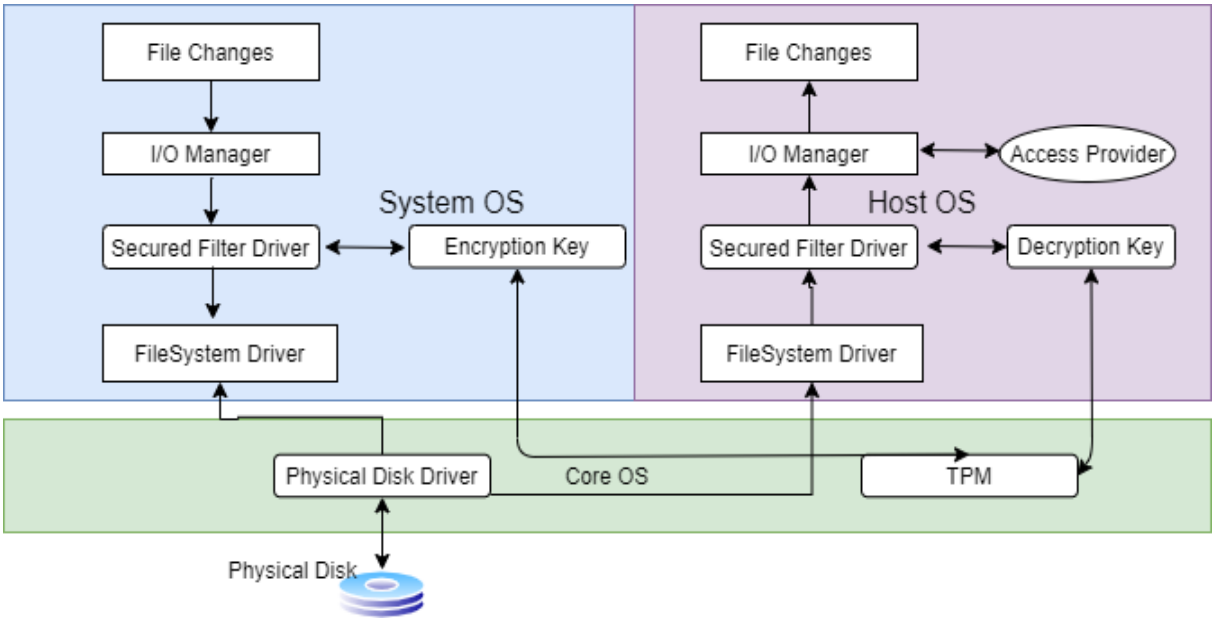

Figure 6.23 High-level structure of our security model

6.3.3 Comparison with existing methods

This section reports the results of comparing the proposed and conventional techniques. The first subsection compares FOTA methods based on data processing and the second subsection has timing execution of different methods.

6.3.3.1 FOTA data processing methods

Several methods are proposed and are in use to improve FOTA data processing. Size reduction may depend on different algorithms, according to the requirement, but overall these methods are mainly divided into three parts:

FOTA Compression method (Figure 6.24)

The Firmware file compresses first on the development/cloud environment side in this method. The best one is used to do that out of all available compression algorithms. Available compression methods like gzip, which has a compression ratio of around 2.7x-3x. For example, if the file size is about 300 MB, it can be reduced to 100-120 MB after compression. Another parameter to consider is compression and decompression speed, which varies from device to device based on the device's processor speed. Using

a regular machine, gzip compression speed is approximately 100MB/s, and decompression speed is approx. 440MB/s. Another available algorithm is The BPE algorithm which provides rapid decoding with a high compression ratio [132,133].

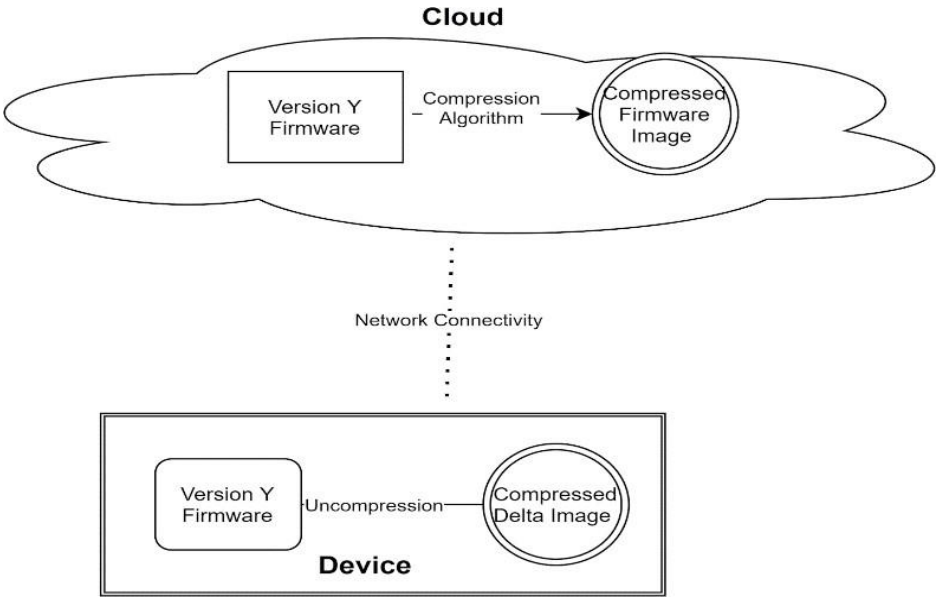

Figure 6.24 FOTA Compression Update

FOTA Delta update (Figure 6.25)

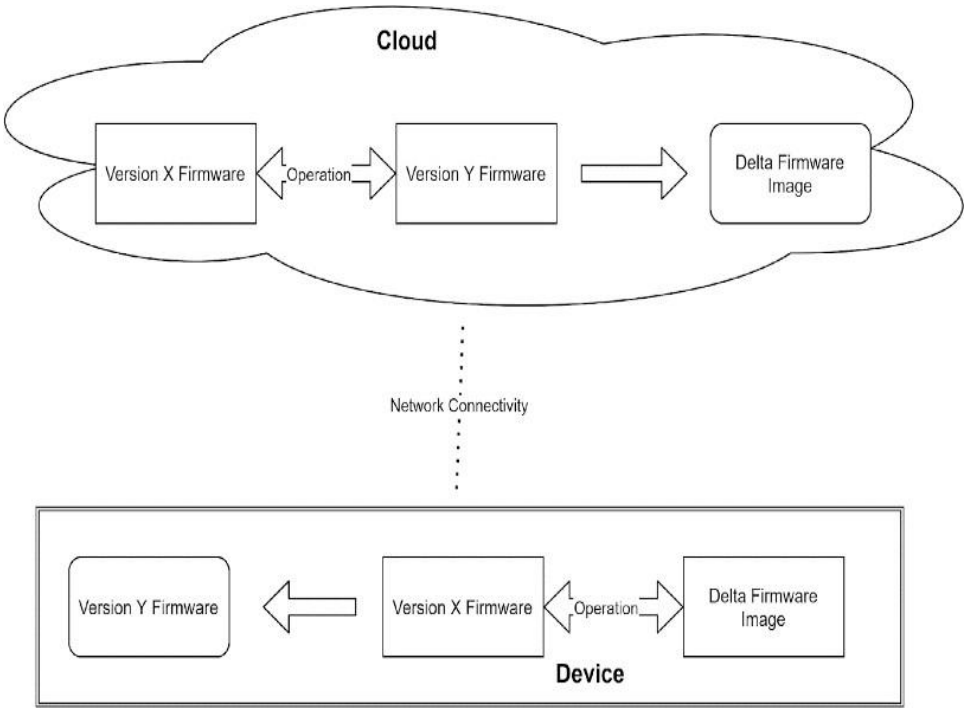

Figure 6.25 Firmware Delta Update

Firmware updates over the air enable the provider to find the essential changes, from an existing firmware version (Firmware version X) to new firmware (Firmware version Y) updated version. And using some specific operation creates a highly compact package called delta update of the updated firmware. This delta image [134] mechanism provides a facility to send minimal required data to save a good amount of network bandwidth. It can save around 30% of bandwidth.

Hybrid Approach (Figure 6.26)

The hybrid approach is the combination of the compressed and the delta approach. It uses both methods, and a compressed delta image sent on a network. It can save around 60% of bandwidth, but it also increases processing speed on both sides (cloud and device). In this method, algorithm complexity also increased. Delta and compression speed depends on the processor speed of the cloud side and gateway/device side.

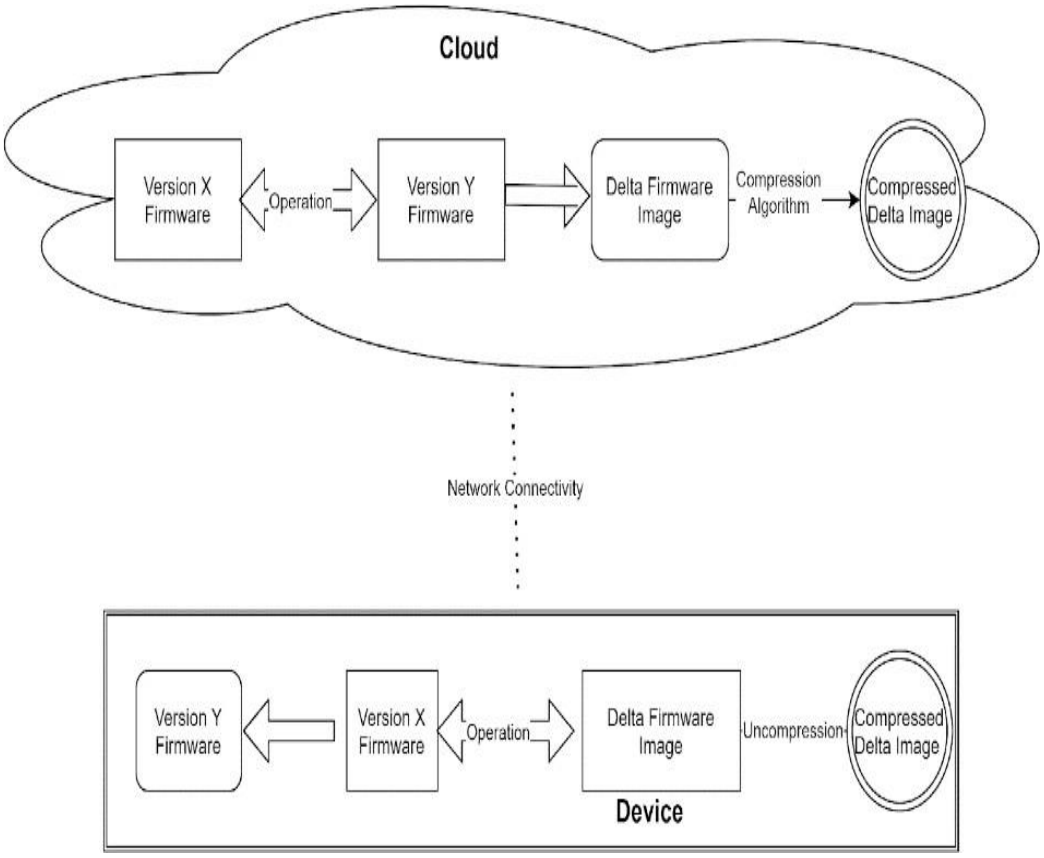

Figure 6.26 Firmware Hybrid Approach Update

Table 6.12 shows the difference between the above-discussed methods and the proposed method.

Table 6.12 Comparison with other FOTA methods

Method / Specification	FOTA Compression Method	FOTA Delta Method	Hybrid Approach	Our Proposed Method
Data Exchange on Network	Medium/High	Medium/High	Medium	Less
Cloud side Processing	High	High	Very High	Low
Gateway/Device side processing	High	High	Very High	High
Usability	Low	High	Low	Noble approach

6.3.3.2 Time Compression

Different vendors discussed different types of FOTA update processes. Standard and Intelligent OTA (Over the Air) is famous [135]. Standard OTA works when devices reconnect with the cloud, while in Intelligent OTA, cloud scans for OTA applicability for devices. The time it takes for a device to receive and install a FOTA (Firmware Over-The-Air) update can depend on a variety of factors. One of the primary factors that can impact FOTA time is the size of the firmware update - generally, larger updates will take longer to download and install. In addition, network speed can play a role in how quickly the update can be downloaded, while the processing power of the device can impact how quickly it can process the update. The availability of the server hosting the update can also affect download time. All the factors discussed above are considered equal when checking FOTA time across all methods.

The following graph (Figure 6.27) shows the time to complete OTA for standard and Intelligent OTA with our proposed method.

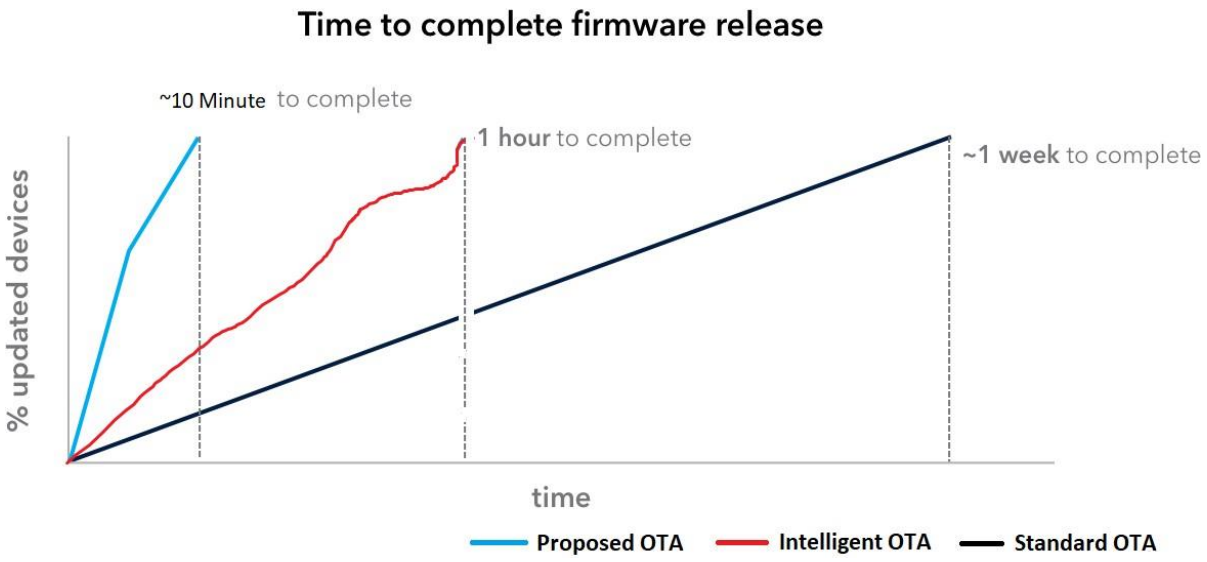

Figure 6.27 Time Comparison graph for different FOTA methods

6.4 Summary

This chapter gives an application of proposed method and discusses the fundamentals of wireless calibration systems, the enabling tools for their implementation. In order to successfully improve the wireless assembly line calibration, a new direction open for improvement. We have shown how the proposed method can save a reasonable amount of time on the assembly line by considering some unique factors that remain unaddressed till now. 802.11 protocol proposed in chapter 5.4 and with uses the testing instrument's signal generation capability to overcome current challenges. It is shown that it is easy to adapt to an assembly line and practical. The technology proposed in this thesis refers to Wi-Fi devices, although it could be intuitive for other wireless technologies such as Bluetooth, UWB (Ultra-wideband), Zigbee, etc. With this approach, Wi-Fi devices can be in the market earliest. This approach also tests other Wi-Fi functionality, and it can help in fundamental Wi-Fi improvement as some other applications of the proposed method are also discussed. The proposed method also reduced complexity at the control PC side.

This chapter also addressed an intermittent use about FOTA update for low powered IoT devices. The swapping of data will diminish from the cloud and device. This benefit can be noticed when the devices are in distant places. Uncertaintyand irregularity under IoT device usage, caused by the low connectivity locations, can

also be minimized with this invention's help. The proposed way is safer as the chance to modulate the firmware binary is significantly less. Because the code is built "on-prem," there is no need to create and maintain the binary for different platforms and operating systems. We have developed a basic template model with standard building blocks and show how the proposed model can be deployed at the production or manufacturing site. We also addressed the security concern of the proposed system and how our system can deal with it.

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

This chapter presents a comprehensive summary of the research work done. It includes the research summary of the work done in Section 7.1, and the limitations of the research study identified in Section 7.2. The chapter also presents the future aspects of the research work performed in Section 7.3 and how the study can help the future researchers in the said domain.

7.1 Research Summary

Some of the major conclusions that can be drawn from the study are:

(i) The proposed solution can save significant power on the STA side. The study shows power consumption for the 802.11n mode with web browsing and idle state. In the wireless network and the Internet of Things world, each Wi-Fi supported 802.11 modes (b, g, n, ac & ax), different bandwidths, and different rates tested for millions of devices. Usually, IoT devices do not require a bulk amount of data transfer, and they send a small amount of data periodically. Therefore, on large-scale proposed model can save significant power. With protocol implementation, the following corner cases are also covered:

   a. Beacon missed via STA

   b. Keep-alive time taken care of

   c. Upper layer timeout taken care of

   d. Delivery of unicast/multicast/broadcast data ensured.

   Moreover, the proposed method saves a good amount of memory at the access point's lower layer side, considerably saving the processor time, which can be used for other tasks. For example, the ARP table can store at the lower layer for fast ARP resolution.

(ii) We have discussed how scan offload to the secondary device can help to improve 802.11 scanning performance by considering some unique factors that remain unaddressed by the existing scan mechanism. The study specifically

identified test cases in which scan offload can help to improve user behaviour, reduce power consumption, and maintain the connection in a roaming environment. Mechanism and advantages discussed.

(iii) In connection improvement, through simulation and results, a new way is proposed for a fast Wi-Fi connection; Things moved from application to driver/firmware level and the number of frames reduced. In this congestion control, from the simulation result, the effect of CW value on overall Wi-Fi throughput is shown. It is shown that how to optimize CW value can work well in the high dense environment. And Access point self is not capable enough to do that without understanding the whole environment. So, AP also needs to talk to each other as they are sharing a common environment. it is described that how an AP shared information about the connected station with other AP in OBSS so a collision can reduce in the highly dense environment. Currently, IEEE 802.11 specification does not define any IE so an AP can talk to other AP.

(iv) Congestion control is a critical aspect of managing network traffic and ensuring reliable communication. Several ways to achieve congestion control discussed in the work, including contention-based, low-power technology, and transmission power feedback mechanisms. In conclusion, an optimal congestion control mechanism for wireless networks involves a combination of these methods to ensure efficient use of network resources, minimize collisions, and improve network performance.

(v) Various Wi-Fi device problem solved via using low-power technology, for example via Bluetooth. Bluetooth is a critical short-range IoT communications Protocol. It has become very needful in computing. A new low-energy Bluetooth is now available, i.e., BLE, an essential protocol for IoT applications. BLE covers the same range as that of Bluetooth; in that same way, it also consumes significantly less power. As we have discussed, our proposed solution can provide the following advantages over standalone Wi-Fi usage:

    a. A regular Wi-Fi station device waking up will not occur on every DTIM, which helps with less energy consumption.

    b. Previously, there was a delay in transmission because of waiting for the STA device to wake up on DTIM; AP can wake up STA without delay.

c. Other packets of Wi-Fi that are connection-related can also be exchanged using this concept.

d. Since the BT channel does not depend on the Wi-Fi operating channel, the BT channel can use to communicate.

e. The BT channel is used for reference here. Other less energy-consuming technology like Zigbee, NFC, UWB (Ultra-Wide Band) etc., can also be used.

f. The proximity data provided by BLE is much more accurate than Wi-Fi, so such a combination (Wi-Fi with Bluetooth) can be advantageous in V2P (Vehicle to Pedestrian) and V2V (Vehicle to Vehicle). As suggested, instead of BT/BLE, UWB can also use in combination with Wi-Fi. Wi-Fi can be used for high throughput things like FOTA, and UWB can be used for positioning systems. UWB uses a multi-sensor using TDOA (Time Difference of Arrival, time of arrival difference) and AOA (Angle of Arrival) positioning algorithm to analyze the accurate position with centimeters accuracy, highest safety and multi-path resolution. The possibility of getting disconnected will also be decreased in a high-volume 802.11 network.

g. The 802.11 connection time will also decrease.

h. As AP can wake up the STA device within a few seconds, there will be no need to buffer data DTIM, leading to shortening the buffer size of AP.

i. Energy consumption will also decrease as 802.11 took much power, saved here.

(vi) Another chapter discusses two topics. First, it addresses an intermittent issue involving the improvement of firmware updates for wireless devices, which can provide valuable insights into the embedded system problem. Second, it presents an application that discusses the fundamentals of wireless calibration systems and the enabling tools required for their implementation which provide a deeper understanding of how the results of the research can be applied in real world.

7.2 Limitations of the Study

The research work done comprised of certain drawbacks and limitations which are discussed as follows:

- The proposed methods are either tested by simulator or physically tested in a system simulator or using prototype Wi-Fi boards, but not in real environments. The proposed methods can experiment with several mobile devices, wireless sensors, notebooks, etc. hence, real-time implementation needs to be done by Wi-Fi chip vendors.
- Wi-Fi packet (Management, Control, and Data) considered to identify valuable scenarios but QoS mechanisms also needs to consider for experimental setup.
- Offloading DHCP frames to MAC layer is limited to DHCP server functionality directly provided by an AP.
- In some cases, the secondary scan radio may not be able to detect interference or other sources of signal degradation, which can affect network performance. Additionally, the secondary radio may not have the same range as the primary radio, which can limit the overall coverage of the network.
- low-power technologies may not be able to handle large numbers of devices, as the limited range and data rates can result in increased congestion and longer delays. As such, congestion control using low-power technologies is best suited for applications that involve a smaller number of devices, with lower data rate requirements, and smaller coverage areas. For example, Bluetooth restricts the number of devices connecting to a single device because of the Logical transport address (LT_ADDR) of the Bluetooth baseband, which only has three bits. Our paper focuses on use-cases where Bluetooth can help, but the increased number of devices depends on chip vendors. Some proposals we can think about like:
    - Divide STA devices into seven groups and the most significant bit of LT_ADDR can be used for the group, and another 2 bits of LT_ADDR can be used to point BT device under the group. So, this way, four times BT device support can increase.

    - Install a greater number of Bluetooth radios under the same SOC. With One Wi-Fi radio, n number of Bluetooth radio can be present on the same hardware. Hardware is a one-time cost while it can permanently save lots of power.

- Limitation of congestion control using collision avoidance is that it does not take into account the amount of traffic generated by different types of devices or applications. As a result, high-bandwidth applications may be given priority over other types of traffic, leading to congestion and reduced network performance for other devices and applications.
- When a device uses a longer listen interval using adaptive listen interval, it will spend more time in sleep mode, resulting in higher latency in receiving data. This can be a problem for user experience, for example watching video can stuck when low battery happens.
- To improve battery usage, STA can where STA takes unfair advantage of the proposed power save method. How will Access Point deal with the situation when it happens?
- Not all devices are compatible with the low power technology used in power save mode, and in some cases, the implementation may not be standardized across different vendors. This can result in interoperability issues and limit the overall effectiveness of the power save mode.

7.3 Future Aspects

The Although the suggested approach significantly improve Wi-Fi performance, it still has some points which must be addressed in the future. Following are the future aspects of the research work performed:

- The use of low-power technology approach should be tested under real environment, hence real-time implementation needs to be done by Wi-Fi/BT chip vendors. Wi-Fi packet (Management, Control, and Data) considered to identify valuable scenarios but QoS mechanisms also needs to consider for experimental setup [136]. Another limitation is with Bluetooth protocol. Bluetooth restricts the number of devices connecting to a single device

because of the Logical transport address (LT_ADDR) of the Bluetooth baseband, which only has three bits. Our paper focuses on use-cases where Bluetooth can help, but the increased number of devices depends on chip vendors.

- During scan offload feature the different device increases the cost, so cost reduction should be taken care of. Hardware can be cheaper with time. In addition, the machine learning approach can decide when to offload the scan to the different device. In future consideration both devices should have the same interpretation for access point RSSI and secondary scan device can be used for other purposes.

- The “adaptive listen interval” proposed method is tested in a system simulator and physically using two Wi-Fi boards, but not in real environments. The proposed method can experiment with mobile devices, wireless sensors, notebooks, etc. Hence real-time implementation needs to be done by Wi-Fi chip vendors. In this method if the STA takes unfair advantage of the proposed method, AP deal behavior can check in future. Currently, IEEE specification does not provide any specific protocol which can change LI. A universal protocol and command can be introduced in the future, providing universal behavior for all the Wi-Fi vendors.

- The proposed DHCP offload approach has the potential to significantly shorten Wi-Fi connection time, it still implemented feature needs support from both side – station and access point, so backward compatibility case needs to tackle by both (Station & AP). Every access point does not have an embedded DHCP server, so in that case, this feature development would be tough. Auth Response and Assoc response consists of failure reason in case anything goes wrong from connection perspective. In case of DHCP failure more reason code needs to be added in Auth Response and Assoc response specification.

- In implementation time mostly success case is covered. It would be interesting to cover all the failures and corner cases in the future.

- Currently, IEEE specification does not provide specific protocol discussed in thesis. A universal protocol and command can be introduced in the future, providing universal behaviour for all Wi-Fi vendors.

References

1. P802.11be Standard for Information technology--Telecommunications and information exchange between systems Local and metropolitan area networks--Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment: Enhancements for Extremely High Throughput (EHT), June 2022.

2. S. Seneviratne et al., "Characterizing Wi-Fi connection and its impact on mobile users: practical insights", WiNTECH, pp. 81-88, 2013.

3. C. Pei, Z. Wang, Y. Zhao, Z. Wang, Y. Meng, D. Pei, Y. Peng, W.Tang, X. Qu, Why it Takes so Long to Connect to a Wi-Fi Access Point?, Cornell University Library, Jan.2017.

4. A. Iyer, C. Rosenberg and A. Karnik, “What is the right model for Wireless channel Interference?” IEEE/ACM Trans. on Networking, vol. 8, no. 5, May 2009

5. M.A. Parvej, S. Chowdhury, N. I. Hia, and M. F. Uddin, “Capture effect on the optimal contention window in IEEE 802.11 based WLANs” in Proc. IEEE ICIEV, 2013.

6. Afaqui, M.S.; Villegas, E.; Aguilera, E. IEEE 802.11 ax: Challenges and requirements for future high efficiency WiF. IEEE Wirel.Commun. 2016, 24, 130–137.

7. Santi, S.; Tian, L.; Khorov, E.; Famaey, J. Accurate Energy Modeling and Characterization of IEEE 802.11ah RAW and TWT.Sensors 2019, 19, 2614. https://doi.org/10.3390/s19112614.

8. Bankov, D.; Khorov, E.; Lyakhov, A.; Stepanova, E. Clock drift impact on target wake time in IEEE 802.11ax/ah network. In Proceedings of the 2018 Engineering and Telecommunication (EnT-MIPT), Moscow, Russia, 15–16 November 2018; pp.1–6.

9. Anand Balachandran, Geoffrey M. Voelker, Paramvir Bahl, and P. Venkat Rangan. 2002. Characterizing user behavior and network performance in a public

wireless LAN. SIGMETRICS Perform. Eval. Rev. 30, 1 (June 2002), 195–205. https://doi.org/10.1145/511399.511359.

10. "G. Castignani, A. Arcia, and N. Montavont, “A study of the discovery process in 802.11 networks,” ACM SIGMOBILE Mobile Computing and Communications Review, vol. 15, no. 1, pp. 25–36, Mar. 2011, doi: 10.1145/1978622.1978626.

11. R. Gupta and V. Singh, "Reduce 802.11 Scanning Time Using Special Device to Provide Scan Results," 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2020, pp. 1276-1278, doi: 10.1109/ICRITO48877.2020.9197842.

12. Lei Wang, “Method and apparatus for accelerated link setup”, US Patent 738,589,Dec 01 2015

13. M. M. Surur and N. Surantha, "Performance Evaluation of Dense WiFi Network Based on Capacity Requirement," in Proc. of 2019 International Conference on Information Management and Technology (ICIMTech), Jakarta/Bali, Indonesia, pp. 466-471, 2019. Article (CrossRef Link)

14. K. Sui et al., "Understanding the impact of ap density on WiFi performance through real-world deployment," in Proc. of Local and Metropolitan Area Networks

15. S. Jin, M. Choi, L. Wang, and S. Choi, "Fast scanning schemes for IEEE 802.11 wlans in virtual ap environments,” Computer Networks, vol. 55, pp. 2520–2533, 07 2011.

16. S. Waharte, K. Ritzenthaler, and R. Boutaba, “Selective active scanning for fast handoff in WLAN using sensor networks,” in Mobile and Wireless Communication Networks (E. M. Belding-Royer, K. Al Agha, and G. Pujolle, eds.), pp. 59–70, 2005.

17. Velayos, H., et al.: Techniques to reduce IEEE 802.11b mac layer handover time. Technical report (2003)

18. Ashraf, F., Kravets, R.H.: Making dense networks work for you. In: 2015 24th International Conference on Computer Communication and Networks (ICCCN), pp. 1–8 (2015)

19. Cicconetti, C., Galeassi, F., Mambrini, R.: Network-assisted handover for heterogeneous wireless networks. In: 2010 IEEE GLOBECOM Workshops (GC Wkshps), pp. 1–5 (2010)

20. R. Syahputri and S. Sriyanto, "Fast and secure authentication in IEEE 802.11i wireless LAN," in Uncertainty Reasoning and Knowledge Engineering (URKE), 2012 2nd International Conference on, aug.2012, pp. 158 –161

21. IEEE standard 802.11. IEEE Std, 802:11r

22. A. Zu and C. Frade, “Pre-Allocation of DHCP Leases: A Cross-Layer Approach,” 2011 4th IFIP International Conference on New Technologies, Mobility and Security, Feb. 2011, doi: 10.1109/ntms.2011.5720663.

23. A. Zúquete and C. Frade, "Pre-Allocation of DHCP Leases: A Cross-Layer Approach," 2011 4th IFIP International Conference on New Technologies, Mobility and Security, Paris, 2011, pp. 1-5, DOI:10.1109/NTMS.2011.5720663.

24. Balasubramanian, N., Balasubramanian, A., Venkataramani, A.: Energy consumption in mobile phones: a measurement study and implications for network applications. In: Proceedings of ACM IMC (2009)

25. J. Manweiler and R. Roy Choudhury, “Avoiding the Rush Hours: WiFi Energy Management via Traffic Isolation,” IEEE Transactions on Mobile Computing, vol. 11, no. 5, pp. 739–752, May 2012, doi: 10.1109/tmc.2011.269.

26. V. Bahl, A. Adya, L. Qiu, E. Shih, and M. Sinclair, “Wake on Wireless - a Case for Multi Radio Wireless LAN,” 2002. Accessed: Jan. 25, 2023. [Online]. Available: https://www.microsoft.com/en-us/research/wp-content/uploads/2016/11/Wake-on-Wireless-A-Case-for-Multi-Radio-Systems.pdf

27. T. Pering, Y. Agarwal, R. Gupta, and R. Want, “CoolSpots,” Proceedings of the 4th international conference on Mobile systems, applications and services, Jun. 2006, doi: 10.1145/1134680.1134704.

28. T. Jin, G. Noubir, and B. Sheng, “WiZi-Cloud: Application-transparent dual ZigBee-WiFi radios for low power internet access,” 2011 Proceedings IEEE INFOCOM, Apr. 2011, doi: 10.1109/infcom.2011.5934951.

29. H. Yomo, Y. Kondo, N. Miyamoto, S. Tang, M. Iwai, and T. Ito, “Receiver design for realizing on-demand WiFi wake-up using WLAN signals,” 2012 IEEE Global Communications Conference (GLOBECOM), Dec. 2012, doi: 10.1109/glocom.2012.6503947.

30. Y. Kondo et al., “Wake-up radio using IEEE 802.11 frame length modulation for Radio-On-Demand wireless LAN,” 2011 IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications, Sep. 2011, doi: 10.1109/pimrc.2011.6140091.

31. Y. Zhang, Z. Song, Y. Tian, and W. Wang, “A Runtime Framework for Context-Sensitive Device-to-Device Communication,” 2017 IEEE 86th Vehicular Technology Conference (VTC-Fall), Sep. 2017, doi: 10.1109/vtcfall.2017.8288152.

32. N. Balasubramanian, A. Balasubramanian, and A. Venkataramani, “Energy consumption in mobile phones,” Proceedings of the 9th ACM SIGCOMM conference on Internet measurement, Nov. 2009, doi: 10.1145/1644893.1644927

33. Fu, X.; Pace, P.; Aloi, G.; Li, W.; Fortino, G. Cascade Failures Analysis of Internet of Things Under Global/Local Routing Mode. IEEE Sens. J. 2022, 22, 1705–1719. https://doi.org/10.1109/JSEN.2021.3133912.

34. Lopez-Aguilera, E.; Demirkol, I.; Garcia-Villegas, E.; Paradells, J. IEEE 802.11-Enabled Wake-Up Radio: Use Cases and Applications. Sensors 2020, 20, 66. https://doi.org/10.3390/s20010066

35. IEEE Std 802.11ba ™ -20 2 1, IEEE Standard for Information Technology, Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 3: Wake -Up Radio Operation , March 2021.

36. Deng, D.-J.; Lien, S.-Y.; Lin, C.-C.; Gan, M.; Chen, H.-C. IEEE 802.11ba wake-up radio: Performance evaluation and practical design. IEEE Access 2020, 8, 141547–141557.

37. Khorov, E.; Kiryanov, A.; Lyakhov, A.; Bianchi, G. A tutorial on IEEE 802.11ax high efficiency WLAN. IEEE Common. Surveys Tuts. 2019, 21, 197–216.

38. Wang, W.; Chen, Y.; Wang, L.; Zhang, Q. Sampleless Wi-Fi: Bringing Low Power to Wi-Fi Communications. IEEE/ACM Trans. Netw. 2017, 25, 1663–1672.

39. Omori, K.; Tanigawa, Y.; Tode, H. Power-saving for wireless stations using RTS/CTS handshake and burst transmission in wireless LANs. In Proceedings of the 2017 14th IEEE Annual Consumer Communications & NetworkingConference (CCNC),Las Vegas, NV, USA, 8–11 January 2017; pp. 708–711.

40. Saeed, A.; Kolberg, M. Towards Optimizing WLANs Power Saving: Context-Aware Listen Interval. IEEE Access 2021, 9, 141513–141523.

41. Lim, T.H.; Rhee, S.H. An Adaptive Power Management Scheme for WLANs using Reinforcement Learning. In Proceedings of the 2019 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 16–18 October 2019; pp. 412–415.

42. Liu, D.; Wang, H.; Zhou, G.; Mao, W.; Li, B. Arbitrating Traffic Contention for Power Saving with Multiple PSM Clients. IEEE Trans. Wirel. Commun. 2016, 15, 7030–7043.

43. Wu, F.; Yang, W.; Ren, J.; Lyu, F.; Yang, P.; Zhang, Y.; Shen, X. Named Data Networking Enabled Power Saving Mode Design for WLAN. IEEE Trans. Veh. Technol. 2020, 69, 901–913.

44. Jin, T.; Noubir, V.; Sheng, B. WiZi-Cloud: Application-transparent Dual Zigbee-WiFi Radios for Low Power Internet Access. Proc. Infocom. 2011, April 2011.

45. Yomo, H.; Kondo, Y.; Miyamoto, N.; Tang, S.; Iwai, M.; Ito, T. Receiver design for realizing on-demand Wi-Fi wake-up using WLAN signals. In Proceedings of the 2012 IEEE Global Communications Conference (GLOBECOM), Anaheim, CA, USA, 3–7 December 2012; pp. 5206–5211. https://doi.org/10.1109/GLOCOM.2012.6503947.

46. Kondo, Y.; Yomo, H.; Tang, S.; Iwai, M.; Tanaka, T.; Tsutsui, H.; Obana, S. Wake-up Radio using IEEE 802.11 Frame Length Modulation for Radio-On-Demand Wireless LAN. In Proceedings of the 2011 IEEE 22nd International Symposium on Personal,Indoor and Mobile Radio Communications, Toronto, ON, Canada, 11–14 September 2011.

47. Zhang, Y.; Song, Z.; Tian, Y.; Wang, W. A Runtime Framework for Context-Sensitive Device-to-Device Communication. In Proceedings of the 2017 IEEE 86th Vehicular Technology Conference (VTC-Fall), Toronto, ON, Canada, 24–27 September 2017; pp. 1–5.

48. Balasubramanian, N.; Balasubramanian, A.; Venkataramani, A. Energy consumption in mobile phones: A measurement study and implications for network applications. In Proceedings of the 9th ACM SIGCOMM conference on Internet measurement, Chicago, IL, USA, 4–6 November 2009.

49. Gupta, P.; Saxena, P.; Ramani, A.K.; Mittal, R. Optimized use of battery power in wireless Ad hoc networks. In Proceedings of the 2010 The 12th International Conference on Advanced Communication Technology (ICACT), Gangwon, Korea, 7–10 February 2010; pp. 1093–1097.

50. Fonseca, M.S.P.; Munaretto, A.; Mendes, C. A resource management framework for 802.11 wireless access networks. Wirel. Netw 2015, 21, 1891–1898.

51. "IEEE Std. 802.11e-2005", Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Quality of Service Enhancements, 2005.

52. M.A. Parvej, S. Chowdhury, N. I. Hia, and M. F. Uddin, “Capture effect on the optimal contention window in IEEE 802.11 based WLANs” in Proc. IEEE ICIEV, 2013.

53. Riyadh Qashi, Martin Bogdan & Klaus Hiinssgen "Case Study: The Effect of Variable Priority Parameters on the QoS of WLANs IEEE 802.11e EDCF" Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference.

54. R. Qashi M. Bogdan and K. Hanssgen "Performance analysis of WLANs for the IEEE 802.11 EDCF in real-time applications " in International Academic Conference of Young Scientists "Computer Science and Engineering 2010

55. J. Sengupta and G. S. Grewal "Performance evaluation of IEEE 802.11 mac layer in supporting delay sensitive services " International Journal of Wireless and Mobile Networks no. 1 2010.

56. T. Nadeem and A. Agrawala "IEEE 802.11 DCF enhancements for noisy environments" in Proceedings of the 15th IEEE International Symposium on Personal Indoor andMobile Radio Communications (PIMRC 2004) vol. 1 pp. 93-97 Sept. 2004.

57. Shao-Cheng Wang and A. Helmy. "Performance Limits and Analysis of Contention-based IEEE 802.11 MAC". Local Computer Networks Annual IEEE Conference on. Vol. 0 pp. 418-425. 2006. USA.

58. T. Yoshiwaka, H. Yomo, and T. Ito, “Wake-Up Channel Selection for On-Demand WiFi Wake-Up Using WLAN Signals,” IEEE Xplore, May 01, 2014. https://ieeexplore.ieee.org/document/7023144 (accessed Jan. 25, 2023).

59. D.-J. Deng, C. -H. Ke, H. -H. Chen and Y. -M. Huang, "Contention window optimization for ieee 802.11 DCF access control," in IEEE Transactions on Wireless Communications, vol. 7, no. 12, pp. 5129-5135, December 2008, doi: 10.1109/T-WC.2008.071259.

60. H. Mishra, R. Gupta, and S. K. Upadhyay, “Systematic review of congestion handling techniques for 802.11 wireless networks,” International Journal of Communication Systems, vol. 33, no. 2, p. e4191, Sep. 2019, doi: 10.1002/dac.4191.

61. S. K. Memon et al., “A survey on 802.11 MAC industrial standards, architecture, security & supporting emergency traffic: Future directions,” Journal of Industrial Information Integration, vol. 24, p. 100225, Dec. 2021, doi: 10.1016/j.jii.2021.100225.

62. Software-Defined Networking for Wi-Fi. White Paper. Available online: https://docplayer.net/9190422-Software-defined-networking-for-wi-fi-white-paper.html (accessed on 10 September 2022)

63. Saldana, J.; Munilla, R.; Eryigit, S.; Topal, O.; Ruiz-Mas, J.; Fernández-Navajas, J.; Sequeira, L. Unsticking the Wi-Fi Client:Smarter Decisions Using a Software Defined Wireless Solution. IEEE Access 2018, 6, 30917–30931.

64. A. Ananthanarayanan and I. Stoica, “Blue-Fi,” Proceedings of the 7th international conference on Mobile systems, applications, and services, Jun. 2009, doi: 10.1145/1555816.1555842.

65. A. Gupta, J. Min, and I. Rhee, “WiFox: Scaling WiFi Performance for Large Audience Environments,” vol. 12, 2012, Accessed: Jan. 25, 2023. [Online]. Available: https://sites.cs.ucsb.edu/~arpitgupta/pdfs/wifox.pdf

66. J. Blobel, F. Menne, D. Yu, X. Cheng, and F. Dressler, “Low-Power and Low-Delay WLAN Using Wake-Up Receivers,” IEEE Transactions on Mobile Computing, vol. 21, no. 5, pp. 1739–1750, May 2022, doi: 10.1109/tmc.2020.3030313.

67. R. Piyare, A. L. Murphy, C. Kiraly, P. Tosato and D. Brunelli, "Ultra Low Power Wake-Up Radios: A Hardware and Networking Survey," in IEEE Communications Surveys & Tutorials, vol. 19, no. 4, pp. 2117-2157, Fourthquarter 2017, doi: 10.1109/COMST.2017.2728092.

68. Xie, Y.; Sun, X.; Chen, X.; Jing, Z. An adaptive PSM mechanism in WLAN based on traffic awareness. In Proceedings of the 10th IEEE International Conference on Networking Sensing and Control (ICNSC), Evry, France, 10–12 April 2013; pp. 568–573.

69. Li, Y.; Zhang, X.; Yeung, K. L. DLI: A dynamic listen interval scheme for infrastructure-based IEEE 802.11 WLANs. In Proceedings of the 2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Hong Kong, China, 30 August–2 September 2015; pp. 1206–1210. https://doi.org/10.1109/PIMRC.2015.7343482.

70. Wi-Fi EasyMesh™ Specification Version 4.0, Nov 2021, [Online]. Available: https://www.wi-fi.org/file/wi-fi-easymesh-specification

71. Meng Li, “Investigating Power Consumption in 802.11 WLANs: Measurement, Visualization, and Improvement,” 2009. [Online]. Available: https://core.ac.uk/download/pdf/225887569.pdf

72. Garcia-Espinosa, Eduardo, Omar Longoria-Gandara, Ioseth Pegueros-Lepe, and Arturo Veloz-Guerrero, "Power Consumption Analysis of Bluetooth Low Energy

Commercial Products and Their Implications for IoT Applications," Electronics, 7(12), p. 386, 2018. Article (CrossRef Link)

73. Firmware Source Code for the Qualcomm Atheros AR9271 USB 802.11n NIC. Available online: https://github.com/qca/openath9k-htc-firmware (accessed on 10 September 2022)

74. NLANR. (2005). Iperf Measuring TCP and UDP Bandwidth Performance. Available online: https://users.informatik.haw-amburg.de/~schulz/pub/Rechnernetze/tools/iperf/iperfdocs_1.7.0.html (accessed on 10 September 2022).

75. Microsoft Windows 10 driver development kit documentation (2017).

76. Wu, A.H.: The Development of WDM Device Drivers Under Windows 2000/XP. Electronic Industry Press, Beijing (2005)

77. Walter, O.: Programming the Microsoft Windows Driver Model. Microsoft Press, America (1999)

78. whitepaper Intel® Centrino® Mobile Technology Wake on Wireless LAN (WoWLAN) Feature.

79. Jia, J., Liu, G., Han, D., wang, J.: A Novel Packets Transmission Scheme Based on Software Defined Open Wireless Platform. Digital Object Identifier https://doi.org/10.1109/ACCESS.2018.2813007.

80. https://learningnetwork.cisco.com/thread/118328

81. markruss, “DebugView - Windows Sysinternals,” docs.microsoft.com. https://docs.microsoft.com/en-us/sysinternals/downloads/debugview

82. R. Sai and K. Vakkantula, “AN4053 Streaming Data Through Isochronous or Bulk Endpoints on EZ-USB ® FX2TM and FX2LPTM.” Accessed: Feb. 05, 2023. [Online]. Available: https://www.cypress.com/file/139866/download

83. A. Pei et al., "Why it takes so long to connect to a WiFi access point," IEEE INFOCOM 2017 - IEEE Conference on Computer Communications, Atlanta, GA, USA, 2017, pp. 1-9, doi: 10.1109/INFOCOM.2017.8057164.

84. S. Seneviratne, A. Seneviratne, P. Mohapatra, and P.-U. Tournoux, “Characterizing WiFi connection and its impact on mobile users,” Proceedings of the 8th ACM international workshop on Wireless network testbeds, experimental evaluation & characterization, Sep. 2013, doi: 10.1145/2505469.2505480.

85. Aviviano, “Miniport drivers - Windows drivers,” learn.microsoft.com. https://learn.microsoft.com/en-us/windows-hardware/drivers/network/ndis-miniport-drivers2 (accessed Feb. 05, 2023).

86. AX1800 Wi-Fi 6 USB adapter, online available at https://www.dlink.com/en/products/dwa-x1850-ax1800-wi-fi-6-usbadapter

87. T. Inaba, K. Ozera, S. Sakamoto, T. Oda, M. Ikeda and L. Barolli, "A Testbed for Admission Control in WLANs: Effects of RSSI on Connection Keep-Alive Time," 2017 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA), Taipei, Taiwan, 2017, pp. 722-729, doi: 10.1109/WAINA.2017.62.

88. S. Radha and N. Ravindran, “A New Solution to Improve the Link Failure Tolerance in Mobile Ad hoc Networks,” IETE Journal of Research, vol. 53, no. 4, pp. 315–327, Jul. 2007, doi: 10.1080/03772063.2007.10876146.

89. Factory floor operations access at: https://docs.microsoft.com/en-us/azure-sphere/hardware/factory-floor-tasks

90. Lukez, John. "Test challenges of Wi-Fi and Bluetooth devices." EE-Evaluation Engineering 49.3 (2010): 30-34.

91. Introduction to a WLAN manufacturing test plan and theory of implementation access at: http://literature.cdn.keysight.com/litweb/pdf/5989-1194EN.pdf

92. K. Gomez, R. Riggio, T. Rasheed and F. Granelli, "Analyzing the energy consumption behaviour of Wi-Fi networks," 2011 IEEE Online Conference on Green Communications, 2011, pp. 98-104, doi: 10.1109/GreenCom.2011.6082515.

93. Calibration process access at: https://download.csdn.net/download/billxy/10147088

94. WiLink8™ Calibrator Tool User's Guide access at: https://www.ti.com/lit/ug/swru562/swru562.pdf

95. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput, IEEE Standard 802.11n, 2009.

96. Szymon Szott, Katarzyna Kosek-Szott, Piotr Gawłowicz, Jorge Torres Gómez, Boris Bellalta, Anatolij Zubow and Falko Dressler "Wi-Fi Meets ML: A Survey on Improving IEEE 802.11 Performance with Machine Learning"- ccs-labs IEEE Communications Surveys & Tutorials, 2022

97. Steer, M. (2010) Transmission Lines. Microwave and RF Design. In: Steer, M., Ed., A System Approach, SciTech Publishing, Raleigh, Chapter 4, 196-197.

98. A.S. V. Medeiros, H. N. Cunha Neto, M. A. Lopez, L. C. S. Magalhães, N. C. Fernandes, A. B. Vieira, E. F. Silva, and D. M. F. Mattos, "A survey on data analysis on large-Scale wireless networks: online stream processing, trends, and challenges," Journal of Internet Services and Applications, vol. 11, no. 1, p. 6, Oct. 2020.

99. M. Morshedi and J. Noll, "A Survey on Prediction of PQoS Using Machine Learning on Wi-Fi Networks," in 2020 International Conference on Advanced Technologies for Communications (ATC), ISSN: 2162-1039, Oct. 2020, pp. 5–11.

100. J. Kunhoth, A. Karkar, S. Al-Maadeed, and A. Al-Ali, "Indoor positioning and wayfinding systems: a survey," Human-centric Computing and Information Sciences, vol. 10, no. 1, p. 18, May 2020.

101. M. Sattarian, J. Rezazadeh, R. Farahbakhsh, and A. Bagheri, "Indoor navigation systems based on data mining techniques in internet of things: a survey," Wireless Networks, vol. 25, no. 3, pp. 1385–1402, Apr. 2019. (visited on 01/27/2021).

102. W. Liu, Q. Cheng, Z. Deng, H. Chen, X. Fu, X. Zheng, S. Zheng, C. Chen, and S. Wang, "Survey on CSI-based Indoor Positioning Systems and Recent Advances," in 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN), ISSN: 2471-917X, Sep. 2019, pp. 1–8.

103. G. Oguntala, R. Abd-Alhameed, S. Jones, J. Noras, M. Patwary, and J. Rodriguez, “Indoor location identification technologies for realtime IoT-based applications: An inclusive survey,” Computer Science Review, vol. 30, pp. 55–79, Nov. 2018.

104. S. Yousefi, H. Narui, S. Dayal, S. Ermon, and S. Valaee, "A Survey on Behavior Recognition Using Wi-Fi Channel State Information," IEEE Communications Magazine, vol. 55, no. 10, pp. 98–104, Oct. 2017.

105. M. Nivaashini and P. Thangaraj, “Computational intelligence techniques for automatic detection of Wi-Fi attacks in wireless IoT networks,” Wireless Networks, vol. 27, no. 4, pp. 2761–2784, May 2021.

106. Kolias, G. Kambourakis, A. Stavrou, and S. Gritzalis, “Intrusion detection in 802.11 networks: empirical evaluation of threats and a public dataset,” IEEE Communications Surveys & Tutorials, vol. 18, no. 1, pp. 184–208, 2016.

107. M. Mamadou, J. Toussaint, and G. Chalhoub, “Survey on wireless networks coexistence: resource sharing in the 5G era,” Mobile Networks and Applications, vol. 25, no. 5, pp. 1749–1764, 2020.

108. S. Bayhan, G. Gür, and A. Zubow, “The Future is Unlicensed: Coexistence in the Unlicensed Spectrum for 5G,” arXiv, cs.NI 1801.04964, Jan. 2018.

109. C. Zhang, P. Patras, and H. Haddadi, “Deep Learning in Mobile and Wireless Networking: A Survey,” IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2224–2287, 2019.

110. B. De Beelde, D. Plets, and W. Joseph, “Wireless Sensor Networks for Enabling Smart Production Lines in Industry 4.0,” Applied Sciences, vol. 11, no. 23, p. 11248, Nov. 2021, doi: 10.3390/app112311248.

111. Liu, Xu, et al. "Performance and Industry Application Test of A High Power Access Point." 2016 International Forum on Management, Education and Information Technology Application. Atlantis Press, 2016.

112. Dialog Semiconductor User Manual DA16200 Mass Production available at: https://www.dialog-semiconductor.com/sites/default/files/2020-11/um-wi-011-da16200_mass_production_user_manual_rev_1v4.pdf

113. Microsoft azure sphere factory-floor operations details available at: https://docs.microsoft.com/en-us/azure-sphere/hardware/factory-floor-tasks

114. Microsoft RF tools details available at: https://docs.microsoft.com/en-us/azure-sphere/hardware/rf-tools.

115. Litepoint IQxel-160 specification available at: https://www.litepoint.com/knowledgebase/iqxel-brochure/

116. MT3620 E-fuse Content Guidelines access at: https://d86o2zu8ugzlg.cloudfront.net/mediatek-craft/documents/mt3620/MT3620_eFUSE_Content_Guideline_V2.1.pdf

117. Anritsu Evaluating WLAN Products Receiver Characteristics available at: https://dl.cdn-anritsu.com/en-en/test-measurement/files/Brochures-Datasheets-Catalogs/Leaflet/mt8862a-leaflet-el8100.pdf

118. MT7612u source code available at: https://github.com/ulli-kroll/mt7612u

119. “Introduction to Wireless LAN Measurements From 802.11a to 802.11ac” access at: http://download.ni.com/evaluation/rf/Introduction_to_WLAN_Testing.pdf

120. Nelson Andrade, Pedro Toledo, Gabriel Guimaraes, Hamilton Klimach, Helga Dornelas, and Sergio Bampi. 2017. Low power IEEE 802.11ah receiver system-level design aiming for IoT applications. In Proceedings of the 30th Symposium on Integrated Circuits and Systems Design: Chip on the Sand SBCCI, Association for Computing Machinery, New York, NY, USA, 11–16. https://doi.org/10.1145/3109984.3110013

121. Wook Bong Lee, Samsung Research “Proposed Draft Text (PDT-PHY): Receive specification: General and receiver minimum input sensitivity and channel rejection”, online, available at https://mentor.ieee.org/802.11/dcn/21/11-21-0013-01-00be-proposed-draft-text-pdt-phy-receive-specification-general-and-receiver-minimum-input-sensitivity-and-channel-rejection.docx.

122. J. Haxhibeqiri, I. Moerman and J. Hoebeke, "Low Overhead, Fine-grained End-to-end Monitoring of Wireless Networks using In-band Telemetry," 2019 15th International Conference on Network and Service Management (CNSM), 2019, pp. 1-5, doi: 10.23919/CNSM46954.2019.9012678.

123. ETSI EN 300 328 V1.8.1 (2012-06) available at: http://www.etsi.org/deliver/etsi_en/300300_300399/300328/01.08.01_60/en_300328v010801p.pdf

124. Rama Sai Krishna Vakkantula Streaming Data Through Isochronous or Bulk Endpoints on EZ-USB® FX2™ and FX2LP™ Sept 2017 access on https://www.cypress.com/file/139866/download

125. Huynh, M. C. A New Over-The-Air Radiated Performance Test System for Multiple-Antenna Wireless Devices for End-of-The-Line Testing in Factories.

126. Wikipedia Contributors, "Hyper-V," Wikipedia, Sep. 20, 2019. https://en.wikipedia.org/wiki/Hyper-V

127. "Configure WiFi Access Points," Ubuntu. https://ubuntu.com/core/docs/networkmanager/configure-wifi-access-points (accessed Feb. 06, 2023).

128. Introduction to Hyper-V on Windows 10 available at https://docs.microsoft.com/en-us/virtualization/hyper-v-on-windows/about/

129. Leshan - an OMA LWM2M implementation in Java, available online at https://github.com/eclipse/leshan

130. OMA Object http://www.openmobilealliance.org/wp/omna/lwm2m/lwm2mregistry.html

131. Trusted Platform Module (TPM), Trusted Compute.Group, Beaverton, OR, USA, 2017. [Online].Available:http://www.trustedcomputinggroup.org/workgroups/trusted-platform-module.

132. R. Kiyohara, S. Mii, M. Mitsuhiro, M. Numao, and S. Kurihara, "A New Method of Fast Compression of Program Code for OTA Updates in Consumer Devices," IEEE Transactions on Consumer Electronics, vol. 55, no. 2, pp.812–817, 2009

133. P. Gage, "A New Algorithm for Data Compression," The C Users Journal, Vol. 12, No.2, pp. 23-38, 1994

134. Arm Firmware image update procedure available at https://developer.pelion.com/docs/device-management/current/updating-firmware/firmware-images.html

135. OTA firmware update available at https://docs.particle.io/tutorials/device-cloud/ota-updates/

136. F. Babich, M. Comisso, M. D'Orlando, and A. Dorni, "Deployment of a reliable 802.11e experimental setup for throughput measurements," Wireless Communications and Mobile Computing, vol. 12, no. 10, pp. 910–923, Sep. 2010, doi: https://doi.org/10.1002/wcm.1026.

137. https://www.ciscopress.com/articles/article.asp?p=102282&seqNum=2

138. Kumar, A., Hussain, J., Chun, A. (2023). Wi-Fi. In: Connecting the Internet of Things . Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-8897-9_3

List of Publications

1. V. Bhargava and N. Raghava, "802.11 practical improvements using low power technology," KSII Transactions on Internet and Information Systems, vol. 16, no. 5, pp. 1735-1754, 2022. DOI: 10.3837/tiis.2022.05.017.
2. Bhargava, V.; Raghava, N.S., “An Enhancement for IEEE 802.11 STA Power Saving and Access Point Memory Management Mechanism”. Electronics 2022, 11, 3914. https://doi.org/10.3390/electronics11233914
3. Vishal Bhargava, N. S. Raghava, “802.11 Wireless devices assembly line process improvement and test time reduction”. communicated Assembly Automation Journal (June 2021) (Communicated).
4. Bhargava, V., Raghava, N.S., “Version Control System Gateway to optimize Firmware over the air (FOTA) update for IoT wireless devices.”. Communicated Wireless Personal Communication Journal (May 2021) (Communicated).
5. Bhargava et al., “A secure IoT gateway framework for fast firmware update” Communicated Malaysian Journal of Computer Science Journal (Dec 2021) (Communicated).
6. V. Bhargava and N. S. Raghava, "Improve Collision in Highly Dense Wi-Fi Environment," 2018 2nd IEEE International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), 2018, pp. 1-5, doi: 10.1109/ICPEICES.2018.8897289.
7. Bhargava, V., Raghava, N.S. (2021). Reduce 802.11 Connection Time Using Offloading and Merging of DHCP Layer to MAC Layer. In: Paiva, S., Lopes, S.I., Zitouni, R., Gupta, N., Lopes, S.F., Yonezawa, T. (eds) Science and Technologies for Smart Cities. SmartCity360° 2020. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 372. Springer, Cham. https://doi.org/10.1007/978-3-030-76063-2_13
8. V. Bhargava and N. S. Raghava, "Offload 802.11 scanning to low power device," 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2022, pp. 1-5, doi: 10.1109/ICRITO56286.2022.9964848.