Universidade de Aveiro
2022

João Francisco Oliveira Soares

Backhaul Steering em Dual Band WiFi Mesh

Backhaul Steering for Dual Band WiFi Mesh

Universidade de Aveiro
2022

João Francisco
Oliveira Soares

Backhaul Steering em Dual Band WiFi Mesh

Backhaul Steering for Dual Band WiFi Mesh

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia de Computadores e Telemática, realizada sob a orientação científica da Doutora Susana Isabel Barreto de Miranda Sargento, Professora Catedrática do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro, e do Doutor Nuno Miguel Abreu Luís, Professor Adjunto do Instituto Superior de Engenharia de Lisboa.

o júri / the jury

presidente / president	Prof. Doutor António José Ribeiro Neves Professor Auxiliar da Universidade de Aveiro
vogais / examiners committee	Prof. Doutor Tiago José Dos Santos Martins Da Cruz Professor Associado da Universidade de Coimbra
	Prof. Doutora Susana Isabel Barreto de Miranda Sargento Professora Catedrática da Universidade de Aveiro

agradecimentos / acknowledgements

Gostaria de começar por agradecer à minha família, especialmente aos meus pais e à minha irmã, pelo incondicional apoio ao longo de toda a minha vida e durante todo o meu percurso académico. Quero demonstrar aos meus pais o meu mais sincero agradecimento pela educação e pela formação que me deram e também por todo o investimento e por toda aposta realizada no meu futuro.

Um especial agradecimento à minha namorada, Beatriz, pelo apoio incondicional em todos momentos, pela preocupação demonstrada, por me motivar e por contribuir para a minha felicidade e bem-estar.

Quero também agradecer a todos os meus amigos pelos bons momentos e pelas memórias criadas ao longo dos anos. Aos meus amigos do laboratório um grande obrigado pelo apoio, pela ajuda e por todos os momentos de descontração partilhados.

Aos meus orientadores, Professora Susana Sargento e Miguel Luís, o meu mais profundo agradecimento por toda a orientação e toda a disponibilidade demonstrada. Quero agradecer também a todas as pessoas envolvidas no projeto onde estive inserido, a todos os membros do grupo de trabalho NAP e ao IT (Instituto de Telecomunicações).

Quero agradecer ao Instituto de Telecomunicações pelas condições de acolhimento e financiamento. Este trabalho é financiado pelo Fundo Europeu de Desenvolvimento Regional (FEDER), através do Programa Operacional Competitividade e Internacionalização (COMPETE 2020) do Portugal 2020 Projeto POWER com o nº 070365 (POCI-01-0247-FEDER-070365).

Palavras Chave

WiFi, Wireless Mesh Network, Backhaul, Optimização, Frequências, Medições, Performance.

Resumo

Ao longo dos anos, a tecnologia WiFi tornou-se uma tecnologia essencial para a maioria das pessoas. O sucesso da indrodução de dispositivos com conectividade WiFi como computadores, smartphones e sensores IoT causou uma maior necessidade de melhores redes WiFi. Como muitas pessoas utilizam estas redes na sua casa, no seu trabalho e em locais públicos, existe uma maior atenção para esta área das telecomunicações. Estas redes começaram a necessitar de melhor serviço e de melhor cobertura. Para resolver este desafio, a WiFi Alliance desenvolveu o WiFi EasyMesh, uma norma que utiliza múltiplos pontos de acesso que permitem ter um setup rápido e fácil (em dispositivos compativeis com WiFi). Esta tecnologia permite criar uma rede mesh para aumentar a cobertura de uma rede WiFi. Para além desta solução, a introdução do standard 802.11ax (WiFi 6) foi bastante importante para melhorar a qualidade dos serviços. Para além do aumento das taxas máximas nas transmições, esta norma também introduziu novas funcionalidades que têm impacto positivo nos utilizadores.

Esta dissertação tem como objetivos mostrar que a utilização de várias frequências (5GHz e 2.4GHz) nas ligações de Backhaul de uma rede mesh podem melhorar o desempenho da mesma, e também desenvolver um mechanismo capaz de alterar e adaptar a rede às suas necessidades. Depois de desenvolver e testar vários cenários, foi comprovado que de facto a utilização de uma ligação em 2.4GHz, além da de 5GHz, poderia melhorar o desempenho da rede. Estes cenários testaram vários aspetos como a quantidade de tráfego na rede, o número de saltos e a frequência utilizada. O mecanismo proposto mostrou ser capaz de identificar cenários onde era necessário alterar a topologia da rede em funcionamento, e também mostrou conseguir realizar as alterações na rede mesh.

Keywords

WiFi, Wireless Mesh Network, Backhaul, Optimization, Frequency Bands, Measurements, Performance.

Abstract

Over the years, WiFi became an essential technology to most people. The success of the introduction of wireless devices with WiFi connectivity like laptops, smartphones, IoT sensors and others, increased the demand for better WiFi networks. Since many people have and use WiFi networks in their homes, jobs and in public places, this is an area that draws a lot of attention. The networks need better service and better coverage. To solve this challenge, WiFi Alliance developed WiFi EasyMesh, a standard for WiFi networks that utilize multiple access points that allow an easy setup and compatibility with WiFi certified devices. This technology is able to create wireless mesh networks over the WiFi protocol to increase the signal coverage. Besides this solution, the release of a new WiFi standard (WiFi 6) was also important. This new release improved the overall rates a WiFi transmission can achieve, and also introduced a set of new features that positively impact the users.

This dissertation's main objectives were to show that the use of several frequencies (2.4GHz and 5GHz) in the backhaul links of a mesh network could increase the overall performance, and to create and test a mechanism that would adapt the network topology when needed. After defining several scenarios and executing them, the obtained results showed that there are in fact situations where the use of a different frequency (2.4GHz) beyond 5GHz would increase the network's performance. The scenarios tested multiple aspects on the mesh network such as the amount of traffic, the number of hops and frequency used by each link. The proposed mechanism showed that it was capable of identifying the need to change the network and actually change it in run-time, so it could benefit from the adoption of a new backhaul topology.

Contents

List of Figures

List of Tables

Glossary

A-MPDU	Aggregated MAC Protocol Data Unit
A-MSDU	Aggregated MAC Service Data Unit
AODV	Ad hoc On-Demand Distance Vector
AP	Access Point
BPSK	Binary Phase-Shift Keying
BSSID	BSS Identifier
BSS	Basic Service Set
BTM	BSS Transition Management
CST	Carrier Sense Threshold
CTS	Clear to Send
DCM	Dual Carrier Modulation
DE	Differential Evolution
DL	Downlink
DPP	Device Provisioning Protocol
DSC	Dynamic Sensitivity Control
EDCA	Enhanced Distributed Channel Access
ELP	Echo Location Protocol
FDE	Fuzzy Differential Evolution
HCCA	Hybrid Coordination Function Channel Access
HE-LTF	HE Long Training Field
HE-MCS	High Efficiency Modulation and Coding Scheme
HE-PPDU	HE Physical Layer Protocol Data Unit
HE-SIG-A	High Efficiency Signal A
HE-SIG-B	High Efficiency Signal B
HE-STF	HE Short Training Field
HE	High Efficiency
IEEE	Institute of Electrical and Electronics Engineers
IPTV	Internet Protocol Television
IoT	Internet of Things
L-SIG	Legacy Signal Field
MAC	Media Access Control
MANET	Mobile Adhoc Network
MCS	Modulation Coding Scheme
MIMO	Multiple Input Multiple Output
MU-MIMO	Multi User Multiple Input Multiple Output
MU	Multi User
NAV	Network Allocation Vector
NRATF	Network Resilience Aware Topology Formation
NSS	Number of Spatial Streams
OBSS	Overlapping Basic Service Set
OFDMA	Orthogonal Frequency-Division Multiple Access
OFDM	Orthogonal Frequency Division Multiplexing
PBC	Push Button Configuration
PHYCCA	PHY Clear Channel Assessment
PHY	Physical
PLR	Packet Loss Rate
PPDU	Physical Layer Protocol Data Unit
PSOHC	Particle Swarm Optimization with Hill Climbing
PSOSA	Particle Swarm Optimization and Simulated Annealing
PTER-ACO	Priority based Trust Efficient Routing using Ant Colony Optimization
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
RCPI	Received Channel Power Indicator
RIFS	Reduced Interframe Space
RSSI	Received Signal Strength Indication
RTS	Request to Send
RTT	Round Trip Time
SA	Simulated Annealing
SNR	Signal to Noise Ratio
SRP	Spatial Reuse Parameter
SSID	Service Set Identifier
STA	Station
STP	Spanning Tree Protocol
SU	Single User
SVM	Support-Vector Machine
TCP	Transmission Control Protocol
TPC	Transmit Power Control
TXOP	Transmit Opportunity
UAV	Unmanned Aerial Vehicles
UDP	User Datagram Protocol
UL	Uplink
URI	Uniform Resource Identifier
WLAN	Wireless Local Area Network
WMN	Wireless Mesh Network
WPA	WiFi Protected Access
WPS	WiFi Protected Setup

CHAPTER 1

Introduction

The access to the Internet has been facilitated throughout the years. The introduction of wireless technology was a big breakthrough and allowed people to access internet services almost anywhere [1]. The number of people that owns mobile devices is increasing, and this increases the number of people that access the internet via Wireless networks (by 2025 it is forecast that 87% of mobile devices will be a smartphone [2]). The wireless network users demand good service and better coverage at the same time. To achieve this, several solutions have been developed and deployed throughout the years.

The improvements in the WiFi technology (IEEE 802.11) have been essential to optimize the communications between the devices in a wireless network. The latest release, known as WiFi 6, had a huge impact, since it increased the maximum rate between two devices to 9.6 Gbits/s, and introduced several new features such as Uplink Multi User Multiple Input Multiple Output (MU-MIMO) and Orthogonal Frequency-Division Multiple Access (OFDMA).

Sometimes, the improvements of the latest releases are not enough to provide the best WiFi experience. One of the most common issues is that the signal coverage may not be as good as the one needed. With that in mind, WiFi Alliance developed a standard called WiFi EasyMesh. This technology is very simple, practical and provides multiple improvements in a wireless network. This technology was designed to increase WiFi signal coverage for residential environments supported by multiple APs. The setup of mesh networks with this standard is very easy because no cables are needed to connect the APs.

The WiFi EasyMesh solution is based on Wireless Mesh Networks that use several frequencies, such as 2.4GHz, 5GHz or even 6GHz, to create backhaul links or to provide service to the devices (fronthaul links). To improve the use of all frequencies, mechanisms have been developed to focus on the fronthaul links, dynamically deciding which frequency to use based on the mesh topology and some other parameters. The backhaul links optimization mechanisms improve the decision of which topology to use, but do not feature the use of several frequency bands.

This work contributes to the evolution of the current solutions developed in the WiFi

EasyMesh technology by showing a different perspective on how mesh network links can be used. The main focus of this dissertation is to analyse the performance of the wireless LAN different topologies, using different frequency bands in the backhaul links, and different traffic usage as well as to improve the way the network assembles and adjusts to changes. Section 1.1 presents the objectives proposed in this work. Section 1.2 enumerates all the contributions of this dissertation. Section 1.3 presents the document organization.

1.1 Objectives

This dissertation has two main objectives centered around the performance of mesh WiFi networks. The first objective is to understand, by designing and testing different scenarios, if the use of different bands simultaneously would contribute to an increase of the performance of a Wireless Mesh Network. The frequency bands used in the backhaul links are 5GHz and 2.4GHz. The second main objective is to receive as input the results from the previous objective, and focuses on the design, implementation and evaluation of backhaul steering algorithms capable of increasing the overall network performance.

1.2 Contributions

Throughout this work several contributions have been made, such as:

- The use of a frequency with different characteristics (2.4GHz) can help in the improvement of the performance a Wireless Mesh Network, mainly in limit scenarios with low Received Signal Strength Indication (RSSI).
- A new set of scenarios were tested for backhaul link optimization.
- It has been developed a mechanism that automatically changes a backhaul link from the 5GHz frequency to the 2.4GHz frequency based on the needs of the network and its current scenario.
- It has been developed a mechanism that automatically changes a backhaul link from the 2.4GHz frequency to the 5GHz frequency based on the needs of the network, its current scenario, and it also avoids excessive steers in the Wireless Mesh Network.

1.3 Document Organization

The organization of the document is the following:

- **Chapter 2 - State of the Art**: it introduces the fundamental concepts of WiFi technology and standards. It also provides an overview of WiFi EasyMesh and a overview of Mesh Network related work.
- **Chapter 3 - Network Measurements over WiFi Mesh**: it describes the testbed, the hardware and the data collection process. An overview of all the metrics is provided for context, and the results are discussed. This chapter identifies the situations where a backhaul steer should be performed to increase the network performance.

- **Chapter 4 - Algorithm for Backhaul Steering**: it presents the proposed backhaul steering algorithm according to the outputs of Chapter 3. An overview of the backhaul steering conditions, as well as the weights for each observed metric, is provided. The tests conducted are described and the results of the performance are discussed.
- **Chapter 5 - Conclusion and Future Work**: it enumerates the conclusions of this work and presents ideas for future work.

CHAPTER 2

State of the Art

The demand for higher coverage and faster and better service in residential networks has increased over the years. WiFi EasyMesh has been seen as a solid approach to this problem. This technology is based on the WiFi (IEEE 802.11) technology.

This chapter starts with a quick overview of the most recent IEEE 802.11 standard, presenting the improvements, the features and details of the Physical (PHY) and Media Access Control (MAC) layers of such version. Section 2.2 presents the WiFi EasyMesh standard, its functionalities and operation modes. Finally, Section 2.3 presents several works addressing the backhaul steering task, and Section 2.4 enumerates the chapter's conclusions.

2.1 IEEE - 802.11

The IEEE 802.11 (commonly known as WiFi) is a standard that specifies protocols to implement Wireless Local Area Networks WLANs. This standard operates in the PHY layer and in the Medium Access Control layer (MAC) of the OSI model.

Several versions have been released to keep improving the overall quality of Wireless Local Area Network (WLAN)s throughout the years. The first version was released in 1997 (known as "legacy") and it featured a 2Mbits/s data rate. The data rate value was increased with the release of the newest versions (see Figure 2.1).

Over the last 32 years (since the first IEEE 802.11 meeting), WiFi has spreaded massively across almost every user's device (from computers to tablets, to smartphones and even to home electronics like televisions, for example). WiFi is now present in every environment from private homes to public spaces like restaurants and airports, to name a few.

This massive adoption shows the success of the technology, but also translates in a demand for higher data rates. The users' needs are getting more demanding due to the use of streaming services, social media and other network applications. To increase the performance of this technology some features were added in the consecutive releases, such as a better use of wireless channels, faster modulation and coding schemes, the introduction of Multiple Input Multiple Output (MIMO) techniques and the use of several antennas (to transmit several

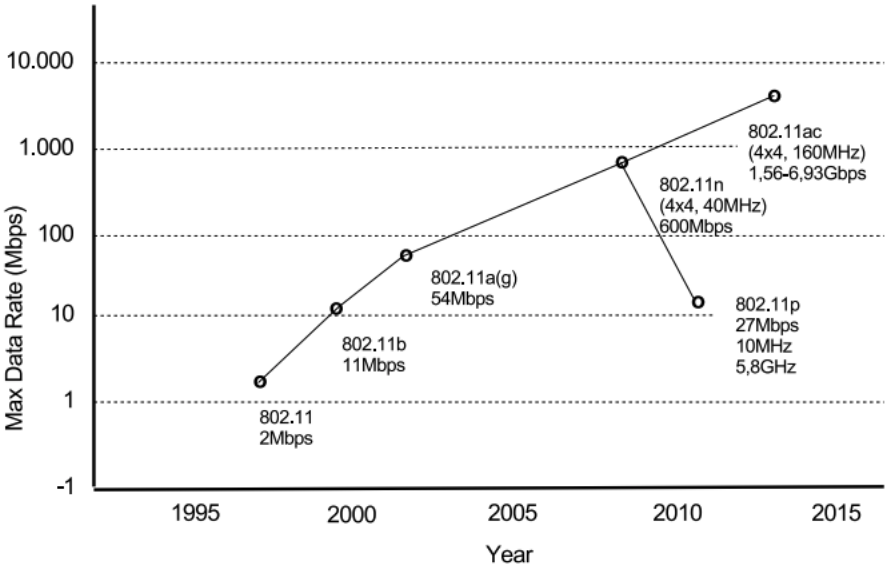

Figure 2.1: Max Data Rate from 802.11 releases [3].

spatial streams between a pair of devices). The latest release is the IEEE 802.11ax, also known as WiFi 6, and it was delivered in 2021. This version features a 9.6 Gbits/s data rate and some breakthrough features that will be explained in detail in the next subsections.

2.1.1 Before IEEE 802.11ax

Over the last years, some amendments have been proposed to increase the nominal data rate of the IEEE 802.11 wireless standard, the most important ones being in the a/b/g/n/ac amendments (see Figure 2.2).

The first ones proposed (802.11 a/b/g) introduced new coding schemes and modulation, and increased data rate from 2 Mbits/s (from the legacy version) to 54 Mbits/s in the 2.4GHz band (802.11g) and in the 5GHz band (802.11a) [5].

The IEEE 802.11n amendment introduced a significant improvement and a really strong step forward for the WiFi standard [6]. Due to the use of new techniques such as the ability to exploit channels with a width of 40 MHz, the usage of higher coding rates and the usage of multiple antennas to transmit up to 4 spatial streams (MIMO), a significant increase in data rates was observed. There were some big changes in the MAC layer like the reduction of overhead in terms of interframe spaces, preambles and control frames. These MAC layer improvements were very important to take advantage of the new PHY layer. In the PHY layer, the IEEE 802.11n introduced a new Reduced Interframe Space (RIFS) and two new aggregation methods, specifically the Aggregated MAC Service Data Unit (A-MSDU)) and the Aggregated MAC Protocol Data Unit (A-MPDU)). The aggregation methods allowed the decoding of some packets even in the presence of small noise bursts which was a big improvement in the transmission reliability.

To reduce the number of collisions, the IEEE 802.11 groups added various contention-free channel access mechanisms. Some solutions were found to resolve the collision problems, but

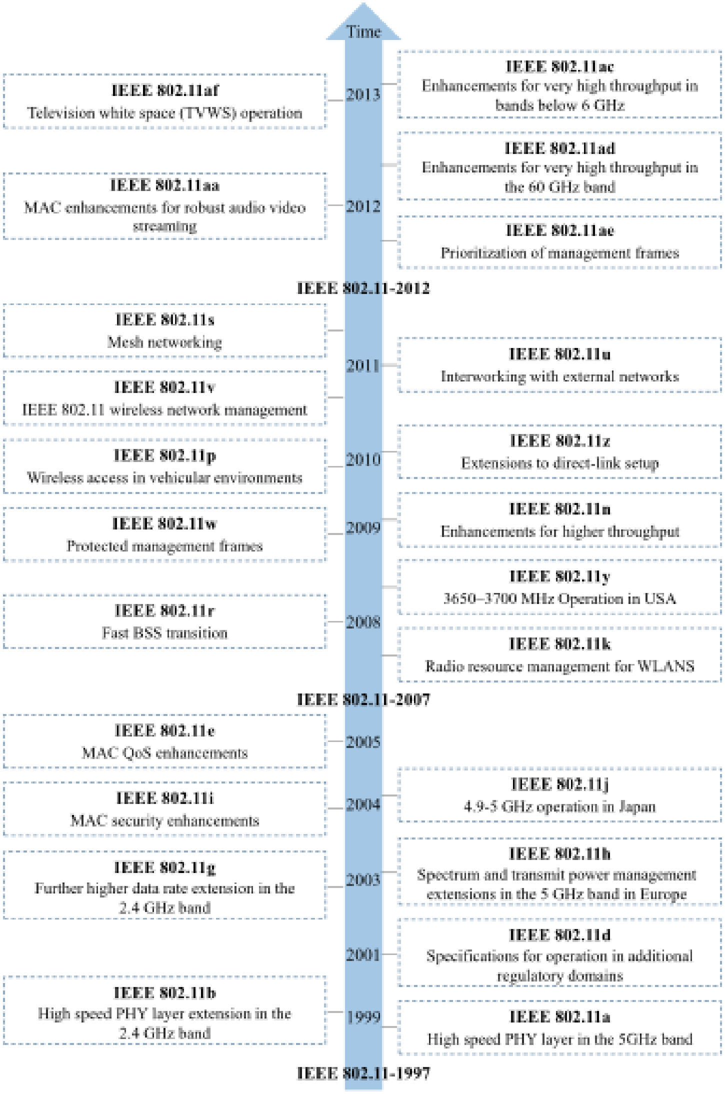

Figure 2.2: IEEE 802.11 Evolution [4]

the deployment of the solutions was never a success due to their inefficiency in scenarios with several Access Point (AP). The problem was partially resolved with the introduction of Hybrid Coordination Function Channel Access (HCCA) Transmit Opportunity (TXOP) Negotiation Mechanism (802.11aa) which allowed different APs to use different time intervals for their transmissions. However this solution only works if the Access Points APs are able to communicate between them.

When it comes to Quality of Service (QoS), the 802.11e was very important because it introduced the Enhanced Distributed Channel Access (EDCA) and HCCA which can differentiate several types of traffic like voice, video and others. EDCA is able to assign priorities to the different types of traffic while HCCA allows to schedule transmissions based on several QoS parameters like packet loss ratio, delay, needed bandwidth and others.

Power saving is a very big deal for some WiFi devices, and the power management, it is done by switching a device between two states (awake and doze). This solution was modified along several 802.11 releases, but the base concept is always the same: switching of the radio of a device on and off.

To conclude, the IEEE 802.11 ac release [7] (the last one before 802.11ax) was introduced with the aim to increase the data rates 10 times when compared to the previous one. The main features were the introduction of Downlink (DL) MU-MIMO, which enables an AP to assign several DL spatial streams to different terminals, widening the transmission bands up to 160 MHz, and the increase of the constellation order to 256-QAM. These improvements allowed the data rate to increase to a theoretical value of 7 Gbits/s.

2.1.2 IEEE 802.11ax

Just like the previous releases from the IEEE 802.11 group, the main objective of the 802.11ax amendment was to increase the nominal rates. To accomplish this, some features were added in the physical and datalink layer [8]. The new PHY protocol includes higher modulation and coding schemes, allowing data rates to be 37% higher (not such a big improvement compared to the growth caused by the 802.11ac). The increase in the user throughput is achieved by a better and more efficient use of the spectrum. Another added feature and perhaps the most important one in the amendment is the introduction of OFDMA (Orthogonal frequency-division multiple access), which is a technique widely used in cellular networks. Apart from these, there were other features that also play a big role in improving WiFi networks. The features will be explained with more detail in the next subsections.

2.1.3 IEEE 802.11ax - Phy Layer

The 802.11ax amendment was launched with some Physical layer enhancements, more specifically in the modulation and in the PHY frame format.

When it comes to the modulation, this amendment inherits several aspects from the previous release (802.11ac). Just like 802.11ac, the new release is grounded on Orthogonal Frequency Division Multiplexing (OFDM) and it allows operations in 20MHz, 40 MHz, 80MHz, 80MHz + 80MHz and 160 MHz.

The duration of OFDM symbols was quadrupled (this increase is very favorable to OFDMA) which makes the transmissions more resilient to inter-user jitter (seen in outdoor scenarios). This change in the duration of the symbols has a big impact in the Uplink Multiple Users transmissions.

This amendment also introduced new modulation techniques in addition to the already used Binary Phase-Shift Keying (BPSK), 16-QAM, 64-QAM and 256-QAM. The first technique is an optional 1024-QAM that should be used in scenarios with perfect channel conditions, usually observed in indoor scenarios. This modulation, combined with error correction codes (convolutional or low-density parity-check), is able to generate data rates with a maximum of 9.6 Gbps (which is only possible with HE-MCS11 with a code rate of 5/6 in a 160 MHz or 80+80 MHz channel with 8 spatial streams) [9].

In addition to the new modulation scheme, the 802.11ax amendment describes an optional Dual Carrier Modulation (DCM) [10]. By allocating the same signal to a pair of tones that are separated significantly apart in the frequency domain, DCM improves transmission robustness.

When it comes to the PHY frame format, the 802.11ax task group defines 4 types of frames, one for the single user transmission, one for the extended range single user transmission, one for the downlink multi-user transmission and one for the uplink multi-user transmission. All of them are based on a frame structure and extend it with specialized fields for each transmission type. In the downlink multi-user case, the frame contains a common preamble defining which tones each receiver should decode to obtain its data field. In a similar process in uplink multi-user transmissions, the preamble is the same for all the Station (STAs), and then, each STA sends the Data fields in a predetermined set of tones.

For all the types of frames, the preamble is duplicated in all 20MHz sub channels in the band used for the transmissions, and it is divided in two parts being the legacy part and the High Efficiency (HE) part. The legacy part is used to enable backwards compatibility and the HE part is used to provide signaling for the new 802.11ax functionality and can only be decoded by devices with 802.11ax (see Figure 2.3). The legacy part contains a field used to synchronize the receiver and the transmitter as well as a Legacy Signal Field (L-SIG) that describes the rest of the frame. The HE part of the frame starts with a repetition of the L-SIG field and has a mandatory High Efficiency Signal A (HE-SIG-A) field and an optional High Efficiency Signal B (HE-SIG-B) field. The HE-SIG-A field contains information about the Modulation Coding Scheme (MCS), the Number of Spatial Streams (NSS), the bandwidth and some other parameters used to be able to correctly decode the rest of the frame. The HE-SIG-A field also carries information like network color, remaining TXOP and a Spatial Reuse Parameter (SRP). In the Uplink (UL) and DL Single User (SU) and in the UL Multi User (MU), the HE-SIG-A field is enough, however in the DL MU case, a HE-SIG-B field is incorporated in the frame preamble. The HE-SIG-B field contains two blocks, one with common information and one with specific user information. The first block (with information to all users) describes the resource allocation of OFDMA, while the second block (with specific information to each user) describes for each resource unit the NSS, the MCS to use and other parameters. In the HE part of the preamble there are also two fields, the HE Short Training Field (HE-STF) and the HE Long Training Field (HE-LTF) that are used for the MIMO. The HE-STF main purpose is to improve the automatic gain control estimation in a MIMO transmission, while the HE-LTF supplies a tool for the receiver to estimate the MIMO channel between the receive chains and the set of constellation mapper outputs.

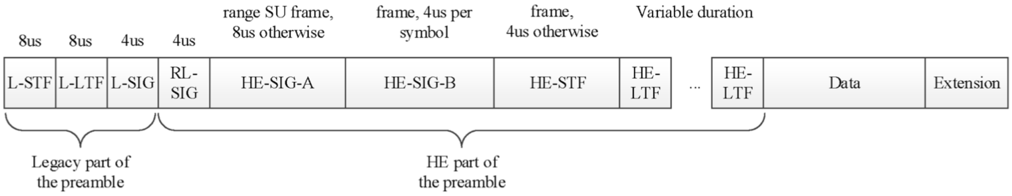

Figure 2.3: 802.11ax PHY frame format [8]

2.1.4 IEEE 802.11ax - MAC Layer

When it comes to the MAC layer, the main objectives of the Task Group AX were the improvement of Spatial Reuse and the improvement of interference management. The improvements in the Spatial Reuse were made possible by PHY Clear Channel Assessment (PHYCCA) modifications, Transmit Power Control (TPC), Basic Service Set (BSS) Color and the use of Multiple Network Allocation Vectors (NAVs).

The legacy versions of IEEE 802.11 use PHYCCA to check if the channel is busy or not. This is done by measuring the energy received. The IEEE 802.11ax decided to embrace the dynamic PHYCCA modifications (see Figure 2.4). These techniques maximize spectrum reuse by allowing multiple concurrent transmissions. In dense deployments, stations may end up assuming that the channel is occupied (due to fixed carrier sensing range), even though many concurrent transmissions may still be viable. This is one of the reasons to add modifications to the PHYCCA. The Dynamic Sensitivity Control (DSC) algorithm was proposed as one of the most important technologies to increase the throughput. The fundamental goal of the DSC system is to optimize the current deployments by appropriately setting the distributed Carrier Sense Threshold (CST) for each node. DSC makes an effort to prevent both very aggressive and conservative behavior, by limiting the increase and drop of CST for a station to a defined area. When paired with the best channel selection, the throughput benefits produced by DSC average more than 20 percent [11] (it increases beyond 40 percent when stations use slow bit rates and send long frames).

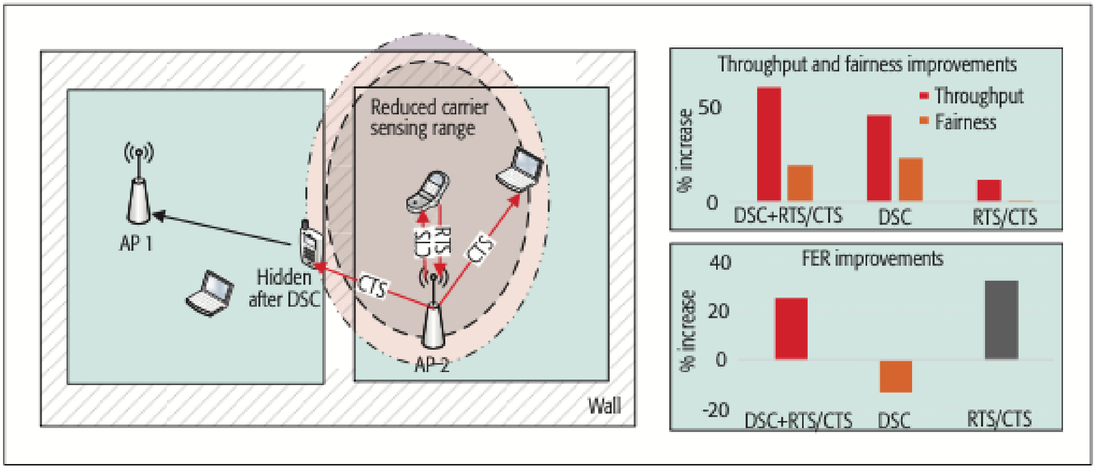

Figure 2.4: TGax proposal for CCA modification and controlled use of RTS/CTS mechanisms [12].

The standardization of per link TPC mechanisms was contemplated to reduce the interference and to increase spatial reuse. The goal of this mechanism is to dynamically tune the lowest possible for stations with the highest path loss to try to reach a target Signal to Noise Ratio (SNR) (good enough to still correctly decode the received frames). The TPC mechanism also changes the transmit power control of non-AP stations based on the RSSI received in the beacon messages (from the associated AP). The IEEE 802.11ax amendment envisions the use of the TPC and the PHYCCA modifications simultaneously to reduce the excessive interference from stations that reduce their carrier sensing range to allow more concurrent transmissions.

BSS Color is an innovative scheme to raise the throughput in dense WLAN networks. To every BSS, a specific color (in-terms of bit defined in the L-SIG field of the PHY header) is assigned. When a station receives frames from a neighbor BSS, it abandons the reception process (assuming that the channel is idle during the transmission) increasing the transmission opportunities. This method was proposed earlier for the 802.11ah standard, but shows good results when applied to the 802.11ax use cases and scenarios.

Virtual carrier sensing is used in 802.11 legacy to fix the collision problems related to hidden nodes. This technique consists in reserving the channel using Request to Send (RTS)/Clear to Send (CTS) handshakes. The neighboring stations, after receiving RTS/CTS frames, set a timer (known as Network Allocation Vector (NAV)) that prevents them from transmitting for a certain time interval. The 802.11ax release adopted the use of two NAV timers. Each station has an intra-BSS NAV and a regular NAV. The intra-BSS NAV is only changed (reseted or increased) by the frames coming from that BSS. The use of these two NAVs increases the spatial reuse by allowing the station to disregard the RTS/CTS frames sent from the Overlapping Basic Service Set (OBSS).

When applied to dense deployments, conventional interference techniques can ease overall network conditions. With this in mind, IEEE 802.11ax aspired to use the RTS/CTS method in a smart way based on channel conditions observed on a per node basis (an AP can use new mechanisms to enable RTS/CTS on its stations). When transmissions are hampered by a hidden node, the affected stations can use this method (remotely enable RTS/CTS) to improve the quality of the transmissions.

2.1.5 Multi-User Enhancements

In the Multi-User techniques, the OFDMA is a strong novelty and it is introduced for Uplink and for Downlink [13]. OFDMA operates on top of OFDM where the subset of carriers are allocated to each user by the base station to accommodate multiple simultaneous transmissions. In dense environments, different stations are allocated to dedicated sub-channels that increase the average user throughput in order to avoid inefficient disputes over shared resources. To reduce the synchronization complexity, the 802.11ax decided to use a specific HE Physical Layer Protocol Data Unit (HE-PPDU) which allows the announcement of scheduling decisions. The methods to allocate RUs at the uplink and the downlink (channel allocation mechanism) are managed by the AP. When transmitting uplink Physical Layer Protocol Data Unit (PPDU)s, IEEE 802.11ax describes an OFDMA-based distributed random access mechanism that randomly chooses resource units supplied by the AP. To start random access at the uplink, a parameter is included in the trigger frame.

Another technique worth mentioning is the Uplink and Downlink MU-MIMO. Although downlink MU-MIMO was already introduced in the 802.11ac standard, the uplink MU-MIMO (triggered with a similar process as in OFDMA transmissions) was introduced in the latest 802.11 release. In a general way, in MU-MIMO transmissions, several stations are overlapped in the same time-frequency resources by exploiting the spatial diversity of the propagation channel allowing multiple stations to communicate simultaneously with the same BSS [14].

The 802.11ax standard also implemented improvements in MU Aggregation, which reduced even more transmissions overhead [15] (MU Aggregation was first introduced in the 802.11n standard).

2.2 WiFi Easy Mesh

WiFi residential networks are demanding better performance because of the use of new applications such as online gaming, virtual reality or high quality video streaming. The improvements made with 802.11ax (WiFi 6) in the PHY and MAC layer are not enough for the average residential environment, therefore WiFi setups with multiple APs are now more common. This type of setups allows better WiFi coverage easing the access to the network by the users (laptops and smartphones) and Internet of Things (IoT) devices. In the configuration of the APs (or extenders) around a house, it is not always possible to provide wired connection to all the nodes, so most of the time the extenders need to connect to each other via a wireless link. In order to standardize the connection and the communications, the WiFi alliance decided to develop a solution called WiFi Easy Mesh [16] based on Wireless Mesh Networks (that uses protocol IEEE 1905), which is used on many released products (routers and extenders).

A Wireless Mesh Network (WMN) consists of a number of wireless nodes organized in a mesh topology [17]. A WMN has several advantages such as being dynamically self-organized and self-configured. It also increases covering range as well as available bandwidth when compared to a regular WLAN.

2.2.1 WiFi Easy Mesh Architecture

In a Multi-AP network there are two types of logical entities: the Multi-AP Controller and the Multi-AP agents. The Multi-AP Controller is a logical entity that controls the fronthaul APs and the backhaul links in the network. The controller acts like a master (Master/Slave architecture) for the network and the others APs, and there can only exist one Multi-AP Controller. The Multi-AP Controller receives the metrics and capability measurements for the fronthaul links, for the backhaul links and for the clients. The controller also provides the onboarding functionality, which allows other Multi-AP devices to enter a Multi-AP network. An example of a Multi-AP can be seen in Figure 2.5.

A Multi-AP Agent is the entity that executes the commands ordered by the Multi-AP Controller (acts like the Slave). A device with a Multi-AP agent has fronthaul APs for the clients to connect and/or a backhaul Station (STA) to support backhaul connectivity (which can be WiFi or Ethernet). There can be more than one Agent in a network and all of them report measurements and capabilities data from the clients, backhaul links and fronthaul links to the Controller or to other Agents. An Agent can change configurations and execute AP control functions only when the Controller orders it.

In a Multi-AP environment, a device can be a Multi-AP Agent, a Multi-AP Controller, or it can be both (a Multi-AP Controller and a Multi-AP Agent at the same time). Two

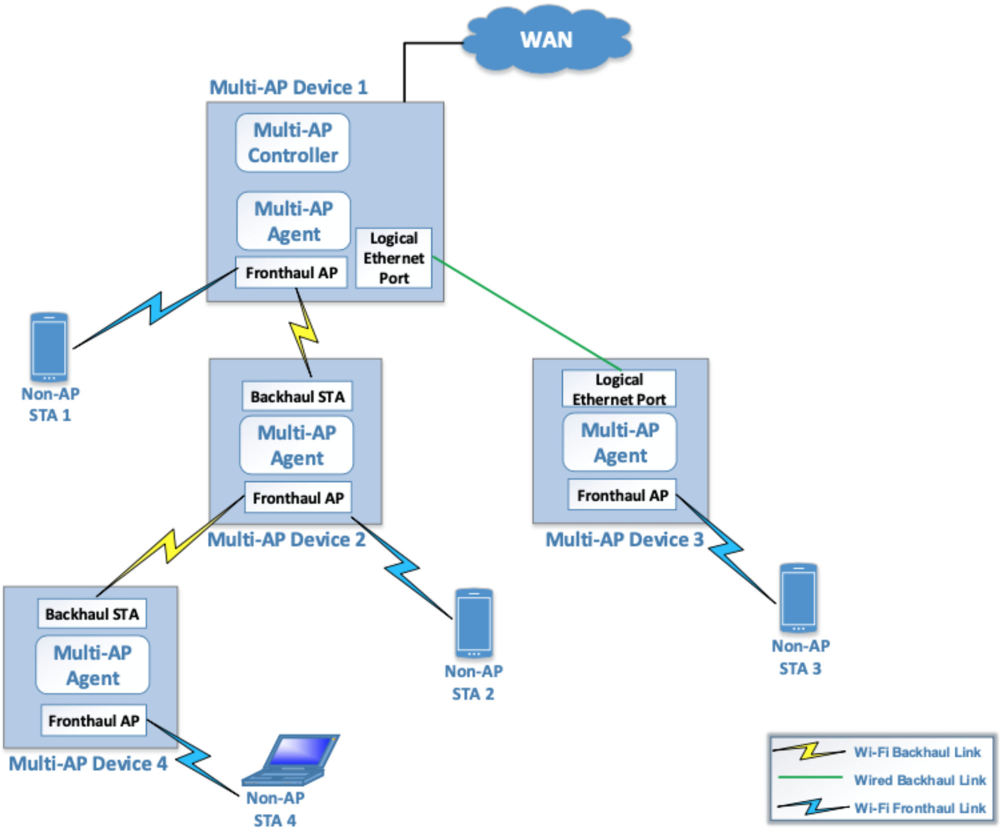

Figure 2.5: Example of a Multi-AP deployment.

Multi-AP Agent devices can be connected to each other via a backhaul link, and there can only be a single link between them at any given time.

Multi-AP devices connect to each other in a tree topology (each “tree branch” can have several devices on it like a daisy chain topology). This ensures that a single backhaul path is established between any two devices in the network.

2.2.2 Multi AP Onboarding

A Multi-AP network must be set up with a single Multi-AP Controller, so normally, if a Multi-AP device has an Agent and a Controller simultaneously, the Controller implementation is disabled and its configuration is saved on the non-volatile memory. This will avoid network onboarding issues later on. For a Multi-AP Agent to discover a Multi-AP Controller, it must send a 1905 AP-Autoconfiguration Search message (the network's specification extends the 1905 AP-Autoconfiguration search/response procedure). After the 1905 message is sent, the Controller responds with a 1905 AP-Autoconfiguration Response message. In these messages some parameters like, profiling information (Multi-AP profiles) and SearchRole (can be Registar and SearchedService) can affect and change how the onboarding occurs.

Onboarding is the set of actions by which a Multi-AP device achieves layer 2 connectivity with a Multi-AP network whether through WiFi or through Ethernet cable. This operation can be done in two ways: the Push Button Configuration (PBC) and the Device Provisioning Protocol (DPP) Onboarding.

The PBC method is used by an enrollee backhaul STA to advertise to a Multi-AP Agent that it is a backhaul STA. This method consists of two steps: BSS configuration and Backhaul STA configuration. In the BSS configuration step, the Multi-AP Agents advertise WiFi Protected Setup (WPS) support, if it is supported, and change some fields in the association request packets such as profiles and fronthaul/backhaul fields to indicate the types of BSS and STAs. The Backhaul STA configuration step starts with a backhaul interface of an Agent exchanging messages during a WPS communication. In these messages, some parameters like a flag to announce that the packet was sent by a backhaul STA, authentication information and credentials, and also backhaul Service Set Identifier (SSID) fields are exchanged to configure the link.

The DPP onboarding procedure allows a Multi-AP Agent that has the DPP Protocol to enter a Multi-AP network through a Controller that implements DPP onboarding. This onboarding method can be done using Ethernet cables or using WiFi (a Multi-AP with the DPP protocol acts like a Proxy). The procedure starts with the Controller receiving bootstrapping information from the Agent (in the form of DPP Uniform Resource Identifier (URI)), and then instructs other Agents to advertise the CCE IE in the beacons and probe responses. After that, when the Controller receives a DPP Presence Announcement frame from an Enrollee Multi-AP Agent, the Controller starts the DPP Authentication. When the authentication is complete, the Multi-AP Agent requests configuration of the network (sending DPP Configuration Protocol Messages).

A Multi-AP can also allocate client devices (STAs) instead of Backhaul STAs. The onboarding process is very similar to the DPP onboarding, but after the authentication is complete, the configuration parameters are set to “sta” and the SSID parameter is set with a BSS that provides fronthaul connectivity for the client.

When a client associates or disassociates itself to/from the network, a 1905 Topology Notification message is sent. These messages allow other Multi-AP Agents to learn of client association, dissociation and re-association events rapidly. The Topology Notification messages are essential for the network to react fast to changes and to prevent outage of services in the STAs.

2.2.3 Capability Reporting and Metric Collection

Capabilities reporting are essential for the Multi-AP Controller and the Multi-AP Agents to acquire radio capability information from Multi-AP Devices, from Backhaul STAs and from Clients.

To report AP capabilities, AP Capability Query and AP Capability Report messages are sent between the Multi-AP device and a Controller or Agents. After an AP Capability Query, a Multi-AP sends an AP Capability Report Message where the basic radio properties are included. All of the report messages must be sent within a second.

On the AP Capability Report Message, parameters like High Throughput support for

802.11 (WiFi), Very High Throughput, High Efficiency Capability (WiFi 6), Traffic Separation or Multiple Virtual Radio Operation Support are encapsulated on AP HT Capabilities fields. Properties like Firmware Version, Software Version, Multiple Execution Environments Support, Channel Scan Capabilities and Channel Availability Check may be included on a report message depending on the type of Multi-AP instance (Multi-AP Agent or Multi-AP device).

The reporting of Client capability and Backhaul STA capability is very similar to AP capability. In the Client/Backhaul Report message, the radio capabilities of Clients/Backhaul STAs are sent to the Controller or to another Agent.

To collect Backhaul link metrics, the 1905 link metric information dissemination protocol is used to query and communicate the information needed. A backhaul link is a link between a Multi-AP Agent AP interface and a Multi-AP Agent Backhaul STA interface, or between two Multi-AP Agents connected by Ethernet. When triggered, a Multi-AP Controller or Multi-AP Agent must send a 1905 Link metric query message to a Multi-AP Agent. The Multi-AP Agent will respond with a 1905 Link metric response message. In this response, the Controller will be able to see metrics like *macThroughputCapacity*, *linkAvailability* or RSSI. The *macThroughputCapacity* is the estimated MAC data rate for the backhaul link (downlink or uplink) if the 100% of channel air time and BSS bandwidth were available. The *linkAvailability* field is the prediction of air time that the backhaul link would consume given the current channel conditions. The RSSI is calculated using the Received Channel Power Indicator (RCPI) of a number of PPDU received.

To collect metrics from an AP in a Multi-AP network, a Multi-AP Controller sends an AP Metrics Query message which must be replied with an AP Metrics Response message. The response message is sent to the Controller and has several useful parameters about AP Metrics, AP Extended Metrics, Radio Metrics, Associated STA Link Metrics and Associated STA Traffic Metrics.

The most important parameters for AP Metrics are:

- BSS Identifier (BSSID) of a BSS
- Channel Utilization
- Number of associated STAs

The most important parameters for AP Extended Metrics are:

- Unicast Bytes Sent Received
- Unicast Bytes Received
- Broadcast Bytes Sent
- Broadcast Bytes Received

The most important parameters for Radio Metrics are:

- Radio Unique Identifier
- Noise
- Transmit/Receive

The most important parameters for Associated STA Link Metrics are:

- Last Data Downlink Rate

- Last Data Uplink Rate
- Utilization Receive
- Utilization Transmit

The most important parameters for Associated STA Traffic Metrics:

- Bytes Sent
- Bytes Received
- Packets Sent
- Packets Received

Channel scanning is also a very useful tool to obtain information on the status of the medium. Channel scanning can be performed when the network devices boot or when the Multi-AP Controller requests it (Channel Scan Request Message) to the Multi-AP Agents in its network. The response from the Multi-AP Agent transmits information from the usage of the medium and from the number of devices and their usage of the channels.

2.2.4 Client Steering

Steering STAs between BSSs in a Multi-AP network is done using Multi-AP control messages. The control messages enable steering functionality on clients that support 802.11v BSS Transition Management (BTM), and also on clients that do not support that standard. The client steering can be performed by a Multi-AP Controller or by a Multi-AP Agent.

A Multi-AP Controller uses the Steering Mandate mechanism to order a Multi-AP Agent to try to steer one or several associated STAs. When triggered, the Controller sends a Client Steering Request message to an Agent. Upon receiving the message, a Multi-AP Agent checks if the STA (that needs to be steered) supports BTM and if the STA's MAC address is the BTM Steering Disallowed STA list. If the STA supports BTM and is not on the disallowed list, then the Multi-AP Agent transmits a BTM Request frame with the BSSID, Operating Class and the Channel Number of the target BSS. If the conditions are not fulfilled, the Agent may try to steer the STA by blocking it from all the other BSSs but the target BSS; and if the STA does not leave the BSS, the Agent might send a Disassociation or Deauthentication frame to the STA (as a last resource procedure).

The Steering Opportunity mechanism allows the Multi-AP Controller to define a time window where a Multi-AP can steer a STA between BSSs. To create a time window, the Controller sends a Client Steering Request message with the Request Mode bit set to zero (which means Steering opportunity) and with a field that specifies the time window itself. When a Agent receives a request, a 1905 Ack message must be sent as response, and if the time window is still open, a Steering Completed message has not been sent and the STA is not on the local steering disallowed STAs list, the Multi-AP Agent may proceed to steer the STA. The Agent checks if the STA (that needs to be steered) supports BTM and if the STA's MAC address is in the BTM Steering Disallowed STA list. If the STA supports BTM and is not on the disallowed list, then the Multi-AP Agent transmits a BTM Request frame with the BSSID, Operating Class and the Channel Number of the target BSS. If the conditions are not fulfilled, the Agent may try to steer the STA by blocking it from all the other BSSs but the target BSS, and if the STA does not leave the BSS, the Agent might send a Disassociation or Deauthentication frame to the STA (as a last resource procedure). This last resource procedure might also be used if a STA that supports BTM sends a BTM response with Reject status code.

The other important steering mechanism is RCPI based. For this mechanism to be used, the Controller of the network sets policies or rules that allow an Agent to perform a steer when needed. When the uplink or downlink RCPI of STA drops below the value defined in the rules, the Agent attempts to steer the STA. This is a reactive mode of operation and is triggered when the quality of a link gets degraded. The Agent needs to find another BSS that is suitable for the STA. To find the new BSS, the Agent takes into account the most recent link metrics collected, received link metrics from the STA, the Controller and other Agents, and the rules or policies for RCPI Steering. After the target BSS is discovered, the Agent checks if the STA (that needs to be steered) supports BTM and if the STA's MAC address is the BTM Steering Disallowed STA list. If the STA supports BTM and is not on the disallowed list, then the Multi-AP Agent transmits a BTM Request frame with the BSSID, Operating Class and the Channel Number of the target BSS. If the conditions are not fulfilled, the Agent may try to steer the STA by blocking it from all the other BSSs but the target BSS, and if the STA does not leave the BSS, the Agent might send a Disassociation or Deauthentication frame to the STA (as a last resource procedure). This last resource procedure might also be used if a STA that supports BTM sends a BTM response with Reject status code.

2.2.5 Backhaul Optimization

In a Multi-AP network, a Multi-AP Agent connects the backhaul STA to a BSS to obtain backhaul connectivity. In this procedure, the Agent can choose the BSS from different candidates during onboarding, which means that the Backhaul configuration might not be the best one possible or the one that the Multi-AP Controller desires. Therefore, the Controller might want to move the Agent to a different BSS.

To make an Agent move to another BSS, the Multi-AP Controller shall send a Backhaul Steering Request Message (to the Agent). When an Agent receives a Backhaul Steering Request, it must reply with a 1905 Ack message and then attempt to connect to another BSS. When the association process is completed or 10 seconds have passed since the reception of the Backhaul Steering Request, the Multi-AP Agent must send a Backhaul Steering Response with the result of the operation encoded in that same message.

When a Multi-AP Agent performs a Backhaul Steer, there might occur a brief data interruption on the STAs attached to that Agent due to the changes on the data path. In some cases, the Fronthaul STAs (network terminals) associated to a BSS might experience full connection interruption if the steer Backhaul STA changes its channel to the channel being used by a fronthaul BSS.

2.3 RELATED WORK

With the evolution of Mesh Networks, it is possible to have an extended WiFi network with better rates, better latency and better overall coverage. Several systems, techniques and methods have been studied and developed throughout the years in order to upgrade and to improve the performance in this type of wireless networks. Backhaul optimization has a huge impact on the network; therefore, it is a big research topic. In other technologies based on mesh networks, there was also an effort to make improvements and some of the concepts explored can (or could) be applied to WiFi Mesh Networks. The main objectives of those researches are:

- Improve the effectiveness of the routing algorithms.
- Improve the backhaul link quality.
- Improve the service quality for terminals.
- Improve the topology of the mesh network.

In [18], there is a different solution from WiFi EasyMesh to establish a mesh network called B.A.T.M.A.N. This solution is a proactive routing protocol for Wireless Ad-hoc Mesh Networks (commonly used for Mobile Adhoc Network (MANET)s) that maintains information about all the nodes existence that are accessible via single-hop or multi-hop. The main strategy is to determine, for each destination of the mesh, a single-hop neighbor that can be used as a gateway (with better performance). This solution tries to calculate the best next hop which make it fast and efficient. To determine the best neighbor, B.A.T.M.A.N. maintains information about the best neighbor calculating metric based on lost packets information between links. The newest version of B.A.T.M.A.N. (B.A.T.M.A.N. Advanced V [19]) introduced Echo Location Protocol (ELP) to collect and update information about old and new neighbors, and also changed the routing decision from packet loss-based to throughput-based (calculates the maximum throughput a link can handle at a given time). In both B.A.T.M.A.N. Advanced IV and V, the time and the processing power needed to calculate the best route is small; however, using only one or two metrics may not be enough to have good accuracy in this type of decisions.

In [20], one of the biggest challenges in WMN is addressed: a node placed in the right position can have a huge performance gap when compared to a node placed in a poor position. The node can have a better link rate, better energy consumption as well as better overall connectivity. To overcome the node placement problem, this study performed several tests using different global optimization search algorithms such as Simulated Annealing (SA), Differential Evolution (DE) and Fuzzy Differential Evolution (FDE). The results show that the FDE algorithm was the best of the tested algorithms. The results also show that FDE achieved the best minimum results, and that a better node placement can improve the performance of the network. Studies like this one are important to optimize the capabilities of the nodes of the network, and sets a base for future investigation.

Although some solutions for node placement have already been developed or discovered, there is always room for improvements. In [21] other algorithms for the node placement problem are implemented and tested. Two hybrid systems consisting of a combination of

Particle Swarm Optimization with Hill Climbing (PSOHC) [22] and combination of Particle Swarm Optimization and Simulated Annealing (PSOSA) were proposed. Both systems were tested in a simulation environment, and the results showed success in solving the placement challenge. The results also showed that the PSOHC system converged to a solution faster than PSOSA. Comparing to a more simple implementation like the FDE, the new algorithms have a better performance and show that different approaches to the problem can result in new and better solutions.

In [23] the focus is in mesh networks with IoT sensors. It is also mentioned that in WMN some nodes (malicious nodes) may transmit compromised and manipulated data to the destination nodes. Therefore, the study proposes a Priority based Trust Efficient Routing using Ant Colony Optimization (PTER-ACO) to achieve data privacy and reliable routing on the WMN. The proposed solution was compared to other popular routing decision methods like the Dijkstra Algorithm (Improved version), Round Trip Time (RTT) based detection and RTS in parameters such as energy latency, network throughput, packet loss rate, computational overhead and latency. The results showed that using PTER-ACO method improves the security, reduces the energy consumed and reduce the Packet Loss Rate (PLR) in malicious nodes.

In [24], a cross-layer routing optimization mechanism is proposed. The mechanism is based on the Ad hoc On-Demand Distance Vector (AODV) protocol, and the main objective is to solve network congestion caused by node load imbalance. This mechanism uses criteria from three different layers being the MAC layer, the physical layer and the network layer. After collecting important information from each of the mentioned layers, all of the metrics are combined into a final layer which then is used to calculate the best routing path. The most important metrics that are used are the number of hops, the current load in the node and the *phy* cost (path). The simulation results showed lower data frame loss rate, lower latency and better throughput, therefore, better routing performance.

The study in [25] proposed a congestion aware multipath routing protocol called EAOMDV-LB. This protocol is based on an AirTime link metric and AirTime Congestion Aware time (both metrics are obtained from radio scans on interfaces). Other metrics used to determine the level of congestion (Rate, parcel misfortune proportion, intra/between stream obstruction, etc) are collected at high frequency, and are used to calculate costs and other metrics. The objective is to redirect the traffic from congested links to other available links and results show positive results. The results of this study outperform the other mechanism to avoid link congestion. Although the results are positive, this protocol can not be applied to multi frequency setups.

The topology used in a mesh network can have a big impact on the network performance. The work in [26] presents a topology optimization method. The algorithm starts by finding the heuristic value for all possible links and it calculates the minimums for the mesh. This algorithm is very fast and light (when speaking of computational power), and has a complexity of N^2 (N being the number of nodes in the mesh). The typologies that are obtained in the algorithm are good enough to be used right away most of the times and a good base to start from otherwise. Running this method gives a better mesh topology that can later be optimized with routing protocols and other decision making mechanisms. This protocol is not able to react to changes in the mesh network, and this lack of adaptation capability is a huge downside. The work developed does not contemplate the use of different frequencies in the mesh links.

In [27] the topology optimization is addressed in a completely different scenario. The WMN is applied to Unmanned Aerial Vehicles (UAV) and, because of that, links must be established based on reliability (crucial in this type of application) to prevent failure. The study proposed two Network Resilience Aware Topology Formation (NRATF) methods that were centralized-NRATF and distributed-NRATF. The methods consist in calculating the 3D position of near by nodes, collecting environmental metrics and then using all the information to calculate the failure probability of the link. After having the link probabilities, the centralized-NRATF method chooses the best links to create a topology, while distributed-NRATF discards the worst links and does all the calculations again until the best topology is found. Both methods outperform the methods used at the time, and each one has some small advantages over the other. These methods are currently applied to UAV (or drones), but can be adapted to other scenarios and mesh networks. The developed work is focused on preventing the failure of the links and, because of that, the connection may not be as good as it can.

In [28] the backhaul links in the WMN do not all transmit in the same frequency (these networks are known as heterogeneous wireless backhaul networks), therefore the topology optimization is more challenging. This study developed a method to optimize the formation of the mesh topology. It consisted in a master/slave architecture (to allow better communication between nodes) that started by collecting several radio metrics from all the interfaces available in the network, and then proceeded to calculate several costs. After that, a variation of the Dijkstra is applied to obtain the best topology. Results show that the proposed solution was able optimize the network topology (topologies with up to 10 hops) even with sub-optimal link conditions. The solution was also resilient to node failures and other types of issues within the network. Future work on this mechanism aims to analyse and optimize QoS policies. The mechanism presented also needs more testing, since the results were obtained in limited and controlled scenarios.

Our previous work with Altice Labs developed both a backhaul and fronthaul link optimization mechanism for WMN. These solutions are being used for WiFi EasyMesh compatible devices. The fronthaul solution optimizes the connections between APs and decides which AP a terminal should connect to and which frequency should it use (2.4GHz or 5GHz) [29]. The backhaul optimization mechanism decides which topology must be used to obtain better performance. This mechanism does not use multiple frequencies for backhaul link and it is not very dynamic. Although the overall performance in the mesh networks is good, there is room for improvement in certain scenarios. These improvements can be achieved with the inclusion of other frequencies for backhaul links.

Machine learning has many applications. In [30] it is used to make fronthaul steering decisions. The main objective is to always have a client connected to the AP that will provide a better quality connection. The steering challenge was addressed as a classification problem; thus, a Support-Vector Machine (SVM) classifier and kernel perceptron classifier were trained based on various network features. The trained SVM model achieved 95% of classification accuracy in identifying the steering conditions. The kernel perception model did not achieve the same level of SVM model, but still produces very good results. This model uses less resources and is faster making it possible to use for real-time steering. The model performs steering instructions in the fronthaul links of a mesh network only. The model cannot be used in the backhaul links because these links are not tolerant to errors.

In [31] backhaul optimization in heterogeneous mesh networks is researched. The objective of this work was to optimize the backhaul of cellular networks. The links used can be optical links between BSS or mmWave links. The authors implemented a policy which takes into account the traffic demands, the user distribution and also the power used. The decision of what link to use is done every time slot. The authors concluded that the presented policies enhanced the performance (and lowered the power consumption). This work showed that backhaul optimization can be done in several mesh technologies; however, due the differences of the used links, the proposed policies cannot be used (but will be taken into account) for other types of transmitting technologies. The properties of the links where this optimization algorithm is applied are very specific and can limit the adaptation to other technologies.

2.4 FINAL REMARKS

This chapter presented the overview of the fundamental concepts used during the dissertation's development. An overview of the characteristics, challenges and provided services was presented for the 802.11ax standard as well for WiFi EasyMesh. The chapter ended with a presentation of the work done in this area that improved the current technology or will/can still improve it.

From the analysis of the WiFi EasyMesh standard and from the developed work in the area, it can be concluded that backhaul optimization was not explored enough and that the use of several frequency bands for backhaul links is rarely taken into account. In order to justify the need for better backhaul link optimization and the need for the use of different bands, multiple scenarios need to be tested and to be analysed. Those tests and conclusions taken from them will be addressed in chapter 3.

CHAPTER 3

Network Measurements over WiFi Mesh

Information about the current state of a WiFi Mesh Network is crucial to guarantee a better service to all stations. Metrics measurements allow the mesh Controller to change the topology in the fronthaul links as well as in the backhaul links. A Multi-AP Controller receives several metrics (RSSI, Phy Rate and others) about the connections between the nodes in the network, processes them and, if needed, triggers changes in the network.

In the fronthaul links, changes in the BSS of a STA and/or the frequency band in use are normal; however, the same does not apply to backhaul links, where changes in the BSS are not as regular, and changes in the frequency band are never seen because usually the links are established solely over the 5GHz band.

This work studies the behavior of a Wifi Mesh Network with backhaul links established in the 2.4GHz and 5GHz frequencies (each link transmitting only on one frequency at a given moment). The addition of the 2.4GHz band introduced some instability and volatility to the metric collection methods.

Several tests and scenarios were defined (using 5GHz backhaul links and both 2.4GHz and 5GHz backhaul links) to see if the introduction of another frequency band could be beneficial to the network.

This chapter presents a description of the hardware used in the work, and will explain the testbed used and all of the changes and workarounds that were made to obtain data from the new mesh network. All the scenarios, tests and collected metrics will be addressed and discussed. Section 3.1 presents all the used hardware, as well as its characteristics. Section 3.2 presents the used testbed and all the metric collection process. Section 3.4 presents all the defined and tested scenarios, and the obtained results. Section 3.5 enumerates the chapter's conclusions.

3.1 HARDWARE

This section consists in the description of the Hardware setup for the WiFi Mesh Network and some extra hardware used to collect the needed metrics.

Section 3.1.1 presents the network nodes (WiFi Smart Extenders) and their properties, and Section 3.1.2 presents the hardware used in the metrics collection process.

3.1.1 Network Node - Extender

To establish a WiFi Mesh Network several nodes were used. These nodes, called EXT-D2260G (see Figure 3.1 and 3.2), were manufactured by Altice Labs S.A. and were equipped with 802.11ax (WiFi 6) capable chips.

Figure 3.1: Exterior look of EXT-D2260G.

Figure 3.2: Interior of EXT-D2260G

The EXT-D2260G emits on two frequencies, 2.4GHz and 5GHz. The 2.4GHz band can be used with dual stream and can achieve a theoretical value of 600 Mbps. The 5GHz can be used with four streams and can achieve a theoretical value of 4800 Mbps. In total the EXT-D2260G can have an aggregated rate of 5.4 Gbps. The 2.4GHz band has a maximum transmit power of +20 dBm, while the 5GHz reaches a maximum of +30 dBm. It also has two GbE LAN ports for wired communications. There is also a serial port that can be used to access to the Extender console. The Extender is powered by a 12V power supply.

This extender supports several IEEE 802.11 standards such as 802.11a/b/g/n/ac/ax, 802.11r fast roaming, 802.11e (WiFi Multimedia), 802.11v and 802.11k. It supports Internet Protocol Television (IPTV), MIMO, Beamforming and EasyMesh. When it comes to security, the nodes support WiFi Protected Access (WPA), WPA2 and WPS (which is used to pair with other nodes and devices).

3.1.2 Metric Collector - APU

The agent used to collect network metrics is implemented using a PC Engines APU. As shown in Table 3.1, the APU runs a Linux distribution operative system which makes it possible to run developed code and scripts. This allows the connection to an Extender to gather all the information and metrics needed to analyse the system's performance. The storage, the memory and the processor present in the APU allows it to process and to save all the collected information. The APU has three Ethernet ports, where one is used to connect with the Extender for metrics collection.

Table 3.1: PC Engines APU characteristics.

Processor	AMD G series gx-412T 1GHz quad Jaguar core with 64bits
Memory	4 GB DDR3-1333 DRAM
Storage	200 GB
Operative System	Linux - Debian 10
Linux Kernel	5.7.10+

3.2 TESTBED

The introduction of an extra frequency (2.4GHz) to be used as backhaul link introduced a new level of complexity for simple in-network tasks such as steers and metric collection. The network became more unstable to work with (because it was a non-final version), turning it more buggy and more prone to errors.

Section 3.2.1 shows, in depth, a view on how the network nodes are configured, how links between nodes are established, and how backahul steering between 2.4GHz and 5GHz are performed. Section 3.2.2 addresses the (new) implemented metric collection methods and some of the challenges encountered, as well as an overall view of the testbed.

3.2.1 2.4 GHz and 5 GHz Backhaul Links

To enable the 2.4GHz and 5GHz frequencies simultaneously for backhaul, several changes were done. In order to make changes in the NVRAM, each Extender was set as an Extender or as a Root AP (there can only be one Root AP in a network). Each frequency is associated to a separate interface, and each interface has sub-interfaces (virtually created) that split backhaul connections from fronthaul connections. Usually, the 2.4GHz frequency is associated with the wl1 interface and the 5GHz frequency is associated to the wl0 interface.

The Root Extender creates two detached networks, one public for fronthaul connections and one hidden for backhaul connections. Stations connect to the network through the public SSID (fronthaul links). The other Extenders can establish a backhaul link to the Root Extender connecting to the hidden SSID using WPS (pressing the WPS button on both nodes), or creating a wired connection between the nodes (after the pairing is done, the connection maintains through wireless communication even if the Ethernet cable is removed).

When two nodes in the WMN connect to each other, two connections get established, one for the 2.4GHz frequency and one for the 5GHz frequency. As mentioned before, only one of the links can be used at a given moment for all the Extenders except for the one defined as the Root AP. By default, the active link is the 5GHz one. Figure 3.3 is a visual representation of an established backhaul link.

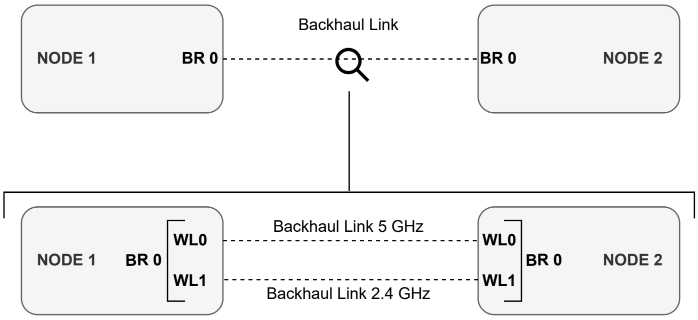

Figure 3.3: Backhaul Link between two nodes.

All the different interfaces in an Extender are combined in a Ethernet bridge. Checking the status of the ports in the bridge will allow to know which backhaul link is being used. The status can be forwarding, blocking or disabled. If the interface wl0 has the state forwarding and the interface wl1 state is blocking or disable, that would translate in the use of wl0 and the blockage of wl1. If both interfaces state is forwarding, the interface chosen to transmit is the one that has more priority (the values and order of priorities can be changed in the Spanning Tree Protocol (STP) of the bridge).

In order to force traffic to go through a specific frequency, one must force the traffic to go through a specific interface. As the interfaces are combined in a bridge port, there is a need to manipulate the behavior of the bridge and its Spanning Tree. The fastest and more efficient way (although not the most elegant) to do that is to shutdown the active interface and to activate the inactive one (sometimes this final step is not needed). This process will make sure that the interface that was transmitting changes its status in the bridge to blocking, and the other stays or changes to forwarding. All of the steps in this process are made by executing commands on the Extender's console or by external scripts.

3.2.2 Metrics Collection

The changes that came with the utilization of the Extender on dual band mode had a huge impact on the existing methods for metrics collection. Due to the use of an Extender as a Root AP (and not an actual gateway), the extraction of the data from the network was not possible. To collect the required data another solution was developed which will be described below.

Each Extender is able to capture and generate all the metrics at a rate of three seconds. To access this information several commands must be executed in the Extender's console. The Root AP accumulates more metrics than an Extender because it gathers information, not only about the direct links it has, but also about the overall state of the mesh network. Some of the commands retrieve metrics from the drivers (information about the interface or the state of the medium), and others retrieve metrics from the Multi-AP controller present in the Root AP.

Because of the lack of storage in the Extender, the outcome of the called commands had to be save somewhere else. The decision was to store the data in APUs. Since the Extenders were not able to run python or other high-level programming languages (limitations on the Linux core), the output of a command had to be stored without any type of processing or parsing.

The execution of commands in the Extender needed to be called by an external machine. With that in mind, it was decided to connect an APU to each Extender via an Ethernet cable (Figure 3.4 shows an example of a possible functional testbed). The APU would then execute a script that establishes a secure connection through SSH between itself and the Extender. The decision to use one APU for each Extender was due to the high traffic volume observed during the metrics collection capable of congesting the wireless links. With the use of some libraries, it was made possible for the APU to execute commands on the Extender and to store the output on a file within the APU's file system. In the APU, a watchdog was deployed to react every time the file or the directory was changed. When triggered, the watchdog would call several functions to parse the text in the file, to organized the information and to save it in a different location (Figure 3.5 illustrates the metric collection process).

The base setup is composed of three Extenders and three APUs (with a wired connection between an Extender and an APU). The mesh network can have different topologies, traffic and other small changes. When executing tests, the APU executes a script with a loop where the metrics are collected and processed. All of the metrics stored have an associated timestamp to allow synchronization when processing the data.

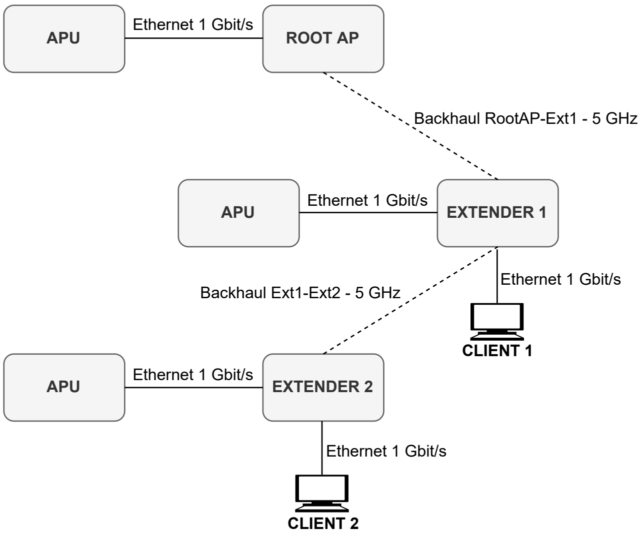

Figure 3.4: Possible setup for metric collection in a Wireless Mesh Network.

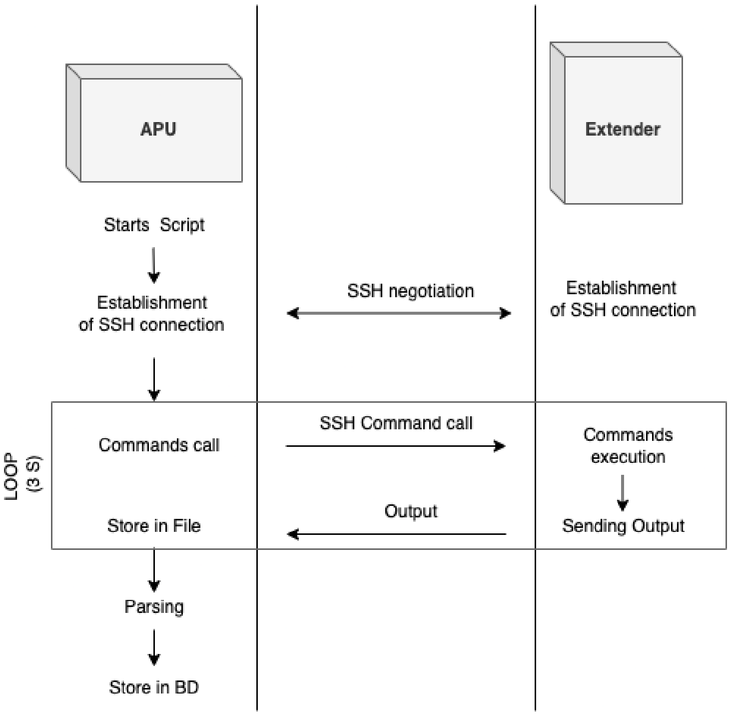

Figure 3.5: Metric collection process between an APU and an Extender.

3.3 METRICS

As mentioned in Section 3.2.2, metrics can be retrieved upon executing commands on an Extender's console. The metrics that are collected by an APU during the execution of scripts can be divided in two groups: the ones provided by the WiFi Easy Mesh Controller, and the ones extracted from the Extenders' drivers.

The main metrics retrieved from the mesh controller were the following:

- RSSI (Received Signal Strength Indication) - measurement of the power present in a received signal. The value is expressed in dBm (decibel-milliwatts) and is associated to each physical link in the network.

- Uplink Phy - is the maximum theoretical value of uplink, in a physical link, at a given moment (represented in Mbits/s).
- Downlink Phy - is the maximum theoretical value of downlink, in a physical link, at a given moment (represented in MBits/s).
- Maximum Phy - is the maximum theoretical value a physical link can ever have (this value is always the same and is not affected by network conditions) (represented in Mbits/s).
- Air Time - the percentage of the medium occupied by a physical link.
- Rt Rate - the amount of traffic flowing through any link (physical link) originated from retransmissions (represented in Mbits/s).
- Ud Rate - the amount of traffic flowing through in the uplink direction at a given moment in a physical link (represented in Mbits/s).
- Dd Rate - the amount of traffic flowing through in the downlink direction at a given moment in a physical link (represented in Mbits/s).

The main metrics retrieved from the Extender's drivers were the following:

- Phy Rate - the maximum theoretical value of a physical link, at a given moment (represented in Mbits/s).
- Retries - the percentage of traffic that needs to be retransmitted (due to errors in the transmission).
- MCS (Modulation and Coding Scheme) - the index used to represent a combination of modulation and coding scheme. Each combination has a unique index.
- NSS (Number of Spatial Streams) - the number of spatial streams being used at a given time.
- TX - percentage of time the station transmitted.
- Inbss - percentage of time a station or Extender is receiving from its own BSS.
- Obss - percentage of time a station or Extender is receiving from other BSS on same/overlapping channel.
- TXOP - percentage of time unused (channel was free and device had the opportunity to transmit but was not used by any).
- GoodTx - number of successful transmissions.
- BadTx - number of unsucessful transmissions.
- Glitch - number of glitches ucode/phy detected.
- Badplcp - number of plcp errors ucode/phy detected.
- Noise - noise in the medium measured in dBm at a given moment.
- Idle - percentage of time spent by the current device inactive.

The metrics that provide information about network rates (Rt Rate, Dd Rate, Ud Rate) are useful to see the current flow of packets and to see which links can have better throughput. Metrics such as Air Time, Air Use, TX, TXOP, Inbss, Obss and Noise can be used to know the channel's state. It is possible to know if there is noise, which devices are generating the noise, how much time a device is transmitting and how much time it is available for new transmissions. Other metrics such as BadTx, GoodTx, Glitch, Badplcp or Retries provide information about the quality, the number of errors and the type of errors of the transmission.

3.4 Scenarios

With the available setup several tests were designed to assess the performance of the network when 2.4GHz and 5GHz backhaul links are used. It was decided to use four different scenarios to observe different behaviors in the WMN. The scenarios are described in the following sections. During the execution of these scenarios iperf3 was used with several arguments such as, -b argument to defined the generated traffic and -u to force UDP traffic when needed.

3.4.1 Scenario 1

Topology

The first scenario consists in a daisy chain topology, using solely 5GHz backhaul links, and a Client, denoted as Client 1, connected to the middle Extender, denoted as Extender 1, through Ethernet. The topology used in Scenario 1 is illustrated in Figure 3.6.

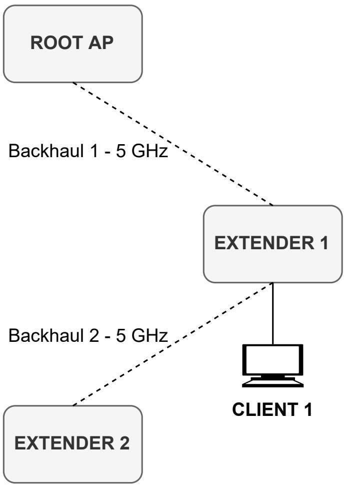

Figure 3.6: Topology of Scenario 1

To generate traffic between the Extender 1 and the Root, iperf 3 was used. To perform the test, an iperf 3 server was initiated on the Root AP and an iperf 3 client was started on the terminal (Client 1). The iperf 3 client was configured to generate as much traffic as possible for 10 uninterrupted minutes. On the backhaul link between Extender 1 and Extender 2, there was no load applied (only the control messages were sent in the link). During the execution of the iperf load, test metrics were collected and stored for analysis.

Results

In this scenario, the throughput values observed in the first link averaged 339 Mbits/s when performing the tests using Transmission Control Protocol (TCP), and 283 Mbits/s using User Datagram Protocol (UDP), as shown in Figure 3.10. Values using UDP had higher variations which then translated in a lower throughput average value. Jitter values (only available in UDP traffic tests) averaged 0.18 milliseconds (as illustrated in Figure 3.8).

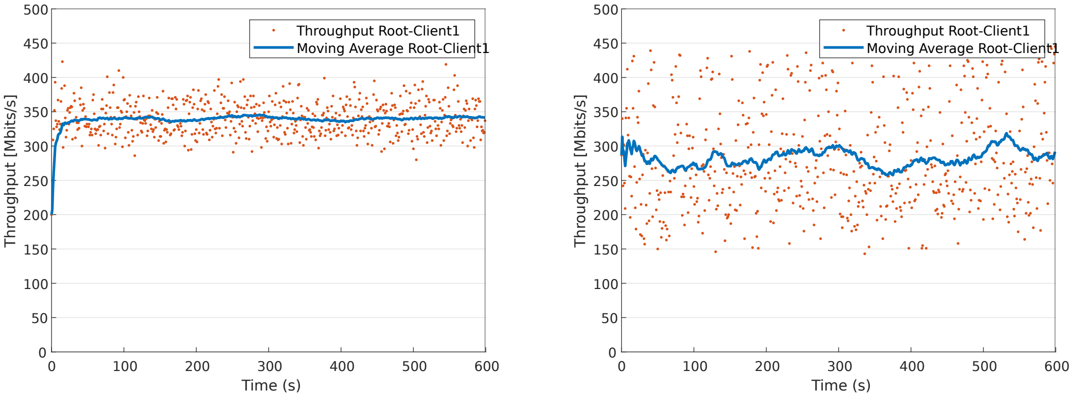

Figure 3.7: Throughput in the backhaul links of Scenario 1 (TCP and UDP).

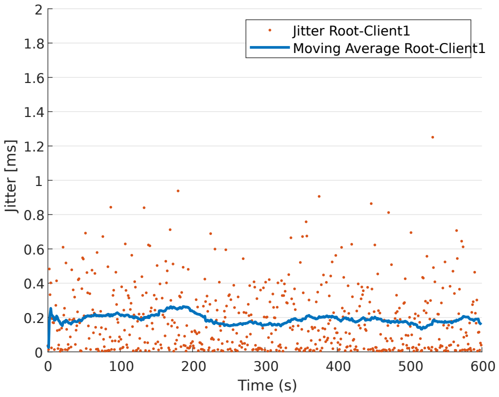

Figure 3.8: Jitter in the backhaul links of Scenario 1 (UDP).

After analysing the remaining metrics from the TCP test, presented in Table 3.2, it was observed that in the medium access metric Air Time the values averaged the 10 to 11 % with no to small variations. The TXOP values observed were 16.5 % when the backhaul link was overloaded with packets (when the link had no traffic the value sets around the 96 % mark), which translates in less opportunities for other devices to transmit. The number of detected errors (measured by BadTx, Glitch and Badplpc) was relatively small, showing that although the backhaul link was getting overloaded, the quality of the transmissions did not suffer much deterioration. When it comes to RSSI, it had a very stable value averaging -59 dBm, and the channel noise had an average value of -78.2 dBm (with almost no deviation). The Inbss and Obss values showed little to no interference from other stations in the network, and also showed stability in the connection during the whole test.

Analysing and comparing the values from the UDP test, it was observed that most of the values were similar (but not so good) to the ones observed in the TCP test. Just like seen in the throughput test, most of the metrics in this test had higher standard deviation which resulted in slightly worst performance. The traffic in the network occupied the medium more as it can be seen in the Inbss values, that had a higher value (82), and in the TXOP value, that had a lower value (9). Comparing the UDP test to the TCP test, the Idle value was also reduced from 32 to 13, which means that the device was active more time (to transmit

Table 3.2: Statistics regarding the metrics collected during the 10 minutes experiment in Scenario 1, for both TCP and UDP (backhaul link RootAP/Extender1).

	TCP		UDP	
Metric	**Average**	**Standard Deviation**	**Average**	**Standard Deviation**
Phy	1594	160.4	1337	136
uPhy	872.9	182.06	870	182.06
dPhy	1611	223.5	1579	249.3
Air Time	10.94	2.2	0.45	0.2
Retries	14.29	17.6	21	20
MSC	5.43	0.91	4.7	1.02
NSS	2.74	0.34	2.7	0.45
RSSI	-59	0.44	-59	0.45
Rt Rate	16.74	9.98	22.4	12.1
Tx	16.6	2.21	4.7	0.77
Inbss	52.1	7.24	82	19.23
Obss	2.05	0.2	2.06	0.3
Txop	16.59	1.69	9	20.03
GoodTx	7.01	1.18	1	0
BadTx	1.14	0.35	0.98	0.12
Glitch	28.9	85.65	16.1	23.05
Badplpc	1.5	5.34	3.3	17.88
Noise	-78.2	0.7	-78	0.73
Idle	32	8.94	13	19.4

less). All these differences in the metrics are justified by the protocol's behavior, one is connection oriented (TCP) and the other is connectionless (UDP). It was also seen that the traffic injected during the test had no negative (drastic) effect on the CPU utilization of the Extenders.

In a general way, all of these values give a reference to how a backhaul link behaves when pushed to the limit. The values show how the channel is used, how much data can be transmitted, the quality of the transmissions, and also allows the comparison with the theoretical values. It is also possible to see the differences the transport layer protocol can have on the performance of the network during this type of test.

3.4.2 Scenario 2

Topology

After knowing how a backhaul link behaves when subjected to large amounts of traffic, the second scenario was designed to observe how the addition of a hop would affect the performance. Just like in the previous scenario, the topology of the WMN consisted in three nodes connected in a daisy chain topology. All the active wireless links between the Extenders were set to the 5GHz frequency. This time a terminal was connected to the last Extender (Extender 2) of the daisy chain using Ethernet to focus only on the wireless backhaul links of the network. The topology used in this scenario is represented in Figure 3.9.

To generate a load that goes through all of the backhaul links, the iperf3 server was initialized on the Root Extender and the iperf3 client was initialized in Client 2. The iperf3 client was configured to send as much traffic as possible for 10 uninterrupted minutes. Metrics

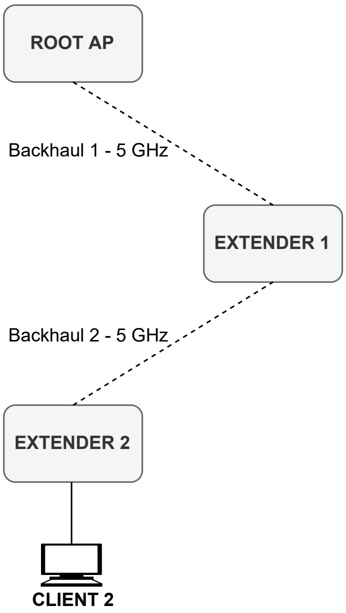

Figure 3.9: Topology of Scenario 2

from the two links were collected during the duration of the whole test and were stored for later processing and analysis.

Results

In this scenario, the throughput values from the transmission between the terminal (Client 2) and the Root AP (where the iperf3 server was running) averaged 127.5 Mbits/s using TCP and 123.2 Mbits/s using UDP. The jitter values observed in the UDP test averaged the 0.32 ms. The introduction of another link (and another hop) reduced the performance by half, approximately.

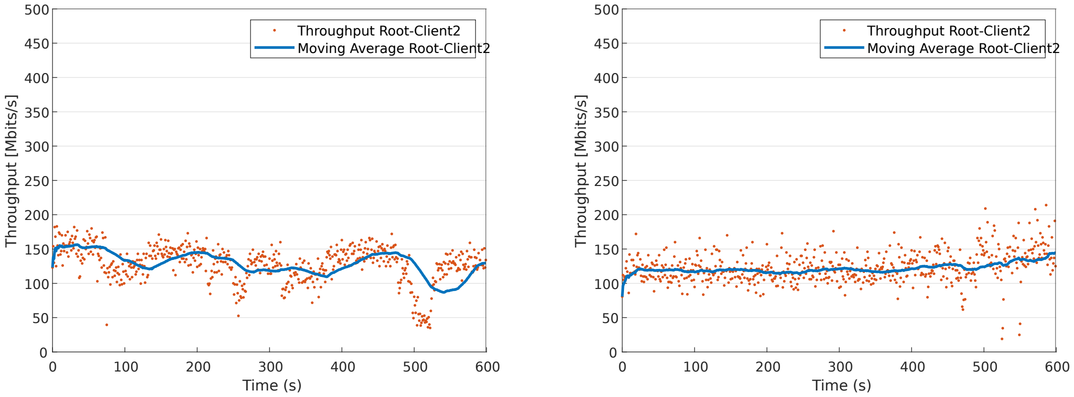

Figure 3.10: Throughput in the backhaul links of Scenario 2 (TCP and UDP).

The theoretical values of the links show that the backhaul link between Extender 1 and Extender 2 would not be able to keep up with performance of the backhaul link between the Root AP and Extender 1. The extracted values from this scenario can be seen in Table 3.3.

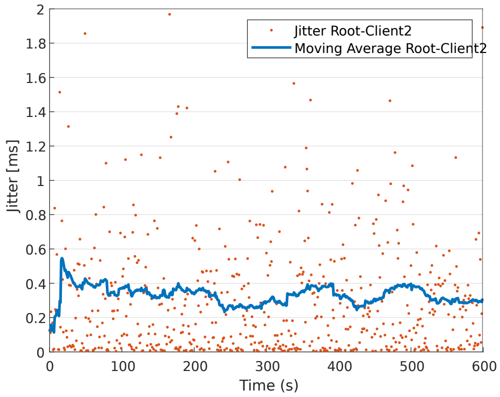

Figure 3.11: Jitter in the backhaul links of Scenario 2 (UDP).

Table 3.3: Statistics regarding the metrics collected during the 10 minutes experiment in Scenario 2, for both TCP and UDP (backhaul link RootAP/Extender1 and backhaul link Extender1/Extender2).

	Root AP - Extender 1 (5GHz)				**Extender 1 - Extender 2 (5GHz)**			
	TCP		**UDP**		**TCP**		**UDP**	
Metric	**Avg.**	**Std. Dev.**	**Avg.**	**Std. Dev.**	**Avg.**	**Std. Dev.**	**Avg.**	**Std. Dev.**
Phy	1553	233	1413	31.18	703	423.3	933.5	117.4
uPhy	881	138.1	864	145.5	425	201.4	477	189.7
dPhy	1614	223.5	1579	249.3	785	457.5	1078	148
Air Time	5.39	1.59	0.84	0.8	5.1	1.43	0.8	0.99
Retries	41.40	29	56	42	39	29.6	65	30
MSC	5.48	1.18	4.2	1.24	3.58	1.25	4.39	0.49
NSS	2.75	0.46	3.2	0.43	1.8	0.69	1.99	0.01
RSSI	-59	0.49	-59.75	0.43	-69.5	0.88	-70	0.75
Rt Rate	39.8	32.7	67.5	40.6	50.5	43.6	64.6	47.3
Tx	7.15	3.01	3.5	0.56	29.8	6.8	33.8	0.75
Inbss	18.9	5.49	26.8	5.15	42.9	10.1	50.5	189.7
Obss	4.32	1.7	2.7	0.48	1.9	0.34	2	148
Txop	21.9	16.36	13.6	13.3	16.1	16.7	9.7	0
GoodTx	3.4	2.25	1	0	21.2	5.5	29.1	0.99
BadTx	2.02	0.87	0.98	0.1	2.07	0.7	1.2	47.34
Glitch	74.2	83.16	116.8	106.3	157	745.1	44.6	35.5
Badplpc	1.27	4.85	8.8	11.5	24.6	38.4	8.5	0
Noise	-78.7	0.71	-78.4	0.54	-76.1	1.4	-76.8	2.2
Idle	27	15.64	16.5	13.05	43.9	11.7	42	7.49

The use of the channel in this scenario was more complex when compared to Scenario 1, because there were more devices being used and there were more links being used in the mesh network. Air time values averaged 5.39% in the link between the Root AP and Extender 1 (Link 1), and 5.1% in the link between Extender 1 and Extender 2 (Link 2). These values showed an even division in two of the available Air Time (the sum of these values is about the value of Air Time in the first scenario). The TXOP values in this scenario averaged 21.9% (TCP) and 13.6% (UDP) in Link 1, and 16.1% (TCP) and 9.7% (UDP) in Link 2. With the TX metric, it was noticeable that the second link spent more time transmitting than the first one, and that the transmission in Link 1 was more effective than in Link 2. TX values averaged 7.15 (TCP) and 3.5 (UDP) in Link 1, and 29.8 (TCP) and 33.8 (UDP) in Link2. The values of RSSI were higher in the first link (-59 (TCP) and -59.5 (UDP)) than in the second link (-69.5 (TCP) and -70 (UDP)), which emphasized even more the better quality of the first link. The same occurred with the Noise metric (with values averaging -78.7 and -78.4 (TCP and UDP respectively) for the first link and -76.1 and -76.8 (TCP and UDP respectively) for the second link).

The introduction of an extra link increased the number of retries (when compared to the link of the first scenario) as well as the number of BadTx, Glitch and Badplpc. These metrics can justify the inferior performance observed in the throughput and jitter parameters as they represent the errors and the necessity of retransmissions. Retry values observed showed 41.40% (TCP) and 56% (UDP) for the first link, and 39% (TCP) and 65% (UDP) for the second link. For the BadTx metric in the first link, the values were 2.02 (TCP) and 0.98 (UDP), while in the second link the values were 2.07 (TCP) and 1.2 (UDP). On the Glitch parameter bigger differences were observed (comparing to scenario 1), and the values were 74.2 (TCP) and 116.8 (UDP) on average for Link 1 and 157 (TCP) and 44.6 (UDP) on average for Link 2. For Badplc in Link 1 the values averaged 1.27 and 8.8 (TCP and UDP respectively), and in the second link the values averaged 24.6 and 8.5 (TCP and UDP respectively). All of the metrics also showed a trend of more errors in the second link when comparing the values with the link between the Root AP and the first Extender. It was also seen that the traffic injected during the test had no negative (drastic) effect on the CPU utilization of the Extenders.

In a general way, the addition of an extra link and an extra hop decreased the overall performance of a client connected to the further Extender. A closer look shows that most metrics reflect this inferior performance through a degradation of the metric values or by an increase of the standard deviation. This scenario showed how the backhaul links perform when traffic has several hops and helped understand how that affects the mesh network.

3.4.3 Scenario 3

Topology

The objective of Scenario 3 was to see how generating great amounts of traffic in both backhaul links at the same time would affect the performance of each link. To accomplish this goal nodes were connected in a daisy chain topology. All the active links between the Extenders and the Root were established in the 5GHz frequency. To apply the traffic load, two Clients were connected (Client 1 to Extender 1 and Client 2 to Extender 2) with Ethernet to focus only on the wireless backhaul links of the WMN. The topology used in this scenario is represented in Figure 3.12.

In this scenario the traffic was generated in a different way when compared to the previous scenarios. To be able to have two different flows of traffic (one from Client 1 to Root AP and other from Client 2 to Root AP), two iperf3 servers were initialized in distinct ports, and an iperf3 client was initialized on each Client. The iperf3 clients were instructed to generate as much traffic as possible in order to saturate the link. Metrics from the two links were collected in the duration of the whole test and were stored for later processing and analysis.

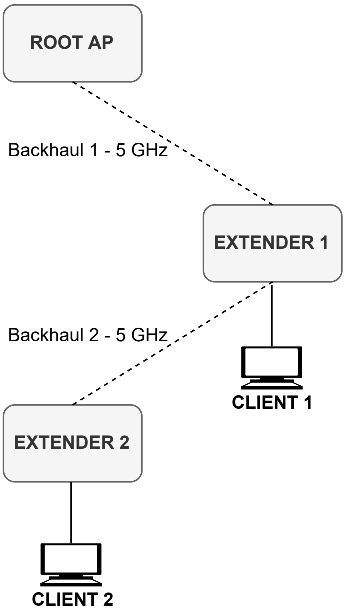

Figure 3.12: Topology of Scenario 3

Results

In this scenario, the throughput values from both terminals were very important because they show how a terminal on the second Extender can influence the performance of a transmission between the first Extender and the Root AP. In the link between the Root AP and Extender 1 (Link 1), represented in Figure 3.13, the throughput values obtained were 220 Mbits/s in a TCP test, and 252 Mbits/s in a UDP test. Comparing these values with the first scenario, it is very clear that the link performance decayed significantly with the introduction of traffic load in the link between Extender 1 and Extender 2. The jitter values obtained in the UDP test, and illustrated in Figure 3.14, were also higher in this scenario, averaging 0.206 ms. In the second link, it was seen an even more drastic decay in the performance. The fact that the first link was being fully (or almost fully) used did not allow large values of traffic from the second link to be transmitted. The second link had an average throughput of 25.6 Mbits/s with TCP loads applied, and an average of 11.3 Mbits/s with UDP loads applied. The jitter values in this link (UDP) were really high due to the amount of lost packets.

The channel access in this scenario was different than in Scenarios 1 and 2. In this case, the metrics collected did not show an even division, with the first link using the channel more time than the second link. Air Time values showed that Link 1 averaged 8.77 while Link 2 averaged 2.29. These Air Time values showed that the first link used 4 times more the medium when compared to the second link, and this difference in the use was reflected in the throughput values. The TXOP values also showed that Link 1 had more opportunities to transmit than Link 2. TXOP values in Link 1 averaged 14.4 (TCP) and 6.5 (UDP), and averaged 9.4 (TCP) and 4.17 (UDP) in Link 2. The TX values showed the same trend of the Air Time and TXOP metrics. Link 2 had significantly higher TX values than Link 1. If these values are compared, it can be concluded that the second link took more time to transmit less

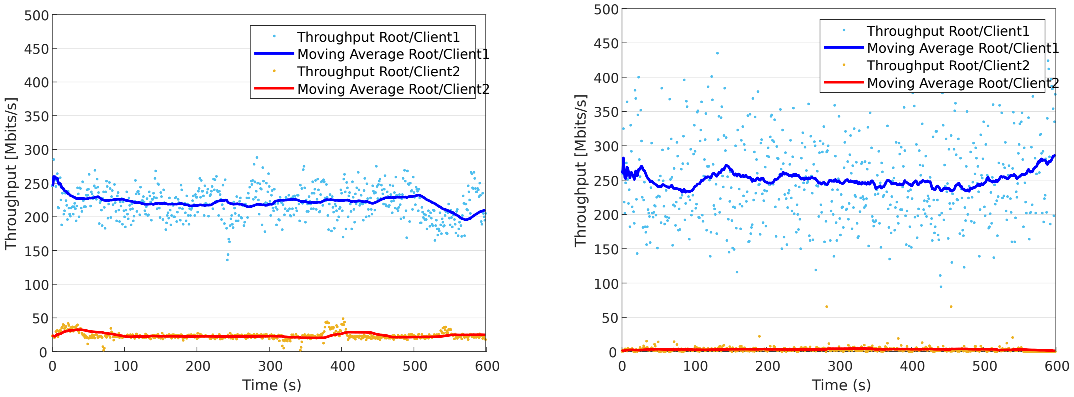

Figure 3.13: Throughput in the backhaul links of Scenario 3 (TCP and UDP).

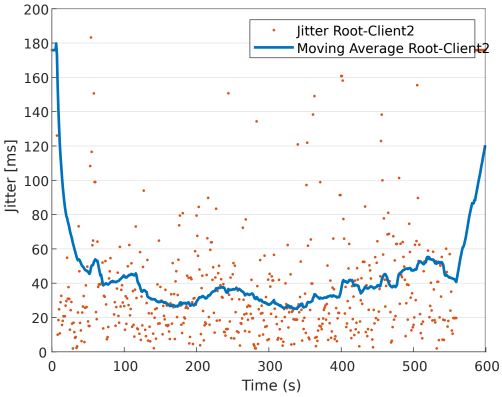

Figure 3.14: Jitter in the backhaul links of Scenario 3 (UDP).

data. The values observed in Link 1 were 11 and 4.14 (TCP and UDP respectively), while the values in Link 2 were 50.3 and 74.5 (TCP and UDP respectively). RSSI values in both links were slightly worst when compared to other scenarios, and averaged -60.4 (TCP) and -60.6 (UDP) dBms in the first link and -72.8 (TCP) and -73.1 (UDP) dBms in the second link. The Noise values observed in the scenario were -78.8 (TCP and UDP) dBm in the first link and -76.5 (TCP) and -83 (UDP) dBm in the second link.

The number of retries also had an increase in its values when compared to the values observed in other scenarios. In Link 1 retries averaged 30% (TCP) and 32% (UDP), and 50% (TCP) and 62% in the second link. These values can, in part, justify the differences in the performance of both links. In metrics like BadTx, Glitch and Badplpc, it was observed once again that the second link was not able to perform as good as the first one. Badtx values averaged 2 (TCP) and 1 (UDP) in the first link, and averaged 1.9 (TCP) and 2.28 (UDP) in the second link. The glitch values observed were 134 (TCP) and 49.7 (UDP) on average for the first link, and 149 (TCP) and 242.2 on average for the second link. The Badplpc metric showed 3 and 2.7 (TCP and UDP respectively) in the first link, and 14.6 and 20.3 (TCP and UDP respectively) in the second link. These three metrics represent bad transmissions and errors encountered on packets, and therefore, justify the performance values obtained in this scenario. It was also seen that the traffic injected during the test had no negative (drastic) effect on the CPU utilization of the Extenders.

In a general way, the results observed in this scenario show that high amounts of traffic in a link closer to the Root of a mesh network negatively affect the performance of links further way. The existence of traffic from the other links also affects the performance of the closer links (even though not as much). These performance issues are observed in all of the metrics and in its standard deviation values. The extracted values from this scenario can be seen in Table 3.4.

Table 3.4: Statistics regarding the metrics collected during the 10 minutes experiment in Scenario 3, for both TCP and UDP (backhaul link RootAP/Extender1 and backhaul link Extender1/Extender2).

	Root AP - Extender 1 (5GHz)				**Extender 1 - Extender 2 (5GHz)**			
	TCP		**UDP**		**TCP**		**UDP**	
Metric	**Avg.**	**Std. Dev.**	**Avg.**	**Std. Dev.**	**Avg.**	**Std. Dev.**	**Avg.**	**Std. Dev.**
Phy	1358	334.7	1252	110.5	816	123	901	251
uPhy	840.6	133.7	836	159.3	379	205.1	304.6	144.2
dPhy	1552	249.6	1373	182	892	236	1072	163
Air Time	8.77	1.95	0.73	0.76	2.29	0.75	0	0
Retries	30	16	32	30	50	28	62	67
MSC	4.7	1.05	5.8	0.75	3.94	0.53	4.4	0.64
NSS	3	0.39	2	0.25	1.99	0.19	2	0.09
RSSI	-60.4	0.51	-60.6	0.59	-72.8	1.05	-73.1	0.37
Rt Rate	37	20.36	31.4	25.4	42.15	35.5	63.5	77.15
Tx	11	3.4	4.14	0.58	50.3	5.5	74.5	14.4
Inbss	39.2	6.3	69	13.25	26.15	5.09	14.3	9.09
Obss	4.14	0.86	2	0.21	1.8	0.39	1.88	0.33
Txop	14.4	9.42	6.5	11.3	9.4	6.5	4.17	14.6
GoodTx	5.39	2.7	1	0	41	5.17	70.3	14.6
BadTx	2	0.7	1	0.15	1.9	0.75	2.28	1.14
Glitch	134	181.2	49.7	85.2	149	731.8	242.2	898.2
Badplpc	3	2.86	2.7	8.1	14.6	21.78	20.3	120.7
Noise	-78.8	0.9	-78.8	0.65	-76.5	2.08	-83	4.55
Idle	23.3	8.9	10.05	10.98	57.7	5.33	76.4	12.05

3.4.4 Scenario 4

Topology

In order to know if there was a situation where using a 2.4GHz backhaul link would translate in a better performance and throughput than in a 5GHz backhaul link, Scenario 4 was designed. The objectives of this scenario were to observe how much traffic a 2.4GHz link could handle and if and how much it would improve the remaining 5GHz backhaul link. It is known that 2.4GHz links do not have the same capabilities as the 5GHz. One of the advantages of the 2.4GHz is that it can transmit between longer distances. In this test scenario, the topology was set to a star architecture with direct links between the first Extender and the Root and between the second Extender and the Root. The link of the second Extender is now set to the 2.4GHz, while the link of the first Extender was set to the 5GHz frequency. To apply traffic loads to the links, a Client was connected to each Extender (Client 1 and

Client 2) using an Ethernet cable. The obtained topology of this scenario is represented in Figure 3.15.

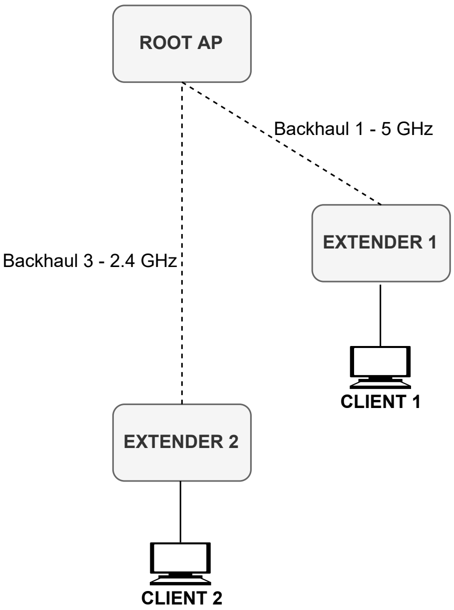

Figure 3.15: Topology of Scenario 4.

The way to generate traffic in the network was the same as in scenario 3. Two iperf 3 servers were initialized in the Root Extender in two different ports, and an iperf3 client was initiated in each Client. The iperf clients were instructed to generate as much traffic as possible for a duration of 10 minutes to saturate both backhaul links. Metrics from the two links were collected in the duration of the whole test and were stored for later processing and analysis.

Results

This scenario was focused on the performance of the 2.4GHz link (Link 3). The throughput on this link, represented on Figure 3.16, averaged 27.7 Mbits/s in the TCP test and averaged 28.1 Mbits/s in the UDP one. These results were very positive if compared to the ones acquired in the previous scenario. In the UDP test the jitter values, illustrated in Figure 3.17, averaged 0.742 ms, although this value was not considered very good, when compared to the one in Scenario 3.

When it comes to medium access, in this scenario the two Extenders did not have conflicts because each one was transmitting on a different frequency. This allows an Extender to use as much of the medium as it needs, improving the rates it can achieve. The 2.4GHz link results show that the air use was really high, reaching values of 74.4. This value shows that most of the medium was being used by the Extender 2. The TXOP metric averaged 7.9 (TCP) and 9.74 (UDP), and show that this link was being used in almost full capacity. The results show high values in the TX in the TCP test (averaging 82), and relatively low values in the UDP test (averaging 8.05). When it comes to RSSI, Link 1 had an average of -67.8 and -68.15 (TCP and UDP respectively) dBm, which show that it can be a well grounded alternative to

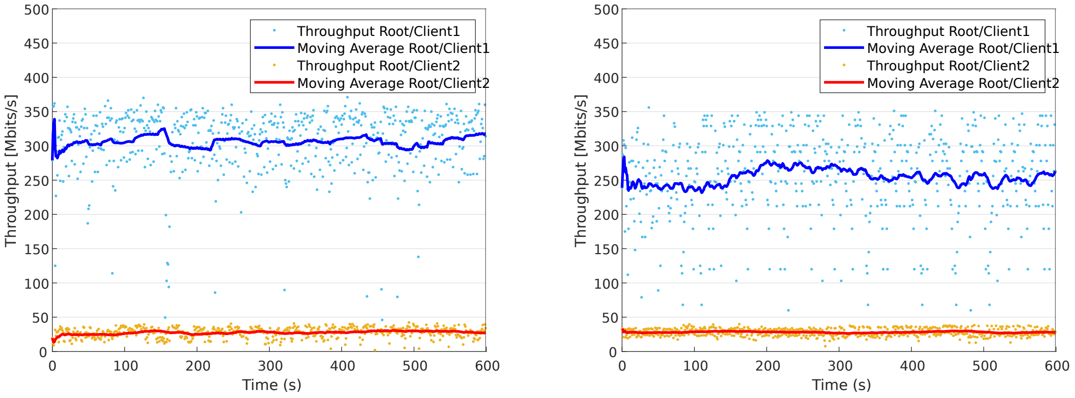

Figure 3.16: Throughput in the backhaul links of Scenario 4 (TCP and UDP).

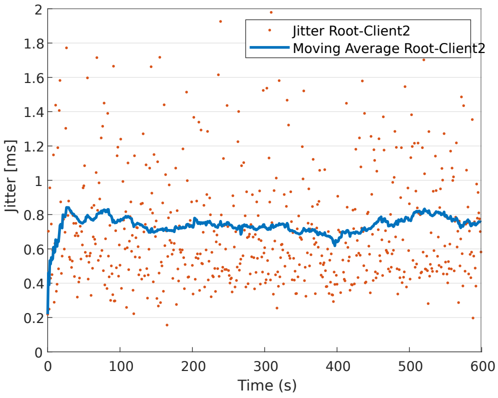

Figure 3.17: Jitter in the backhaul links of Scenario 4 (UDP).

the 5GHz link. The Noise value in this link was also smaller than in other scenarios (because the 2.4GHz frequency was not being used so much inside and outside of the mesh network).

The number of retries in the 2.4GHz frequency was not very high during the tests with 17% (TCP) and 32% (UDP). The retries values show that the link is reliable and robust. The BadTx, Glitch and Badplpc metrics did not change much from what was seen in other scenarios. BadTx averaged 8 in the TCP test and 0.87 in the UDP test. In the Glitch metric, values averaged 204 and 156 in the TCP and UDP test respectively. The Badplpc values, averaged 0 for TCP and 0.24 for UDP, which show almost no errors in the packets.

Looking at the performance results in the link using the 5GHz frequency (Link 1), it was noticeable a slight enhancement. The throughput values obtained were 306 Mbits/s (TCP) and 384 Mbits/s (UDP), which were better than in scenario 3. The jitter values of the UDP test also showed an improvement and average of 0.08 ms.

In a general way, it was possible to notice that the 5GHz band was less congested, allowing Extender 1 to use more of the medium. Values like Air Time, TXOP and TX justify this behavior. The RSSI and Noise values also showed less usage of the frequency band and showed better results. The metrics like Retries, BadTx, Glitch and Badplpc, that show the quality of the transmissions, did not show many differences from the other scenarios. It was also seen that the traffic injected during the test had no negative (drastic) effect on the CPU utilization of the Extenders.

After observing the results obtained in this scenario, it was possible to conclude that the use of a 2.4GHz backhaul link can increase the overall performance of the network, increasing the bandwidth in both links, as well as reducing the jitter values of the replaced link. The 2.4GHz link uses most all the medium time it needs because there is less interference. The 5GHz also benefits from not having to share the medium, and gets values closer to the ones observed in the first scenario. The extracted values from this scenario can be seen in Table 3.5.

Table 3.5: Statistics regarding the metrics collected during the 10 minutes experiment in Scenario 4, for both TCP and UDP (backhaul link RootAP/Extender1 and backhaul link RootAP/Extender2).

	Root AP - Extender 1 (5GHz)				**Root AP - Extender 2 (2.4GHz)**			
	TCP		**UDP**		**TCP**		**UDP**	
Metric	**Avg.**	**Std. Dev.**	**Avg.**	**Std. Dev.**	**Avg.**	**Std. Dev.**	**Avg.**	**Std. Dev.**
Phy	1348	322.1	1285	102.4	52.2	9.2	45.2	14.4
uPhy	844.5	111.3	849	119.3	47.8	7.1	51.8	12
dPhy	1572	234.6	1384	178	54.74	0.08	52.5	17.7
Air Time	22.1	1.8	0.73	0.76	74.4	16	0.33	0.49
Retries	24.7	12	25.1	11.1	17	16	32	50
MSC	5.7	1.1	5.9	0.75	4.8	1.48	7.18	18.6
NSS	3.1	0.49	2.6	0.35	1	0.12	1.04	0.17
RSSI	-59	0.6	-59.4	0.2	-67.8	0.92	-68.15	0.49
Rt Rate	14.7	8.3	20.4	12.4	20.92	1.48	26.9	24.13
Tx	17.2	3.3	5.74	1.08	82	9.4	8.05	0.47
Inbss	83.1	16.5	81	13.22	2	0.49	74.05	11.35
Obss	2.07	0.8	2	0.22	3.85	1.21	3.98	1.73
Txop	15.6	8.32	7.53	10.2	7.9	9.7	9.74	11.09
GoodTx	8.01	1.7	1	0	68.09	8.8	1	0
BadTx	1.12	0.2	1	0.12	8	3.49	0.87	0.33
Glitch	32	81.2	19.7	75.1	204	35.6	156	952.4
Badplpc	2	2.05	3.7	1.2	0	0.43	0.24	0.7
Noise	-78.9	0.7	-78.9	0.7	-83.3	1.06	-85.4	2.05
Idle	29.3	1.9	39.05	9.1	82.1	3.5	17.2	10.86

3.5 FINAL REMARKS

This chapter presented the testbed used for the networks measurement over a WiFi mesh network. It described the used hardware and the way data was collected. All the data was collected from the Extenders to an APU that stored and processed the obtained metrics. All the metrics were explained, which allowed a more clear view at the obtained results.

After explaining the metric properties and the four scenarios, the analysis of the data from all of the four scenarios showed that using the 2.4GHz frequency for a backhaul link could be useful in some situations. The 2.4GHz frequency can have better performance (when the channel is not congested) if some level of saturation exists in the 5GHz backhaul link of the mesh network. These results validate the need for an automatic steering mechanism that will be presented and evaluated in the next chapter.

CHAPTER 4

Algorithm for Backhaul Steering

In the previous chapter we have shown that, using a 2.4GHz backhaul link can bring benefits to a mesh WiFi network if the 5GHz backhaul links, used in a daisy chain topology, are overloaded. With this in mind, a backhaul steering mechanism to detect these situations and change the topology of the mesh network has been developed.

This chapter presents all the steps regarding the development of the proposed backhaul steering algorithm. Section 4.1 addresses the algorithm conditions and their parameters. Section 4.2 presents the weight distribution of the conditions and the tests performed to obtain the weights. Section 4.3 details the backhaul steering algorithms. Section 4.4 presents the tests and the obtained results regarding the steering process. Finally, Section 4.5 enumerates the chapter's conclusions.

4.1 Algorithm Conditions and Metric Relations

In order to implement the steering algorithm, some conditions were defined to identify situations and scenarios where the change in the backhaul links would enhance the network performance. To define these conditions, the metrics collected in the scenarios of the previous chapter were reanalysed to establish relations between several parameters and the quality of the mesh network.

Some of the metrics were instantly discarded due to the lack of useful information in the identification of the target scenario, such as MCS or the NSS. These metrics show information about modulation and number of spatial streams, but do not represent the status of the network in a way that would allow distinguishing the need to change the frequency band of a given backhaul link. The metrics that express theoretical rate values such as Phy, uPhy and dPhy were not use, since they show an estimate of the maximum rate a link can handle at a given moment, and not the actual rate that is flowing through the links. The Inbss and Obss metrics were excluded because they are low level metrics collected from the drivers, and their accuracy might not be as good as the one needed for this type of solution. The Retries metric looks like it would be a very good parameter to use in some conditions at first sight; however, due to the high standard deviation and instability, it was not considered when implementing the algorithm. GoodTx, Glitch and BadTx were also discarded due to some instability on the values (represented in the standard deviation). The metrics with high standard deviation could not be included in the algorithm, because the amount of spikes seen in the values could trigger an undesired steer (that could reduce the quality of the network). Beside these reasons, in most of the metrics enumerated before, there was no direct relation between their variations and trends with the throughput values observed.

The RSSI metric is very useful to understand the physical properties (signal strength) of a link at a given time in the mesh network. After analysing in more detail the RSSI measured in the backhaul links, illustrated in Table 4.1, it was concluded that some patterns and trends of the RSSI values could be used to determine (or help determine) whether or not the backhaul steer would be worth. This analysis also shows that, during periods of higher traffic on any link, the links' RSSI increases (more on the link between the Extender 1 and the Extender 2). Comparing Scenario 2 with Scenario 3, it was observed that, in the link between the Root AP and Extender 1, the differences in the values were very small (less than 1 dBm). On the other side, the differences observed in the link between Extender 1 and Extender 2 were more noticeable (in some moments higher than 3 dBm). This metric will be used in the steering decision process, as explained in the following sections.

Table 4.1: Per minute RSSI values, in dBm, observed in the backhaul links of Scenarios 1, 2 and 3.

	Minutes									
	0-1	**1-2**	**2-3**	**3-4**	**4-5**	**5-6**	**6-7**	**7-8**	**8-9**	**9-10**
Scn 1 Link 1	-59.40	-59.55	-59.44	-59.44	-59.77	-59.88	-59.88	-59.88	-60.00	-60.00
Scn 2 Link 1	-59.40	-59.50	-59.30	-59.47	-59.60	-59.60	-59.63	-59.52	-59.30	-59.90
Scn 3 Link 1	-60.40	-60.02	-60.02	-60.40	-60.66	-60.40	-60.40	-60.40	-60.70	-60.40
Scn 2 Link 2	-68.10	-69.70	-69.70	-69.80	-69.70	-69.30	-69.90	-70.2	-69.00	-69.95
Scn 3 Link 2	-70.00	-72.90	-73.00	-72.80	-73.00	-73.20	-73.20	-73.40	-73.40	-73.10

TXOP can be used to understand the current state of the links, since it informs on how many opportunities there were to transmit. Even before performing a detailed analysis, it is very clear that the more a link is occupied, the less opportunities are, and that translates in a smaller TXOP percentage. After analysing Table 4.2, it is clear that, in scenarios where there is more traffic, more hops and more interference, the TXOP values tend to be smaller when compared to scenarios with empty channel. After comparing Scenario 3 to Scenarios 1 and 2, it was observed that TXOP values were smaller in all the links. The values observed in the link between the Root AP and Extender 1 were 2% smaller (on average) when compared to Scenario 1, and 4-5% smaller (on average) when compared to the second scenario. On the link between Extender 1 and Extender 2, the differences between them were even higher. This metric will be used in the steering decision process, as will be explained in the following sections.

Table 4.2: Per minute TXOP values, in percentage, observed in the backhaul links of Scenarios 1, 2 and 3.

	Minutes									
	0-1	**1-2**	**2-3**	**3-4**	**4-5**	**5-6**	**6-7**	**7-8**	**8-9**	**9-10**
Scn 1 Link 1	29.50	14.90	15.63	15.18	14.80	15.30	14.09	15.36	15.60	15.36
Scn 2 Link 1	34.60	17.68	18.47	18.20	18.21	17.68	17.36	17.63	17.57	41.50
Scn 3 Link 1	24.60	13.40	12.60	13.30	12.80	12.66	13.80	14.10	13.20	13.80
Scn 2 Link 2	20.90	11.14	11.28	12.42	14.20	14.30	11.60	11.80	13.60	40.06
Scn 3 Link 2	15.90	8.65	8.85	8.85	8.55	8.90	8.65	8.85	9.10	8.70

The Idle metric can be used to determine how much time a device is inactive. After looking in detail to the values of Table 4.3, a similar behavior to the TX metric was identified. It was also identified a relationship between TXOP and Idle metrics where the difference of the values of each metric is maximum when the network is under huge amounts of traffic, and would benefit from using a different topology with other frequency bands in some backhaul links. This metric will be also used in the steering decision process.

Table 4.3: Per minute Idle values, in percentage, observed in the backhaul links of Scenarios 1, 2 and 3.

	Minutes									
	0-1	**1-2**	**2-3**	**3-4**	**4-5**	**5-6**	**6-7**	**7-8**	**8-9**	**9-10**
Scn 1 Link 1	43.00	30.36	31.60	30.60	30.67	30.81	30.36	30.81	31.18	30.80
Scn 2 Link 1	39.20	22.57	23.78	24.30	23.10	22.70	24.47	22.84	22.36	45.35
Scn 3 Link 1	33.90	22.00	21.30	22.13	21.46	21.26	22.26	23.00	24.20	22.40
Scn 2 Link 2	49.00	40.00	41.70	42.50	41.70	42.00	42.04	42.33	37.52	60.33
Scn 3 Link 2	64.50	55.75	57.50	58.55	56.55	59.35	54.60	57.15	57.40	56.50

Other useful metric to understand the current state of an Extender is the TX. This metric shows how much time of the whole active time the device is transmitting. After careful evaluation and analysis of this metric, it was concluded that, in fact the TX metric could be useful to identify the need for a steer in the backhaul link and a consequent topology change. The TX values in the link between Extender 1 and Extender 2 tend to be much higher than in the link between the Root AP and the first Extender (when both links are being used). In fact, the higher the traffic in the first link, the higher the TX value gets in the second link. In Scenario 2, the difference between the two links averages a value close to 25, while the third scenario averages almost 40. The TX values between the two Extenders in scenario 2, averaging 30%, are smaller than the values from scenario 3 (50% on average), and this difference can also be used to identify a case where a steer would be recommended. This metric can also be related with the amount of traffic to identify a situation where a device tries to transmit more traffic than what the link can handle. As a consequence, this metric will be also included in the backhaul steering process.

The link rate (can be the UdRate value, the DdRate value or even the sum of both, and term Rate will be used in the rest of the document to mention these metrics) is the most

Table 4.4: Per minute TX values, in percentage, observed in the backhaul links of Scenarios 1, 2 and 3.

	Minutes									
	0-1	**1-2**	**2-3**	**3-4**	**4-5**	**5-6**	**6-7**	**7-8**	**8-9**	**9-10**
Scn 1 Link 1	14.15	16.90	17.36	16.20	17.18	16.90	17.45	16.45	16.45	16.50
Scn 2 Link 1	6.90	7.10	7.57	8.42	6.57	6.68	9.30	7.20	6.63	5.15
Scn 3 Link 1	11.00	10.80	10.40	10.80	10.70	10.80	10.40	10.90	13.80	10.60
Scn 2 Link 2	29.60	30.70	32.40	32.50	30.00	29.70	32.30	33.00	25.60	22.40
Scn 3 Link 2	50.70	49.10	50.80	51.60	50.15	52.60	47.90	50.00	50.55	49.80

relevant metric when it comes to understand the current state of the network. It is also the best way to identify the need for a topology change. After looking in depth to the values presented in Table 4.5, it is clear that this metric is essential to identify each scenario at a given moment because it is the best way to know how the links are being used and the network performance. Another reason why this metric must be included in the backhaul steering process lies in the stability and consistency of the obtained values (no big variations were seen), which would reduce the number of false positives of a condition using this parameter. Thus, it is clear that, in Scenario 3, the link rate values of the backhaul link between the Extender 1 and 2 are considerably reduced, while the values in the backhaul link between the Root AP and Extender 1 are very high. This relation between the link rates of the two backhaul links will be the base of the most important condition in the whole backhaul steer algorithm.

Table 4.5: Per minute Rate values, in Mbits/s, observed in the backhaul links of Scenarios 1, 2 and 3.

	Minutes									
	0-1	**1-2**	**2-3**	**3-4**	**4-5**	**5-6**	**6-7**	**7-8**	**8-9**	**9-10**
Scn 1 Link 1	265.1	364.6	357.0	361.7	367.6	360.2	360.4	363.3	361.33	359.40
Scn 2 Link 1	127.0	136.0	148.2	144.1	124.6	121.9	142.5	146.8	92.3	111.5
Scn 3 Link 1	240.0	283.0	271.0	287.0	275.0	278.0	267.0	273.0	272.0	271.0
Scn 2 Link 2	127.2	138.3	151.9	151.4	143.1	144.4	151.6	153.0	112.2	118.3
Scn 3 Link 2	28.6	27.2	29.2	25.7	28.2	25.2	25.6	28.6	27.7	28.0

The proposed backhaul steering algorithm can be divided into two complementary processes: one to perform a 5GHz to 2.4GHz frequency and topology steer, and another one to make the inverse decision. For that, the conditions for each mechanism need to be adjusted to its purpose. We will now introduce the conditions used in each steering processes.

4.1.1 5GHz to 2.4GHz backhaul steer

RSSI-based condition

The first condition used in the steering process is related with the difference of RSSI values in each backhaul link of the daisy chain topology, and is given by

$$RSSI_{Root-Ext1_{5GHz}} - RSSI_{Ext1-Ext2_{5GHz}} > RSSI_{threshold}. \tag{4.1}$$

To define the $RSSI_{threshold}$, the difference between the RSSI of the two backhaul links, observed in Table 4.1, was taken into account. It was observed that the difference between the links in the second scenario was never higher than 10 dBm. On the other side, in the third scenario, the difference was almost always higher than 10 dBm (being usually around 12 to 13 dBm). The first value proposed for the variable was 12 dBm, which would trigger the condition when the difference was 13 or higher. After some tests, it was observed that sometimes, the condition was not triggered when it should, so after several adjustments, this value was lowered to trigger the condition whenever the difference between the RSSI observed in both links is larger than 10 dBm.

TXOP-based condition

The TXOP metric shows the channel's occupation. With that in mind, and after analysing Table 4.2, the TXOP condition was designed to use the TXOP values from the 5GHz backhaul links to identify a steering need. The proposed condition for the TXOP is given by

$$TXOP_{Root-Ext1_{5GHz}} < TXOP_{threshold_1} \text{ and } TXOP_{Ext1-Ext2_{5GHz}} < TXOP_{threshold_2}. \tag{4.2}$$

In this condition, two thresholds had to be defined, one regarding the link between the Root AP and the Extender 1, and another one for the link between Extender 1 and Extender 2. It was seen that the TXOP value in the link between the Root AP and Extender 1, during scenario 3, was always higher than 12, and that in the second scenario, these values were higher that 17. The value for $TXOP_{threshold_1}$ needed to be comprehended between these values in order to differentiate and separate the two scenarios and, with that in mind, the first proposed value was 17. This value was too close to the limit, and after some preliminary tests, it turned out to fail more than expected. After some experiments, experimenting values in the 12-17 interval, the one that showed to be the best was 15. This value allows more tolerance to variations in the link. The link between Extender 1 and 2 had a very similar behavior to the other link but with lower values. During the second scenario, the values were never lower than 11, while in the third scenario the values were not higher than 9. After some tests, $TXOP_{threshold_2}$ was set to 10. In summary, the final thresholds regarding this condition were set to 15 for $TXOP_{threshold_1}$, and 10 for the $TXOP_{threshold_2}$.

Idle-based condition

The Idle and TXOP condition was defined after analysing that the difference between these two metrics had a peak during periods of great traffic in the link between Extender 1 and Extender 2 (as shown in Tables 4.3 and 4.2). Therefore, another condition was set, which is given by

$$Idle_{Ext1-Ext2_{5GHz}} > TXOP_{Ext1-Ext2_{5GHz}} + Idle_{threshold}. \tag{4.3}$$

Comparing the values from all the scenarios, it was observed that the difference between the two metrics (Idle and TXOP) is higher whenever both backhaul links are congested. The difference between the two metrics in Scenario 3 is close to 45, where in the second one the difference is around 30. With these values in mind and, after some tuning, the final value proposed for the $Idle_{threshold}$ was 40 in order to give some tolerance to small variations of values.

Rate-based condition

As mentioned before, the Rate is the most important metric to identify steering necessities. Thus, a fourth condition was proposed using the Rate values of both backhaul links, and is based on the analysis of Table 4.5, as follows:

$$Rate_{Root-Ext1_{5GHz}} > Rate_{threshold_1} \text{ and } Rate_{Ext1-Ext2_{5GHz}} < Rate_{threshold_2}. \tag{4.4}$$

Looking at the link between the Root AP and Extender 1, we can observe that, on the third scenario, the values were always larger than 240 Mbits/s, and in the second one the values were never larger than 150 Mbits/s (the first scenario was not taken into account because there was no traffic on the link between Extender 1 and 2). With these values in mind, the value for the $Rate_{threshold_1}$ was set to 240. The second part of the condition is the one that really determines whether it is worth changing the frequency of the backhaul link. The process to choose the threshold value is the following: if the current rate of the link was smaller than the one that can be achieved in the 2.4GHz, then the condition should be triggered. After looking at the values obtained in Scenario 4, the chosen value was 30. The final threshold values for this condition are 240 for $Rate_{threshold_1}$ and 30 for $Rate_{threshold_2}$.

TX-based condition

The TX condition was defined after analysing the behavior of the TX metric (presented in Table 4.4), and comparing it to the Rate metric in the link between Extender 1 and Extender 2 as follows:

$$TX_{Ext1-Ext2_{5GHz}} > TX_{threshold} \text{ and } Rate_{Ext1-Ext2_{5GHz}} < Rate_{threshold_2}. \tag{4.5}$$

As seen in the previous condition (rate condition), the value for the $Rate_{threshold_2}$ is 30. The same process of the rate condition was used to get this value. After analysing the TX values, it was observed that they were higher in the link between Extender 1 and 2. It was also noticed that the values were even higher when the link between the Root AP and Extender 1 was congested. The main goal with this condition is to identify when the device is spending time transmitting, but the rate values are not reflecting it, which translates in inefficient transmissions, leading to, for example, more retransmissions. Looking at the data from the link between the two Extenders on Scenario 3, the chosen value was 50 because the TX value was higher than that (almost at all times).

4.1.2 2.4GHz to 5GHz backhaul steer

Rate-based condition

The Rate metric in the backhaul link between the Root AP and Extender 1 shows if more traffic on that link can be handled. Therefore, following the Rate information in Table 4.5 and observing the previous Rate condition regarding the 5GHz to 2.4GHz steering mechanisms, the following condition has been proposed:

$$Rate_{Root-Ext1_{5GHz}} < Rate_{threshold_3}. \tag{4.6}$$

The rate condition needs a threshold that could ensure a tolerance interval to avoid excessive steer commands. With this in mind and knowing that the value used to steer from 5GHz to 2.4GHz was 240 Mbits/s, the chosen value was 200 Mbits/s. Using this value for the variable ensures that, when the steer command is executed, the rate value of the link of the Extender 2 increases. At the same time this value guarantees that another steer, from 5GHz to 2.4GHz, will not be needed right away.

TXOP-based condition

To define the TXOP condition, Table 4.2 was reanalysed, along with a tolerance interval needed to avoid excessive steers. The TXOP condition to steer a backhaul link from 2.4GHz to 5GHz is given by

$$TXOP_{Root-Ext1_{5GHz}} > TXOP_{threshold_3}. \tag{4.7}$$

To ensure a hysteresis interval and to guarantee that the steer would in fact enhance the performance of the mesh network, the $TXOP_{threshold_3}$ was set to 25.

TX-based condition

The TX metric reflects the amount of time a device is transmitting, and this information can be used to understand whether or not the device can transmit more than what is currently transmitting. Then, the following condition has been proposed

$$TX_{Root-Ext1_{5GHz}} < TX_{threshold_3}. \tag{4.8}$$

To ensure a tolerance interval and to guarantee that the mesh network will benefit from a steer, and following the TX results presented in Table 4.4 for the different scenarios, $TX_{threshold_3}$ was set to 10.

4.2 Condition Weights

After defining the conditions used to determine when a topology change in a mesh WiFi network shall occur, in terms of backhaul links, we must set the weight of each condition. All the conditions are prompt to errors and false positives may trigger an unnecessary backhaul steer. These false positives emphasize the need for several conditions, instead of a single one. To prove this, and to obtain a performance ranking of the several conditions, multiple tests were conducted. These tests addressed several possible scenarios that can happen in the mesh network.

To determine the weight of each condition regarding a possible steer from 5GHz to 2.4GHz, several network configurations were tested. These tests are different from the ones presented and discussed in the previous chapter because they introduce variations in the channel load during the test. The network nodes (Root AP and the two Extenders) in the mesh network were connected in a daisy chain topology, using solely 5GHz backhaul links, with Client 1 connected to Extender 1 through Ethernet and Client 2 connected to Extender 2 also through Ethernet (as illustrated in Figure 3.4). During the test, the traffic injected used the TCP protocol. In detail, during the test the network conditions changed every 60 seconds for a total of 7 minutes. In the first minute there was no traffic in the backhaul links (Clients 1 and 2 were silent). In the second minute there was 100 Mbits/s being applied in the link between the Root AP and the Extender 1, through Client 1, and no traffic flowing through the link between Extender 1 and Extender 2. During the third minute, the load in the Root AP-Extender 1 link increased to 200 Mbits/s, caused by Client 1, and in the fourth minute the load applied in the same link was the maximum the link could handle. Still, up to this point, no traffic was injected on the link between Extender 1 and Extender 2. Minute 5 marks the injection of traffic between the two Extenders, through Client 2, and no traffic is injected by Client 1. During the sixth minute the load injected by Client 1, and consequently in the backhaul link between the Root AP and Extender 1 was maximum, and Client 2 was injecting 12 Mbits/s, to be transferred through link between Extender 1 and 2. The final minute is characterized by having both Clients injecting the maximum traffic allowed in the network. For ease of understanding, Figure 4.1 illustrates the loads injected by each Client during the experiment.

We will now analyze how the RSSI behaves in the aforementioned test. Figure 4.2 illustrates the RSSI observed in the link between the Root AP and Extender 1, and the link between the two Extenders, as well as the load observed in the same backhaul links. During the first two minutes, the condition regarding the RSSI, namely the difference between the RSSI of two links being lower than 10, was not triggered. In the third minute, the condition triggered several times even thought there was no need for a steer. The same happened in the rest of the test due to the injection of traffic. The degradation of the signal throughout the test increased the amount of false positives. The obtained values were considered positive, since this condition was triggered in the final minute, and in the ones it wrongly activated, it showed that there were physical conditions good enough to connect the Extender 2 to the Root AP

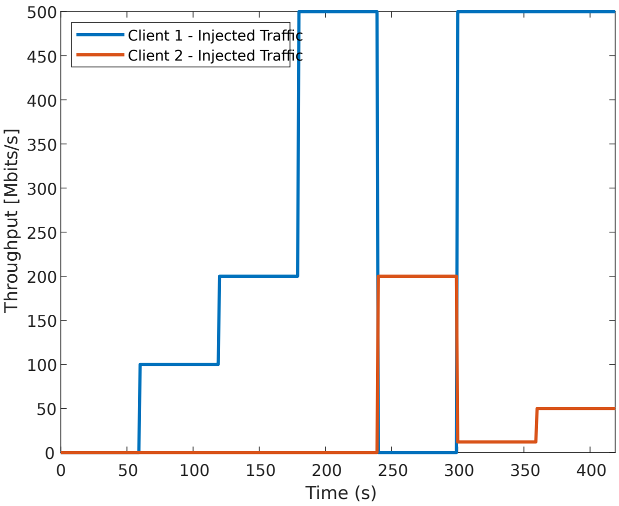

Figure 4.1: Injected traffic (5GHz to 2.4GHz steering process).

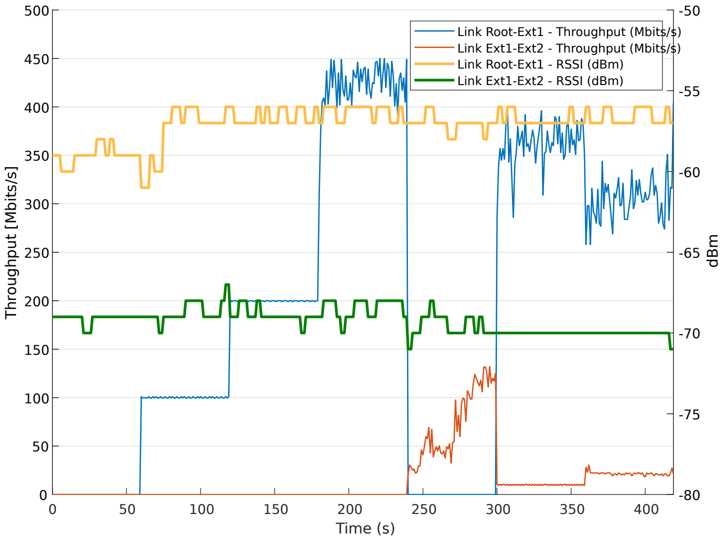

Figure 4.2: RSSI metric behavior (5GHz to 2.4GHz steering process).

via a 2.4GHz link.

We will now analyze how the Idle metric behaves in the same test. Figure 4.3 illustrates the Idle values observed in the link between the Root AP and Extender 1 and in the link between the two Extenders, as well as the load observed in the same backhaul links. The TXOP values of the same links are illustrated in Figure 4.4. During the first three minutes, the condition regarding the Idle and also the TXOP metric was not triggered. When the load in the link between the Root AP and the Extender 1 increased (minute 4), some cases of false positives were seen. In the fifth minute the condition's behavior was as expected, not being triggered, and in the final two minutes, the condition was activated, meaning that in the sixth minute there were false positive values. In the last minute, the Idle condition triggered a steer intention as it was supposed.

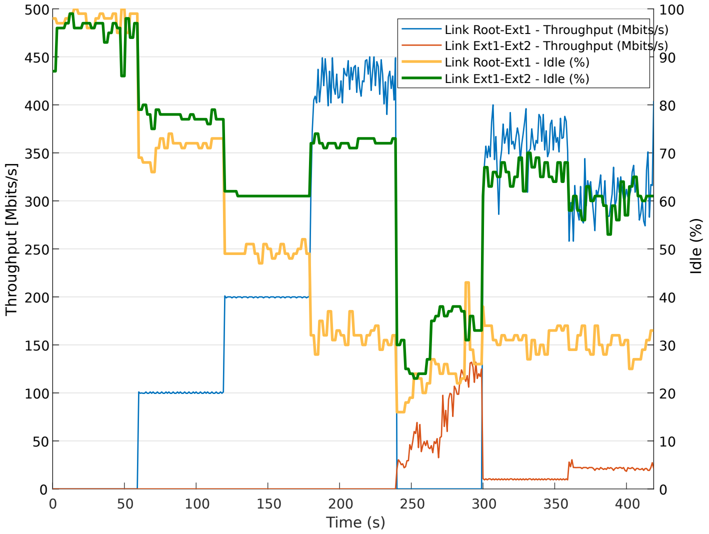

Figure 4.3: Idle metric behavior (5GHz to 2.4GHz steering process).

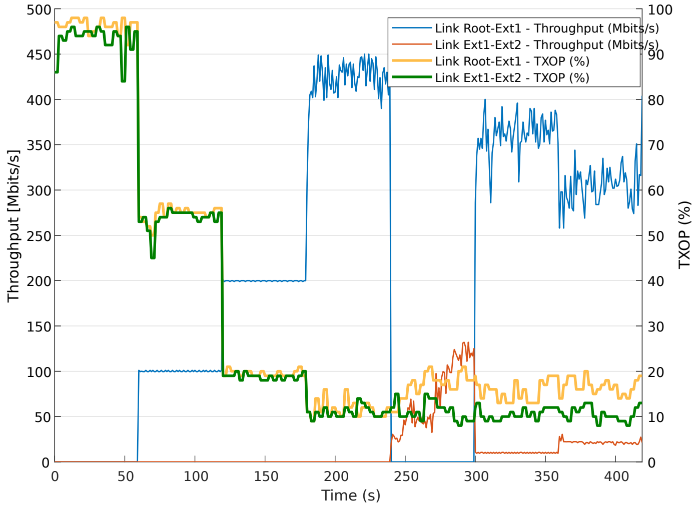

Figure 4.4: TXOP metric behavior (5GHz to 2.4GHz steering process).

We will now analyze how the TXOP metric behaves. Figure 4.4 illustrates the TXOP values observed in the link between the Root AP and Extender 1 and the link between the two Extenders, as well as the load observed in the same backhaul links. In the TXOP condition, the number of false positive results (triggered when no expected) was reduced. During the first 3 minutes, the output of the condition was correct. In fact this condition only failed in minute 4 and 5 where it wrongly decided to change the topology of the mesh network. During the final two minutes (minute 6 and 7), this condition was able to correctly identify the need for a steer (being triggered only on the seventh minute). This condition can have a positive impact in the decision, and the weight attributed to that will show exactly that.

To analyse the Rate condition, one must follow the throughput curves presented in any figure between Figure 4.2 and Figure 4.5 because the Rate metric of a given link is given

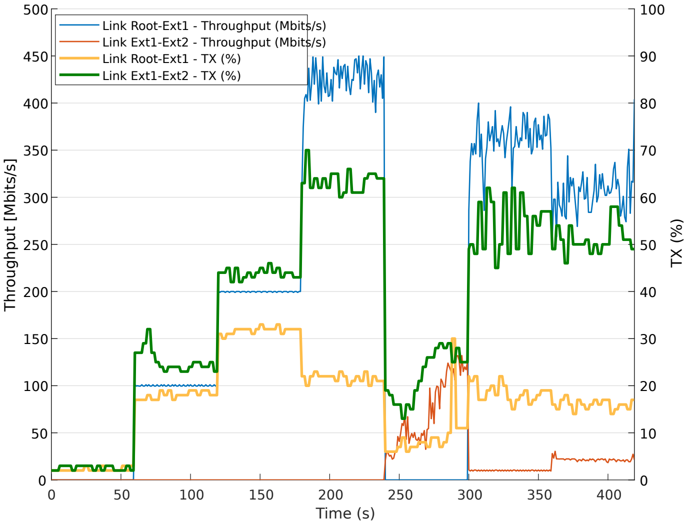

Figure 4.5: TX metric behavior (5GHz to 2.4GHz steering process).

by the throughput observed in that link. During the first minute, the condition had no false positives and the same happened until the sixth minute. The only moment when this condition failed to give the right output was during minute number 6, where the rate applied in the link between the Extender 1 and Extender 2 was lowered on purpose. In that situation, the network would not benefit from any change. Such wrong output shows that, by looking to a single metric, in a single condition, it may lead to undesired steering decisions, and therefore it is hard to have a steering algorithm with good performance. In the final 60 seconds, the Rate condition did output a right decision.

We will now analyze how the TX behaves. Figure 4.5 illustrates the TX values observed in the link between the Root AP and Extender 1 and the link between the two Extenders, as well as the load observed in the same backhaul links. The Tx metric condition's behavior was very similar to the one seen for the Rate's condition. In the first minute of the test, the output was right and the same happened for the following 4 minutes. Only during the sixth minute, the TX condition was wrongly triggered. In the final 60 seconds, the condition identified correctly the need for a topology and frequency change. The behavior of this condition was not as good as the Rate condition because, during the tests, some spikes in the TX values were very close to the decision threshold, a situation to be considered during the weight distribution for the different conditions.

At this point it was clear that a single condition would not be enough to correctly decide whether to steer or not. Each condition eventually outputs a wrong decision, and even though some conditions perform better than others, it is expected that the algorithm could benefit all them. Giving the same importance to a condition that outperforms others would not be fair and could harm the overall decision process. With the obtained results in mind, the conditions were ranked in order to decide which one was more important. The most important one was the Rate condition, followed by the TX condition. These two conditions showed a better performance only failing in one of the situations. The TXOP condition is very important because it did not fail where the previous two did. The RSSI and Idle conditions were the ones with the worst performance and sit on the bottom of the conditions rank. It was decided that the sum of the weights would be equal to 100 points. To each condition, a weight was given if the condition was fulfilled at a given moment and another smaller weight was given if the condition was fulfilled for the average value of the last 15 seconds (in order to avoid a wrong steer caused by a spike in the parameters). The final weight distribution for the 5GHz to 2.4GHz steering algorithm was the following:

- Rate condition (instant) - 30 points;
- Rate condition (average) - 10 points;
- TX condition (instant) - 15 points;
- TX condition (average) - 5 points;
- TXOP condition (instant) - 15 points;
- TXOP condition (average) - 5 points;
- RSSI condition (instant) - 5 points;
- RSSI Condition (average) - 5 points;
- Idle condition (instant) - 5 points;
- Idle condition (average) - 5 points.

In a similar process as the one described before, we tried to understand which weights to give to each condition of the 2.4GHz to 5GHz backhaul steering process. For this case, a new topology was built with one Root AP and two Extenders, with Extender 1 connected through 5GHz backhaul link to Root AP, and Extender 2 connected to Root AP through a 2.4GHz backhaul connection (as illustrated in Figure 3.15). Clients 1 and 2 are connected to Extender 1 and 2, respectively, through Ethernet. The overall test duration was five minutes, with the TCP traffic load injected by the Clients changing every minute. During the whole duration of the test, the link between the Root AP and Extender 2, through 2.4GHz, was being pushed to the limit. In the first minute, the link between the Root AP and the first Extender had a maximum load being applied, through Client 1. In the second minute, Client 1 decreased the load down to 220 Mbits/s, and again in the third minute down to 180 Mbits/s. In the fourth minute, the load was reduced to 100 Mbits/s. In the fifth and final minute, Client 1 was silent with the only load in the network being applied by Client 2 in the 2.4GHz backhaul link between the Root AP and the Extender 2. The traffic variations in the 5GHz link were designed to show limit cases of the conditions defined. For ease of understanding, Figure 4.6 illustrates the loads injected by each Client during the experiment.

Looking at the performance of the Rate condition, the results obtained were according to the expected. The condition did not recognize the need for a steer during the first two minutes (where the rate values were higher than 200 Mbits/s), and triggered a steering indication in the rest of the test (minutes 3, 4 and 5). This condition performance was within its purpose, since it guarantees the existence of the hysteresis interval and the improvement of the mesh network performance.

We will now analyze how the TXOP metric behaves in the test presented before. Figure 4.7 illustrates the TXOP values observed in the link between the Root AP and Extender 1, as

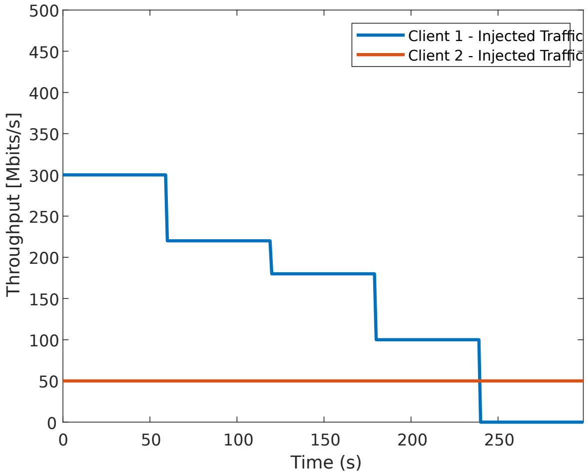

Figure 4.6: Injected Traffic (2.4GHz to 5GHz steering process).

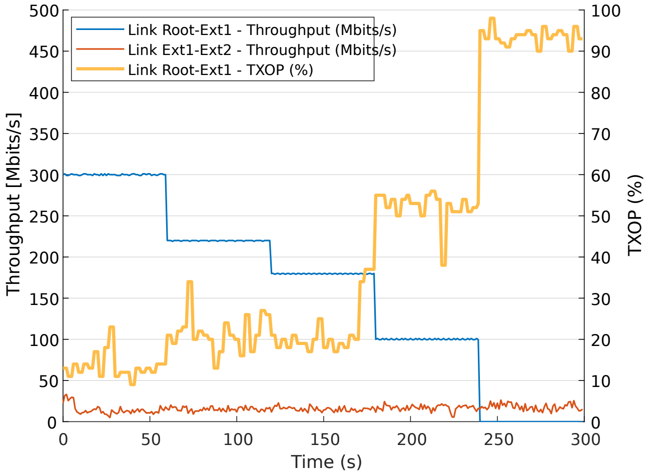

Figure 4.7: TXOP metric behavior (2.4GHz to 5GHz steering process).

well as the load observed in all active backhaul links. Analysing the TXOP metric condition, it was easy to observe that the performance was very similar to the one observed in the Rate condition. During the first two minutes, the TXOP condition did not recognise the need to change the backhaul links. In the other minutes (3, 4 and 5), the condition recognised that a steer would be useful for the performance of the mesh network.

Finally, we will now analyze how the TX metric behaves. Figure 4.8 illustrates the TX values observed in the link between the Root AP and Extender 1, as well as the load observed in all active backhaul links. The TX condition had a behavior slightly different than the one in other conditions. In the first 3 minutes, the condition was not triggered, showing that it did not recognise a steer situation. In the rest of the test (4 and 5 minutes), the condition activated and indicated a steering situation.

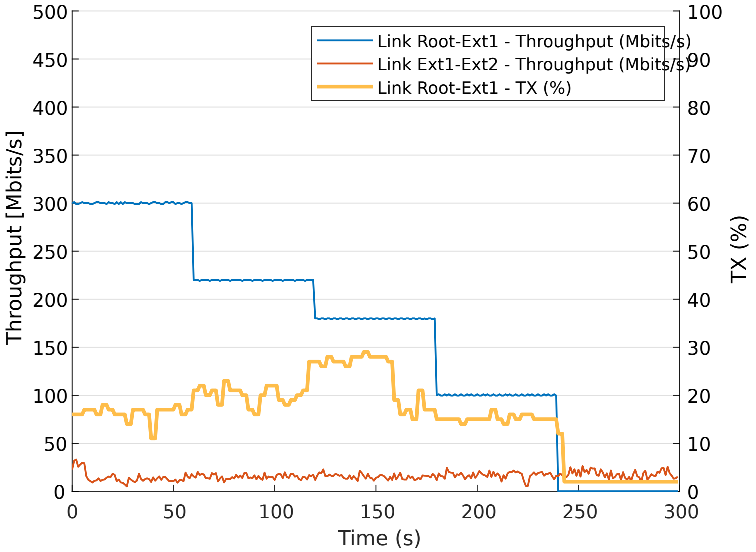

Figure 4.8: TX metric behavior (2.4GHz to 5GHz steering process).

To wrap up all the steering conditions designed for the 2.4GHz to 5GHz steering process, we decided to weight each condition with a weight total sum of 100 points. All conditions had a similar performance, indicated the need for a steer at almost the same situations. The weight distribution in this mechanism was more balanced (all conditions have similar weights). Since the Rate condition was the best condition to identify every scenario, its weight was slightly higher than the other conditions. The TXOP and TX conditions had the same weight assigned to both. The final weight distribution for the 2.4GHz to 5GHz steer mechanism was the following:

- Rate condition (instant) - 30 points;
- Rate condition (average) - 10 points;
- TX condition (instant) - 20 points;
- TX condition (average) - 10 points;
- TXOP condition (instant) - 20 points;
- TXOP condition (average) - 10 points.

4.3 Backhaul Steering Algorithms

As mentioned before, the algorithm has two different mechanisms, one to steer a backhaul link from 5GHz to 2.4GHz, and another one to the opposite direction. The existence of these two mechanisms translates in two separate algorithms, one for each mechanism, presented in Algorithm 1 and Algorithm 2. Each algorithm can be divided into three parts. The first part is the collection of data and metrics to feed the algorithm. As mentioned before, metrics are collected every three seconds due to the available setup. The collection is done by calling scripts that establish *ssh* connections to the Extenders on the mesh network to retrieve data and to parse it. During the execution of the code, the last 15 seconds of network information are stored in vectors (3 vectors in total, one for each monitored link). After collecting metrics, the second phase starts. This phase consists of testing all the conditions that were defined before to update a score variable. Each condition is tested for the latest available results, and if the condition is satisfied, the condition is tested again for the average value of the last 15 seconds. In case of success, the score is updated again. After the validation of all the conditions, the final score is calculated and compared with the steer decision threshold, defined as $Steer_{threshold_5}$ or $Steer_{threshold_{2.4}}$, depending if it is a steer from 5GHz to 2.4GHz, or the opposite, respectively. When the score value satisfies the $Steer_{threshold}$ value, a script that executes the needed commands (at the Root AP of the mesh network) is called and the mesh network adapts and changes its topology. In the cases where a steer is not justified, the algorithm waits (sleeps) for a while (time needed to have new metrics). All these three subsections in the code are inside an endless loop and, in the beginning of each loop, the score variable is set to 0. The program runs in this loop until it is interrupted by an external stop command. The variable values used in the code were the ones defined in section 4.1. The value used in the condition of the score ($Steer_{threshold_5}$) is 80 for the 5GHz to 2.4GHz mechanism, because this value does not allow a steer whenever there is a spike (bigger then the sum of the weights in a instant condition) in the metrics of the Extenders. For the 2.4GHz to 5GHz mechanism, the defined score value defined ($Steer_{threshold_{2.4}}$) is 70 (for the same reason). The execution of each mechanism is done according to the current topology of the mesh network, and the two are never executed simultaneously.

Algorithm 1: 5GHz to 2.4GHz backhaul steering algorithm.

```
Input: Network Metrics
Output: Steer Decision
$Link1 \leftarrow$ metrics of backhaul link 1 (Root AP-Extender 1 on 5GHz);
$Link2 \leftarrow$ metrics of backhaul link 2 (Extender 1-Extender 2 on 5GHz);
$Link3 \leftarrow$ metrics of backhaul link 3 (Root AP-Extender 2 on 2.4GHz);
while True do
    score = 0
    if $Link1[rate] > Rate_{threshold_1}$ and $Link2[rate] < Rate_{threshold_2}$ then
        score = score + 30
        if $avg(Link1[rate]) > Rate_{threshold_1}$ and $avg(Link2[rate]) < Rate_{threshold_2}$ then
            score = score + 10
        end
    end
    if $Link1[tx] > TX_{threshold}$ and $Link2[rate] < Rate_{threshold_2}$ then
        score = score + 15
        if $avg(Link1[tx]) > TX_{threshold}$ and $avg(Link2[rate]) < Rate_{threshold_2}$ then
            score = score + 5
        end
    end
    if $Link1[txop] > TXOP_{threshold_1}$ and $Link2[txop] < TXOP_{threshold_2}$ then
        score = score + 15
        if $avg(Link1[txop]) > TXOP_{threshold_1}$ and $avg(Link2[txop]) < TXOP_{threshold_2}$ then
            score = score + 5
        end
    end
    if $Link1[rssi]$ - $Link2[rssi] < RSSI_{threshold}$ then
        score = score + 5
        if $avg(Link1[rssi])$ - $avg(Link2[rssi]) < RSSI_{threshold}$ then
            score = score + 5
        end
    end
    if $Link2[idle] > Link2[txop] + Idle_{threshold}$ then
        score = score + 5
        if $avg(Link2[idle]) > avg(Link2[txop]) + Idle_{threshold}$ then
            score = score + 5
        end
    end
    if $score > Steer_{threshold_5}$ then
        5 GHz TO 2.4 GHz STEER
    end
    else
        sleep(3) // Wait for the metrics to bee refreshed
    end
end
```

Algorithm 2: 2.4GHz to 5GHz backhaul steering algorithm.

```
Input: Network Metrics
Output: Steer Decision
$Link1 \leftarrow$ metrics of backhaul link 1 (Root AP-Extender 1 on 5GHz);
$Link2 \leftarrow$ metrics of backhaul link 2 (Extender 1-Extender 2 on 5GHz);
$Link3 \leftarrow$ metrics of backhaul link 3 (Root AP-Extender 2 on 2.4GHz);
while True do
    score = 0
    if $Link1[rate] < Rate_{threshold_3}$ then
        score = score + 30
        if $avg(Link1[rate]) < Rate_{threshold_3}$ then
            score = score + 10
        end
    end
    if $Link1[tx] < TX_{threshold_2}$ then
        score = score + 20
        if $avg(Link1[tx]) < TX_{threshold_2}$ then
            score = score + 10
        end
    end
    if $Link1[txop] > TXOP_{threshold_3}$ then
        score = score + 20
        if $avg(Link1[txop]) > TXOP_{threshold_3}$ then
            score = score + 10
        end
    end
    if $score > Steer_{threshold_{2.4}}$ then
        2.4 GHz TO 5 GHz STEER
    end
    else
        sleep(3) // Wait for the metrics to refreshed
    end
end
```

After developing the whole algorithm, several performance tests were conducted. Three tests were defined, two to analyse the individual performance of each mechanism, and one to test the algorithm as a whole.

To analyse the performance of the 5GHz to 2.4GHz steer mechanism, several situations and network variations were tested. The nodes in the mesh network were connected in a daisy chain topology with two Clients connected to the Extenders (Client 1 connected to Extender 1 and Client 2 connected to Extender 2). The test started with one minute where no traffic was running through the backhaul links of the mesh. In the following 3 minutes (minute 2, 3 and 4), the only traffic applied was on the link between the Root AP and the Extender 1, through Client 1. During these 3 minutes, the injected traffic varied between 100 Mbits/s, 200 Mbits/s and the maximum load the backhaul link could handle. In the minute number 5, there was no traffic injected in the link between the Root AP and the first Extender, and the link between the two Extenders was pushed to the limit (traffic injected through Client 2). In the sixth minute, both backhaul links had traffic being injected (the link between the Root AP and Extender 1 had maximum traffic injected, while the link between the two Extenders had a 12 Mbits/s load being injected). During the remaining time, both backhaul links had maximum traffic being injected.

Figure 4.9 represents the performance during the tests. This figure represents the throughput of each link, as well as the score that was being calculated and used for the steer decision during the execution of the test.

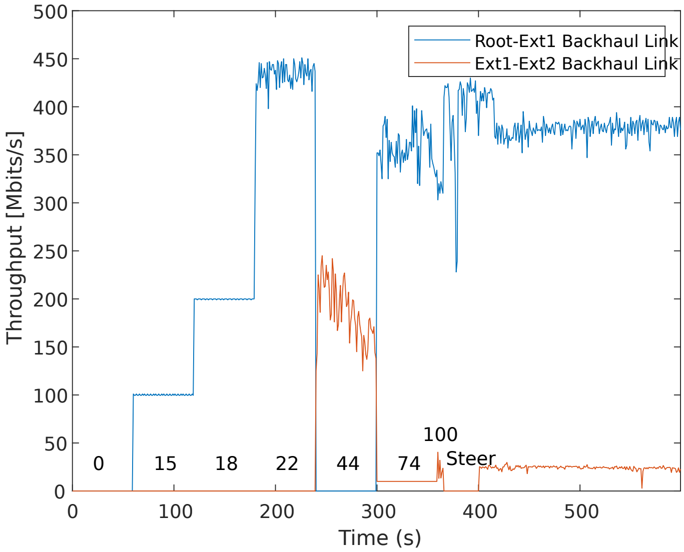

Figure 4.9: Backhaul 5 GHz to 2.4 GHz Steer Algorithm Test (TCP).

Looking at the obtained results, it is possible to observe that the algorithm performed as expected, since the steer decision was taken in the right time. During the first 60 seconds, the score was 0 since no conditions were activated. When the traffic in the link between the Root AP and Extender 1 started to increase, it was observed that the score value increased.

The values in the following 3 minutes were the ones expected and averaged 15 in the first minute, 18 in the second and 22 in the final minute. In the fourth minute, a bigger increase in the score was spotted. During this part of the test, the score value averaged 44 points, 22 more than what was seen before. In the next minutes (6 and 7), the score value increased significantly, just like it was expected, since it is the scenario where a steer would ideally happen. When the load of the link between the Extenders was not the maximum it could handle, it was seen that the score value averaged 74, which is really close to the limit that was defined earlier. Even though the average value was 74, it never surpassed the 80 points limit and never triggered a steer decision. When scenario 3 was applied with no variation, the score went up to the 100 points, which was more than enough to trigger the steer. Looking at the score values and at the timing of the steer decision, it can be concluded that the performance is good, and that it correctly identifies the situation where the network benefits from changing the used band in the link.

When the steer occurred, a service outage was seen for around 30 seconds. The steer process takes (in the worst scenario) 5 seconds to be concluded. The remaining 25 seconds of no service in the link between the two Extenders can be justified by the tool that was used to generate traffic in the links. As referred before, iperf3 was used and during this test the traffic was generated using TCP packets. Since TCP is a connection based protocol and since the connection was changed, it took around 25 seconds to reestablish the connection again (this behavior does not mean that there was no connection between Root and Extender 2). Looking at the throughput values obtained, a significant difference can be seen before and after the steer. Before the steer, the average throughput in the link between the Root AP and Extender 1 was 317 Mbits/s, and 25.3 Mbits/s on the link between Extender 1 and 2. After the steer, the throughput values were higher than before. The link between the Root AP and the Extender 1 throughput value went up to 368.4 Mbits/s, while the link that changed the frequency band averaged 28.4 Mbits/s. As it was observed by looking at the values, both links performance was improved, showing even more that using the 2.4GHz can be a valid option in certain situations. The throughput went up 16% in the link that did not change, and went up 12% in the link that suffered a change. The standard deviation on the values decreased significantly (it can be seen in figure 4.9).

To test the 2.4GHz to 5GHz steering mechanism, another test was conducted. The nodes were configured in a star topology, where the Extender 1 and Root AP were connected through 5GHz and Extender 2 and Root AP were initially connected through 2.4GHz, and a Client was connected to each Extender. The test had the duration of 4 minutes and 20 seconds, and featured several variations on the network. During the whole test, the backhaul link of the Extender 2 had as much traffic has it could handle being injected (through Client 2). The other link, during the first minute, had a maximum load being applied by Client 1. In the second minute, the injected load was reduced to 220 Mbits/s, and in the third minute the load was reduced to 180 Mbits/s. For the remaining time, the injected load was 100 Mbits/s.

Looking at the obtained results (represented in Figure 4.10), it is observed that the algorithm behaved and performed as expected, since the steering was triggered when it was

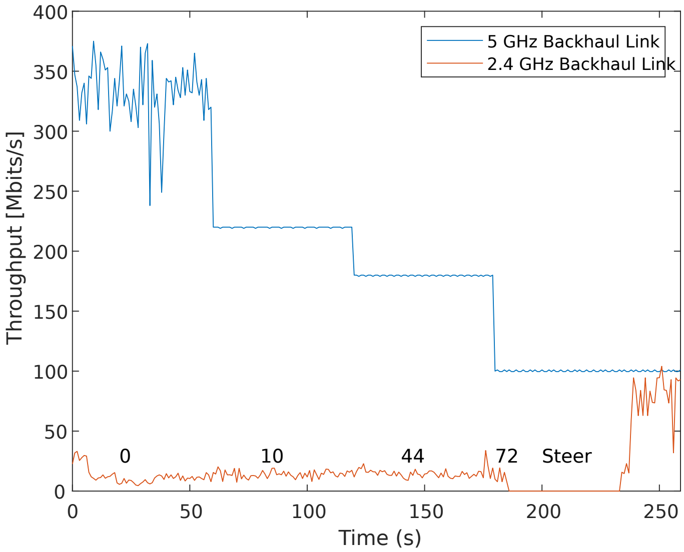

Figure 4.10: Backhaul 2.4 GHz to 5 GHz Steer Algorithm Test (TCP).

supposed, by moving the Extender 2 to be connected to Extender 1 through 5GHz. It is easy to notice that the tolerance interval implemented is fulfilling its purpose, and it can be concluded that it will help to avoid excessive steer commands. During the first minute, the score value was 0 and increased the next minutes. In the following 3 minutes, the values increased from 10, to 44 and then to 72. The last value was enough to trigger a steer command that changed the link to the 5GHz frequency.

After executing and analysing the performance results of the steer mechanism, it was decided to test both at the same time. To test the whole developed algorithm, a combination between the two previous tests was designed. The objective was to see how the performance of the algorithm and the mesh network would be in a situation where both types of steer would be needed. The test was conducted two times, one with the steering algorithms and another without the algorithms, to see if there was a true improvement and, if there was, how big was the improvement. The starting topology in this test was in a daisy chain with a Client connected to each Extender. This test had a duration of 11 minutes in total. In the first minute, the backhaul links had no traffic being injected. In the next 3 minutes, the traffic was applied in the link between the Root AP and the Extender 1 (each minute the traffic was increased), through Client 1. In the fifth minute, the traffic was injected to the link between the two Extenders by Client 2. From the sixth minute and until the end of the test, the backhaul link of the Extender 2 had maximum traffic (Client 2 was injecting as much traffic as it could). During the minutes 6, 7 and 8 the amount of traffic in the link of the Extender 1 was increased (to induce a 5GHz to 2.4GHz steer) and stayed at the maximum for another minute (minute 9). Then, the traffic values were gradually reduced (to see the 2.4GHz to 5GHz steer happen) until the end of the test.

The results, presented in Figures 4.11 and 4.12 show (again) that there were situations where the steer commands from the algorithm improved the quality and the performance

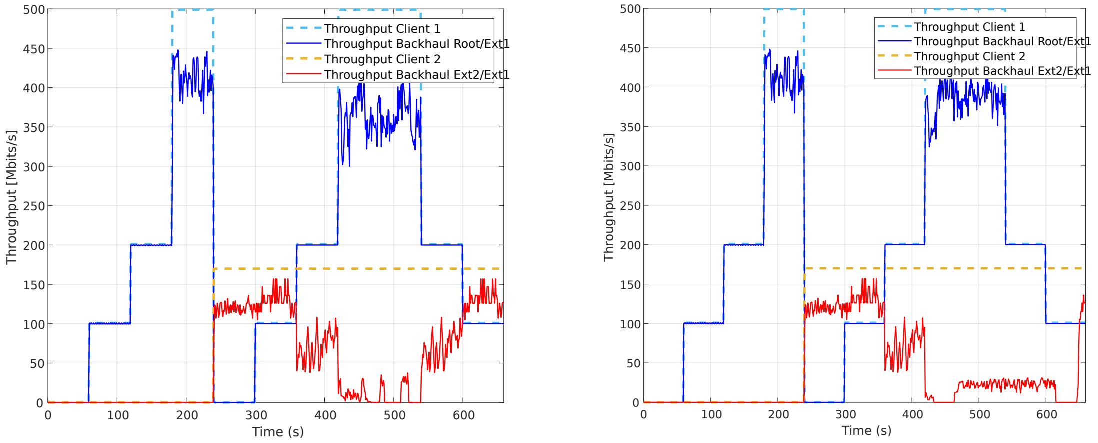

Figure 4.11: Backhaul steer algorithm test (TCP).

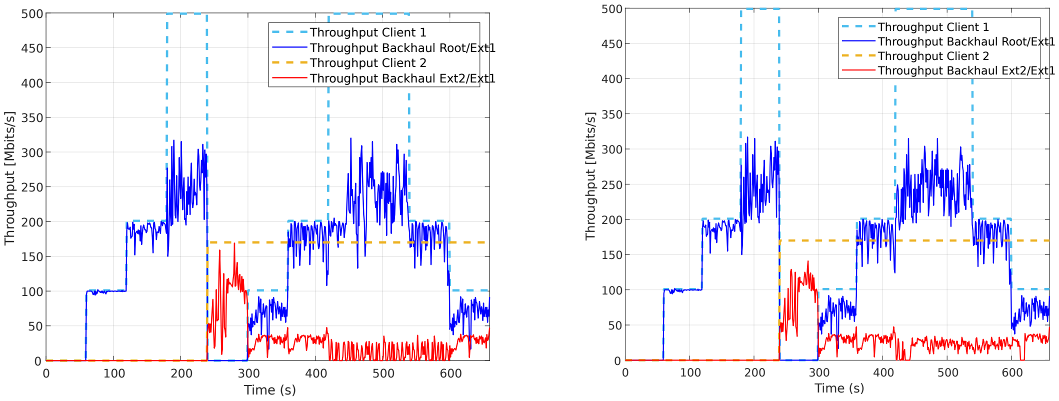

Figure 4.12: Backhaul steer algorithm test (UDP).

of the network. It was seen that, even with the service outage associated with the steering processes, the overall performance improved (specially in the UDP test). The throughput of all of the backhaul links showed higher and more consistent values. It was also observed that, when using the algorithm, there were no periods without connection (besides during the steer command). Opposite to that, the test without the algorithm running showed several intervals without service. It can also be seen that, in some specific cases, the mechanism to steer from the 2.4GHz to the 5GHz does not allow the network to take full advantage of the available resources (a downside to avoid excessive steers). Considering that the steer process implemented is not optimized and its time can be greatly reduced, we can conclude that using the proposed algorithms and enable the steering to the 2.4GHz band when really needed presents better results.

4.5 Final Remarks

This chapter explained the whole process behind the proposed algorithms. The conditions, its weights and all the tests needed to tune the process were described and justified.

After analysing the results obtained from the proposed algorithms, it was observed that they were able to identify correctly the situations where a steer would be needed. The link performance also showed a significant improvement in the overall network performance (in the bandwidth of the links and in their jitter values). The improvements were observed in both links (16% in the first link and 12% in the second link when the network used 2.4GHz links), and helped to concluded that the overall performance of the algorithms was good and that all proposed objectives have been accomplished. The direct comparison between using and not the algorithms showed that the algorithms provide performance advantages, and the only downsides are related to the need to reduce the steering times. The overall performance shows that the proposed algorithms can in fact be beneficial to these dual-band networks.

CHAPTER 5

Conclusions and Future Work

The main objective of this dissertation was to improve and to optimize the backhaul links' performance in a WMN in daisy chain topologies. Since in the available setup the devices were capable of operating with several frequency bands in the backhaul links, 5GHz and 2.4GHz, it was decided to find situations where the use of a band with less capacity but with higher coverage (such as 2.4GHz) could be used. To do that four scenarios were defined in order to look at the properties of the backhaul links and how they could change in certain conditions. Several traffic variations were tested in the setup and several different topologies were experimented. It was concluded that, in a given situation, there is a scenario where a backhaul link in 2.4GHz could be used to overcome the saturation of the 5GHz backhaul link. It was shown that, on a daisy chain, the number of hops restrains the amount of traffic that can be transmitted from a Client to the Root AP. The scenarios also showed that saturating a link would reduce and limit the performance of the links further way from the Root AP. When all of these mentioned conditions were put together, the properties of the 2.4GHz frequency (longer distance transmissions) stood out as a way of avoiding the aforementioned downsides. The use of a 2.4GHz allowed to reduce the number of hops between the Client and the Root AP, reducing the noise in the channel (if the transmissions are all in the same frequency, the conflicts between them increase). The use of the 2.4GHz provided an improvement for all the backhaul links: the ones that did not change benefited from the reduction of interference and load, and the one that changed benefited from the reduction of hops and also from the lack of noise in the band.

After showing that using a 2.4GHz backhaul link in a mesh network with daisy chain topologies can enhance the overall network performance, the next objective was to automate a steering mechanism to change the topology as needed. Two mechanisms were developed, one to identify steer needs from 5GHz to 2.4GHz, and one to identify the opposite situations (2.4GHz to 5GHz). To develop the solutions several metrics were analysed and some relations between metrics were noticed. From that, several conditions were proposed to identify the situations where steering and changing frequency bands would increase the network performance. The metrics used in the mechanism were RSSI, Idle, TXOP, TX and Rate. After testing the conditions and adapting the parameters, some weight distribution tests were conducted in order to determine how much a condition should weight in the algorithm. These tests, that consisted in deploying several different network scenarios, provided a clear view of the behavior of the conditions. The test results allowed to understand which conditions needed to have more impact on the algorithm.

After the execution of the final performance tests, results showed that the algorithm was able to identify both steer scenarios (5GHz to 2.4GHz and 2.4GHz to 5GHz) and situations correctly. Several network situations and different traffic injection was applied to the network in order to see how the algorithm performed. The algorithm triggered steering commands when it was supposed to, and that improved the performance of the mesh network.

The time needed for the network to adapt to the execution of a steer command can be long (5 seconds or more), making it one of the few downsides of a topology change. Still, this behavior occurs because the instructions used to perform a backhaul steer were not the optimal ones, but the ones available in the hardware and software used in this work, and this will be improved in the future.

Even with the steering time downside, a look at the performance values captured during the tests showed an improvement of around 12-16% in each active link of the network (when using a 2.4GHz backhaul link). Another important conclusion is that the algorithms is able to prevent excessive steer commands (even if sometimes it does not take advantage of all of the network resources).

In conclusion, the presented results show an enhance on the mesh network performance by using a different frequency in backhaul links. This process of decision (whether to change or not the frequency used) can be implemented for different setups with more nodes and different characteristics to improve other mesh networks.

5.1 Future Work

The WiFi and mesh network scenarios are in constant improvement and expansion. The work developed in this dissertation can be, in the future, improved. Some examples are now introduced for discussion:

- **Scenarios with more and different nodes can be studied**: Studying new scenarios with a different number of network devices could improve the conditions defined in the steering mechanism as well as discover other situations where the use of a different frequency band can improve the overall network performance;
- **Improve the bandwidth steering process**: The results obtained could be much improved if the steering time can be reduced. It is assumed that this steering process is not optimal, and that it can be improved;
- **Application and adaptation of the developed mechanism to a setup with tri-band capabilities (WiFi 6E)**: WiFi 6E is the newest version of the WiFi standard that uses three frequencies (2.4GHz, 5GHz and 6GHz). These tri-band capable devices will be incorporated in EasyMesh solutions, making it almost mandatory to use the new

added frequency to improve the backhaul performance on this kind of networks; This will require an overall new approach for the optimization of the topology and frequencies used in the different links;

- **Development of Machine Learning solution using the collected data**: Machine Learning can have great performance in solutions that need to predict what is going to happen in the future. A well trained machine learning model would be able to change the network topology before the network actually needs it. This would reduced the negative impact of the steer process.

References

[1] *Internet usage worldwide – statistics & facts | statista*, Accessed: 2022-10-26. [Online]. Available: `https://www.statista.com/topics/1145/internet-usage-worldwide/`.

[2] *Smartphone sales worldwide 2007-2021 | statista*, Accessed: 2022-10-26. [Online]. Available: `https://www.statista.com/statistics/263437/global-smartphone-sales-to-end-users-since-2007/`.

[3] A. F. Rochim and R. F. Sari, "Performance comparison of ieee 802.11n and ieee 802.11ac," in *2016 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, 2016, pp. 54–59. DOI: `10.1109/IC3INA.2016.7863023`.

[4] H. A. Omar, K. Abboud, N. Cheng, K. R. Malekshan, A. T. Gamage, and W. Zhuang, "A survey on high efficiency wireless local area networks: Next generation wifi," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2315–2344, 2016. DOI: `10.1109/COMST.2016.2554098`.

[5] D. Vassis, G. Kormentzas, A. Rouskas, and I. Maglogiannis, "The ieee 802.11g standard for high data rate wlans," *IEEE Network*, vol. 19, no. 3, pp. 21–26, 2005. DOI: `10.1109/MNET.2005.1453395`.

[6] E. Perahia, "Ieee 802.11n development: History, process, and technology," *IEEE Communications Magazine*, vol. 46, no. 7, pp. 48–55, 2008. DOI: `10.1109/MCOM.2008.4557042`.

[7] E. H. Ong, J. Kneckt, O. Alanen, Z. Chang, T. Huovinen, and T. Nihtilä, "Ieee 802.11ac: Enhancements for very high throughput wlans," in *2011 IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications*, 2011, pp. 849–853. DOI: `10.1109/PIMRC.2011.6140087`.

[8] E. Khorov, A. Kiryanov, A. Lyakhov, and G. Bianchi, "A tutorial on ieee 802.11ax high efficiency wlans," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 197–216, 2019. DOI: `10.1109/COMST.2018.2871099`.

[9] Q. Qu, B. Li, M. Yang, *et al.*, "Survey and performance evaluation of the upcoming next generation wlans standard-ieee 802.11 ax," *Mobile Networks and Applications*, vol. 24, no. 5, pp. 1461–1474, 2019.

[10] J. Liu *et al.*, "Reliable dual sub-carrier modulations (dcm) for he-sig-b and data," vol. 12, p. 25, Sep. 2015. [Online]. Available: `https://mentor.ieee.org/802.11/dcn/15/11-15-1068-01-00ax-reliable-transmission-schemes-for-he-sig-b-and-data.pptx`.

[11] M. S. Afaqui, E. Garcia-Villegas, E. Lopez-Aguilera, G. Smith, and D. Camps, "Evaluation of dynamic sensitivity control algorithm for ieee 802.11ax," in *2015 IEEE Wireless Communications and Networking Conference (WCNC)*, 2015, pp. 1060–1065. DOI: `10.1109/WCNC.2015.7127616`.

[12] D.-J. Deng, Y.-P. Lin, X. Yang, *et al.*, "Ieee 802.11ax: Highly efficient wlans for intelligent information infrastructure," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 52–59, 2017. DOI: `10.1109/MCOM.2017.1700285`.

[13] Q. Qu, B. Li, M. Yang, and Z. Yan, "An ofdma based concurrent multiuser mac for upcoming ieee 802.11ax," in *2015 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, 2015, pp. 136–141. DOI: `10.1109/WCNCW.2015.7122543`.

[14] R. Liao, B. Bellalta, M. Oliver, and Z. Niu, "Mu-mimo mac protocols for wireless local area networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 162–183, 2016. DOI: `10.1109/COMST.2014.2377373`.

[15] Z. Machrouh and A. Najid, "High efficiency ieee 802.11ax mu-mimo and frame aggregation analysis," in *2018 International Conference on Advanced Communication Technologies and Networking (CommNet)*, 2018, pp. 1–5. DOI: `10.1109/COMMNET.2018.8360264`.

[16] W.-F. Alliance, "Wi-fi easymeshtm specification version 4.0," 2021.

[17] L. Song, D. Hatzinakos, and X. Wang, "Wireless mesh infrastructure for ubiquitous voice and video," in *2008 5th IEEE Consumer Communications and Networking Conference*, 2008, pp. 1267–1268. DOI: `10.1109/ccnc08.2007.302`.

[18] D. Seither, A. König, and M. Hollick, "Routing performance of wireless mesh networks: A practical evaluation of batman advanced," in *2011 IEEE 36th Conference on Local Computer Networks*, 2011, pp. 897–904. DOI: `10.1109/LCN.2011.6115569`.

[19] *Batman v - batman-adv - open mesh*, Accessed: 2022-09-14. [Online]. Available: `https://www.open-mesh.org/projects/batman-adv/wiki/BATMAN_V`.

[20] G. Merlin Sheeba and A. Nachiappan, "Fuzzy differential evolution based gateway placements in wmn for cost optimization," in *Intelligent Systems Technologies and Applications*, S. Berretti, S. M. Thampi, and S. Dasgupta, Eds., Cham: Springer International Publishing, 2016, pp. 137–145.

[21] S. Sakamoto, K. Ozera, M. Ikeda, and L. Barolli, "Implementation of intelligent hybrid systems for node placement problem in wmns considering particle swarm optimization, hill climbing and simulated annealing," *Mobile Networks and Applications*, vol. 23, no. 1, pp. 27–33, 2018.

[22] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm intelligence*, vol. 1, no. 1, pp. 33–57, 2007.

[23] S. K. Narayana and N. T. Hosur, "Priority based trust efficient routing using ant colony optimization for iot-based mobile wireless mesh networks," *International Journal of Intelligent Engineering and Systems*, vol. 15, no. 2, pp. 99–106, 2022.

[24] J. Shi, J. Yu, H. Li, and R. Zong, "Aodv protocol optimization based on cross-layer design of wmn," in *2011 7th International Conference on Wireless Communications, Networking and Mobile Computing*, 2011, pp. 1–4. DOI: `10.1109/wicom.2011.6040282`.

[25] G. D VENU and S. SOUJANYA, "Enhancing the performance of wireless mesh network (wmn) for congestion aware load balancing," *Journal of Engineering Sciences*, vol. 9, no. 5, 2018.

[26] A. Kershenbaum, P. Kermani, and G. Grover, "Mentor: An algorithm for mesh network topological optimization and routing," *IEEE Transactions on Communications*, vol. 39, no. 4, pp. 503–513, 1991. DOI: `10.1109/26.81738`.

[27] Z. Gu, J. Zhang, and Y. Ji, "Topology optimizing in fso-based uavs relay networks for resilience enhancement," *Mobile Networks and Applications*, vol. 25, no. 1, pp. 350–362, 2020.

[28] M. Kretschmer, P. Batroff, and G. Ghinea, "Topology forming and optimization framework for heterogeneous wireless back-haul networks supporting unidirectional technologies," *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 698–710, 2013, ISSN: 1084-8045. DOI: `https://doi.org/10.1016/j.jnca.2012.12.022`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S108480451200269X`.

[29] P. Jesus, N. Silva, J. Ribeiro, J. Pereira, M. Luís, and S. Sargento, "Method and system for enhanced steering and traffic load balancing in wireless mesh networks," 2019, WO publication WO2021045633A1.

[30] B. Kuskonmaz, H. Ozkan, and O. Gurbuz, "Machine learning based smart steering for wireless mesh networks," *Ad Hoc Networks*, vol. 88, pp. 98–111, 2019, ISSN: 1570-8705. DOI: `https://doi.org/10.1016/j.adhoc.2019.01.005`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S1570870518308965`.

[31] A. Mesodiakaki, E. Zola, R. Santos, and A. Kassler, "Optimal user association, backhaul routing and switching off in 5g heterogeneous networks with mesh millimeter wave backhaul links," *Ad Hoc Networks*, vol. 78, pp. 99–114, 2018, ISSN: 1570-8705. DOI: `https://doi.org/10.1016/j.adhoc.2018.05.008`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S1570870518302397`.