



HAL
open science

Performance Enhancement in Wi-Fi Networks using Machine Learning

Ibrahim Sammour

► **To cite this version:**

Ibrahim Sammour. Performance Enhancement in Wi-Fi Networks using Machine Learning. Machine Learning [cs.LG]. Université Clermont Auvergne, 2023. English. NNT : 2023UCFA0041 . tel-04393773

HAL Id: tel-04393773

<https://theses.hal.science/tel-04393773v1>

Submitted on 15 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



PhD Thesis

Defense date: 27/06/2023

By: Ibrahim Sammour

Performance Enhancement in Wi-Fi Networks using Machine Learning

The jury is composed of the following members:

Andrzej Duda Full Professor, Grenoble INP-Ensimag	reviewer
Abed Ellatif Samhat Full Professor, Faculty of Engineering - Lebanese University	reviewer
Véronique Veque Full Professor, Université Paris-Saclay	reviewer, jury president
Cherifa Boucetta Associate Professor, Université de Reims Champagne-Ardenne	examiner
Gerard Chalhoub Associate Professor, Université Clermont Auvergne	supervisor
Gil De Sousa Research Engineer, CR INRAé	co-supervisor

A thesis submitted in fulfillment of the requirements of PhD degree
LIMOS UMR 6158 CNRS Laboratory
Doctoral School of Science for Engineering
Clermont Auvergne University, France

Acknowledgments

This work was the outcome of three years of research, involving the participation of numerous individuals who contributed to its success.

First, I express my gratitude to my family, specifically my father Adnan, my mother Mariam, and my brothers and sister (Hassan, Nour, and Khalil), for their unwavering support throughout this journey.

I extend my sincere appreciation to my director Gerard Chalhoub, whose assistance and encouragement were invaluable during the academic pursuit of this thesis and during challenging times. I would also like to thank my co-supervisor Gil De Sousa for his presence and support throughout my work.

I am thankful to the members of the jury for their time and valuable feedback, starting with the reviewers Prof. Véronique Veque, who also served as president of the jury, as well as Prof. Abed Ellatif Samhat and Prof. Andrzej Duda. I also extend my gratitude to Dr. Cherifa Boucetta who served as an examiner in the jury.

Additionally, I would like to express my gratitude to my colleagues, Ali, Mouna, Samar, Ines, Theo, Lea, Omaira, Amani, and Wassim, for enabling a great atmosphere in the office.

Abstract

Wireless technologies are used in various applications due to their ease of deployment, and inherent capabilities to support mobility. Wireless Fidelity (Wi-Fi) is one of the popular wireless technologies. The rapid growth of Wi-Fi networks, the increasing demand for network capacity in terms of bandwidth, and the growing number of users cause the densification of deployed networks. Wi-Fi networks suffer from deterioration of the Quality of Experience (QoE) in dense deployments. This thesis examines different ways to improve the performance of Wi-Fi networks. It focuses, on one hand, on the deployment of Wi-Fi networks taking into account the capacity of the technology and the needs of the application, and on the other hand, on Reinforcement Learning (RL) based approaches for adapting the amount of transmitted data and improving the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol.

As part of this thesis, we first analyze existing rate adaptation algorithms and their impact on the performance of Wi-Fi networks. Then, we study the best practices for deploying these networks within the context of smart farming, including the number of required access points and their location to maximize coverage and performance in a remote monitoring and control application for mobile agricultural robots. The deployment technique leverages Wi-Fi features such as beamforming. In the second part, we exploit reinforcement learning mechanisms to dynamically adapt application profiles based on network conditions and user requirements. Finally, we exploit RL mechanisms to improve the CSMA/CA protocol by adapting the backoff duration to the network state.

Through simulation results, we demonstrate that taking into account the impact of Multiple Input Multiple Output (MIMO) antennas in the Wi-Fi network deployment strategies reduces the required number of access points and achieves better coverage. A deployment strategy that takes into account the network capacity results in a trade-off between the number of access points, coverage, required antennas per access point, and performance. Our reinforcement learning models bring a significant improvement in Wi-Fi network performance. These models allow dynamic adaptation of application profiles based on network conditions and user requirements, resulting in better network throughput and fewer data losses. Moreover, the model used at the CSMA/CA algorithm level has led to fewer access delays, improved overall throughput, fewer collisions, and better fairness among users.

Through this research work, we demonstrate, using a numerical simulation evaluation approach, the effectiveness of reinforcement learning to improve the performance of different aspects of a Wi-Fi network. The results show the potential for significant performance improvements and better user experience in Wi-Fi networks through the application of these optimization methods. Future work should continue to explore and refine these techniques to ensure that Wi-Fi networks can meet the ever-increasing demand for reliable, high-quality wireless connectivity.

Résumé

Les technologies sans fil sont utilisées dans diverses applications en raison de leur facilité de déploiement et de leurs capacités inhérentes à prendre en charge la mobilité. Le Wi-Fi est l'une des technologies sans fil populaires. La croissance rapide des réseaux Wi-Fi, la demande croissante de capacité de réseau en termes de bande passante et le nombre croissant d'utilisateurs provoquent une densification des réseaux déployés. Les réseaux Wi-Fi souffrent de la détérioration de la QoE dans les déploiements denses. Cette thèse examine différentes façons d'améliorer les performances des réseaux Wi-Fi. Elle se concentre, d'une part, sur le déploiement des réseaux Wi-Fi en tenant compte de la capacité de la technologie et des besoins de l'application, et d'autre part, sur les approches basées sur l'apprentissage par renforcement pour l'adaptation de la quantité de données transmises et l'amélioration du protocole CSMA/CA.

Dans le cadre de cette thèse, nous analysons d'abord les algorithmes d'adaptation de débit existants et leur impact sur les performances d'un réseau Wi-Fi. Ensuite, nous étudions les meilleures pratiques pour le déploiement de ces réseaux au sein d'une exploitation agricole, notamment le nombre de points d'accès requis et leur emplacement pour maximiser la couverture et les performances dans le cadre d'une application de surveillance et de contrôle à distance de robots agricoles mobiles. La technique de déploiement tire parti des fonctionnalités du Wi-Fi tel que le beamforming. Dans une seconde partie, nous exploitons des mécanismes d'apprentissage par renforcement pour adapter dynamiquement les profils d'application en fonction des conditions du réseau et des exigences des utilisateurs. Nous exploitons également les mécanismes de RL afin d'améliorer le protocole CSMA/CA en adaptant la durée du backoff selon l'état du réseau.

A travers les résultats de simulation, nous démontrons que l'utilisation du MIMO dans la stratégie de déploiement de réseaux Wi-Fi permet de réduire le nombre de points d'accès requis et d'obtenir une meilleure couverture. Une stratégie de déploiement tenant compte de la capacité du réseau aboutit en un compromis entre le nombre d'antennes, la couverture, les points d'accès requis et les performances.

Nos modèles d'apprentissage par renforcement apportent une nette amélioration des performances des réseaux Wi-Fi. Ces modèles ont permis une adaptation dynamique des profils d'application en fonction des conditions du réseau et des exigences des utilisateurs, entraînant un meilleur débit du réseau et moins de pertes de données. De plus, le modèle utilisé au niveau de l'algorithme CSMA/CA a entraîné moins de retards d'accès, un débit global amélioré, moins de collisions et une meilleure équité entre les utilisateurs.

A travers ce travail de recherche, nous démontrons, à travers une démarche d'évaluation par simulation numérique, l'efficacité de l'apprentissage par renforcement pour améliorer les performances de différents aspects d'un réseau Wi-Fi. Les résultats montrent le potentiel d'améliorations significatives des performances et de l'expérience utilisateur des réseaux Wi-Fi grâce à l'application de ces méthodes

d'optimisation. Les travaux futurs devraient continuer à explorer et à affiner ces techniques pour s'assurer que les réseaux Wi-Fi puissent répondre à la demande toujours croissante d'une connectivité sans fil fiable et de haute qualité.

Acronyms

A-MPDU Aggregated MAC Protocol Data Unit.

ACK Acknowledgements.

AI Artificial Intelligence.

AP Access Points.

API Application Programming Interface.

BER Bit Error Rate.

BPSK Binary Phase Shift Keying.

BSS Basic Service Set.

C-ITS Cooperative Intelligent Transport Systems.

CC Convolutional Coding.

CSI Channel State Information.

CSMA/CA Carrier Sense Multiple Access with Collision Avoidance.

CTS Clear to Send.

CW Contention Window.

DIFS Distributed Interframe Space.

DL Deep Learning.

DQL Deep Q-Learning.

DRL Deep Reinforcement Learning.

DSS/CCK Direct Sequence Spread Spectrum/Complementary Code Keying.

EIRP Effective Isotropic Radiated Power.

ESS Environmental Signal Strength.

EWMA Exponential Weighted Moving Average.

FLR Frame Loss Rate.

ICSMA/CA Intelligent Carrier Sense Multiple Access with Collision Avoidance.

IEEE Institute of Electrical and Electronics Engineers.

IOT Internet of Things.

ISM Industrial, Scientific, and Medical.

LDPC Low-Density Parity Check.

MAC Medium Access Layer.

Mbps Megabits per second.

MCS Modulation and Coding Scheme.

MDP Markov Decision Process.

MIMO Multiple Input Multiple Output.

MINLP mixed-integer nonlinear programming.

MiRA MIMO Rate Adaptation.

ML Machine Learning.

MPDU MAC Protocol Data Unit.

MSDU MAC Service Data Unit.

MU-MIMO Multi-User MIMO.

MutFed Mutual Feedback.

NACK Non-Acknowledgements.

NICs Network Interface Cards.

NS-3 Network Simulator 3.

OBSS Overlapping Basic Service Set.

OFDM Orthogonal frequency-division multiplexing.

OFDMA Orthogonal Frequency Division Multiple Access.

OFRA On-Demand Feedback Rate Adaptation.

OMNeT++ Objective Modular Network Testbed.

PD Preamble-Detection.

PER Packet Error Rate.

PLR Packet Loss Ratio.

PPO Proximal Policy Optimzation.

QAM Quadrature Amplitude Modulation.

QoE Quality of Experience.

QoS Quality of Service.

QPSK Quadrature Phase Shift Keying.

RAA Rate Adaptation Algorithms.

RAM Random Access Memory.

RAMAS Rate Adaptation for Multi-Antenna System.

RF Radio Frequency.

RL Reinforcement Learning.

RSSI Received Signal Strength Indicator.

RTS Request to Send.

RUs Resource Units.

SIFS Short Interframe Space.

SINR Signal to Interference and Noise Ratio.

SIRA SNR-aware Intra-frame Rate Adaptation.

SISO Single Input Single Output.

SNR Signal to Noise Ratio.

STRALE STandard-compliant and mobility aware PHY RAte and A-MPDU LEngth adaptation.

TRPO Trust Region Policy Optimization.

TWT Target Wake Time.

Wi-Fi Wireless Fidelity.

WLANs Wireless Local Area Networks.

WSN Wireless Sensor Network.

List of Figures

2.1	A summary of the main features of various Wi-Fi standards.	9
2.2	An example illustrating how CSMA/CA operates.	11
2.3	An example illustrating how beamforming operates.	12
2.4	A channel with input x and noise n	21
2.5	A MIMO channel decomposed into s parallel independent subchannels.	22
2.6	Comparison between the capacity of SISO and MIMO systems.	23
2.7	An illustration of the basic RL functionality.	27
3.1	Throughput of a station moving away from an access point at a speed of 6 m/s.	35
3.2	Overall throughput received at the access point.	36
3.3	MAC Protocol Data Unit (MPDU) Frame Loss Ratio.	36
3.4	A-MPDU Frame Loss Ratio.	37
3.5	Mean value of the modulation and coding schemes (MCS) used by the nodes.	37
3.6	Average throughput of sink nodes in an Ad hoc network.	39
3.7	MPDU Frame Loss Ratio.	39
3.8	A-MPDU Frame Loss Ratio.	40
3.9	Mean value of the MCS selected by sender nodes.	40
3.10	Grid Deployment.	43
3.11	Deployment of the access points at the edge of the field.	43
3.12	A graphical tool used to calculate the number of required access points and visualize the deployment	45
3.13	Grid deployment: number of access points inside the field for a medium and high quality video intervention.	46
3.14	Edge deployment: number of access points inside the field for a medium and high quality video intervention.	46
3.15	Scenario 1 (data rate independent deployment): Throughput and number of access points needed for a medium quality stream.	50
3.16	Scenario 1 (data rate independent deployment): End-to-End delay for a medium quality stream.	50
3.17	Scenario 2 (data rate dependent deployment): Throughput and number of access points needed for a medium quality stream.	51
3.18	Scenario 2 (data rate dependent deployment): End-to-End delay for a medium quality stream.	51
3.19	Scenario 3 (edge deployment): Throughput and number of access points for medium quality stream.	52
3.20	Scenario 3 (edge deployment): End-to-End delay for a medium quality stream.	52
3.21	Scenario 1 (data rate independent deployment): Throughput and number of access points needed when sending a high-quality stream.	53

3.22	Scenario 1 (data rate independent deployment): End-to-End delay for a high quality stream.	54
3.23	Scenario 2 (data rate dependent deployment): Throughput and number of access points when sending a high-quality stream.	55
3.24	Scenario 2 (data rate dependent deployment): End-to-End delay for a high quality stream.	55
3.25	Scenario 3 (Edge deployment): Throughput of the system in the special deployment with high quality streaming.	56
3.26	Scenario 3 (edge deployment): End-to-End delay for a high quality stream.	56
3.27	Impact of the number of access points on the interference versus the number of robots in the field for a 2x2 system.	58
3.28	Throughput obtained through estimation for a field of area $2km^2$. . .	59
3.29	Throughput obtained through simulation for a field of area $2km^2$. . .	59
3.30	Throughput obtained through estimation for a field of area $5km^2$. . .	60
3.31	Throughput obtained through simulation for a field of area $5km^2$. . .	60
4.1	Communication between NS-3 and the DRL model	65
4.2	The structure of the proposed approach using the PPO algorithm . .	68
4.3	Actor and critic losses during the training phase of the DRL model .	69
4.4	Return (cumulative reward)	70
4.5	The goodput of nodes transmitting using a single application profile.	72
4.6	Overall goodput obtained in the network using the trained DRL model, E2E-MAC, and single application profile.	73
4.7	Ratio of collisions obtained in the network using the trained DRL model, E2E-MAC, and single application profile.	73
4.8	A detailed representation of ICSMA/CA training phase	75
4.9	Actor and critic losses during the training phase of the ICSMA/CA .	76
4.10	Normalized throughput in the network in terms of the offered load in Mb/s.	78
4.11	Collisions in the network in terms of the offered load in Mb/s. . . .	78
4.12	Access Delay (ms) in the network in terms of the offered load in Mb/s.	79
4.13	Fairness in the network in terms of the offered load in Mb/s	80

List of Tables

2.1	Summary of existing Rate Adaptation Algorithms (RAA).	19
2.2	802.11ac MCS lookup table [1].	20
3.1	Modulation and coding schemes.	34
3.2	Summary of simulation parameters.	35
3.3	Video parameters.	44
3.4	Simulation parameters table for the capacity-aware deployment. . . .	49
4.1	Simulation parameters table for the application profile optimization. .	71
4.2	Simulation parameters table for the CSMA/CA optimization.	77

Contents

Acknowledgments	i
Abstract	ii
Résumé	iii
List of Acronyms	vii
List of Figures	ix
List of Tables	x
1 Introduction	1
1.1 Context: I-Site CAP 20-25 Project	1
1.2 Wireless Networks	2
1.2.1 Wireless Technologies	2
1.2.2 Wi-Fi Networks	3
1.2.3 Performance of Wi-Fi Networks	4
1.3 Objectives and Research Methodology	4
1.4 Thesis Contributions	5
1.5 Manuscript Structure	6
2 State of the Art	7
2.1 Wi-Fi Standards and Features	7
2.1.1 Evolution of Wi-Fi Standards	7
2.1.2 802.11ac and Beyond	8
2.1.3 Modulation and Coding Schemes	9
2.1.4 Carrier Sense Multiple Access with Collision Avoidance	10
2.1.5 Beamforming	11
2.2 Wi-Fi Problems and Limitations	13
2.2.1 Congestion in Modern Wi-Fi Applications	13
2.2.2 Manufacturer-Specific Features of Wi-Fi	14
2.3 Rate Adaptation Algorithms	15
2.3.1 Explicit Feedback Algorithms	15
2.3.2 Implicit Feedback Algorithms	16
2.3.3 Hybrid Adaptation	17
2.4 Wi-Fi Channel Modeling	18
2.4.1 Propagation Models of Wi-Fi Channels	18
2.4.2 Wi-Fi Coverage	20
2.4.3 Wi-Fi Channel Capacity	21
2.5 Wi-Fi Networks Deployment	23
2.5.1 Indoor Environments	23

2.5.2	Outdoor Environments	24
2.6	Reinforcement Learning in Wireless Networks	25
2.6.1	The Core Concept Behind Reinforcement Learning	26
2.6.2	Deep Reinforcement Learning	29
2.6.3	Leveraging Reinforcement Learning to Optimize Wi-Fi Networks	30
2.6.4	Using Network Simulators with Reinforcement Learning for Wi-Fi Optimization	31
3	Contribution 1 - Wi-Fi Networks Deployment	32
3.1	Introduction	32
3.2	Selecting the Most Effective Rate Adaptation Algorithm	32
3.2.1	Simulation Setup	32
3.2.2	Scenario 1 - Interference Free Network	34
3.2.3	Scenario 2 - Infrastructure Network	35
3.2.4	Scenario 3 - Ad hoc Network	38
3.2.5	Lessons Learned	41
3.3	Wi-Fi Deployment with Beamforming	42
3.3.1	Deployment Technique	42
3.3.2	Deployment Scenarios	44
3.3.3	Capacity Aware Deployment	45
3.4	Performance Evaluation of Deployment Strategies	47
3.4.1	Network Performance Evaluation	47
3.4.2	Empirical Performance Evaluation	57
3.4.3	Deployment Feasibility Study	58
3.5	Conclusion	61
4	Contribution 2 - Reinforcement Learning in Wi-Fi Networks	62
4.1	Introduction	62
4.2	Algorithm and Implementation	62
4.2.1	Deep Reinforcement Learning Algorithm	62
4.2.2	Implementation of the Training and Evaluation Scenarios	64
4.3	Application Profile Selection Using Deep Reinforcement Learning	64
4.3.1	Problem Formulation	65
4.3.2	Model Design	66
4.3.3	Model Training	67
4.3.4	Model Evaluation	70
4.4	Intelligent CSMA/CA using Deep Reinforcement Learning	72
4.4.1	Problem Formulation	74
4.4.2	Model Design	74
4.4.3	Model Training	75
4.4.4	Model Evaluation	76
4.5	Complexity	79
4.6	Conclusion	80
5	Conclusions and Future Directions	81
5.1	Conclusions	81
5.1.1	Evaluating and Selecting the Best Rate Adaptation Algorithm	81
5.1.2	A Study on Capacity-Aware Deployment	81
5.1.3	Wi-Fi Performance Optimization using Reinforcement Learning	82
5.2	Future Directions	82
5.2.1	Potential Contributions to Rate Adaptation Algorithms	83

5.2.2	Possible Enhancements to the Capacity-Aware Deployment . . .	84
5.2.3	Enhancement of Reinforcement Learning Approaches for Wi-Fi Optimization	85
	Publications	87
	Bibliography	96

Chapter 1

Introduction

Wireless communication technologies have become an increasingly popular method for establishing connectivity between connected nodes without requiring extensive cable deployment. Such technologies are critical for many applications, especially ones that require mobility, including vehicular networks for Cooperative Intelligent Transport Systems (C-ITS) [2, 3], mobile-connected engines for precision farming [4, 5], and a wide range of applications used on mobile phones and that rely on cellular networks. The development of wireless technologies continues to evolve to meet the demands of modern applications and maintain acceptable Quality of Service (QoS) and Quality of Experience (QoE) for end users.

In this chapter, we begin in section 1.1 by introducing the context of this thesis which is an I-Site CAP 20-25 project. We then provide an overview of wireless networks, specifically focusing on Wi-Fi networks and their applications, in section 1.2. The main objectives of the thesis and our research methodology are outlined in section 1.3, followed by a discussion of our contributions in section 1.4. Finally, we summarize the structure of this manuscript in section 1.5.

1.1 Context: I-Site CAP 20-25 Project

The work done in this thesis is part of a research activity related to smart farming funded by the I-Site CAP 20-25 Project. One of the building blocks of smart farming is autonomous robots. These robots are equipped with onboard intelligence, allowing them to execute tasks on fields autonomously, the fields are of areas that range from $1km^2$ to $5km^2$. There are currently 5 functional robots, the number will increase to 20 robots in the short run, and it is expected to reach 100 robots in the long run. During the task execution, robots send through a wireless communication system periodical data such as localization, state of the robot, images, and sometimes video streams. Indeed, in some cases, robots need human intervention to assess specific situations where their onboard intelligence is not able to make decisions (unexpected obstacles, faulty behavior, etc.). Human operators are expected to take control of these robots remotely and guide them through the situation. This task requires a reliable video stream from the robot to the control stations. In other words, The communication flow during the remote takeover needs to benefit from a certain level of QoS that offers a satisfactory QoE. The latter is the main focus of this thesis.

Many wireless technologies currently exist. Each technology has its own technical characteristics in terms of coverage, cost, energy consumption, throughput, delay, etc.

1.2 Wireless Networks

Wireless technologies have experienced significant advancements in the past decades, enabling the creation of highly efficient communication networks. Wi-Fi networks have emerged as a ubiquitous technology that offers seamless connectivity to the internet and facilitates communication across a wide range of devices. This section provides an overview of wireless technologies and Wi-Fi networks. Additionally, we explore various applications of Wi-Fi networks and discuss their performance.

1.2.1 Wireless Technologies

The field of wireless networks encloses various technologies. In what follows, we list some of the popular wireless technologies that offer data rates that meet the needs of smart farming mobile robots applications:

- Cellular networks [6] which represent a crucial component of modern telecommunication systems, providing wide connectivity and enabling seamless data transmission across large areas. Advances in cellular technology, such as the transition from 4G to 5G, have significantly enhanced network capacity, data rates, and latency, paving the way for a wide array of emerging applications. In general, 4G LTE networks can provide speeds of up to 100 Mbps, while 5G networks can provide speeds of up to 10 Gbps [7].
- Wi-Fi [8] is a popular way to provide wireless communication, offering high-speed data transfer for various devices. Recent advancements in Wi-Fi standards, such as the introduction of Wi-Fi 6 and Wi-Fi 7 with data rates up to 30 Gbps, have led to improved network efficiency, increased throughput, and reduced latency, thereby supporting the growing demands of data-intensive applications.
- Bluetooth [9] is a widely used wireless technology that enables short-range communication between devices, such as connecting headphones, speakers, or wearables to smartphones and computers. Its low power consumption and ease of use make it ideal for a variety of applications. Bluetooth technology has seen recent improvements, such as the introduction of Bluetooth 5, which can achieve a maximum data rate of up to 2 Mbps. [10].
- WiMAX [11] is a wireless technology that provides high-speed connectivity over a wide area. It can provide data rates of up to 70 Mbps.
- Zigbee [12] is a wireless technology extensively used in home automation and industrial applications. It operates at low power levels and enables efficient transmission of small data volumes over short distances, achieving data rates of up to 250 Kb/s.
- LP WAN (Low-Power Wide Area Network) technologies [13], such as LoRaWAN, have been specifically designed to enable long-range communication while minimizing power consumption. LoRaWAN offers data rates up to 27 Kb/s. The actual data rates depend on the specific network configuration and environmental factors.
- Satellite communication [14] enables the transmission of data over vast distances. The achievable data rates in satellite communication can vary based

on factors such as the type of satellite system deployed and the frequency band used reaching up to multiple Gb/s. However, Satellite communications suffers from delay due to the long distances travelled.

Ongoing advancements in wireless communications have led to the development of innovative applications, such as the Internet of Things (IoT) [15], which includes smart homes [16], smart farming [17], and C-ITS [18].

In this work, we decided to focus on Wi-Fi technology because it offers a good compromise between performance and availability. It is an autonomous system that can be deployed anywhere without the need for a telecommunication operator coverage. Furthermore, it offers high data rates that can support a wide range of modern applications.

1.2.2 Wi-Fi Networks

Wi-Fi is a widely adopted wireless communication technology integral to a variety of devices. Wi-Fi networks are critical in wireless communication systems by offering internet connectivity and data transmission capabilities to various devices, including smartphones, laptops, tablets, and smart appliances. Wi-Fi primarily operates on multiple frequency bands, such as 2.4 GHz and 5 GHz, presenting a range of channels to facilitate multiple devices in varying settings. Typically, Wi-Fi networks adhere to the IEEE 802.11 standards, which have undergone numerous modifications to improve network efficiency and speed.

Initially introduced in 1997, the first Wi-Fi standard, IEEE 802.11, provided data rates reaching 2 Mbps. Since then, several subsequent versions have emerged, each enhancing data rates, coverage, and network effectiveness. Some well-known Wi-Fi standards include 802.11n, 802.11ac, and 802.11ax. Each standard incorporates new technologies and methodologies to optimize Wi-Fi network performance, such as Multiple Input Multiple Output (MIMO), beamforming, and Orthogonal Frequency Division Multiple Access (OFDMA).

Two primary types of Wi-Fi networks exist infrastructure mode and ad-hoc mode. Infrastructure mode, the more popular type, involves devices connecting to an Access Points (AP) responsible for administering the wireless network and enabling connectivity to the internet. In contrast, ad-hoc mode permits devices to connect directly, forming a decentralized wireless network without requiring an AP.

The extensive adoption of Wi-Fi networks has generated an increasing demand for uninterrupted, high-speed, and reliable connectivity. This has led to the creation of various network management and optimization strategies, such as load balancing, adaptive channel allocation, and QoS prioritization. These approaches aim to ensure that Wi-Fi networks can accommodate the growing number of devices and data-intensive applications while maintaining optimal performance.

Wi-Fi networks have been adopted in many modern applications, which led to many ongoing research work and development efforts aiming at enhancing network performance. Emerging technologies that can influence the future of Wi-Fi networks include Wi-Fi 6E, which extends the Wi-Fi 6 standard to the 6 GHz frequency band, offering additional channels and increased capacity for Wi-Fi networks. Moreover, developing next-generation Wi-Fi standards, such as IEEE 802.11be (Wi-Fi 7), seeks to improve data rates, latency, and network efficiency, paving the way for even more sophisticated applications and use cases.

1.2.3 Performance of Wi-Fi Networks

Wi-Fi networks have become vital in many applications across various sectors. Smart farming, in particular, has witnessed significant advancements due to Wi-Fi networks, which enable the deployment of cutting-edge technologies for precision agriculture [4]. These technologies encompass remote irrigation management, livestock monitoring, and real-time soil analysis, leading to more efficient resource use, higher crop yields, and a reduced environmental impact.

In smart cities, Wi-Fi networks facilitate the integration of intelligent systems that optimize traffic flow, enhance public safety, and improve energy management [19]. Wi-Fi networks connect devices throughout the urban environment and provide city planners and administrators with valuable insights to streamline city services and make more informed decisions.

In the industrial sector, Wi-Fi networks play a crucial role in implementing Industry 4.0 solutions, which involve using interconnected devices, sensors, and robots for real-time data analysis, predictive maintenance, and automated decision-making, thereby increasing efficiency and reducing operational costs [20].

Transport systems also benefit from Wi-Fi networks, which provide passengers with reliable internet connectivity during their journeys, whether on buses, trains, or airplanes. This enhanced connectivity not only improves the travel experience but also allows for real-time tracking and communication between vehicles and central control centers, contributing to greater operational efficiency and safety.

In large venues like stadiums [21], Wi-Fi networks offer attendees high-speed internet access, enabling them to share live updates, access event information, and enjoy immersive experiences that enhance their enjoyment of the event.

Wi-Fi technology has been employed in a multitude of domains and applications across various domains. However, despite their numerous advantages, Wi-Fi networks often suffer from performance challenges that can impact the QoS provided to users. Factors such as network congestion, interference, and the increasing number of connected devices can lead to reduced data rates, increased delay, and even connectivity issues. In order to maintain high-performance Wi-Fi networks, optimization strategies are necessary to tackle these challenges effectively. Optimizing Wi-Fi networks involves various approaches on different network layers, including data rate adaptation, deployment optimization, QoS prioritization, and congestion control to ensure reliable connections even in dense network environments.

Many current research works on Wi-Fi networks have been carried out to incorporate Machine Learning (ML) principles into their network stacks to provide reliable communications. ML allows devices to learn to solve problems without being explicitly programmed. These methods include supervised, unsupervised, and reinforcement learning. RL has become one of the most important machine learning research areas as it enables models to achieve specific goals while maximizing their rewards over time. RL agents can explore, interpret, act, and learn from their environment through trial and error to optimize their cumulative reward. Wi-Fi networks have integrated these techniques into their operations to enhance their performance.

1.3 Objectives and Research Methodology

This thesis aims to enhance certain aspects of Wi-Fi networks for better QoS in the context of smart farming with dense scenarios as part of the I-Site CAP 20-25

project. Our main research objectives are listed below:

- Enhance Wi-Fi deployments through various deployment strategies by leveraging Wi-Fi features such as beamforming. We aim to improve network efficiency while minimizing deployment complexity and cost.
- Enhance Wi-Fi communication protocols, especially in terms of throughput, in order to offer the required QoS that meets the QoE of the users.

We selected a simulation-based evaluation approach to evaluate our contributions. Simulation offers ease of control of the simulation environment parameters and yields valuable results. Due to the unavailability and complexity of experimental equipment, conducting experiments can take time and effort. By simulating various scenarios, we were able to assess the effectiveness of our proposed solutions in a controlled environment, providing us with comprehensive data and insights into the performance of our proposals. This approach allowed us to refine and improve all our contributions and to train our RL models, leading to a more accurate evaluation of our proposed solutions.

1.4 Thesis Contributions

The contributions of this thesis investigate techniques to optimize Wi-Fi network performance. In what follows, we list our main contributions, note that the first and second contributions are combined into a single chapter in the manuscript due to their strong correlation:

1. In the first step of the thesis, we studied rate adaptation algorithms that adjust the data rate on the lower layers of Wi-Fi. Various algorithms have been proposed over the years, each utilizing unique strategies to determine the optimal data rate. We categorized the algorithms according to the way they behave. Then, we picked and evaluated four algorithms from the literature to determine their performance in terms of throughput and packet loss under different network conditions. We listed their strengths and weaknesses and the lessons learned from the study. Finally, we select the most effective algorithm based on our results. This contribution has been published in a journal paper [22].
2. In the second step of the thesis, we conducted a comparison study of the performance of Wi-Fi networks using MIMO and SISO systems. We also investigated the main objective of the study, which is the impact of beamforming on coverage and capacity. To optimize network performance for users, we explored deployment concerns such as equipment selection and access point placement strategies. We evaluated various deployment configurations through simulation and an empirical model that we developed. Our findings have been submitted to a journal for publication, and we are currently awaiting the outcome of the review process.
3. In the last step of the thesis, we employed deep reinforcement learning to optimize Wi-Fi network performance in two contexts. First, we designed a model to learn the most suitable application profiles for optimal network performance in dense Wi-Fi networks, where network saturation can lead to significant performance degradation. We evaluated the model through simulation compared

to the baseline rate and another work from the literature. This work has been published in a conference paper [23]. Second, we used deep reinforcement learning to optimize the random backoff decision of the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol, which manages access to the wireless medium in Wi-Fi networks. The model was trained and evaluated in the simulator. This work has been submitted to a journal, and we are currently awaiting the outcome of the review process.

1.5 Manuscript Structure

The manuscript is structured into five main chapters. **Chapter 1** provides an overview of wireless networks, with a focus on Wi-Fi networks, and summarizes the thesis methodology and contributions. In **chapter 2**, a literature review is presented, which covers the Wi-Fi standards, features, and limitations, as well as literature directly related to the thesis contributions. **Chapter 3** is devoted to our first contribution, which explores Wi-Fi network deployment, including the evaluation and selection of the most effective rate adaptation algorithm and a study on capacity-aware deployment. In **chapter 4**, we delve into our second contribution, which concentrates on employing reinforcement learning in Wi-Fi networks, comprising application profile selection using deep reinforcement learning and intelligent Carrier Sense Multiple Access with Collision Avoidance (ICSMA/CA) using deep reinforcement learning. **Chapter 5** concludes the manuscript and suggests possible future research directions.

Chapter 2

State of the Art

In this chapter, we explore various aspects of Wi-Fi networks and their features that are relevant to our thesis. We commence with section 2.1, outlining Wi-Fi standards along with their features such as Modulation and Coding Scheme (MCS), MIMO, CSMA/CA, and others that significantly influence the performance and reliability of Wi-Fi networks. Section 2.4 delves into Wi-Fi channel modeling in outdoor environments. We discuss the challenges and limitations of Wi-Fi networks in section 2.2. In section 2.3, we examine rate adaptation algorithms from existing literature and classify them based on their behavior. Section 2.5 evaluates indoor and outdoor deployment strategies found in the literature. Finally, we provide a comprehensive explanation of reinforcement learning types and their functioning, as well as their applications in Wi-Fi networks.

2.1 Wi-Fi Standards and Features

Wireless Fidelity (Wi-Fi) has undergone rapid changes since its initial standard was released in 1997. Over time, the evolution of Wi-Fi standards has been driven by the need to support increasing data rates and devices. To achieve these goals, new modulation techniques, frequency bands, and technologies have been added to each new Wi-Fi standard. Moreover, the standards had to adapt to changes in the wireless environment, such as interference from other wireless devices and the demand for supporting various advanced applications and use cases. In the following section, we will explore the major Wi-Fi standards.

2.1.1 Evolution of Wi-Fi Standards

Wi-Fi standards form the basis of Wi-Fi technology and define the specifications and requirements for Wireless Local Area Networks (WLANs). These standards are maintained by the Institute of Electrical and Electronics Engineers (IEEE) and are commonly referred to as IEEE 802.11 standards [24].

The first Wi-Fi standard, 802.11, was introduced in 1997 and provides data rates of up to 2 Mbps. Although initially designed for business and scientific use, as the demand for wireless connections grew, the Wi-Fi Alliance was created in 1999 to develop commercial Wi-Fi standards.

In 1999, Wi-Fi 1, also known as the 802.11b standard [25], was introduced by the Wi-Fi Alliance [26]. Wi-Fi 1 uses Direct Sequence Spread Spectrum/Complementary Code Keying (DSS/CCK) modulation schemes for data transmissions. It provides support for different speeds up to 11 Mbps. This standard quickly became popular

and the default choice for wireless networking, utilizing the 2.4 GHz frequency band with good coverage in indoor environments.

Wi-Fi 2, also known as the 802.11a standard, was introduced as another extension of the original 802.11 standard in 1999 [27]. It uses the 5 GHz frequency band with Orthogonal frequency-division multiplexing (OFDM) transmissions and offers data rates of up to 54 Mbps. Although not backward compatible with 802.11b, it provided better performance in environments with significant interference.

Wi-Fi 3, also known as the 802.11g standard, was released in 2003 [28]. It provides backward compatibility with 802.11b and faster data rates of up to 54 Mbps. Although still utilizing the 2.4 GHz frequency band, it uses a more efficient multi-carrier modulation scheme for improved performance.

Wi-Fi 4, also known as the 802.11n standard, was introduced in 2006 [29]. Wi-Fi 4 was a significant improvement over previous standards due to the employment of MIMO, 40 MHz channels in the physical layer, and frame aggregation in the Medium Access Layer (MAC) layer. Channel width of 40 MHz provides double the data rate of a single 20 MHz channel of Wi-Fi 3. Wi-Fi 4 allows for up to four spatial streams, with a maximum theoretical throughput of 600 Mbps. The modulation and coding schemes available in this standard include Binary Phase Shift Keying (BPSK), Quadrature Phase Shift Keying (QPSK), 16Quadrature Amplitude Modulation (QAM), and 64QAM. Moreover, Frame aggregation enables the combination of multiple MAC Service Data Unit (MSDU)s or MAC Protocol Data Unit (MPDU)s to reduce overheads and increase the application level data rate across multiple frames. Additionally, Wi-Fi 4 is compatible with previous standards.

In 2013 and 2019, two significant advancements in wireless networking were introduced: 802.11ac [30] and 802.11ax [31]. These standards represented a significant improvement over previous standards and brought with them several features and enhancements, which we will explore in the next subsection.

2.1.2 802.11ac and Beyond

In 2013, Wi-Fi 5, also known as 802.11ac, was a major improvement over the previous Wi-Fi standards. Wi-Fi 5 operates exclusively in the 5 GHz band and offers data transfer speeds of up to 3.5 Gbps, which is more than double the speed of Wi-Fi 4. Wi-Fi 5 includes wider bandwidth of up to 160 MHz, more MIMO spatial streams of up to 8, downlink Multi-User MIMO (MU-MIMO) up to 4 clients leading to reduced network congestion and better performance in dense areas, and high-density modulation of up to 256QAM at 3/4 and 5/6 coding rates. Wi-Fi 5 uses Beamforming, which allows the antenna to transmit radio signals in a specific direction to a particular device, strengthening the signals in that region and resulting in a more robust and stable connection, especially at longer distances.

WiFi 6, also known as 802.11ax, was introduced in 2019. It operates on both the 2.4 GHz and 5 GHz frequency bands. Wi-Fi 6 uses a denser modulation scheme, namely the 1024QAM, to provide data rates of up to 9.6 Gbps, which is nearly three times faster than WiFi 5. Additionally, WiFi 6 uses the OFDMA technology, which divides the wireless channel into smaller sub-channels, leading to more efficient use of the wireless spectrum. Moreover, WiFi 6 enhances downlink MU-MIMO technology, allowing up to 8 simultaneous user connections per access point, which is useful in dense areas. Wi-Fi 6E expands the existing Wi-Fi 6 standard and allows access to a new 6 GHz band. Finally, Wi-Fi 6E (wave 2) introduces uplink MU-MIMO.

WiFi 7, also known as 802.11be, was introduced in 2023 [32]. The new standard will operate on the 6 GHz frequency band. This standard offers data rates of up to

30 Gbps and support channels that are up to 320 MHz wide. Additionally, Wi-Fi 7 introduces improved MU-MIMO technology, improved multi-access point coordination, 4096QAM, 16 spatial streams, and other features to boost performance, efficiency, and security.

Figure 2.1 summarizes the main features of the most recent Wi-Fi standards.

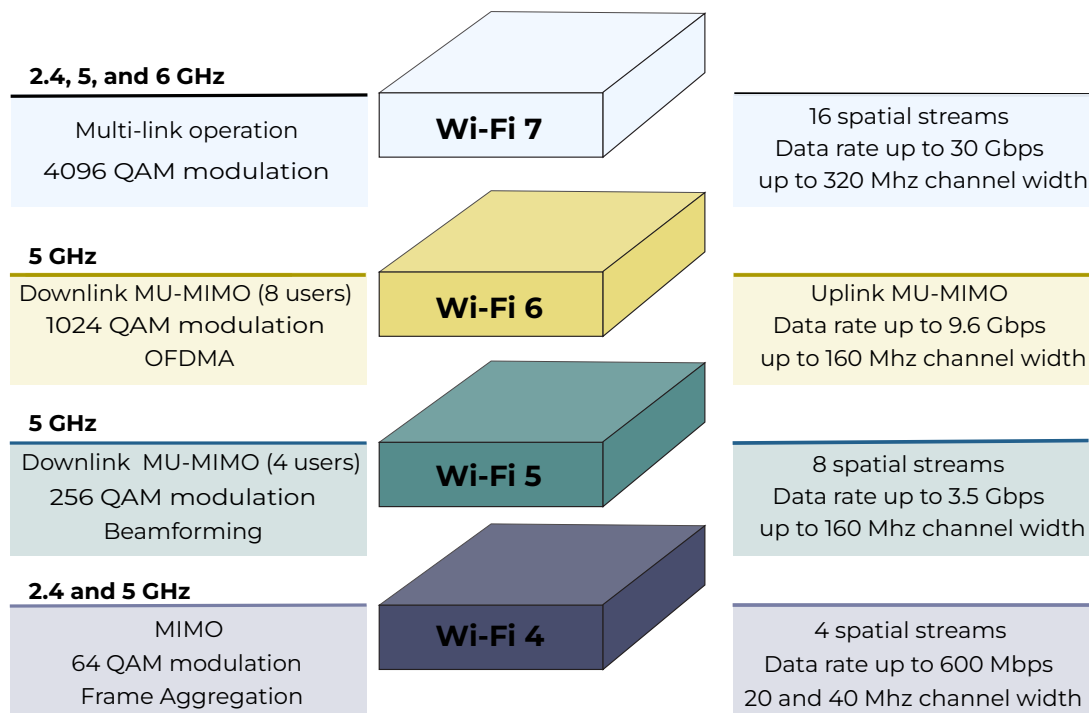


Figure 2.1: A summary of the main features of various Wi-Fi standards.

In the following subsections, we will delve into several fundamental features of Wi-Fi that are frequently improved upon in Wi-Fi standards and have a significant impact on the performance of Wi-Fi networks. Namely, the MCS, CSMA/CA, and beamforming.

2.1.3 Modulation and Coding Schemes

Modulation and coding schemes are fundamental components of Wi-Fi that enable wireless signals to be transmitted and received through the air. Modulation involves the conversion of digital data into an analog signal that can be transmitted over the air, while coding involves the addition of error correction codes to digital data to ensure accurate reception and decoding by the receiver. The selection of modulation and coding schemes used in Wi-Fi is crucial in determining the data rates, range, and reliability of wireless connections.

The MCS used in Wi-Fi technology is defined by IEEE 802.11 standards. The earliest Wi-Fi standards, such as 802.11g, used relatively simple modulation schemes such as BPSK and QPSK, which enabled data rates of up to 54 Mbps. However, their low data rates and susceptibility to interference restricted their practical applications. In contrast, more recent Wi-Fi standards, such as 802.11n, 802.11ac, and 802.11ax, use more sophisticated modulation schemes like 64-QAM and 256-QAM, enabling data rates of up to 1 Gbps, 9.6 Gbps, and 10 Gbps, respectively. These schemes offer higher data rates, but they are more complex than earlier schemes.

Aside from modulation schemes, Wi-Fi also employs various coding schemes to ensure the accuracy and reliability of wireless transmissions. One of the commonly used coding schemes in Wi-Fi is Convolutional Coding (CC), which adds redundancy to transmitted data, allowing errors in the received signal to be corrected. More recent Wi-Fi standards use advanced coding schemes like Low-Density Parity Check (LDPC) coding[33] and Turbo coding[34]. LDPC coding is a highly efficient scheme that can achieve reasonable error correction performance, while Turbo coding can achieve even higher data rates than LDPC coding.

The combination of MCS used in Wi-Fi determines the achievable data rates, range, and resistance to interference. More complex MCS generally offer higher data rates but require stronger and more stable wireless signals. As a result, Wi-Fi devices using advanced MCS may have shorter ranges and be more sensitive to interference than those using simpler schemes.

The choice of MCS used in Wi-Fi has a significant impact on the performance and reliability of wireless connectivity. As the Wi-Fi technology evolves, the development of new MCS plays a vital role in enhancing the performance of wireless networks.

In the following subsection, we will delve into the CSMA/CA method in Wi-Fi responsible for the multiple access to the Wi-Fi medium.

2.1.4 Carrier Sense Multiple Access with Collision Avoidance

In a Wi-Fi network, multiple devices compete for airtime to transmit data. CSMA/CA is a method used in Wi-Fi networks to avoid collisions between multiple devices trying to communicate over the same wireless channel [35].

A device using CSMA/CA listens to the wireless channel before transmitting any data packet. The device listens for a Distributed Interframe Space (DIFS) duration to ensure that the channel is idle. If the channel remains idle, the device generates a random backoff in the range of $[0, CW_{min}]$. The random backoff duration corresponds to the number of slots the device should wait before transmitting its packet. The backoff mechanism is used to reduce the probability of multiple devices sending their packets at the same time, causing collisions. Note that each Wi-Fi standard defines the slot duration. The device decrements its backoff duration in each slot if the channel remains idle and freezes the countdown if the channel is sensed busy. At the end of the backoff duration, the device transmits its packet.

CSMA/CA does not prevent collisions completely. It is possible for multiple devices to complete their backoff duration countdowns simultaneously and transmit their packets, resulting in a collision. To mitigate this issue, the Contention Window (CW) is doubled after each collision, which widens the range of the random backoff duration selected by each device, making another collision less probable. This process continues until the data is successfully transmitted or the maximum number of re-transmissions is reached. Figure 2.2 shows an example functionality of CSMA/CA. While device 1 is transmitting the packet, device 2 senses the channel as busy and waits until the transmission ends. After successfully receiving the packet, the receiver waits for a Short Interframe Space (SIFS) duration and sends back an Acknowledgements (ACK). Then, device 1 senses the channel as idle for DIFS and then generates a random backoff duration. At the same time, device 2 continues its backoff duration countdown. The two devices transmit their packets at the same moment since the backoff durations countdown reaches zero in the same time slot. Thus, a collision occurs and each device picks a new random duration from a double

size CW.

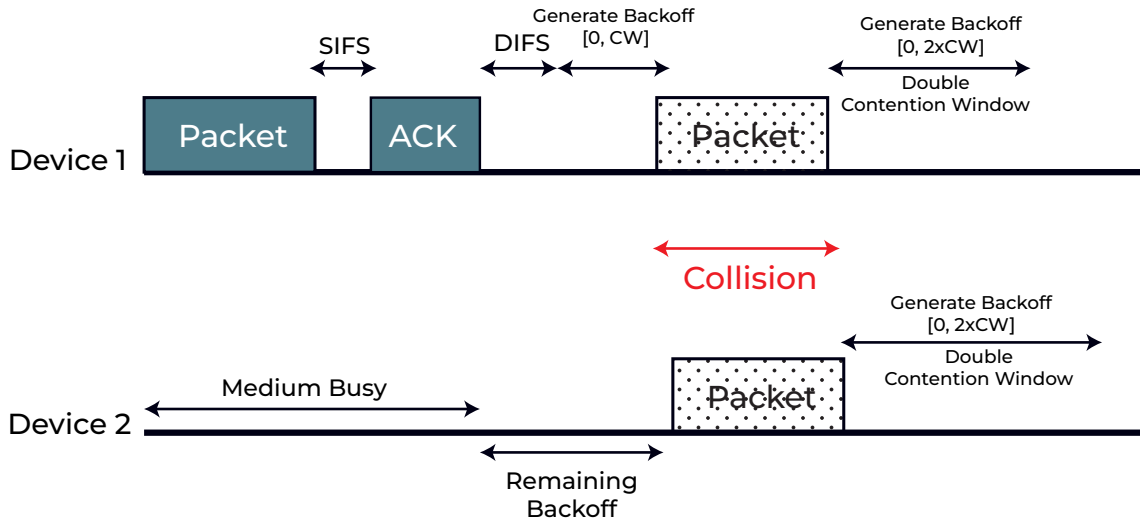


Figure 2.2: An example illustrating how CSMA/CA operates.

One of the main advantages of CSMA/CA is that it allows multiple devices to share the same wireless channel without requiring central coordination. This makes Wi-Fi networks flexible and easy to set up, as devices can join and leave the network at any time without requiring any manual configuration. CSMA/CA is often combined with other techniques such as channel bonding, beamforming, and QoS to further improve the reliability and efficiency of Wi-Fi networks. Channel bonding allows devices to use multiple channels simultaneously to increase the available bandwidth, while beamforming uses multiple antennas to direct the wireless signals toward the intended receivers, reducing interference and improving the signal quality. QoS prioritizes traffic based on the application type, ensuring that high-priority data such as voice and video data are transmitted with minimal delay and packet loss. CSMA/CA can also be optionally supplemented by the exchange of Request to Send (RTS) packet to the receiver. The RTS packet includes the length of the data to be transmitted and the time duration required for the transmission. If the receiver is ready to receive the data, it responds with a Clear to Send (CTS) packet. The CTS packet also shows how long the data is and how long the transmission needs to take. With the RTS/CTS exchange, a device can take over the wireless channel for a certain amount of time and stop other devices from sending data.

In the following subsection, we will explore the concept of beamforming and the benefits it offers to Wi-Fi networks.

2.1.5 Beamforming

Beamforming is a sophisticated signal processing technique in modern Wi-Fi communication systems to improve wireless performance, coverage, and capacity [36]. It involves the adaptive control of antenna radiation patterns, enabling the transmission of radio signals in a targeted direction to maximize the received signal strength at the receiver while minimizing interference with other devices. Beamforming is achieved by employing multiple antennas at the transmitter and receiver. The transmitted signals from each antenna element are weighted and combined, exploiting the constructive and destructive interference of the radio waves to form a focused beam toward the receiver. The weights applied to each antenna element can be adjusted to

steer the main lobe of the radiation pattern toward the desired direction, optimizing Signal to Interference and Noise Ratio (SINR) and overall link quality. Figure 2.3 shows the basic functionality of beamforming where a main lobe is directed from a transmitter toward a receiver. The rest are low-energy undesired sidelobes called backlobes.

Beamforming techniques have been integrated into various Wi-Fi standards. Wi-Fi 5 introduced standardized beamforming techniques called explicit and implicit beamforming. Explicit beamforming requires explicit feedback from the receiver, which provides information about the channel state to the transmitter. The transmitter then adjusts the signal accordingly to optimize the beamforming process. Implicit beamforming, on the other hand, relies on the transmitter's observation of the received signal without any explicit feedback from the receiver.

Beamforming offers several benefits in Wi-Fi, such as increased signal strength and range, improved throughput, reduced interference, and energy efficiency. By focusing the signal toward the intended receiver, beamforming increases the received signal strength, thereby improving the overall range and coverage of the Wi-Fi network. The enhanced SINR resulting from beamforming allows for higher data rates and increased network throughput. By directing the main lobe of the radiation pattern toward the desired user, beamforming helps minimize interference with other devices in the network. Lastly, beamforming allows for more efficient use of the transmitted power, as the energy is concentrated towards the intended receiver, reducing overall energy consumption and prolonging the battery life of wireless devices.

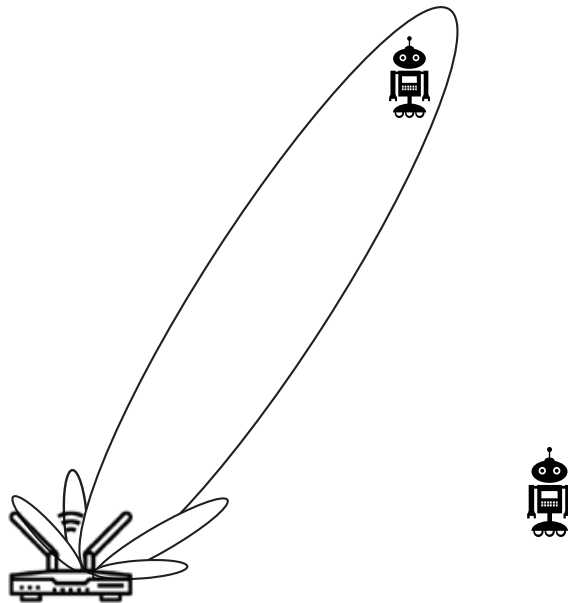


Figure 2.3: An example illustrating how beamforming operates.

According to [37], the gain of a MIMO system that uses beamforming can be expressed as:

$$\begin{aligned} G_t &= G_{t0} + 10 \log_{10} (n_t) \\ G_r &= G_{r0} + 10 \log_{10} (n_r) \end{aligned} \quad (2.1)$$

where n_t and n_r are the number of transmit and receive antennas respectively, G_{t0} and G_{r0} are the gains of a single transmit and receive antenna respectively. Thus using 2 antennas provides a gain of around 3 dB, and using 4 antennas, a gain of around 6 dB.

Wi-Fi standards are subject to regulations regarding transmit power [38]. When MIMO was first introduced, regulators imposed a limit based on the antenna array gain. Regulatory rules are typically a cap on Effective Isotropic Radiated Power (EIRP). EIRP includes the array gain in both the US and Europe [37].

The gain achieved when using beamforming results in a significant improvement in the coverage area of the wireless network. Thus, it becomes essential to take into account the benefits of beamforming during the deployment of Wi-Fi networks.

Over the years, Wi-Fi standards have introduced numerous features and enhancements. Despite these advancements, certain problems and limitations persist in how these features are defined and function. In the following section, we will discuss some of these problems and limitations.

2.2 Wi-Fi Problems and Limitations

The availability of Wi-Fi access points has become a necessity in any public, business, or commercial place. The growing number of Wi-Fi devices and the growing number of places where Wi-Fi networks are used have led to a significant increase in the density of wireless networks, making performance problems harder to solve. Hence, Wi-Fi faces increasing challenges regarding the QoS. Understanding these problems is crucial for applications that rely on Wi-Fi, and addressing these challenges enhances the speed and reliability of the Wi-Fi network. In the following section, we will explore common Wi-Fi problems and limitations.

2.2.1 Congestion in Modern Wi-Fi Applications

Wi-Fi networks are facing a significant challenge in the form of increased demand for high bandwidth data transmissions while the capacity of the wireless spectrum remains limited. Congestion has become a common issue among Wi-Fi applications due to this. It occurs when the number of devices attempting to communicate over a wireless channel exceeds its capacity, causing data packets to become lost, delayed, or dropped. This can lead to slower data rates, increased latency, and even disconnections. For instance, Streaming high-quality videos can consume a lot of bandwidth, slowing down the speeds of other devices in the network. Additionally, smart home devices, such as security cameras, require an internet connection. As more of these devices are added to a network, they can consume significant bandwidth, increasing the congestion.

Wi-Fi channel access management is a major issue in Wi-Fi. When a device wants to send data over the network, it must first listen for other devices transmitting on the same channel. Transmission can be delayed if many devices are trying to access the network simultaneously. The CSMA/CA protocol allows multiple devices to share the same wireless channel while avoiding collisions. However, it is imperfect, and collisions can still occur, especially in dense networks. This can slow down the network because devices must wait a random amount of time before sending data. CSMA/CA has limitations and challenges, such as the hidden node problem and the exposed node problem. The hidden node problem occurs when two devices cannot detect the transmissions of each other, leading to collisions when they both try to transmit data simultaneously. The use of RTS/CTS can solve this problem, but it also introduces additional overhead on the performance of the network. The exposed node problem occurs when a device refrains from transmitting data, even though the

channel is idle, because it detects ongoing transmissions from other devices. This leads to inefficient use of the available bandwidth.

Interference from other Wi-Fi devices can delay channel access. Neighboring Wi-Fi networks and Wi-Fi devices can all cause interference, making it more difficult for devices to assess an idle channel and access the network. These issues and limitations are essential to consider as they impact the QoS in Wi-Fi networks. To overcome some of these challenges, it is possible to enhance specific features without implementation specified in the standard. We explore some of these features in the following subsection.

2.2.2 Manufacturer-Specific Features of Wi-Fi

Despite the standardization efforts, there are still some features of Wi-Fi that are left for the manufacturer to decide. These features are not specified in the standard, and each manufacturer can implement them as they want. The variety of implementations leads to a varying performance of Wi-Fi among devices. We list some of the manufacturer-specific features:

- Wi-Fi standards define QoS mechanisms for prioritizing different types of traffic, such as voice or video. However, the exact implementation of these mechanisms is left up to the manufacturer. Some manufacturers may implement additional QoS mechanisms or prioritize traffic differently than others.
- Wi-Fi standards do not specify algorithms for selecting which channel to use. Different manufacturers may use different algorithms or have different settings for channel selection. The algorithm is responsible for selecting the optimal channel for the Wi-Fi network, taking into account factors such as interference from other wireless networks, signal strength, and available bandwidth. If the channel selection algorithm is not effective, it can lead to channel congestion and poor network performance.
- Wi-Fi standards specify power management functions for Wi-Fi devices, but it does not specify algorithms or techniques that manufacturers should use to conserve power. Manufacturers can implement their own power management strategies to optimize battery life and reduce power consumption. Manufacturers must find a balance between conserving battery power and providing reliable and efficient data transmission and reception.
- Rate adaptation algorithm determines the data rate at which devices can transmit data over the Wi-Fi network. Based on the measured network conditions, the algorithm switches between different MCS values, channel width values, and spatial streams. Different manufacturers use different algorithms to adapt to the network conditions and optimize the data rate for better performance.

Manufacturer-specific features play a critical role in the performance of Wi-Fi networks, and there is always room for improvement, one of the most important features are the rate adaptation algorithms. RAAs help us understand the ways in which Wi-Fi networks adapt to changing network conditions and how they optimize their performance. These algorithms are critical components of Wi-Fi networks and play a key role in determining the data rates that can be achieved under various network conditions. By understanding these algorithms, we can gain insights into how Wi-Fi networks operate and how they can be optimized for specific applications and use cases. In the following section, we will discuss these algorithms in detail.

2.3 Rate Adaptation Algorithms

RAA is an important feature of Wi-Fi that is not specified in the standard but is left for the manufacturer to decide. These algorithms are responsible for adjusting the data rate of a Wi-Fi transmission based on the channel conditions. In other words, RAA determine the maximum speed at which data can be transmitted over a wireless link.

The objective of RAA is to achieve the highest possible throughput while maintaining a stable and reliable wireless connection. The algorithms are designed to take into account the capabilities of the devices involved in the communication, such as the MCS values, channel widths, number of spatial streams, and guard interval lengths supported by the devices.

The implementation of RAA can have a significant impact on the performance of Wi-Fi devices. A poorly designed algorithm can result in low throughput, frequent re-transmissions, and unstable connections, leading to a frustrating user experience. On the other hand, a well-designed algorithm can achieve high throughput, low latency, and stable connections, improving the user experience and enhancing the overall performance of the network.

RAAs can be classified into various categories according to the metrics that they use to evaluate the channel or link quality, such as frame loss and SINR in [39], or consecutive transmission count, frame loss ratio, transmission time, throughput, SINR, bit error rate, and combined metrics in [40]. We chose to classify RAAs into three categories:

- Explicit Feedback: RAAs base their adaptation on the feedback of the receiver
- Implicit Feedback: RAAs base their adaptation on the information available on the sender side
- Hybrid: which is a category that combines information from the feedback of the receiver and information available to the sender

In what follows, we will describe each of these categories by selecting representative RAAs of each category. Table 2.1 summarizes the information about the different RAA algorithms presented in this section.

2.3.1 Explicit Feedback Algorithms

Explicit Feedback is a receiver-driven rate adaptation scheme where the receiver makes a decision based on its estimation of the channel conditions and relays it back to the sender via different approaches using control frames, such as CTS and ACK.

On-Demand Feedback Rate Adaptation (OFRA) [41] is a receiver-based RAA, where the channel quality is estimated at the receiver based on SINR values. The receiver selects the optimal bit rate from a lookup table created previously. It contains a set of thresholds at which data rates should be changed. This information is returned to the sender on demand while using ACK frames. In the case of ACK-less traffic, OFRA uses a specially designed feedback frame. OFRA presents some limitations, such as modifying the ACK frame that violates the standard and introducing additional overhead with the special feedback frame sent at the lowest data rate.

SNR-aware Intra-frame Rate Adaptation (SIRA) [42] selects two rates for a single Aggregate MAC Protocol Data Unit (Aggregated MAC Protocol Data Unit (A-MPDU)) transmission. It finds the starting symbol “T” when the rate should be changed. When the condition $SINR_i < SINR_{th}$ is met, the symbol “T” is found. $SINR_{th}$ is the minimum SINR at which the theoretical Bit Error Rate (BER) of the primary rate is less than 10^{-4} . Subsequently, “T” is fed back to the sender via the BlockAck. The main drawback of SIRA is that it only determines two rates for an aggregated frame, which may not be enough for a fast-changing channel.

An Ideal RAA is implemented in the famous network simulator NS-3. This RAA initially creates a table of SINR and MCS pairs. The SINR thresholds in this table ensure selecting an MCS that leads to a Bit Error Rate (BER) below a certain value. For example, the default value is 10^{-5} , and the SINR is fed back from the receiver to the transmitter via a perfect out-of-band mechanism. The main drawback of this mechanism is the use of an out-of-band channel for sending back the feedback, which is not available in the Industrial, Scientific, and Medical (ISM) bands used by IEEE 802.11.

2.3.2 Implicit Feedback Algorithms

Implicit Feedback is a sender-driven rate adaptation scheme usually based on Packet Error Rate (PER). The main idea is for the sender to select an appropriate data rate based on the PER observed on his side. This mainly requires ACK to enable the sender to calculate PER.

STandard-compliant and mobility aware PHY Rate and A-MPDU Length adaptation (STRALE) [43] jointly adapts the physical rate and A-MPDU length. After receiving a BlockAck, it calculates the optimal A-MPDU length for the highest throughput of the last A-MPDU transmission. Subsequently, the next A-MPDU length of the next transmission is determined using the Exponential Weighted Moving Average (EWMA). The difference between the latest and the newly calculated A-MPDU length is compared to a certain threshold. If it is greater than the threshold, STRALE decides on MCS and A-MPDU length after checking if the throughput with one lower MCS index using the latest A-MPDU length is better than staying at the same MCS index with the newly calculated A-MPDU length. The main drawback of STRALE is that it does not consider interference.

Minstrel-HT [44] relies on three parameters: channel width, guard interval, and several streams to create group rates. Each group contains eight different data rates represented by the MCS index. Minstrel uses probing to determine the best data rate. This process consists of two periods: sampling and non-sampling periods. During the sampling period, Minstrel-HT selects a random data rate from all the available data rates in each group. If a data rate results in higher throughput than the previous one, it is used for subsequent MPDU transmissions. Otherwise, Minstrel-HT keeps using the previous data rate. Throughput is calculated based on Frame Loss Rate (FLR) while considering EWMA. Minstrel-HT collects three data rate options at the end of the sampling period: best throughput, second best throughput, and highest probability of successful transmission. Subsequently, in the non-sampling period, MPDUs are sent using the best data rate until the maximum number of re-transmissions is reached in the case of packet loss. Then, the second-best data rate is used. Similarly, the best probability data rate is used if the second-best data rate experiences packet loss. The evaluation in [44] showed that in some conditions, Minstrel-HT fails to enhance throughput, especially in a non-fading channel when the quality of the channel changes back from bad to good.

MIMO Rate Adaptation (MiRA) [45] is a rate adaptation used for MIMO channels. It overcomes MPDU loss by applying a zigzag rate adaptation between intra-mode and inter-mode. MiRA first performs probing on the rate in MIMO intra-mode. If goodput is not increased in intra-mode, MiRA zigzags to inter-mode MIMO. The probing mechanism only starts if significant changes occur in the measured moving average goodput of the current rate. The probing interval of MiRA is also adapted, which limits the probing number when goodput is low. MiRA also considers frame aggregation and Block Acknowledgement schemes when performing the best data rate probing. It also includes a collision-aware mechanism where the sender detects collision if it satisfies the condition that the aggregate frame has experienced at least one retry. The loss ratio of its sub-frames is less than 10%. If collision exists, it triggers the adaptive RTS/CTS mechanism. The main drawback of MiRA is the introduction of overhead when using the RTS/CTS mechanism.

Rate Adaptation for Multi-Antenna System (RAMAS) [46] is a credit based approach. The data rates are grouped into two groups: modulation and enhancement groups. The modulation group consists of different MCS values. The enhancement group consists of spatial stream, guard interval, and channel width. RAMAS uses credit-based algorithms, which rely on the packets' success and failure statistics, to adapt these groups independently of each other and combine the results together to decide the overall feature set. In each group, different rules are applied to increase or decrease the data rate sequentially. The main drawback of RAMAS is that it performs poorly because its credit-based scheme is conservative in adapting the number of streams and aggressive in adapting the MCS. This mismatch causes RAMAS to often operate at sub-optimal settings with single stream and high MCS values leading to higher losses and reduced performance, as shown in the evaluations in [47].

Damysus [48] addresses 802.11ax exploiting the Basic Service Set (BSS) Color Scheme. It increases transmission opportunities by using adaptive Overlapping Basic Service Set (OBSS)/Preamble-Detection (PD) thresholds, leading to a higher contention inside a BSS and jointly adjusting the transmit power level. A statistical study is done during an interval of 100ms and a cycle of 1 second, where packet transmissions' success and failure are recorded and compared to the success and failure thresholds. Depending on the statistical results collected, it is then decided whether to increase or decrease either the rate, the OBSS/PD threshold, or the transmission power. The main drawback of Damysus is relying on packet loss ratio thresholds. In [49], several experiments were done to verify that no single best Packet Loss Ratio (PLR) threshold can help achieve the maximum throughput.

2.3.3 Hybrid Adaptation

In Mutual Feedback (MutFed) [50], the SINR is measured on the receiver side. After 10 frames, a new rate selection decision is made to stay at the same rate or change the rate. This decision is based on a table that maps SINR threshold ranges to rates. The newly selected rate is fed back to the sender by sending an ACK with the proposed rate. On the transmitter side, if two frame transmissions fail, the transmitter automatically lowers the MCS for the following transmission. The main drawback of MutFed is relying on a static threshold to lower MCS values which can frequently occur in a dense network.

EasiRA [51] measures the link quality by two means. First, it calculates the FLR and combines it with mobility and other sensor information. Secondly, it obtains the Environmental Signal Strength (ESS) information to help differentiate the causes of

packet loss. When a packet cannot be successfully received due to bit errors, the receiver sends a special control frame, named "Non-Acknowledgements (NACK)", to the transmitter to inform it that it may suffer a collision. If the transmitter does not receive an ACK or a NACK, it reduces the rate. Finally, it combines the random and deterministic rate adaptation mechanisms together. The main drawback of EasiRA is that it tries to identify collisions while using external information, such as sensor-hints and ESS, which may not be available on all devices.

RAAs play a crucial role in adjusting the data rate between access points and users based on channel conditions. Many of these algorithms use metrics such as SINR information to make decisions on MCS. Maintaining a high SINR is essential to achieve high data rates. Thus, effective WiFi deployments could consider metrics like SINR as a significant design parameter. In the following section, we will explore channel modeling, which is considered a step before deployment that takes into account SINR based on the MCS requirement of the application.

2.4 Wi-Fi Channel Modeling

Wi-Fi channel modeling plays a crucial role in predicting and analyzing the performance of Wi-Fi networks during the planning phase of Wi-Fi deployments. It provides a mathematical representation of how radio signals propagate through different environments, both indoor and outdoor. Channel models aid in determining the optimal placement of access points, estimating coverage and capacity, and predicting the quality of service for users. In this section, we will explore the propagation model equations of Wi-Fi channels in an outdoor environment. We will examine the effects of this model on both coverage and capacity, taking into account SINR constraints based on the application requirements.

2.4.1 Propagation Models of Wi-Fi Channels

Channel modeling is one of the key parts when planning Wi-Fi deployments, as it plays an important role in its performance. Before confirming the planning of the system, accurate propagation characteristics of the environment should be known. Signals transmitted over the Radio Frequency (RF) channel are subject to losses due to fading, scattering, and penetration, among others. The inaccuracy of the estimated Received Signal Strength Indicator (RSSI) values can be reduced if the propagation characteristic of the channel can be predicted using an appropriate model that best suits the environment of the area of interest.

A propagation model is a set of mathematical equations used for representing the characteristics of an RF signal of a given scenario and environment. These mathematical equations are used for estimating the received signal strength in a particular environment. The models are usually designed for distinct scenarios due to factors such as the distance between the transmitter and receiver, obstacles, or weather conditions. An understanding of the path loss in a given geographical area provides better network planning, it is important for determining the best access points locations as well as estimating their coverage area based on link budget calculations.

Path loss is the decrease of an electromagnetic wave power density as it spreads or travels through a medium, and it is given by [52]:

$$L_p = L_0 + 10\alpha \log_{10}(d) + X \quad (2.2)$$

Table 2.1: Summary of existing RAA.

Name	Category	Metrics	Pros	Cons	Modify Standard
Ideal	Explicit Feedback	SINR	Instantaneous feedback and rate decision	Uses perfect out-of-band mechanism for feedback	Yes
OFRA	Explicit Feedback	SINR	Feedback is sent only when a new rate is selected	Additional frame is sent for feedback at the lowest rate	Yes
SIRA	Explicit Feedback	SINR	Uses 2 rates for a single A-MPDU transmission to adapt with channel changes	In a fast-changing channel, we may need more than 2 rates for a single A-MPDU transmission to adapt to channel changes	No
STRALE	Implicit Feedback	Throughput	Prefers to use an optimal A-MPDU length instead of decreasing the rate	Does not consider interference	No
Minstrel-HT	Implicit Feedback	Throughput	It can select a high rate with a significant FLR as long as it can achieve the best throughput	In some conditions, it fails to enhance the throughput [44]	No
MiRA	Implicit Feedback	Throughput	Collision-based decision making and probing interval adaptation	Introduces overhead when using the RTS/CTS mechanism	No
RAMAS	Implicit Feedback	PLR	It is efficient in MIMO where it selects between different values of transmission metrics	Slow in adapting the number of streams and premature MCS adaptation	No
Damysus	Implicit Feedback	PLR	Performs well in dense environments in infrastructure mode	Is not applicable in Ad Hoc mode	No
MutFed	Hybrid	SINR,FLR	Distinguishes the cause of frame loss combined with SINR decision	In a dense environment many frame losses cause the algorithm to send at lower rates	Yes
EasiRA	Hybrid	ESS,PLR	Combines external sensors and ESS to detect collisions and uses this information with the PLR for rate selection	It tries to identify collisions using external information, such as sensor-hints and ESS which may not be available in all the devices	No

where L_p is the path loss for a position p , d is the distance at which we are calculating the path loss, L_0 is the power loss at a reference point in the coverage field of the antenna at a small distance d_0 from the transmitting antenna, α is the path loss exponent, and X is the fading component.

The link budget is the analysis of signal loss/gain factors during signal propagation. It is the calculation of all the gains and losses that occur during signal transmission through a wireless medium. It expresses the attenuation in transmitted signal power caused by propagation losses and antenna gains. The link budget equation can be expressed as:

$$P_r = P_t + G_t + G_r - L_p \quad (2.3)$$

where P_r is the received power, P_t is the transmitted power, G_t and G_r are the antenna gains at the transmitter and receiver respectively, and L_p is the path loss.

In Wi-Fi deployment planning, one has to comply with the application needs, such as minimum data rate and maximum latency. These constraints affect the deployment of access points and the choice of technology used. A minimum SINR has to be available at the receiver to offer the required data rate. The SINR equation is expressed as:

$$SINR = \frac{P_r}{N} \quad (2.4)$$

Where N represents the noise and interference at a given time and position, P_r is the received power expressed as RSSI.

SINR is an important parameter in the selection of the MCS index, which depends on the modulation type, the coding rate, the number of spatial streams, the channel width, and the guard interval. A combination of these parameters represents a unique MCS index. Table 2.2 shows a part of the MCS lookup table, which is used to find at which data rate two devices can communicate based on minimum SINR thresholds and RSSI values.

MCS	Modulation	Coding	40 MHz			
			Data Rate (Mbps)		Min. SINR	Min. RSSI (dBm)
			800 ns	400 ns		
0	BPSK	1/2	13.5	15	5	-79
1	QPSK	1/2	27	30	8	-76
2	QPSK	3/4	40.5	45	12	-74
3	16-QAM	1/2	54	60	14	-71
4	16-QAM	3/4	81	90	18	-67
5	64-QAM	2/3	108	120	21	-63
6	64-QAM	3/4	121.5	135	23	-62
7	64-QAM	5/6	135	150	28	-61
8	256-QAM	3/4	162	180	32	-56
9	256-QAM	5/6	180	200	34	-54

Table 2.2: 802.11ac MCS lookup table [1].

2.4.2 Wi-Fi Coverage

In a deployment where we need to guarantee a minimum data rate, a minimum SINR value should be considered as a target. Interference level is a factor that

cannot be controlled because we are dealing with Industrial, Scientific, and Medical (ISM) frequency bands. The estimation for reaching the target SINR is done based on the propagation model. Hence the SINR constraint can be written as:

$$SINR \geq s_{th} \quad (2.5)$$

where s_{th} is the minimum (target) SINR sufficient for using a MCS, which makes it possible to send data at the minimum data rate required by the application specifications.

Using equations 2.2, 2.3, and 2.4 we get:

$$\log_{10}(d) \leq \frac{P_t + G_t + G_r - L_0 - N - s_{th}}{10\alpha} \quad (2.6)$$

where d is the maximum coverage distance of an access point that ensures having the data rate requirement at the edge of a coverage cell.

2.4.3 Wi-Fi Channel Capacity

Channel capacity gives an upper bound on the data rate for reliable communication, as stated by Shannon's theorem [53]. It is the maximum attainable mutual information between the transmitter and the receiver. The network is capable of serving more users and higher data rates per user when its capacity increases. In the case when the channel is not deterministic, each time the channel is used employs an independent realization of the mutual information matrix H [54].

It should be noted that capacity is a limit to the error-free bit rate that is provided by information theory. Any wireless system can only achieve a bit rate that is only a fraction of that capacity. For a channel without shadowing, fading, or inter-symbol interference, Shannon proved that the maximum possible bit rate on a given channel of bandwidth B is given by [55]:

$$C = B \log_2(1 + SINR) \quad (2.7)$$

Consider a channel shown in figure 2.4 with:

$$y = gx + n \quad (2.8)$$

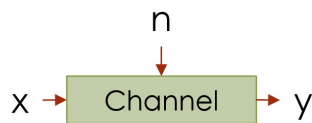


Figure 2.4: A channel with input x and noise n .

where g is the channel response, it reduces the SINR as the signal loses power while propagating through the channel. The capacity equation of a Single Input Single Output (SISO) system in equation 2.7 becomes:

$$C = B \log_2(1 + SINR|g|^2) \quad (2.9)$$

MIMO systems consist of different correlated sub-channels. However, different techniques can be used to decompose the MIMO channel into m parallel independent sub-channels, as shown in figure 2.5, such as singular value decomposition [56].

Hence the capacity of each sub-channel becomes:

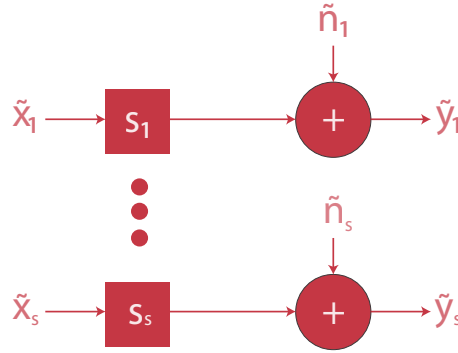


Figure 2.5: A MIMO channel decomposed into s parallel independent subchannels.

$$C_m = \log_2 \left(1 + s_m^2 \frac{P_m}{N} \right) \quad (2.10)$$

where s_m is the channel response of the m_{th} sub-channel and P_m is the power allocated to the m_{th} transmitting antenna. The objective is to maximize the overall capacity by the optimal power allocation over the sub-channels. Channel estimation becomes handy here to extract the Channel State Information (CSI). CSI represents the effect of the channel, such as scattering, fading, and power decay with distance. In case CSI is not known, the power is distributed equally over all the transmitting antennas:

$$P_1 = \dots = P_S = \frac{P}{N} \quad (2.11)$$

In case the CSI information is known at the transmitter, we allocate power to maximize the capacity given by equation 2.12, considering that the sum of all transmitting power does not exceed a P_{limit} :

$$C = \sum_{m=1}^S (C_m) \quad (2.12)$$

$$P_1 + \dots + P_S = P_{limit} \quad (2.13)$$

P_{limit} is the maximum power allowed according to the regulations of the country. The water-filling algorithm is one of the best solutions for such a problem [57]. In this algorithm, the transmitter allocates more power to the stronger sub-channels and less or no power to the ones with worse channel conditions [58]. Solving the above problem with water filling gives:

$$P_m = \max \left(\mu - \frac{N}{s_m^2}, 0 \right) \quad (2.14)$$

where μ is selected to satisfy the condition in equation 2.13.

Figure 2.6 is an example of capacity enhancement when using a MIMO system with spatial multiplexing under different SINR conditions.

Wi-Fi channel modeling is essential to understanding how Wi-Fi signals propagate and the expected performance of the Wi-Fi networks. Coverage and capacity study based on application requirements drive the planning for Wi-Fi deployment. In the following section, we will review the available Wi-Fi deployment strategies from the literature.

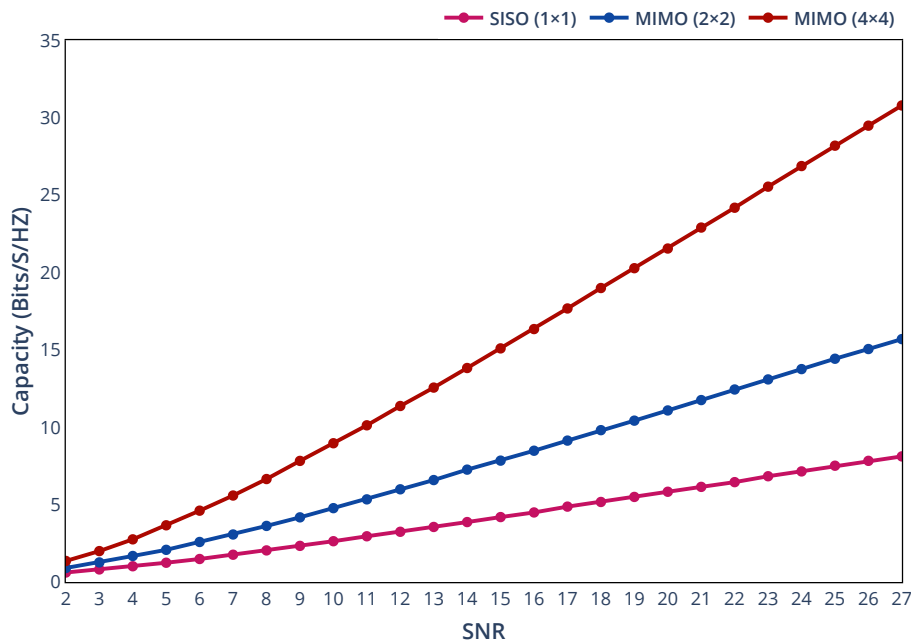


Figure 2.6: Comparison between the capacity of SISO and MIMO systems.

2.5 Wi-Fi Networks Deployment

Wi-Fi APs are the backbone of any Wi-Fi network deployment, responsible for transmitting and receiving wireless signals. APs are typically installed in specific locations throughout an indoor or outdoor location, allowing users to connect to the network from anywhere within range. APs can be deployed in a variety of configurations based on the environment and expected user density. The deployment of APs is critical to ensuring acceptable coverage and performance of Wi-Fi networks.

Indoor Wi-Fi networks are typically deployed in buildings such as offices, homes, schools, and hospitals, where there is a high demand for wireless connectivity. However, outdoor Wi-Fi networks are deployed in public spaces, ranging from high-density stadiums and campuses to agricultural fields. In this section, we will review previous AP deployment strategies in indoor and outdoor environments.

2.5.1 Indoor Environments

Authors in [59] investigate the network performance in a targeted building and propose an optimization plan for deploying AP devices based on received power and path loss parameters obtained from Wireless InSite software [60]. The investigation took into account the effect of building materials and frequency sensitivity. The initial analysis of network coverage and performance showed discouraging results in covering the entire building, driving the proposal of an optimization plan that uses a different model with AP devices of higher power and antenna gain. The proposed plan reduces the number of devices needed from 13 to 8, increases received power, reduces path losses, minimizes interference effects, and expands coverage area at a lower implementation cost.

Authors in [61] proposed a decision support tool as a solution for WLANs deployment optimization in indoor settings. Genetic algorithm and simulated annealing were used to determine the optimal number and placement of APs for optimal network coverage. Parameters used in the study were antenna type, channels, wall structures, and throughput. The authors claim that this approach is superior to

traditional trial-and-error methods and can improve WLANs performance significantly.

Authors in [62] used a multi-objective genetic algorithm with the aim of minimizing the number of APs while maximizing coverage over a whole planning area. The developed algorithm is integrated into an engineering tool called WiFiSim, which allows for the investigation of complex issues related to the design of WLANs. The tool is compared with a previous approach based on a mono-objective genetic algorithm, and the results show that the multi-objective genetic algorithm achieves better performance. The paper concludes that the developed algorithm and tool can provide an efficient and scalable solution for the design and optimal deployment of WLANs, saving time and cost for network technicians.

Authors in [63] use a central controller with sensors deployed in different locations inside the area of interest that measure RSSI values. The central controller analyzes the measurements and updates the configuration of the APs. The simulation results showed that the proposed solution can effectively identify and solve coverage problems related to interference and coverage holes. The authors concluded that their method improves coverage robustness and saves the cost of human intervention.

Mono-objective and multi-objective Tabu searches were used in [64] to optimize the number, transmit power, and location of the access points in an indoor setup with the help of constraints concerning coverage, interference, and QoS. Results show the mono-objective search performs better in terms of computational time but requires several launches to achieve the desired trade-off. However, the multi-objective search is promising but comes with more complexity in terms of time and computation.

In [65], the authors developed a mathematical model to determine the optimal number and location of APs WLANs. They used the Discrete Gradient optimization algorithm to solve the problem, with the objective of minimizing the average path loss and maximum path loss received by any receiver. The results showed that the size of the design area and the number and locations of users affect the location and number of APs needed. However, the authors concluded that some manual tuning of the algorithm parameters is needed to produce better results.

2.5.2 Outdoor Environments

Authors in [66] focus on the coverage problem in outdoor wireless network design and propose methods for near-optimal coverage using greedy algorithms based on RSSI. The study compares various wireless network coverage planning scenarios in an imaginary town subdivided into areas. APs are switched between on and off by the proposed algorithm. The authors conclude that the proposed algorithm is simple, fast, and provides good sub-optimal solutions.

Authors in [67] provide an analysis of the Wi-Fi network in a large university campus. The study involves coverage predictions and statistical analysis of data from existing APs. The results show areas with weak or no Wi-Fi signal coverage and over-crowded access points. Based on these findings, the paper recommends installing new access points in some widely used locations around the campus to improve signal quality. The paper also discusses the disparity between theoretical and real-world aspects of WLANs deployment, including the maximum number of clients per access point and the ideal placement of access points that depends, in their opinion, on the applications used by the users and the co-channel interference.

Authors in [68] combine predictive models and geospatial analysis to aid in the design and placement of Wi-Fi APs in an outdoor environment. The study com-

compares different geospatial analysis techniques and signal strength prediction models based on RSSI. RSSI values were used to predict the coverage of the APs. Authors developed an empirical model to assist in predicting the coverage of APs for better deployment.

The work in [69] discusses the use of beamforming to improve the performance of WLANs in outdoor environments. The study focuses on the performance of adaptive antenna array beamforming in a real-life outdoor environment. The tests showed that beamforming can significantly improve throughput rates and increase range in near-line-of-sight situations. In addition, beamforming can enable communication in non-line-of-sight situations where it would otherwise not be possible. According to the authors, the gain in throughput depends on the radio conditions and the implementation of the beamforming algorithm. Moreover, they conclude that beamforming can counteract degradations and improve WLANs deployment.

Authors in [70] explore the deployment of WLANs in an outdoor environment where multiple APs form a cluster and are connected through wireless repeaters. The paper proposes using the Mixed-integer nonlinear programming (MINLP) optimization approach to determine the optimal number of APs in a cluster and the best separation distance between them to maximize the throughput between the APs and mobile devices. The paper compares uniform spacing to the increasing spacing deployment strategies and concludes that the increasing spacing strategy outperforms the uniform spacing strategy in terms of throughput achieved.

The literature presents numerous studies that have analyzed the deployment of Wi-Fi networks using various techniques, such as evolutionary algorithms and nonlinear programming, and based on commonly used metrics such as RSSI. The impact of beamforming was also assessed in these studies. The main objective of these studies was to enhance the capacity and coverage of WLANs while reducing the number of access points required. However, optimizing the placement of access points is not enough to fully optimize the performance of Wi-Fi networks, and various other features of Wi-Fi still have room for improvement. One promising modern technique for achieving this optimization is the use of ML techniques, particularly Reinforcement Learning (RL). In the following section, we will discuss the added value of ML, with a specific focus on RL in addressing the challenges faced by Wi-Fi networks.

2.6 Reinforcement Learning in Wireless Networks

ML is a branch of artificial intelligence that involves making algorithms and models that allow computers to learn from data and make decisions or predictions without being explicitly programmed. The goal of ML is to develop algorithms that can automatically identify patterns and relations in data and use them to make predictions or decisions. The process of learning usually requires the algorithm to be trained using a substantial collection of data, either labeled or unlabeled. Labeled data is characterized by annotations or tags associated with specific classes or categories, whereas unlabeled data lacks any predefined labels. Machine learning includes supervised, unsupervised, semi-supervised, and reinforcement learning.

- Supervised learning requires a dataset with labels to train a model to predict outcomes on new data. If there is prior knowledge about the wireless network environment and the data can be labeled, supervised learning algorithms can be used for solving network problems such as resource allocation [71], localization [72], and security [73].

- Unsupervised learning involves finding patterns or structures in an unlabeled dataset. The learner is provided only with inputs without the outputs, while learning is performed by finding similarities in the input data. It has been used in the wireless networking domain for data aggregation [74], node clustering in Wireless Sensor Network (WSN) [75], and data clustering [76].
- Semi-supervised learning combines labeled and unlabeled data. It is employed when there is a small quantity of data that has been labeled and a large amount of unlabeled data. This type of learning is useful because it can help reduce the expenses associated with creating a fully labeled training dataset, particularly in situations where labeling every instance is impossible. Many challenges has been addressed using semi-supervised learning in areas such as anomaly detection [77], signal recognition [78], and security [79].
- RL involves training an agent to make decisions in an environment and learning from rewards and punishments. The agent observes the state space of the environment and takes action. Over time, the agent learns actions that maximize its rewards. RL has been applied to a wide range of challenges in wireless networks, such as resource management [80], rate adaptation [81], channel access [82].

In scenarios involving wireless optimization problems, the environment can exhibit complexity, and finding the optimal solutions can depend upon numerous factors that are challenging to model explicitly. In such instances, relying on pre-existing datasets and employing supervised, unsupervised, or semi-supervised learning techniques may prove ineffective. Moreover, RL agents can learn from their interactions with the environment without prior knowledge of its dynamics, and it can handle a large set of states by employing Deep Learning (DL). Deep learning is a branch of machine learning that involves a neural network architecture. These networks aim to learn from data sets. A neural network takes an input and processes it through hidden layers, and produces an output.

RL is often preferred over supervised/unsupervised/semi-supervised learning in this context since it involves learning through trial-and-error interactions with an environment. This trial-and-error process enables the agent to discover actions that maximize the reward. Furthermore, RL can handle scenarios where actions have delayed effects on the rewards, making it well-suited for optimizing wireless networks, where actions taken at a particular time may significantly impact future network performance. RL is capable of rapidly solving a variety of optimization issues. It offers computation-efficient solutions to a wide range of optimization problems, including Markov Decision Process (MDP) and non-convex optimization problems that are difficult to solve. RL does not rely on precise environment modeling. Through experimenting in an interactive environment, RL agents can learn indirect information about network dynamics from raw high-dimensional data. By doing so, RL can then learn correlations between different factors, such as network metrics [83].

In this section, we will explore the concept behind RL, the methods of training and evaluation, and examples of leveraging RL in Wi-Fi networks.

2.6.1 The Core Concept Behind Reinforcement Learning

RL is a machine learning approach where an agent acquires behavior in an environment by taking actions and receiving feedback in the form of rewards or penalties. The fundamental concept behind RL is to empower the agent to learn from its own

experiences rather than depending on a predetermined set of rules. RL allows the agent to learn appropriate behavior through iterative experimentation. RL consists of the following components:

- **Environment:** The environment is the external world in which the agent operates. It can be anything from a simulation to a physical environment, like a robot in an agricultural field. The environment provides the agent with *observations* and *rewards* based on its actions.
- **Agent:** The agent is the entity that interacts with the environment. The agent performs actions in the environment based on its *observations* and receives *rewards* or *penalties* based on the outcome of its actions.
- **State:** The *state* is the current environment situation that the agent observes.
- **Action:** The *action* is the decision made by the agent based on its current state. It is the output of the agent's decision-making process and determines the next state of the environment.
- **Reward:** The *reward* is the feedback the agent receives from the environment based on its actions. It is a scalar value that indicates the quality of the action taken by the agent.

Figure 2.7 shows an illustration of the basic RL functionality.

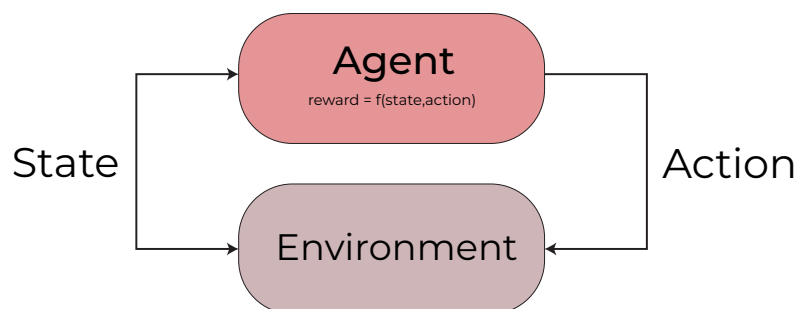


Figure 2.7: An illustration of the basic RL functionality.

The goal of reinforcement learning is for the agent to learn a policy, a function that maps states to actions. The policy determines the actions of the agent in each state to maximize its cumulative reward over time. The expected cumulative reward, which is the total rewards the agent has gained over time, represents the quality of the current policy. The policy is updated based on the rewards received after each action. RL algorithms use various techniques to update the policy, such as value-based, policy-based, and actor-critic methods:

- **Value-based methods** rely on the estimation of the state-action value function, which is a function that estimates the expected cumulative reward for taking a particular action in a particular state. This value is known as Q-Value. The agent updates its policy by selecting the action with the highest state-action value, which is known as the Q-Value. Some examples of value-based methods are Q-Learning and Deep Q-Networks (DQN). What follows is a list of the pros and cons of value-based methods:
 - They can learn in continuous action spaces by using function approximation methods, such as neural networks, to represent the Q-function, such as in DQN.

- They are not suitable for handling large state and action spaces as they can lead to high memory requirements and computational costs.
 - They can suffer from overestimation or underestimation of Q-values, which can lead to sub-optimal policies.
- Policy-based methods optimize the policy function by adjusting its parameters to maximize the expected cumulative reward. These methods typically use gradient descent to update the policy iteratively. Some examples of policy-based methods are REINFORCE and Deterministic Policy Gradient (DPG). What follows is a list of the pros and cons of policy-based methods:
 - They can handle stochastic policies, which are policies that output a probability distribution over actions rather than deterministic actions.
 - They can converge to a global optimum under certain conditions, such as when the policy is continuously differentiable and the optimization problem is convex.
 - They can be computationally expensive, especially when using high-dimensional state and action spaces and complex policy representations.
 - Actor-critic methods combine value-based and policy-based methods by using separate networks to estimate the state-action value and the policy function. The actor-critic method learns to improve the policy by using the state-action value function as a baseline. Some examples of policy-based methods are Advantage-Actor-Critic (A2C) and Deep Proximal Policy Optimization (PPO). What follows is a list of the pros and cons of actor-critic methods:
 - They can converge faster than policy-based or value-based methods because they learn both the policy and the value functions simultaneously.
 - They can incorporate prior knowledge of the environment by initializing the value function with a pre-existing estimate or using transfer learning techniques.
 - It can be difficult to tune hyperparameters, such as the learning rate and the discount factor, which can affect the stability and convergence of the algorithm.
 - The actor can overfit the current policy if the hyperparameters are not tuned correctly.

The hyperparameters mentioned earlier are a set of parameters that define the learning algorithm and that need to be specified by the developer. They have a big impact on the performance and convergence of the method. Here are some examples of hyperparameters:

- Learning rate: it controls the step size of the updates to the model during learning. Increasing the learning rate can help in converging faster, but it may also lead to instability and oscillations.
- Discount factor: it determines the importance of future rewards. A higher discount factor favors future rewards more than immediate rewards. This parameter can have a big impact on finding the optimal value. If badly tuned, it will converge to a local optimum instead of finding the global optimum.

- Exploration rate: it determines how much the agent should explore the action space instead of exploiting the current values that maximize the reward. A high exploration rate can help the agent better fine-tune the policy but can also lead to inefficient learning and slow convergence.
- Neural network architecture: in deep reinforcement learning, the neural network architecture can significantly affect the performance of the learning algorithm. The number of layers, the number of units per layer, and the activation functions are examples of hyperparameters that can be tuned.
- Replay buffer size: in deep reinforcement learning, a replay buffer is a memory that stores the experiences of the agent in the form of transitions, which consist of the current state, action taken, next state, and reward received. The size of the replay buffer is a hyperparameter that can affect the sample efficiency, the memory requirements, the learning stability, and the convergence of the algorithm.
- Batch size: it determines the number of experiences sampled from the replay buffer at each iteration of training. Large batch size can lead to more stable learning but requires more memory and computational capacities.

RL enables machines to learn and adapt to complex environments in a way that was previously not possible. However, reinforcement learning is still an active area of research, and many challenges need to be addressed. The integration of deep learning with reinforcement learning has led to the development of a new field known as DRL, which has emerged as a major advancement in the RL domain.

2.6.2 Deep Reinforcement Learning

RL has several limitations that can make applying it in complex and high-dimensional environments challenging. One of the main limitations of traditional RL is the "curse of dimensionality", which refers to the exponential increase in the number of possible states and actions as the dimensionality of the problem increases [84]. This can make it difficult for an RL agent to explore the environment and learn an optimal policy. Deep Reinforcement Learning (DRL) can help address this limitation by using deep neural networks as a function approximator to build on the ideas of RL. This lets the agent learn from high-dimensional input data, like images or sounds, and apply its knowledge to new situations.

The exploration-exploitation trade-off is another limitation of traditional RL. The main difference between exploration in RL and DRL is the complexity of the exploration strategies. Traditional RL methods use simple exploration strategies like epsilon-greedy or softmax exploration [85], in which the agent chooses actions randomly with a certain probability of success. These strategies for exploring are easy to use and work well in simple settings with clear action spaces. On the other hand, DRL algorithms can use more sophisticated exploration strategies that take advantage of the power of deep neural networks. They can use uncertainty estimation to guide exploration, where the agent chooses uncertain actions or high variance in their predicted value [86]. For example, the agent may choose to explore a new part of the state space if it is uncertain about the value of the current state or if the value of the current state is highly variable across different deep neural network predictions. DRL algorithms can also use intrinsic motivation [87] and

curiosity-based [88] approaches to encourage the agent to explore novel parts of the environment.

DRL has been used to overcome challenges in Wi-Fi networks. Wi-Fi channels are known to be highly unstable, and numerous promising DRL solutions have been proposed in the literature as many modern DRL algorithms, such as Proximal Policy Optimization (PPO) [89] and Trust Region Policy Optimization (TRPO) [90] were designed to work in unstable environments. In the following subsection, we will discuss some of these DRL solutions.

2.6.3 Leveraging Reinforcement Learning to Optimize Wi-Fi Networks

The work of [91] proposes a solution for power control in Wi-Fi using distributed DRL by adapting the transmit power based on the observed states of each mobile node in a multi-hop Wi-Fi network. The observed states include the transmit power, link quality, and signal strength. The Deep Q-Learning (DQL) model was trained offline and evaluated using the Alfa AWUS036NHA USB wireless adapter. Evaluation tests showed that the DRL-based power control achieved significant improvements in energy efficiency (up to 24%) and throughput (up to 22%) compared to fixed power allocation schemes.

Double DQL was used for rate adaptation on the physical layer of Wi-Fi networks in [92]. The state representation included metrics such as MCS and RSSI. The agent relied on the goodput to calculate the reward and select a new triplet (MCS, channel width, number of spatial streams) predefined profiles. The authors used online training in which the model continues to learn even after deployment. The agent was deployed on Intel 802.11ac Network Interface Cards (NICs). It outperformed the Intel and Linux default rate adaptation algorithms by more than 200%.

Authors in [93] used DQL to enhance the performance of CSMA/CA in dense WLANs by observing the backoff values, performing actions to increase or decrease them, and learning from a reward calculated based on the probability of collisions. The DQL model was trained offline and evaluated using the Network Simulator 3 (NS-3) simulator. Results showed enhancement in terms of throughput, channel access delay, and fairness compared to other mechanisms in the literature.

Authors in [94] used DQL to check if clients actually benefit from participating in MU-MIMO. The metrics used in the offline training were CSI and SINR. Experimental results using a commodity AP showed that the additional implementation of the pre-screening algorithm alone, without otherwise modifying MU-MIMO client grouping or link parameter selection algorithms, can improve system throughput by up to 40% when half of the clients are mobile and to maintains throughput improvement of around 20% when 50% to 75% of the clients are mobile.

In [95], authors improve the data rate obtained by Wi-Fi clients using a client-AP association scheme based on DQL. The proposed method takes into account the application demands of the user and link capacity. The offline training and evaluation were performed in a homemade simulation environment developed with Python. Results showed improvements over standard signal strength-based association regarding throughput and ensuring application requirements.

The complexity of deployment and limited availability of equipment has led to a limited number of studies testing DRL models on real hardware. Instead, the majority of research in the literature has used network simulators for training and evaluating DRL models.

2.6.4 Using Network Simulators with Reinforcement Learning for Wi-Fi Optimization

Network simulators help to emulate the way a network operates. They do this by defining modules that represent network protocols, radio interfaces, and wireless channels. They are one of the leading evaluation methods in wireless networks and have an important place in academia and industry [96]. The simulation tools and results create an environment where those concerned in these fields can test, compare, and improve their proposals. Authors in [97] show some overviews of the most popular network simulators for modeling wireless network technologies.

When it comes to RL, network simulation tools can be used to represent the environment that the agent interacts with to learn. The learning accuracy of the model will depend on how well the network simulators can make the agent behaves realistically. Simulation tools offer great flexibility in enabling the agent to interact with any type and use case of the environment. Whether during (1) Offline training, where the model does not undergo any further modifications after the training is done. (2) Online training, where the model continues to learn even after being deployed, to explore states it did not encounter during its initial training. Objective Modular Network Testbed (OMNeT++) and NS-3 are among the most used simulators in the literature.

The introduction of tools such as NS-3 Gym has simplified the integration of reinforcement learning with network simulators. NS-3 Gym [98] provides a flexible and convenient interface between NS-3 simulator and reinforcement learning algorithms, allowing researchers to easily explore and optimize network protocols. Similarly, Tensorflow [99] has a C++ Application Programming Interface (API), which enhances the OMNeT++ simulator with machine learning capabilities by offering a framework that can incorporate various ML algorithms.

Chapter 3

Contribution 1 - Wi-Fi Networks Deployment

3.1 Introduction

This chapter presents our initial contribution toward enhancing Wi-Fi deployment in the context of smart farming. We concentrate on two interconnected topics: rate adaptation algorithms and Wi-Fi access point deployment strategies. In section 3.2, we examine and test rate adaptation algorithms found in the literature. We refine the "Ideal Rate Adaptation Algorithm" to ensure compatibility with the Wi-Fi standard and discuss the lessons learned. In section 3.3, we propose Wi-Fi access point deployment based on beamforming technology while considering the data rate requirements of the application. We utilize the modified ideal rate adaptation algorithm selected in section 3.2 in this section. In section 3.4, we evaluate through simulation our deployment strategies in various scenarios with distinct application requirements. Subsequently, we create an empirical model and compare its results to the simulation results. Finally, we conclude this chapter in section 3.5.

3.2 Selecting the Most Effective Rate Adaptation Algorithm

Several rate adaptation algorithms have been proposed and implemented over the years, ranging from simple fixed-threshold methods to more complex machine learning based approaches. These algorithms employ different strategies to determine the optimal data rate, such as monitoring the number of successful or failed transmission attempts and estimating the channel conditions based on SINR or RSSI. In this section, we evaluated four different rate adaptation algorithms from the literature. The primary focus of this analysis was to determine their performance in terms of throughput and packet loss across various network conditions. These algorithms, which represented a range of approaches to rate adaptation, were tested to help us understand their relative strengths and weaknesses and select the most effective one. We then discuss the lessons learned from the results of this evaluation.

3.2.1 Simulation Setup

In this subsection, we assess the performance of representative algorithms from each RAA category (implicit, explicit, and hybrid) under mobility scenarios. The algo-

rithms include Ideal, Strale[43], MinstrelHT[44], and MutFed[50]. Each algorithm represents a different category. Minstrel-HT and STRALE are implicit feedback algorithms, with Minstrel-HT being a commonly used algorithm implemented in the Linux kernel. It aimed to select a sampling rate resulting in the highest throughput and probability of successfully delivering frames. On the other hand, STRALE could serve as an extension algorithm to Minstrel-HT, enhancing its performance by adapting the number of MPDUs in aggregated frames without reducing the transmission rate. To our knowledge, STRALE had not been tested in a dense environment. Among the few hybrid RAAs in the literature, MutFed was the most generic algorithm compared to the other hybrid algorithms proposed for specific scenarios. The idea behind MutFed was promising since it could distinguish the reason for packet loss and relied on SINR. These algorithms offer a comprehensive understanding of RAAs as they rely on various metrics and approaches commonly used in the literature. We used the NS-3 network simulator to compare and evaluate the performance of the algorithms. Ideal and Minstrel-HT algorithms were already implemented in the simulator. We modified the original implementation of the Ideal algorithm in the simulator to ensure a fair comparison among the selected algorithms. We included feedback in the reserved bits of the BlockAck[100], which was sent back to the transmitter on the same communication channel. We assumed all nodes had only one communication channel for data and control traffic exchange.

We implemented Strale and MutFed algorithms and selected specific metrics to evaluate their performance, including throughput, selected MCS values, FLR based on MPDUs, and FLR based on A-MPDUs.

- Throughput: provides a global view of the achieved performance and is calculated at the physical layer.
- Selected MCS index: gives insight into the different choices made by RAAs and their impact on performance.
- FLR (MPDU): provides an overview of overall lost MPDUs using the BlockAck information.
- FLR (A-MPDU): relies on the number of times the transmitter needed to retransmit the whole A-MPDU frame.

Table 3.1 presents the different modulation and coding schemes of IEEE 802.11ac.

We evaluated all of these performance metrics based on the number of nodes in the network. To achieve this, we increased the number of nodes while keeping the deployment surface constant. This node increase resulted in higher traffic load and interference levels, allowing us to assess how the algorithms behaved under increased interference conditions.

We evaluated the algorithms in three different scenarios:

- Interference-free network: highlights the impact of link degradation on RAAs due to mobility. In this scenario, a mobile node is moving away from a static access point. This allowed us to evaluate the efficiency of RAAs in adapting the rate according to RSSI values without interference.
- Infrastructure network: represents the most commonly used deployment mode. In this scenario, we evaluate the behavior of RAAs in a standard deployment with one access point through which all network traffic needs to pass to be relayed to a wired network. There is only one receiver, and the SINR values

Table 3.1: Modulation and coding schemes.

MCS index	Modulation Type	Coding Rate
0	BPSK	1/2
1	QPSK	1/2
2	QPSK	3/4
3	16-QAM	1/2
4	16-QAM	3/4
5	64-QAM	2/3
6	64-QAM	3/4
7	64-QAM	5/6
8	256-QAM	3/4
9	256-QAM	5/6

for each link with the other mobile nodes in the network vary based on their mobility patterns.

- Ad Hoc network: the most complex deployment in terms of interference where the receivers are randomly spread throughout the network, and interference levels are highly unstable due to mobility. All nodes need to adapt their rates depending on the channel conditions and network dynamics, such as node density and mobility.

In the infrastructure and ad hoc scenarios, direct connectivity was ensured for the duration of the simulation to avoid routing protocol impact. This allowed us to concentrate on interference and mobility impacts on RAAs. Each simulation result presented is the mean value of 50 simulations, with the bars in the graphs representing the standard deviation. Table 3.2 summarizes the network parameters of the simulation. A simulation duration of 30 seconds was sufficient to ensure randomness in the movement in the nodes within the square boundaries and for the selected algorithms to converge. We used Log-distance and Nakagami-m loss models to make the simulations more realistic regarding link quality and stability. We preferred UDP to TCP for traffic generation to avoid TCP overhead and its rate adaptation. As for packet size and mobility speed, we did not study their impacts, and the chosen values were representative of average to big-sized frames and relatively fast-moving nodes.

3.2.2 Scenario 1 - Interference Free Network

In this scenario, we consider a simple network configuration with only one communication link. One stationary AP and one mobile station moved away from the AP at a speed of 6 m/s, as stated in Table 3.2. The primary purpose of this scenario was to evaluate the efficiency of RAAs under the influence of mobility in an interference-free network.

We measured the throughput of the four algorithms as the station moved progressively away from the AP. As shown in Figure 3.1, although the results of all tested algorithms were similar, the Ideal and MutFed RAAs performed slightly better than the MinstrelHT and Strale algorithms. The Ideal algorithm detected channel changes faster than other algorithms and adapted the rate accordingly due to its fast feedback and decision-making capabilities. MutFed had a slightly lower throughput than Ideal due to the algorithm taking decisions every ten frames. As

Table 3.2: Summary of simulation parameters.

Parameter	Value
Simulation time	30 s
Runs	50
WLAN standard	IEEE.802.11ac
Path loss model	Log-distance
Fast fading loss model	Nakagami-m
Traffic	UDP
Packet size	1500 Bytes
Data Rate	60 Mbps (Application Layer)
Mobility model	Random Walk 2d Mobility Model
Mobility speed	6 m/s
Topology size	Square of boundaries (100, 100, -100, 100)

a result, when channel conditions deteriorated, the decision was not made immediately, resulting in frame losses. Minstrel-HT and STRALE achieved lower throughput as these algorithms take some time to lower the rate when needed, as their decision-making process relies on random probing, FLR in the case of MinstrelHT, and A-MPDU size adaptation in the case of Strale.

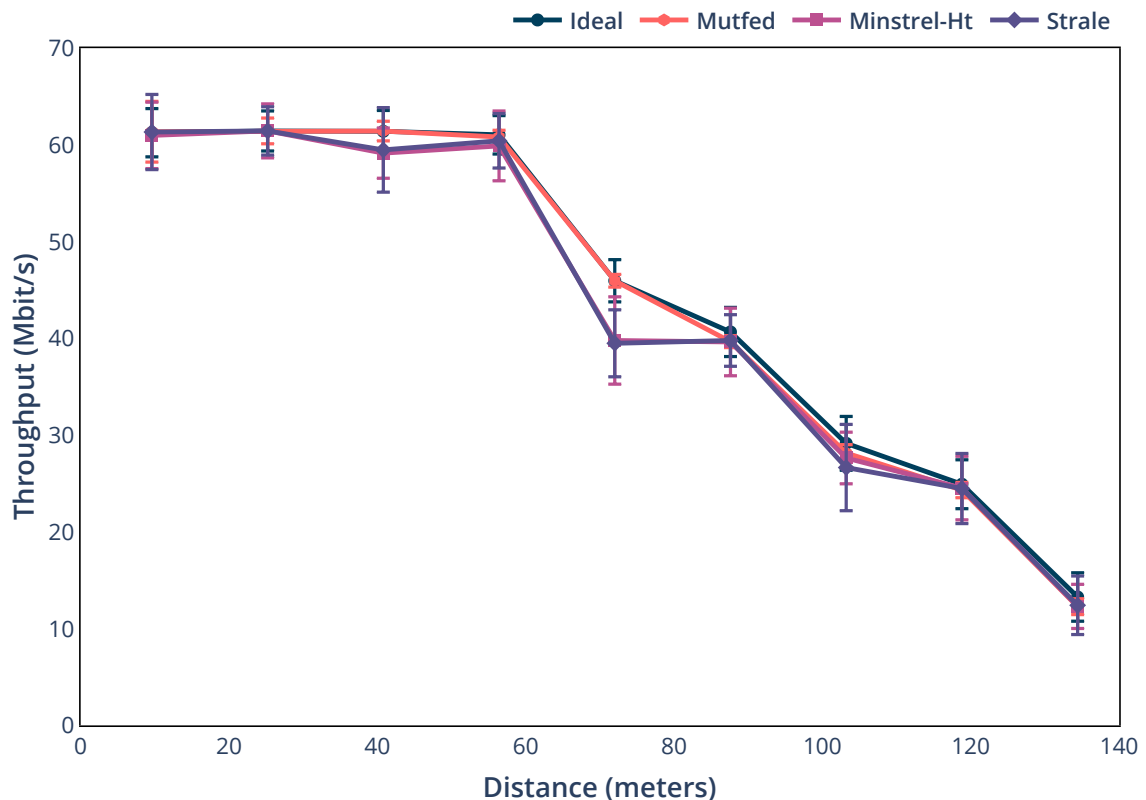


Figure 3.1: Throughput of a station moving away from an access point at a speed of 6 m/s.

3.2.3 Scenario 2 - Infrastructure Network

In this scenario, we consider an infrastructure mode network, with a stationary access point positioned at the center of a square field, and all stations moving ran-

domly while sending constant traffic of 60 Mb/s to the access point. The number of stations is gradually increased from 5 to 50.

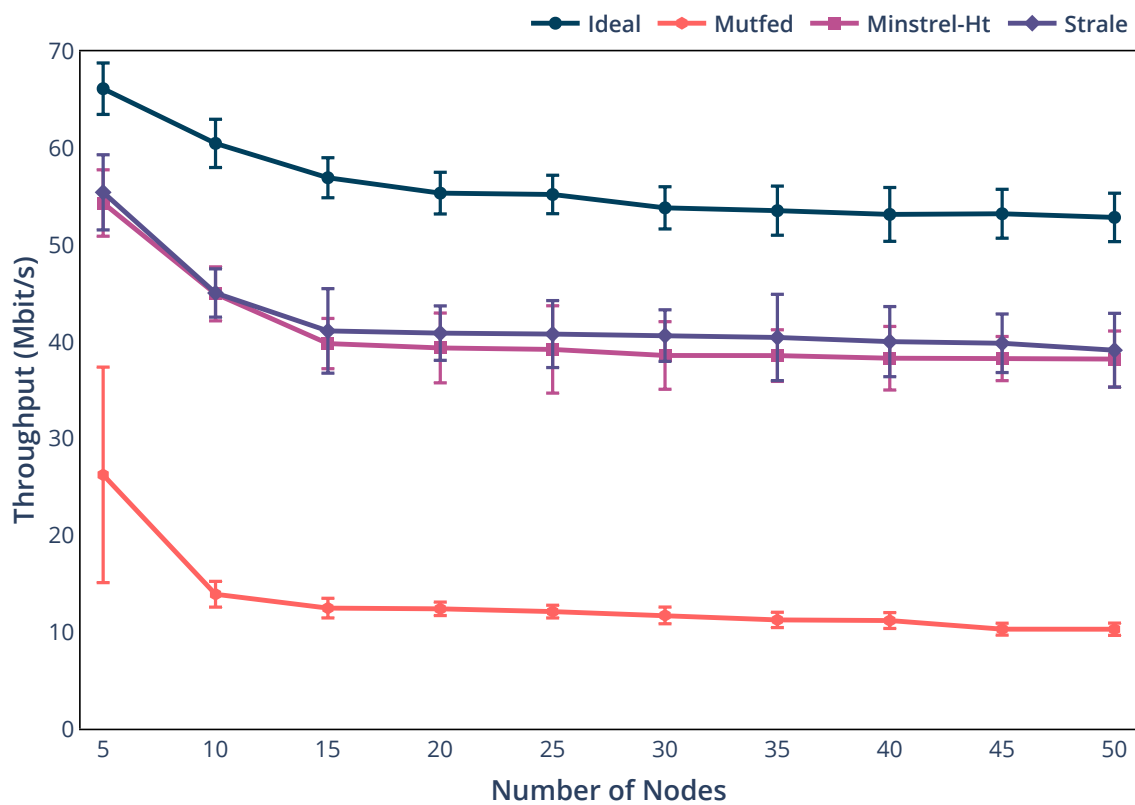


Figure 3.2: Overall throughput received at the access point.

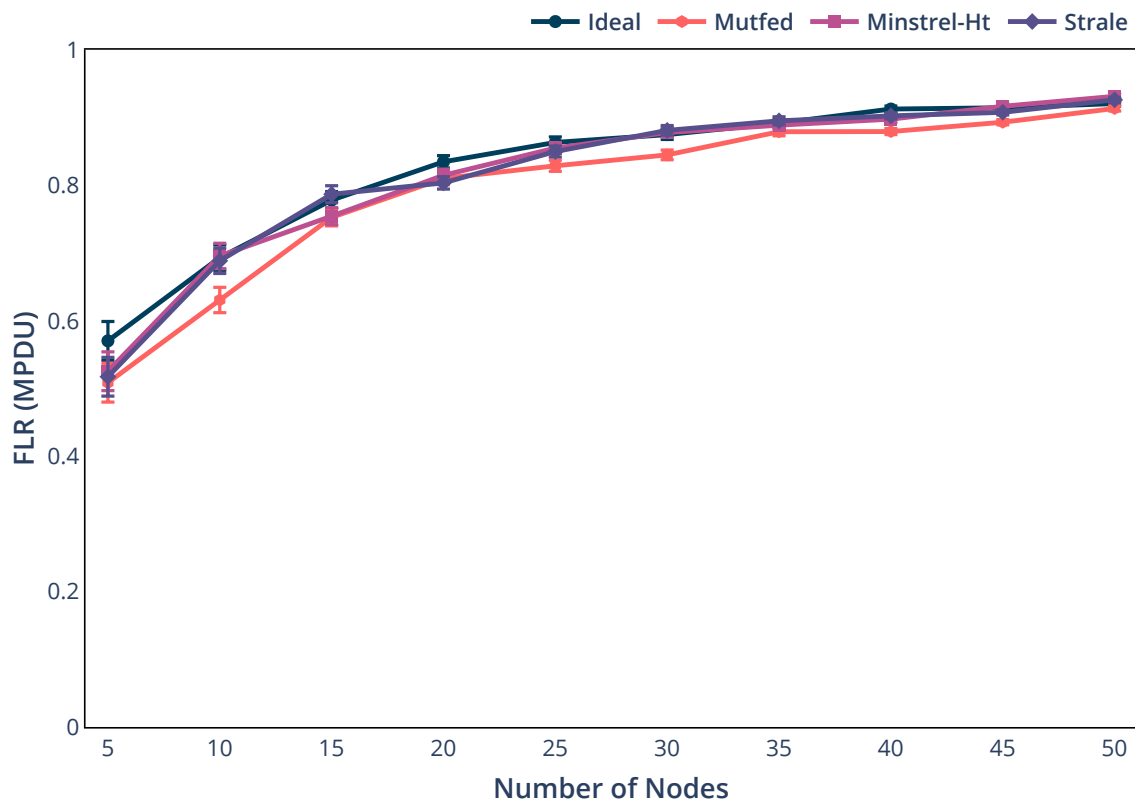


Figure 3.3: MAC Protocol Data Unit (MPDU) Frame Loss Ratio.

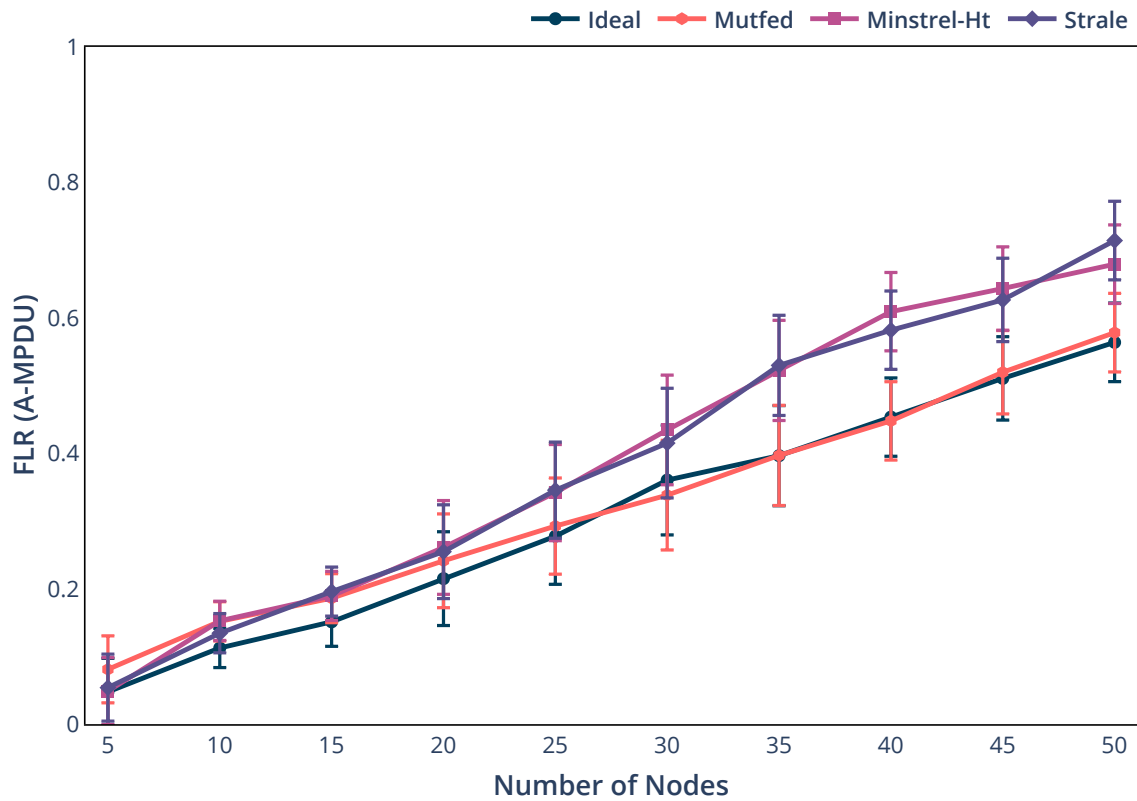


Figure 3.4: A-MPDU Frame Loss Ratio.

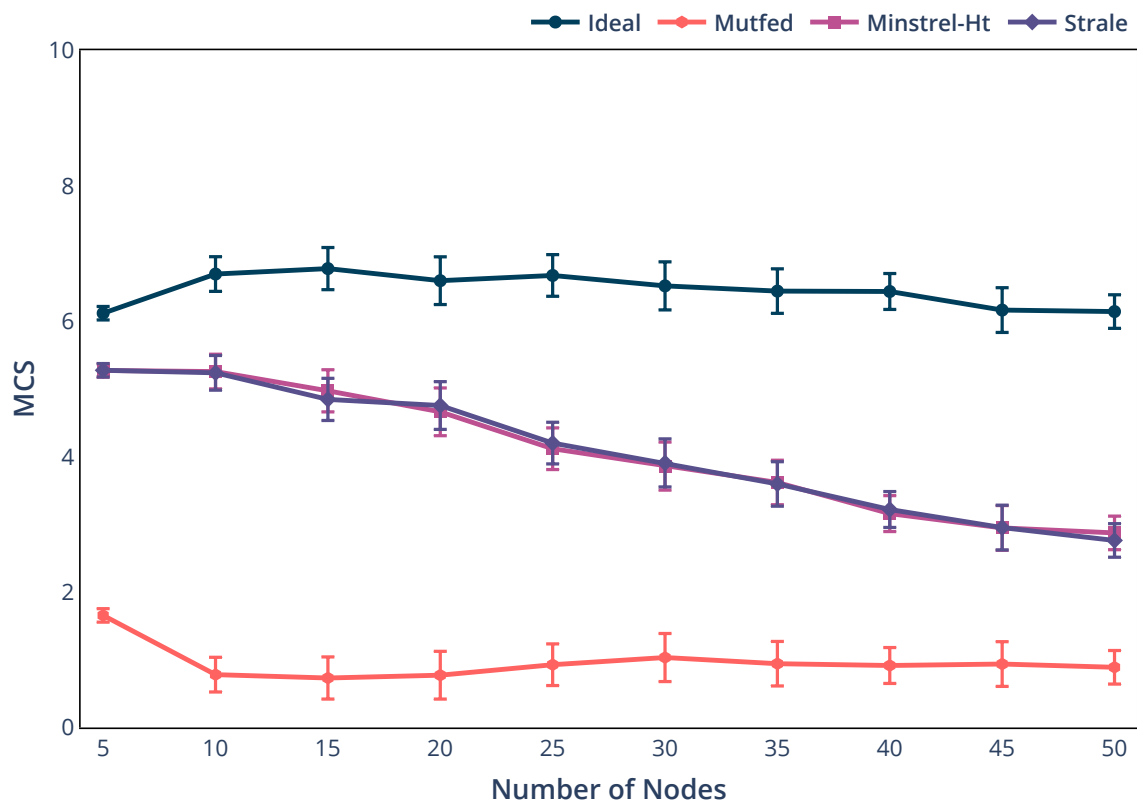


Figure 3.5: Mean value of the modulation and coding schemes (MCS) used by the nodes.

The physical throughput received at the access point is shown in Figure 3.2. The Ideal RAA performed better than the other algorithms due to its quick reaction to

changes in channel conditions. The two implicit feedback algorithms (Strale and Minstrel-HT) take longer to recover after channel conditions change. The Strale algorithm performed slightly better than Minstrel-HT because it seeks an optimal A-MPDU size instead of reducing the rate.

FLR based on MPDU losses and A-MPDU losses are shown in Figures 3.3 and 3.4, respectively. High MPDU FLR values are observed because all stations are within the transmission range of each other, increasing interference as node density increases. A-MPDU FLR is a false-positive rate increase in a fast-changing channel, which can occur when the transmitter increases the rate. However, the channel conditions deteriorate before sending the frame. Figure 3.5 shows the average MCS index values selected by the stations. Ideal RAA selected higher MCS index values with almost the same MPDU FLR and lower A-MPDU FLR, leading to higher achieved throughput. MutFed had the worst performance among the tested algorithms due to the frequent rate reduction caused by frame losses.

Figure 3.4 indicates that as node density increases (>20, for instance), causing the interference level to rise, the A-MPDU FLR gap between SINR-reliant and other RAAs also widens. This is because SINR values better reflect the channel condition.

3.2.4 Scenario 3 - Ad hoc Network

In this scenario, we consider an Ad hoc network with randomly placed nodes in a square field. The nodes move randomly at 6 m/s and change their direction every 3 seconds. Half of the nodes are traffic generators, and the other half are sink nodes, with a constant UDP traffic rate of 60 Mb/s flowing toward the sink nodes. We gradually increase the number of nodes from 5 to 50 to evaluate the performance of the rate adaptation algorithms under increasing node density and interference.

Figure 3.6 presents the overall average physical throughput achieved by all the sink nodes, while Figures 3.7 and 3.8 show the FLR based on MPDU losses and A-MPDU losses, respectively. Figure 3.9 shows the selected MCS values by the stations.

The stations using the Ideal RAA achieve the highest throughput among the studied algorithms, thanks to its fast adaptation to the changing channel conditions. However, in scenarios where frame losses occur, the lack of feedback from the receiver prevents the transmitter from adapting the rate, leading to more frame losses until the channel conditions improve. False-positive MCS rate decisions can also result in more A-MPDU frame losses in some cases, as seen in scenario 2.

We also observe that the average MCS values in Ad hoc mode are slightly lower than in infrastructure mode, which explains the lower throughput achieved in Ad hoc mode. Additionally, the FLR results suggest a high interference level in this scenario, leading to higher MPDU FLR values. The A-MPDU FLR results also show that using SINR values for rate adaptation improves performance in high interference-level scenarios.

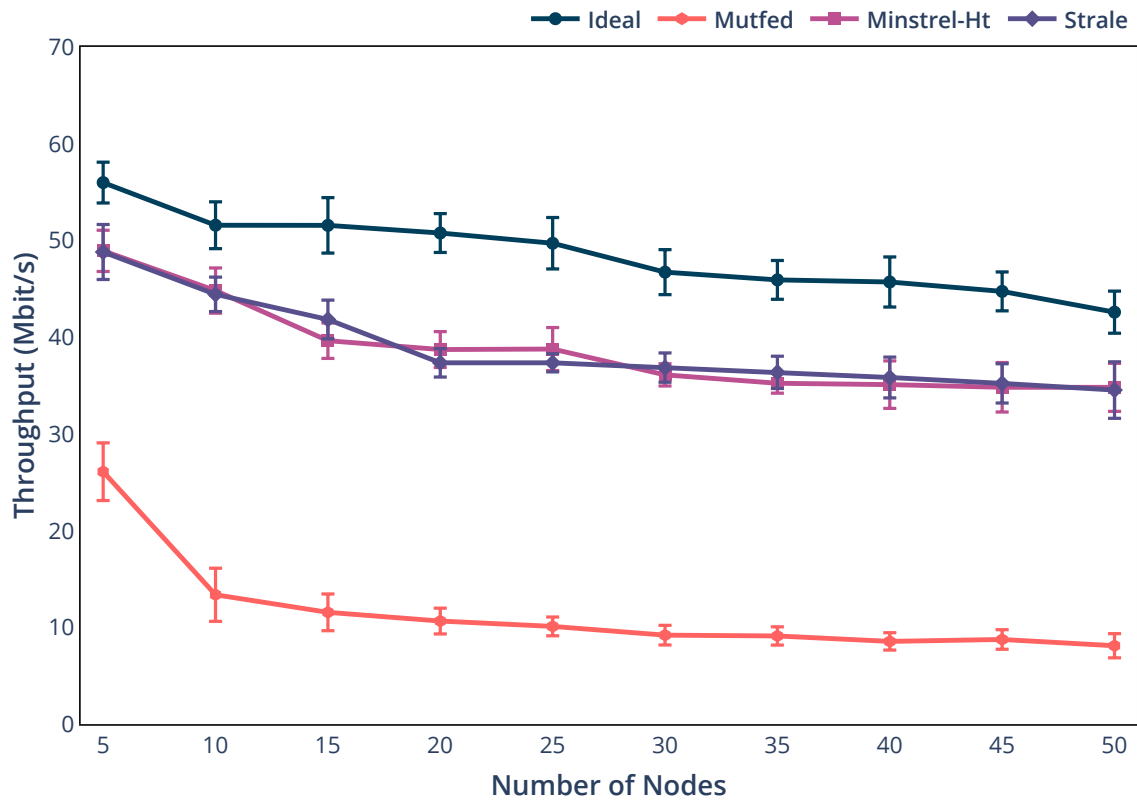


Figure 3.6: Average throughput of sink nodes in an Ad hoc network.

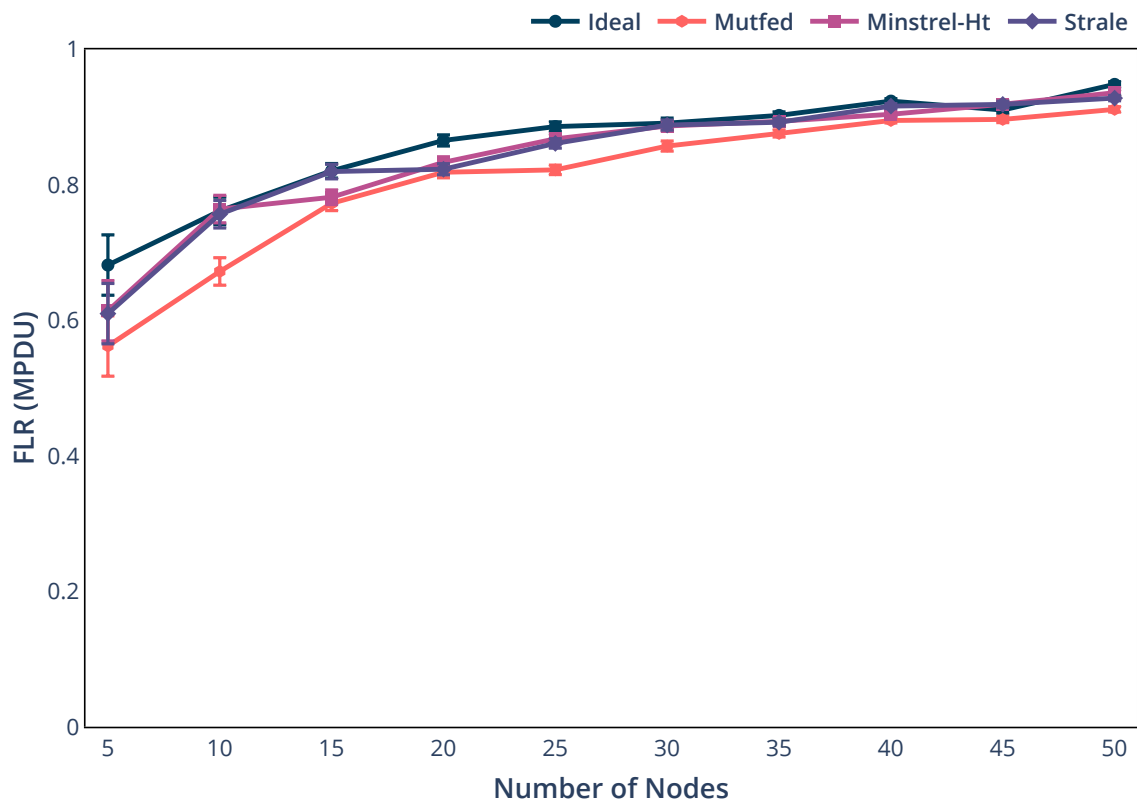


Figure 3.7: MPDU Frame Loss Ratio.

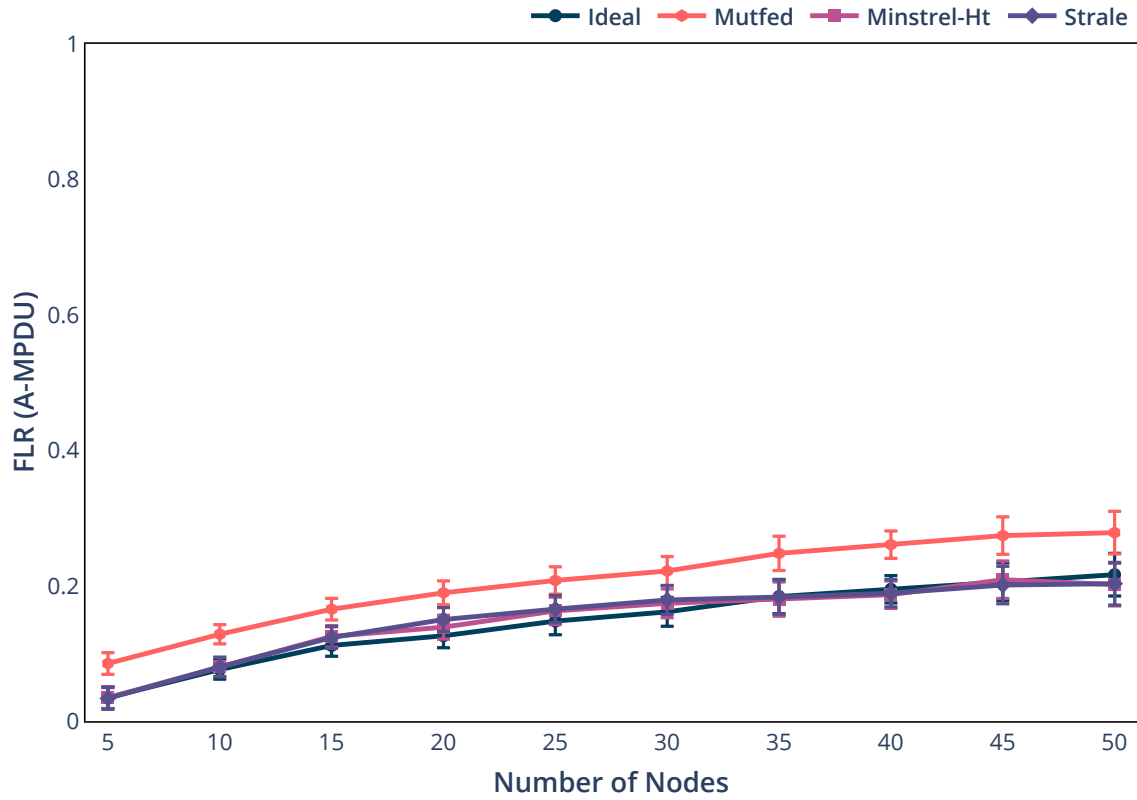


Figure 3.8: A-MPDU Frame Loss Ratio.

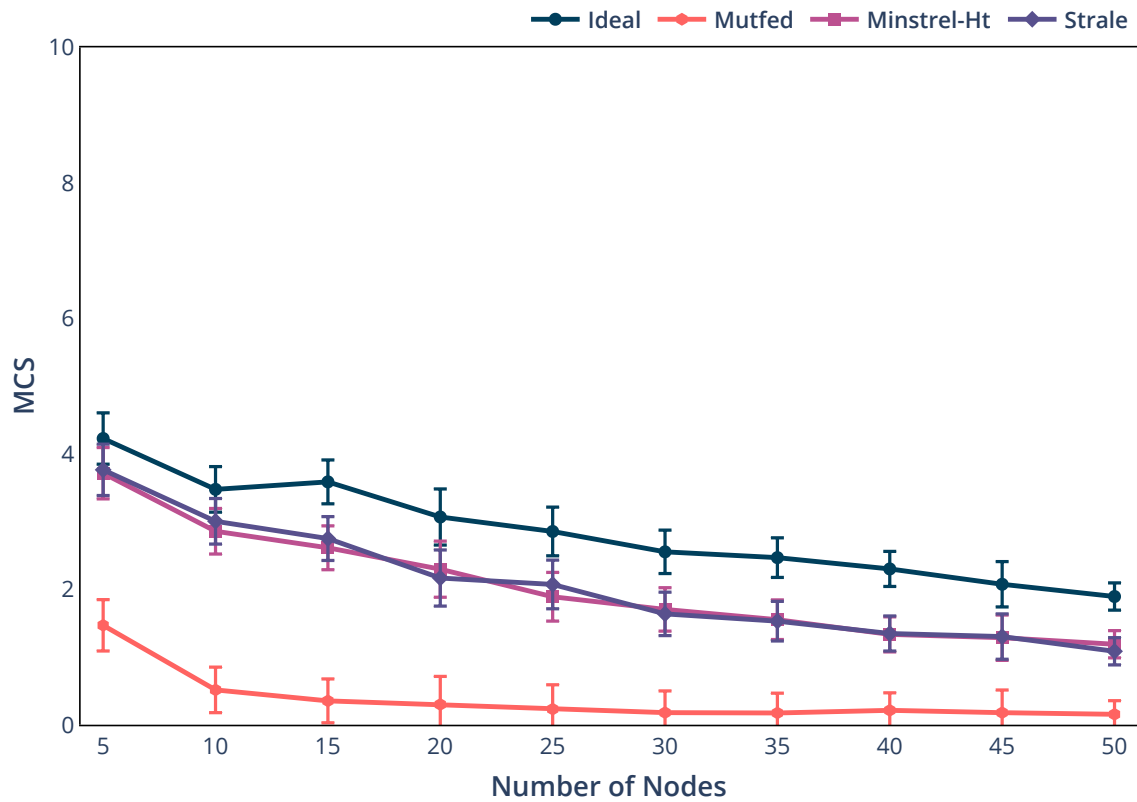


Figure 3.9: Mean value of the MCS selected by sender nodes.

It is important to note that in scenarios 2 and 3, all nodes are in in the communication range of each other, causing the interference level to increase with each new

sender node added to the network. As a result, we observe high FLR values in both scenarios. Specifically, the FLR based on MPDUs counted in BlockAck reaches over 80% in both scenarios when the number of nodes exceeds 20. The FLR based on A-MPDUs also increases gradually with the number of nodes in the network.

3.2.5 Lessons Learned

In this subsection, we share the lessons learned from our study of the existing rate adaptation algorithms and the simulation results.

Explicit Feedback

Most explicit feedback approaches rely on physical layer metrics, mainly SINR, as seen in Chapter 2. However, for such an approach to be used on real devices, several conditions must be met, such as having hardware that provides SINR values and a method for relaying the feedback to the transmitter that does not violate the standard.

Furthermore, the simulation results indicate that more than relying solely on SINR values for decision-making may be required. In some cases, the lack of feedback may result in multiple frame losses until the channel quality is suitable for the current rate. One possible solution to this issue may be implemented on the transmitter side.

Although explicit feedback algorithms, such as Ideal RAA, outperformed other algorithms in a dense environment in both ad hoc and infrastructure modes, much work remains to minimize FLR and achieve better overall performance. Combining the current explicit approach with implicit approaches, such as changing A-MPDU size, which results in lower channel occupancy, and collecting statistical information at the transmitter that aids in the decision-making of future transmissions, may be considered to reduce FLR.

Implicit Feedback

The implicit feedback algorithms commonly use frame loss ratio and probing rates different from the selected rate. However, caution must be exercised when using random probing rates due to the risk of excessive frame losses. This may result in slow convergence of the algorithm towards the optimal rate.

Although implicit feedback algorithms have shown promising results in a mobile, collision-free environment, their performance has not been found to be superior to other categories. To improve their performance, it may be useful to detect the degree of mobility and investigate its direct effects on frame transmission results. In dense environments, it is essential for RAAs to accurately estimate the cause of frame loss and rely on different metrics to make more precise rate decisions. Additionally, a rate-changing method should be implemented to increase the rate when channel conditions improve and decrease it when they deteriorate. However, achieving this is a challenging task, as the algorithm needs to have real-time estimates of conditions such as SINR.

Hybrid Approach

The simulation results indicated that decreasing the rate after failed transmissions in a dense environment can result in longer transmission times, reduced throughput, and increased channel occupation time, which affects all nodes in the network. One

possible solution to this problem is to use a sliding window approach that can predict future channel conditions and adjust transmission accordingly.

Designing an efficient hybrid approach has proven to be a challenging task, with few existing studies focused on it. The simulation results showed that the Mufed hybrid algorithm performed worse than other algorithms, mainly due to its method of decreasing the bit rate, which leads to decreased throughput, increased interference, and frame loss. Ideally, a hybrid algorithm could be built on top of an explicit algorithm by incorporating additional metrics, such as collecting statistics on frame loss ratio, to adapt the number of MPDUs in an A-MPDU.

Based on our performance evaluation results, we decided to use the modified ideal rate adaptation algorithm as the RAA in the rest of our work since it outperformed the rest of the algorithms in the literature. This contribution was published in a journal paper [22].

An in-depth understanding of rate adaptation algorithms offers valuable insights into the performance and adaptability of Wi-Fi networks in different network conditions. This knowledge can help optimize Wi-Fi networks for specific applications and use cases, and drive the deployment of Wi-Fi networks in accordance with application requirements by considering MCS and SINR requirements. In the following section, we will focus on the deployment of Wi-Fi networks using MIMO and beamforming in the context of smart farming, taking into account the specific requirements of the application.

3.3 Wi-Fi Deployment with Beamforming

Our study focuses on the comparison of the performance of WLANs between MIMO and SISO systems. We investigate the impact of beamforming in terms of coverage and capacity. Deploying Wi-Fi networks involves a range of techniques and considerations, including equipment selection and access point placement. In this section, we discuss the different techniques for Wi-Fi deployment, our Wi-Fi deployment scenarios, and the factors involved in calculating the number of access points, taking into account coverage and capacity based on the application requirements.

3.3.1 Deployment Technique

Inspired by the deployment techniques used in cellular networks, mainly the grid approach such as in [101], we investigate the efficiency of a grid-based deployment with and without considering a target data rate. An access point has a coverage range R . The grid cell size is selected depending on R in a way that the coverage of the access point completely covers the grid cell, as shown in figure 3.10. This ensures complete radio coverage at any point inside the grid cell. Note that in real deployment, the coverage of the access points is not a perfect circle due to many factors, such as environmental factors and antenna radiation patterns.

The radius of coverage of an access point which determines the grid cell size is dynamic and depends on the target data rate of the application. Our deployment approach follows 4 steps:

- We determine a target SINR value based on the target data rate of the application.
- We calculate the coverage radius of the access point based on this target SINR value.

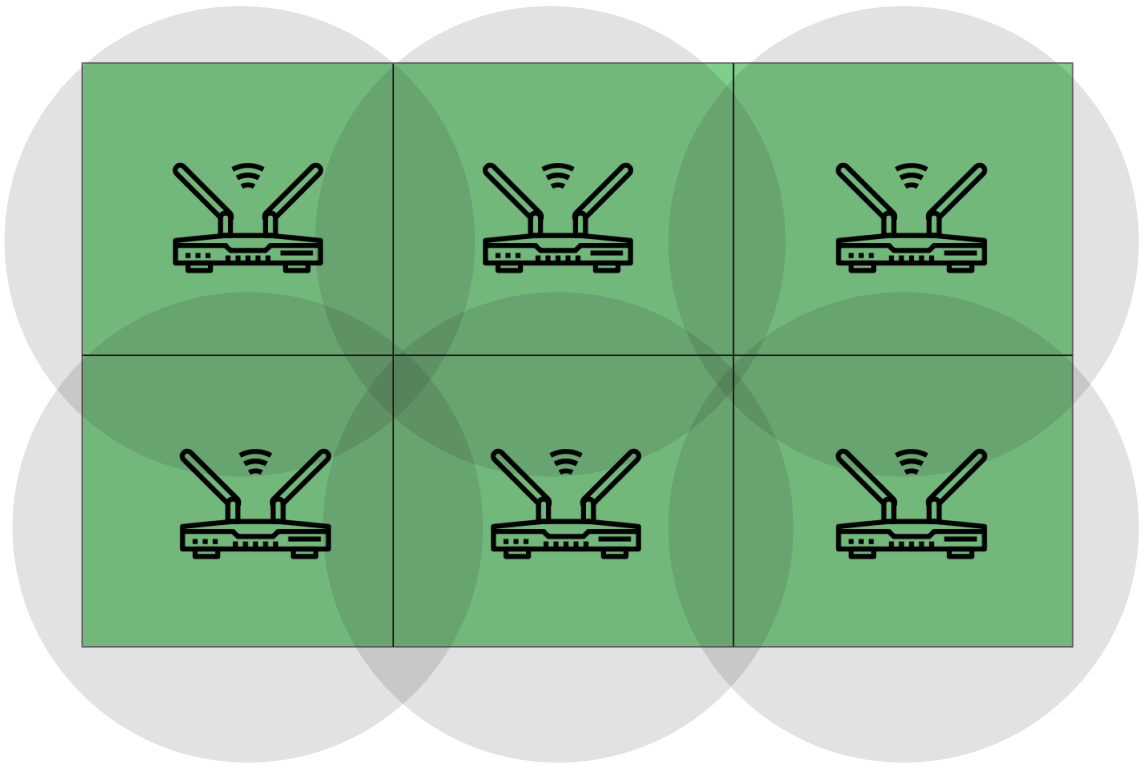


Figure 3.10: Grid Deployment.

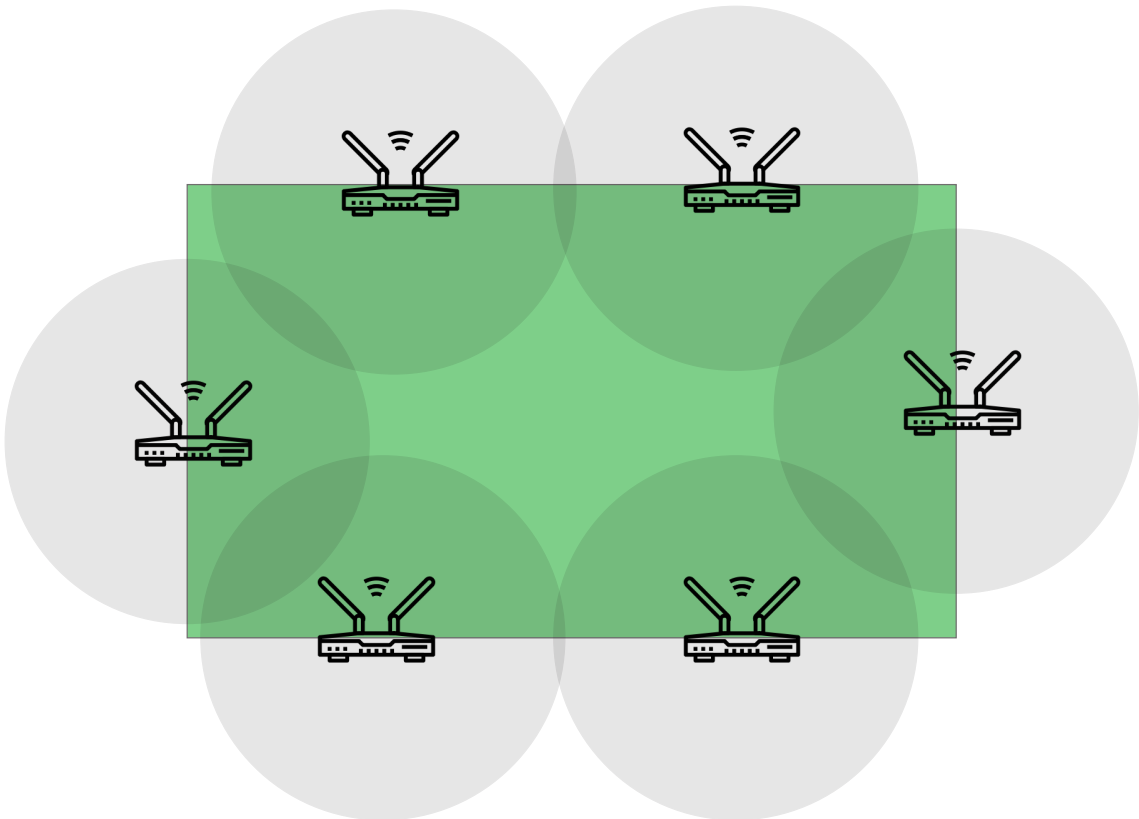


Figure 3.11: Deployment of the access points at the edge of the field.

- We calculate the grid cell size.
- We fill the area of interest with cells horizontally and vertically until the area of interest is completely covered.

Note that, in this study, we can install access points at the desired positions. In real life, this might not be possible due to obstacles. Nevertheless, we can easily include constraints for positioning access points depending on the field for a specific deployment.

3.3.2 Deployment Scenarios

In this study, we consider 3 different deployment scenarios: *data rate independent*, *data rate dependent*, and *edge deployment*.

Our first scenario covers the use case of sending at 1Mbits/s from any point in the field. We call this scenario *data rate independent deployment*. The aim is to deploy the necessary number of access points that ensures coverage of the whole area by applying a grid positioning.

In the second scenario, we consider a *data rate dependent deployment* where the required data rate is taken into consideration. In this scenario, we apply a grid positioning of the access points. The number of required access points is based on the range calculated in equation 2.6.

The installation of access points inside a field is not a simple task. It requires a power source and network cables for every access point. Additionally, placing the access points introduces additional obstacles in the field for mobile robots that are performing tasks that may dynamically change. Hence a special deployment technique is studied in the third scenario, which we call *edge deployment*, where APs are deployed at the edge of the area as shown in figure 3.11.

A human operator from its control station may decide at any time to monitor a the actions of a robot in real-time or intervene when a problem occurs: when a robot is stuck or when some faulty behavior is detected, for example. The quality of the video may be critical in some operations that require a clear vision of the environment. The maximum quality of the video can be determined during the design phase since it is related to the hardware being installed and the type of application. Thus, for each of the scenarios, two main use cases are studied considering an intervention with medium and high quality video requirements. We used the "CCTV Calculator" [102] to calculate the minimum data rate needed for high and medium quality video streaming. The video parameters used in the following sections are shown in table 3.3.

Parameter	Value
Resolution	1920 x 1080
Compression Type	MPEG4
Frame Rate	30
Data Rate - High Quality	95.7 Mbits/s
Data Rate - Medium Quality	25.4 Mbits/s

Table 3.3: Video parameters.

3.3.3 Capacity Aware Deployment

Using the MCS table 2.2, we can determine the minimum SINR achievable at the receiver, which allows sending at an MCS providing the data rate required by the application.

We developed a small graphical tool using Unity[103] to calculate the number of required access point and visualize their deployment as shown in figure 3.12. The tool takes into account transmit power, number of antennas and antenna gain, noise power, pathloss exponent, target SINR, field dimensions, overlapping percentage between access points, and whether to use beamforming or not.

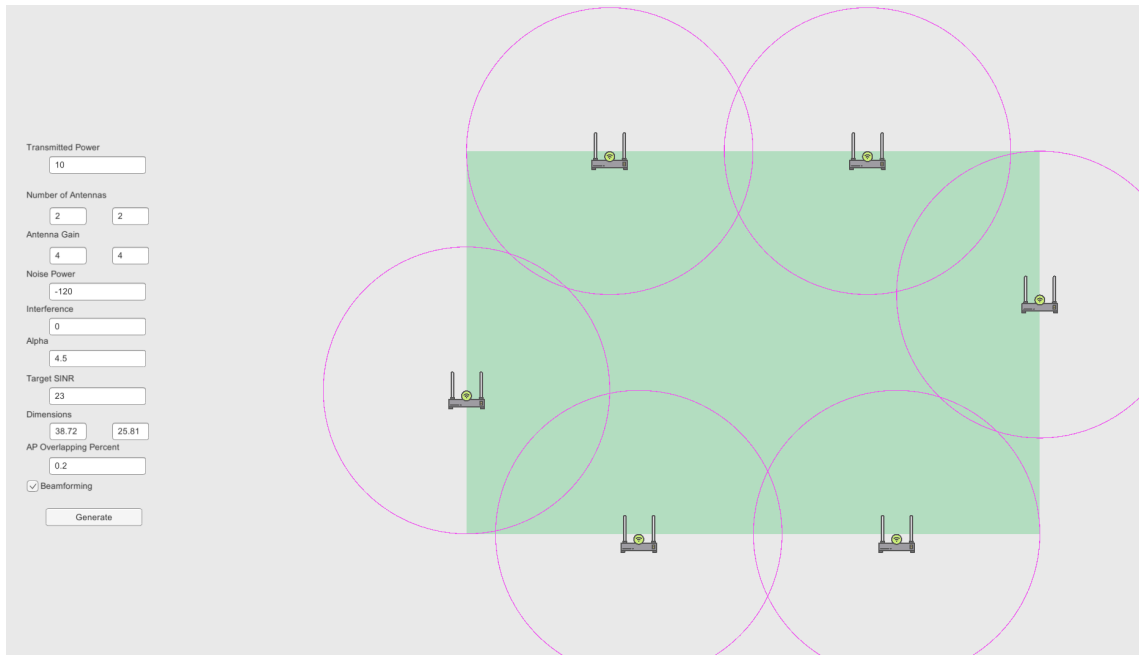


Figure 3.12: A graphical tool used to calculate the number of required access points and visualize the deployment

Two 1x1, one 2x2, and one 4x4 MIMO access points are needed to cover the area allowing a minimum data rate of 1Mbits/s . Our deployment strategy considers a minimum SINR of 5 at any point in the field, allowing robots to use "MCS index = 0".

For the two other scenarios, both are data rate dependent. We considered a minimum SINR value of 8 for the medium-quality use case and a minimum SINR of 21 for the high-quality use case. These SINR values allow robots to send a data rate that respects the application needs, 25.4 Mbits/s and 95.7 Mbits/s , respectively. Note that these SINR values are related to the use of a channel width of 40 MHz . Other values are required for other channel widths.

Using this information, and based on equation 2.6, we calculate the required number of access points for deployment. The required number of access points in the first scenario is found to be close in both the SISO and MIMO cases. Hence, no significant improvement in using beamforming when sending at low data rates: two SISO access points are required to cover the area, whereas only one access point with 2x2 or 4x4 MIMO antennas is sufficient to cover the same area.

The number of access points required for the medium-quality video use-case is shown in figure 3.13. Four 1x1 access points are required to cover the field, while only one 4x4 access point is sufficient to ensure complete coverage of the field and allow a robot at any point in the field to use $MCS = 1$, which is enough to send at

a data rate of 25.4 Mbits/s. The difference in the number of access points can be translated into a cheaper and less complex deployment.

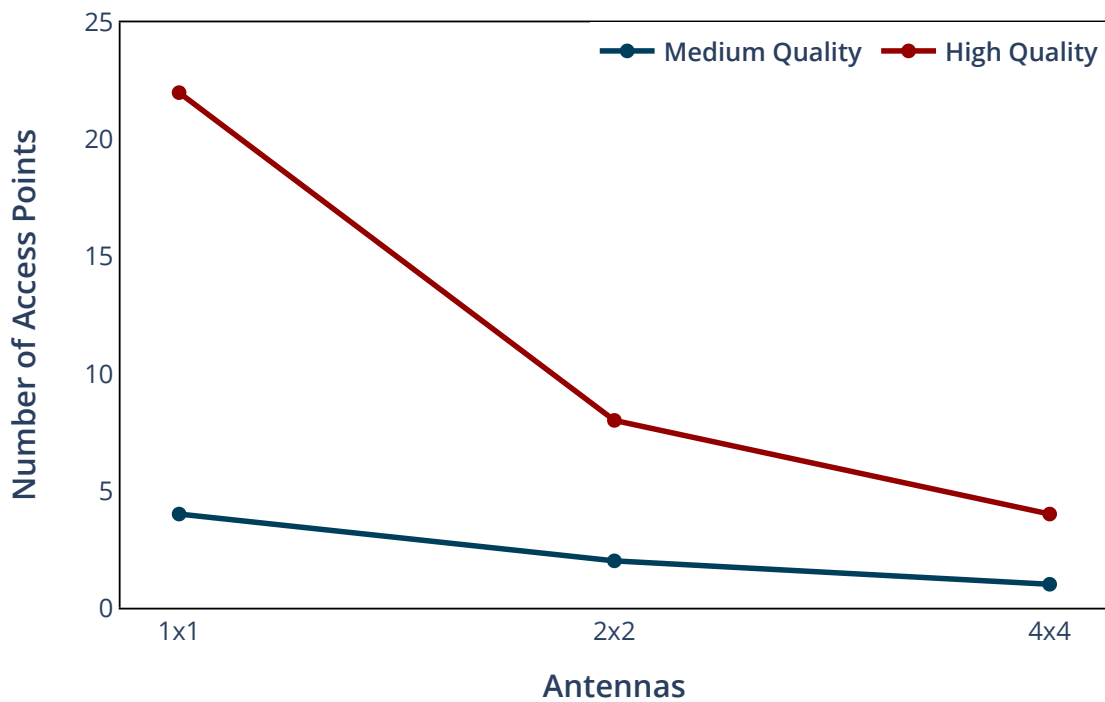


Figure 3.13: Grid deployment: number of access points inside the field for a medium and high quality video intervention.

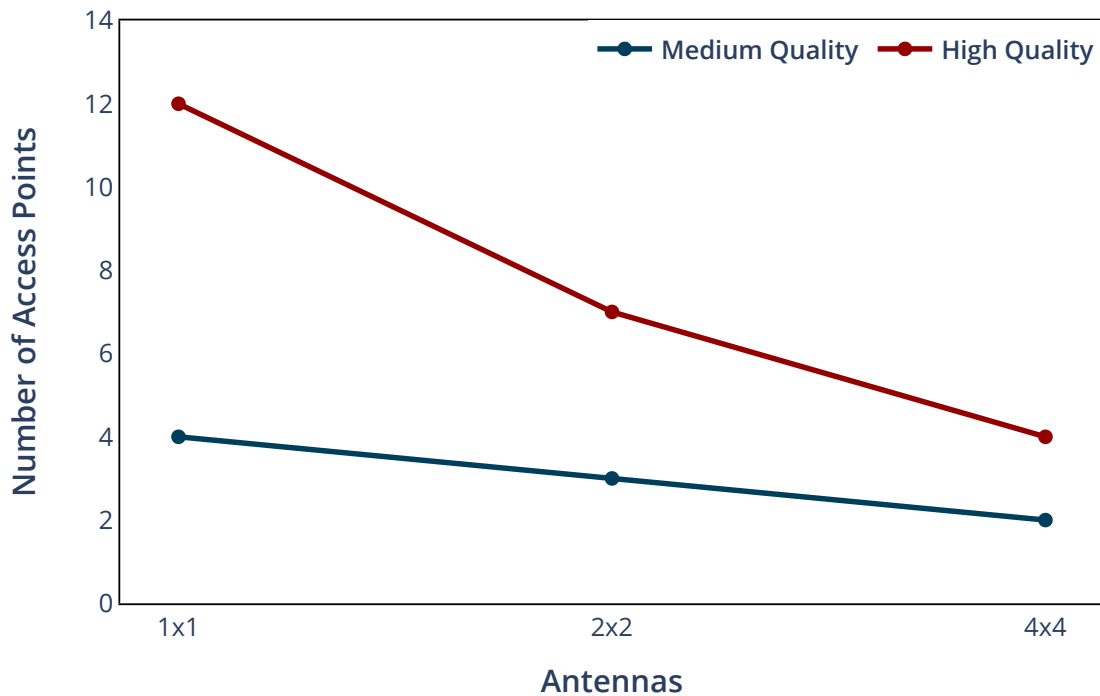


Figure 3.14: Edge deployment: number of access points inside the field for a medium and high quality video intervention.

In the third scenario, where APs are deployed on the edge of the field for a medium-quality video intervention requirement, the number of access points is shown

in figure 3.14. The access points cover the whole field when using SISO and MIMO technologies, with a slight difference in the required number of access points. The non-covered area is 0%, which means that even though the access points are deployed around the field, robots can send the required MCS from any point inside the field without needing a relay node. This is not the case when a high-quality video intervention is required. We can see a big difference in the number of access points required when using each technology. Eleven access points are required to cover the area for 1x1, while 12.63% of the total area is not covered. Seven access points are needed for the 2x2 MIMO system, with 3.63% of the area not covered. Four 4x4 MIMO access points are enough to maintain full coverage of the whole field.

In the following section, we will present the simulation results of the Wi-Fi networks deployment.

3.4 Performance Evaluation of Deployment Strategies

The number of access points for deployment is an essential criterion in respecting the required data rate. In real scenarios, a network includes multiple robots. Hence, a performance evaluation is necessary to help understand the impact of mobility and interference on the quality of communication.

In this section, we evaluate the deployment techniques outlined in 3.3 within the context of a specific application use case involving mobile robots in a Smart Farming scenario. We conduct our tests through simulation and derive an empirical model by offering a generalization of our approach for any field size and comparing it to simulation results.

3.4.1 Network Performance Evaluation

We conduct a performance study for each deployment scenario. We compare the achieved performance with data rate-independent and data rate-dependent deployments. In the data rate-independent deployment, we use the maximum coverage range of access points without considering the application or the required data rate for reliable video streaming. This means that the considered range only guarantees to send data at the minimum MCS value (MCS index = 0). The number of access points deployed is the same as the ones calculated in the first case (1 Mbits/s). In the case of data rate-dependent deployment, the access points coverage is calculated according to equation 2.6, considering the minimum SINR value to send at the required data rate.

Extension of the available Wi-Fi module in NS-3

Our evaluation procedure is based on a simulation study. We used NS-3 simulator which offers a detailed module for Wi-Fi but lacks important aspects needed in our study. In this section, we discuss the features we implemented in the simulator for the sake of our study, namely the beamforming procedure, handover, and MCS negotiation.

Beamforming is not implemented in the current version of the NS-3 simulator and, according to our knowledge, is not on the timetable of the designers yet. Thus, we decided to implement a beamforming behavior in the simulator. In a single

data exchange, a transmitter called the *beamformer* starts to measure the quality of the channel used to communicate with the receiver, known as the *beamformee*. The result of the measurement, which is called the sounding procedure, is used to direct the energy toward the receiver. In a Single User MIMO (SU-MIMO), the main lobe represents the total energy transmitted in the direction of the receiver while no energy (depending on the accuracy of the hardware) is transmitted in other directions. This main lobe has a beamwidth that depends on the type of antenna used.

What follows is a brief summary of the features that we added to the NS-3 Simulator:

- A beamforming transmission behavior. The transmission consists of a main lobe of a triangular shape directed toward the receiver with no energy in other directions.
- A handover procedure allowing the robots to switch to the access point providing a better SINR than the currently associated access point.
- Replace the beamforming sounding procedure by the airtime it takes, which is around 500 microseconds[37].
- Integration and use of the Modified Ideal Rate Adaptation Algorithm[22].

Description of Simulation Scenarios

We consider that robots are sending video streams for 30% of the time. In the remaining 70%, robots send 1 Mbits/s of data. This proportion is chosen to emulate a real intervention. Indeed, this emulates the event where the robot is taken over by a human operator taking into consideration the time needed to remotely solve the issue. Multiple robots are deployed in an outdoor environment. Each robot is deployed at a random position and follows a random path. In a real-life scenario, a robot will have a predefined trajectory. Knowing the trajectories in advance would help optimize further the positioning and the number of deployed APs. However, we adopted random trajectories as a worst-case scenario.

Each scenario is repeated 40 times with different initial positions and random paths. The number of robots deployed is increased gradually from 4 to 20 in each scenario. Note that, in a typical scenario, only a limited number of robots are deployed in one field (1 to 4 robots). In this study, we focus on the challenges that will occur due to interference and channel overload in case more robots are deployed.

We consider a wired link between the APs and the control station. Multiple simulations of different scenarios were carried out while varying the number of mobile robots. The simulation parameters are shown in Table 3.4.

In the following sub-sections, we investigate the impact of the number of APs and their configurations on throughput and delay depending on the number of users in the network. The investigation includes using SISO and MIMO with a different number of antennas. The scenarios are classified according to different application requirements (high and medium-quality video streaming) and different field sizes. We evaluate two access points deployment strategies for each scenario, one considering the application's data rate requirement and the other being a classical deployment (which does not take into account application requirements). The number of access points in each simulation scenario is based on the analytical results of previous section.

Parameter	Value
Simulation Time	60 seconds
Runs	40
WLAN Standard	IEEE.802.11ac
Path Loss Exponent	2.5
Antenna Gain	6 dbi
Channel Width	40 MHz
Noise Level	-120 dBm
Reference Loss	46.6 dB
Fading Factor	random(0, 2dB)
Mobility Model	Random Walk 2d Mobility Model
Mobility	6 m/s
Topology Size	1km ²

Table 3.4: Simulation parameters table for the capacity-aware deployment.

Medium Quality Video Streaming Application

Figure 3.15 shows the throughput obtained in a data rate independent deployment scenario. The throughput obtained in SISO is higher than that of MIMO with beamforming. Two main reasons explain this result. First, the number of APs is larger in the SISO case (2 APs), which leads to fewer packet losses and more channel access time for each robot, as depicted in figure 3.16. Second, a data rate of 25 Mbits/s requires a low SINR value at the receiver, as indicated in table 2.2. Thus, robots maintain connectivity most of the time since the MCS requirement is low.

The significant difference in the throughput between the 2x2 and 4x4 antenna APs is due to the beamforming gain. When using the 4x4 system, we have a gain of approximately 6 dBs (equation 2.1), while in the 2x2 system, the gain is around 3 dBs. This gain difference has a direct impact on the power received. This means that data can be received at a higher SINR when using a 4x4 system for a given relative position between the transmitter and the receiver. Thus, a higher MCS value can be selected at certain locations, which explains the better throughput obtained in the 4x4 system case.

Note the impact of interference on the performance of the network when the number of robots increases. Additional robots increase the traffic load and thus increase the interference rate and collision probabilities. Thus, a decrease in throughput occurs when the number of robots increases.

Figure 3.17 shows the throughput obtained in a data rate dependent deployment scenario. The 1x1 system provides better throughput than the 2x2 and 4x4 systems. This explains the need for multiple APs to handle the increased density of robots. The same applies to the edge deployment scenario due to the full coverage and the close number of APs. The throughput results of the edge deployment scenario are shown in figure 3.19. However, the difference in the number of access points between the two scenarios is seen in the lower delay for the 2x2 and 4x4 systems as shown in figures 3.18 and 3.20 due to more channel access time combined with the beamforming gain. Beamforming gain leads to receiving stronger signals allowing the use of higher MCS values.

A larger number of access points allows a better distribution of the robots among them. Results show that the beamforming gain may not have a significant added value in applications with medium data rate requirements compared to SISO sys-

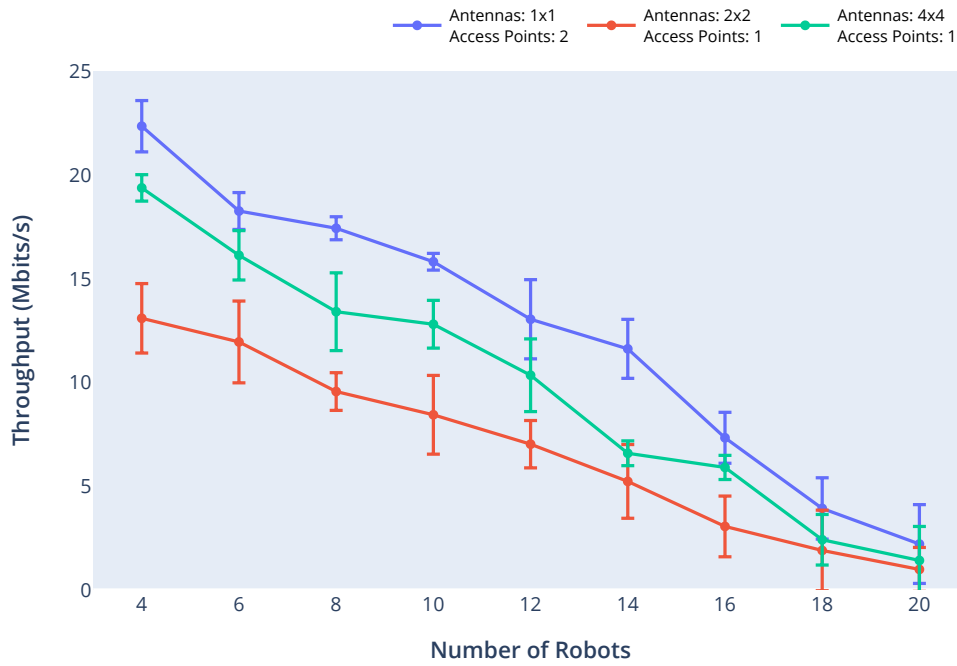


Figure 3.15: Scenario 1 (data rate independent deployment): Throughput and number of access points needed for a medium quality stream.

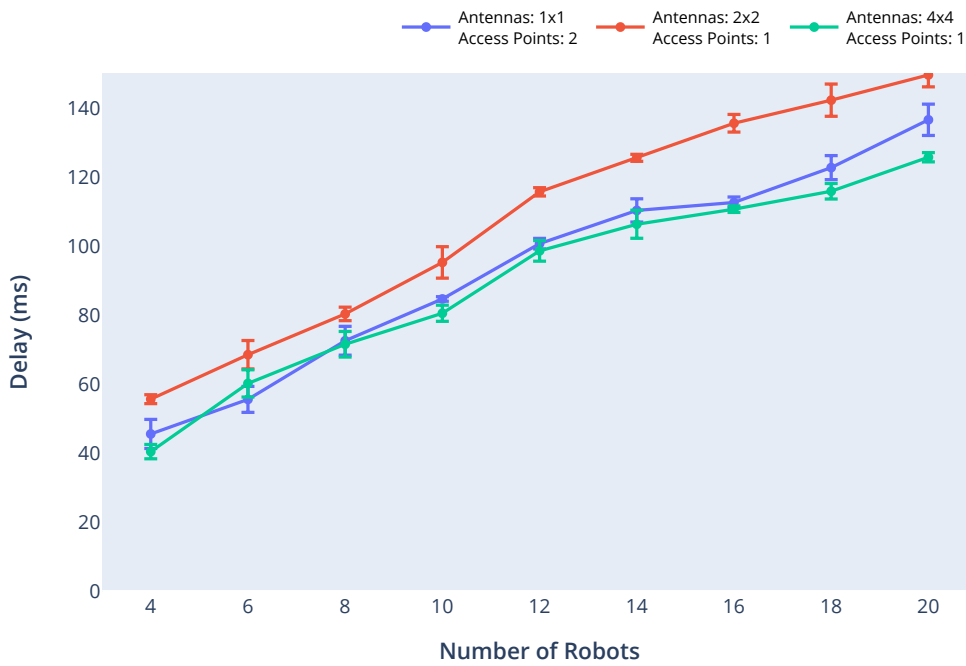


Figure 3.16: Scenario 1 (data rate independent deployment): End-to-End delay for a medium quality stream.

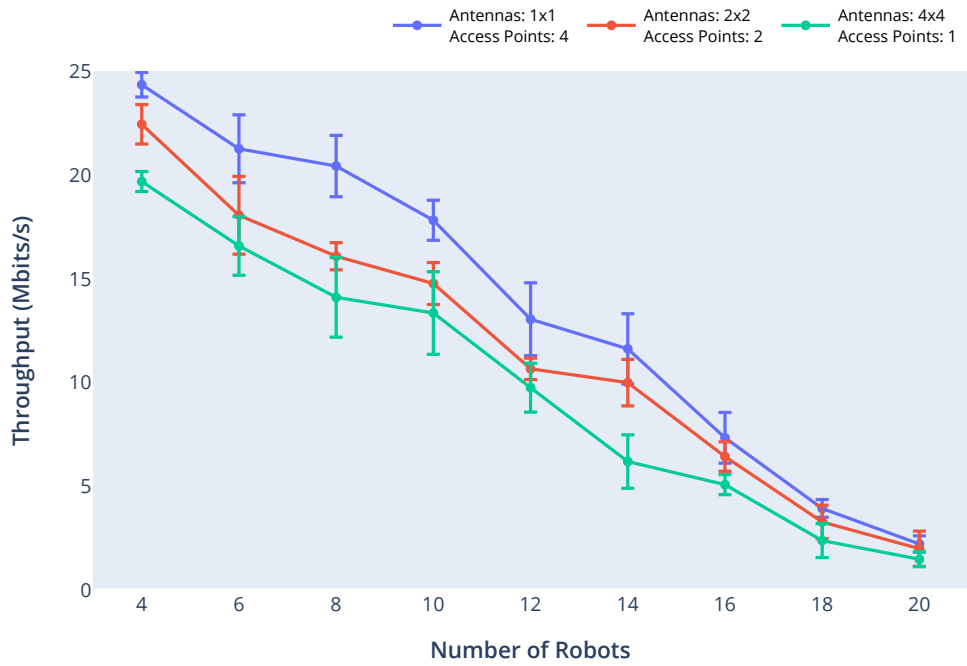


Figure 3.17: Scenario 2 (data rate dependent deployment): Throughput and number of access points needed for a medium quality stream.

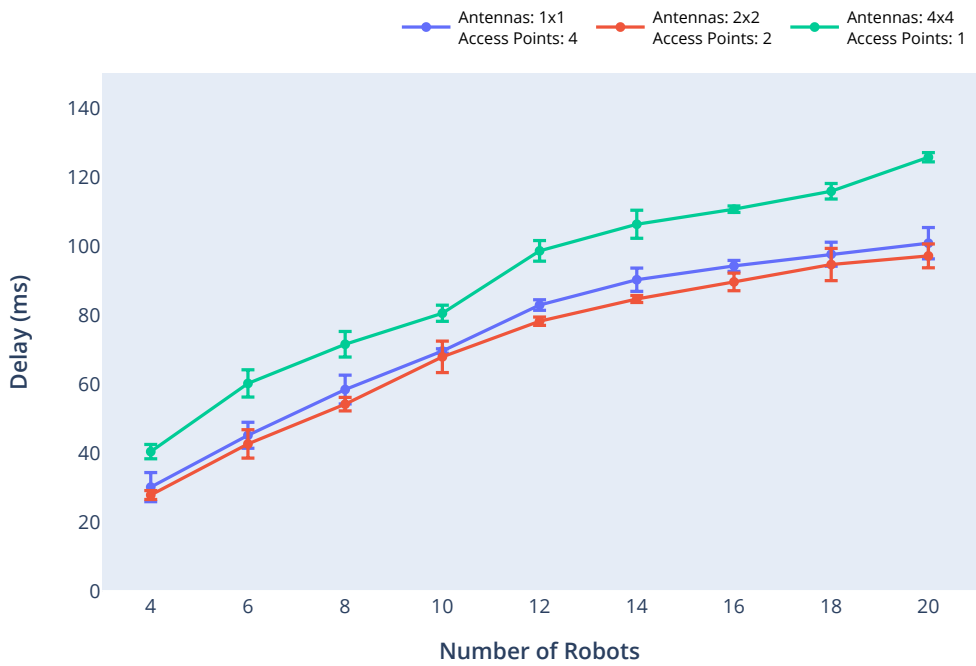


Figure 3.18: Scenario 2 (data rate dependent deployment): End-to-End delay for a medium quality stream.

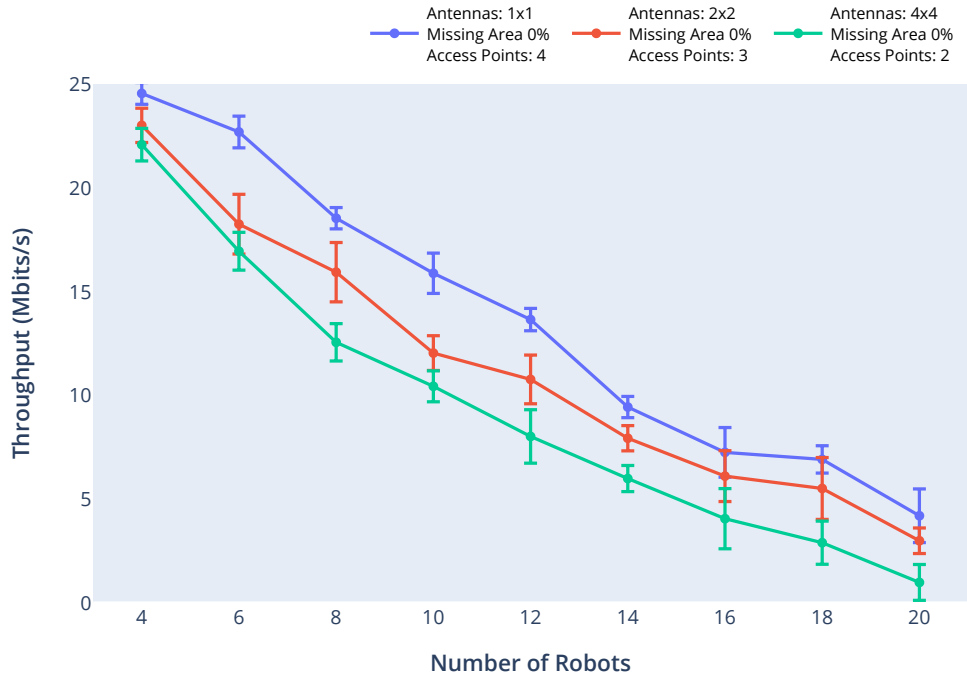


Figure 3.19: Scenario 3 (edge deployment): Throughput and number of access points for medium quality stream.

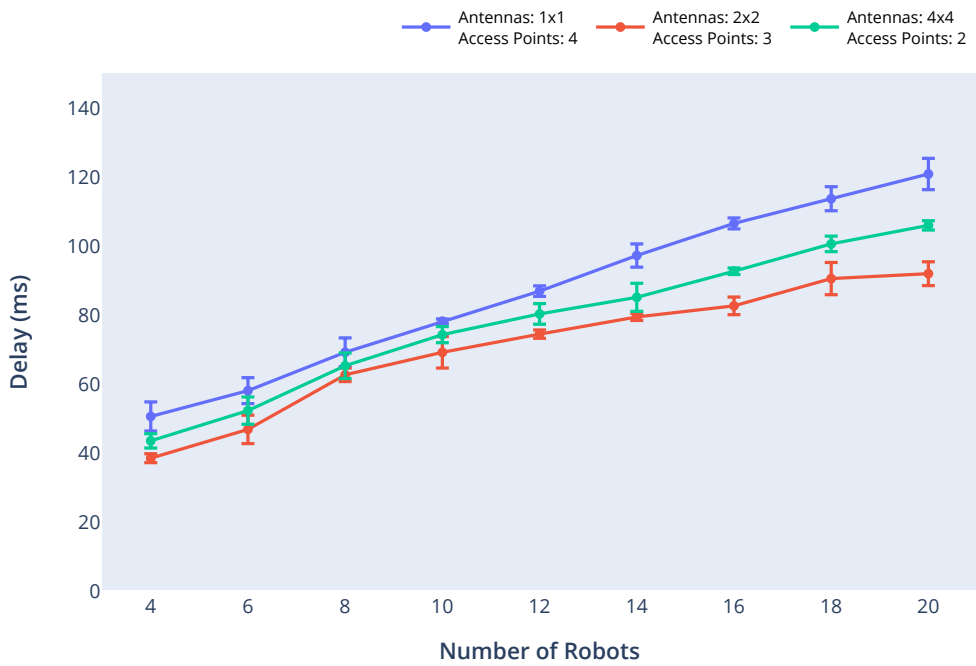


Figure 3.20: Scenario 3 (edge deployment): End-to-End delay for a medium quality stream.

tems. Furthermore, The results show no significant difference in the number of deployed APs between different systems, as seen in figure 3.13. Thus, A cost-capacity study should be made to further aid in the choice of the technology of the deployed APs.

High Quality Video Streaming Application

This section presents results for the use-case of high quality video streaming application requirements. Figure 3.21 shows the throughput obtained in a data rate independent deployment scenario. The robots are far from reaching the required data rate since they cannot use the target MCS in many parts of the deployment field. Although the required data rate cannot always be achieved in such deployment, the impact of beamforming and its gain is notable as it extends the range in which the MCS requirement can be satisfied. This explains the better throughput achieved by the 4x4 and 2x2 systems over the 1x1 system.

The impact of beamforming on the end-to-end delay is shown in figure 3.22. 2x2 and 4x4 systems have the same number of deployed access points but the 4x4 system achieves a lower delay due to the selection of higher MCS values.

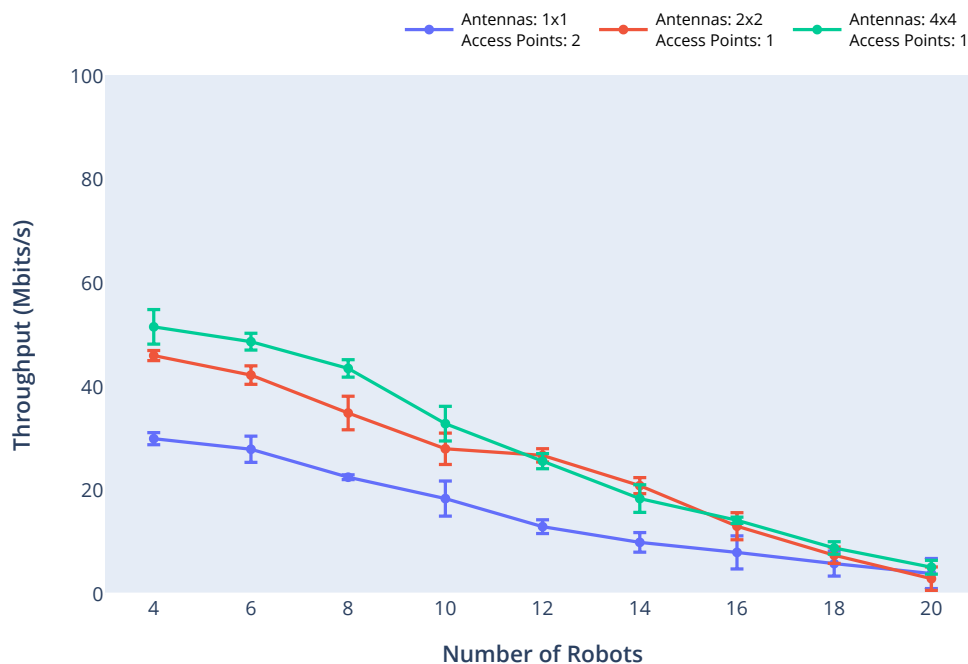


Figure 3.21: Scenario 1 (data rate independent deployment): Throughput and number of access points needed when sending a high-quality stream.

In a data rate dependent deployment (scenario 2), the field is fully covered by the access points complying with the SINR requirement, hence, as shown in throughput results of figure 3.23. With the presence of a few robots, the performance of SISO and MIMO is close due to having a large number of access points relative to the number of robots and an SINR-dependent coverage zone of each access point. As the number of robots increases, We observe a difference in the obtained throughput due to the difference in the number of access points. More access points mean fewer robots are associated with each access point. Thus, better load balancing between

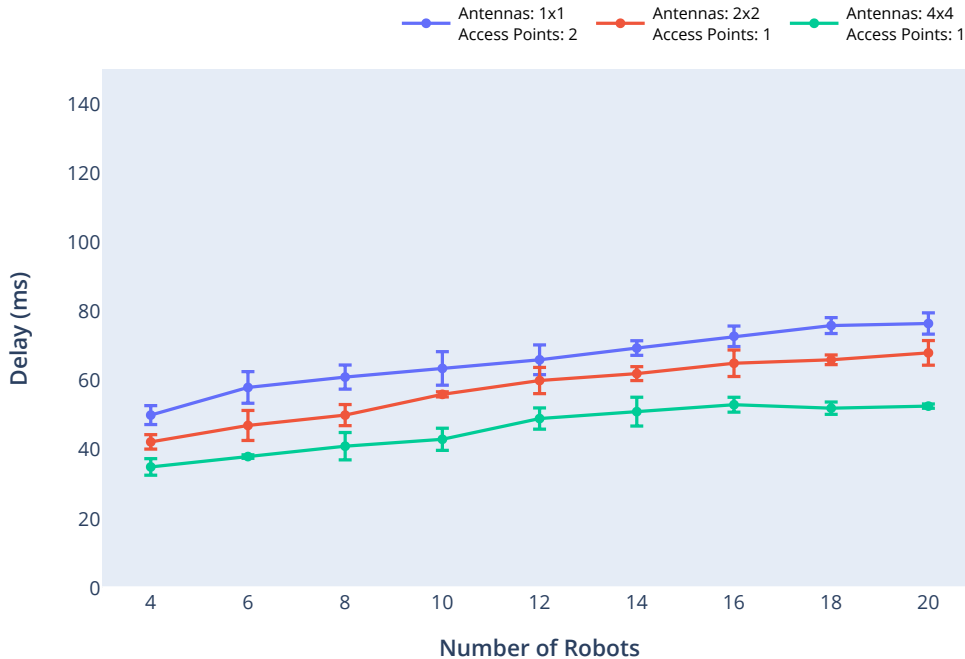


Figure 3.22: Scenario 1 (data rate independent deployment): End-to-End delay for a high quality stream.

access points. The significant difference in the number of access points between the SISO and MIMO systems leads to a lower end-to-end delay in the SISO case, as shown in figure 3.24 due to lower channel occupancy as we increase the number of robots. The impact of a large number of access points overcomes, in this case, the impact of the beamforming gain on the end-to-end delay.

In the third scenario, access points deployed at the edge of the field, figure 3.25 shows the enhancement obtained from the usage of beamforming on the network performance. The network with beamforming outperformed the SISO deployment due to the difference in the non-covered area. In this type of deployment, the larger number of access points did not result in better throughput. Certain areas in the field do not allow sending at the application required MCS due to low SINR, which is the case of 2x2 and 1x1 systems. The end-to-end delay results are presented in figure 3.26. We have a lower delay for the 2x2 system over the 4x4 and the SISO systems due to the larger number of access points and the beamforming gain. Based on these results and the ones shown in figure 3.24, we conclude that as the number of access points is closer in different systems, the beamforming gain impact becomes more significant. This causes a less end-to-end delay for MIMO systems with a larger number of antennas. Note that one of the added values of beamforming is reducing interference, especially with multiple access points deployed at the edge of the field.

The better performance in 1x1 over 2x2 over 4x4 systems comes at a cost. The large gap between the number of access points required means a more expensive deployment cost for the SISO over the MIMO systems. This cost is not limited to the price of the access points but also to the increased complexity of the deployment in terms of cables and additional obstacles in the field. We should note that the

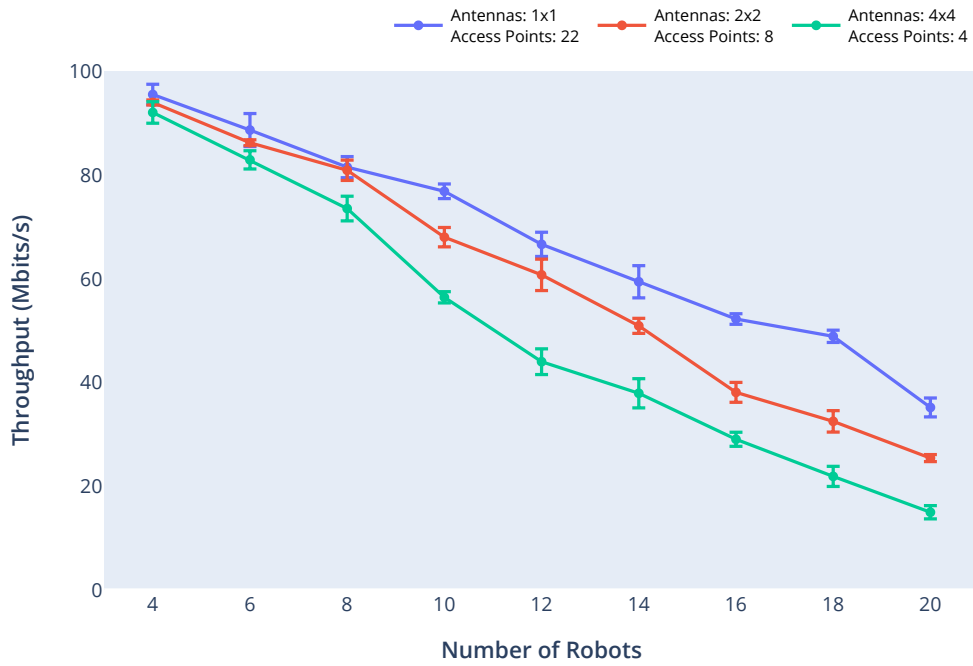


Figure 3.23: Scenario 2 (data rate dependent deployment): Throughput and number of access points when sending a high-quality stream.

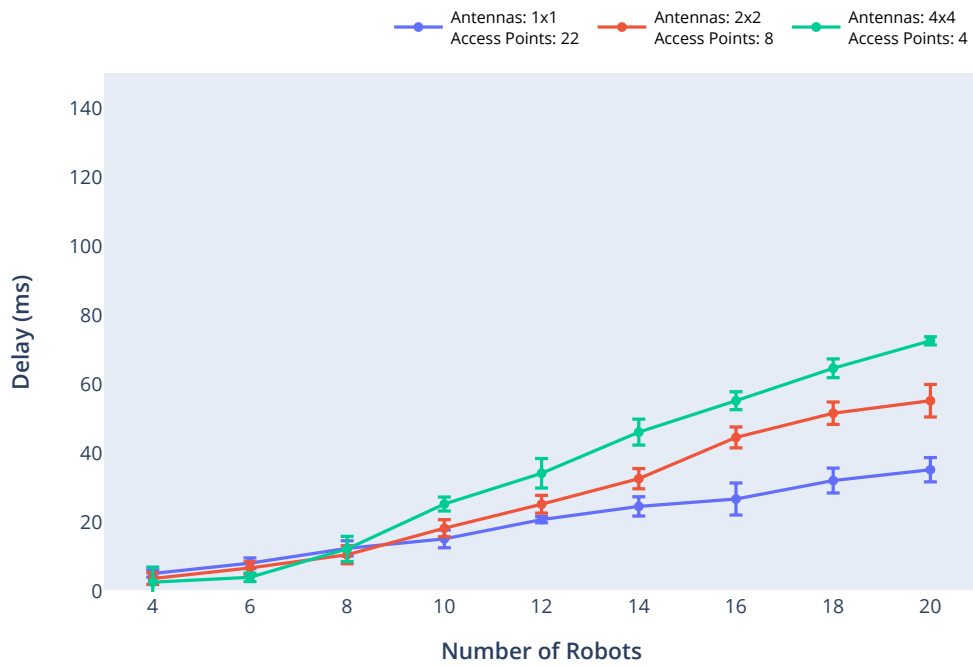


Figure 3.24: Scenario 2 (data rate dependent deployment): End-to-End delay for a high quality stream.

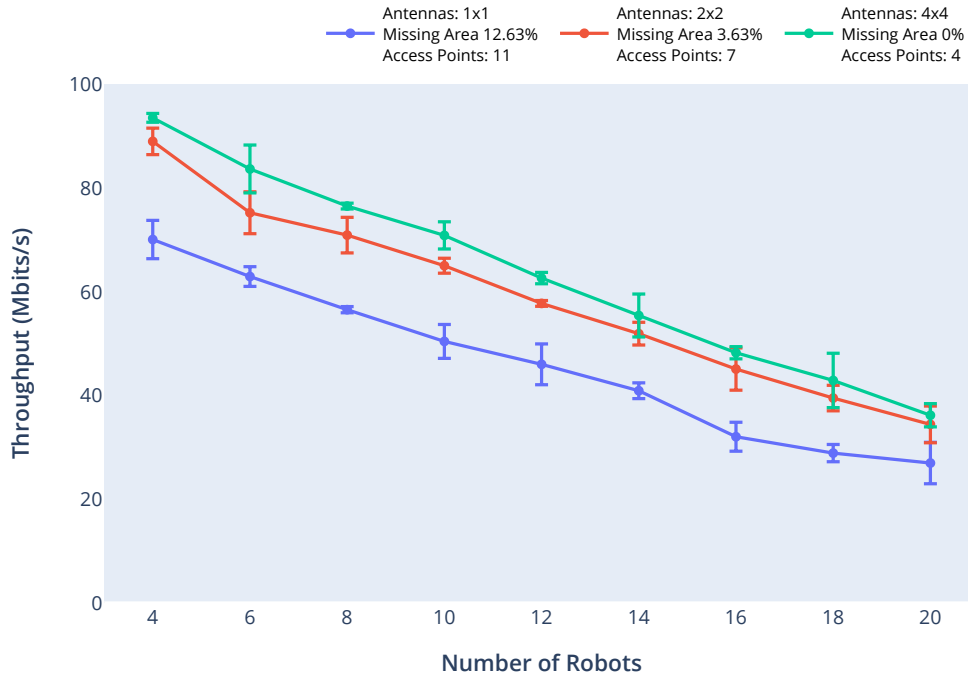


Figure 3.25: Scenario 3 (Edge deployment): Throughput of the system in the special deployment with high quality streaming.

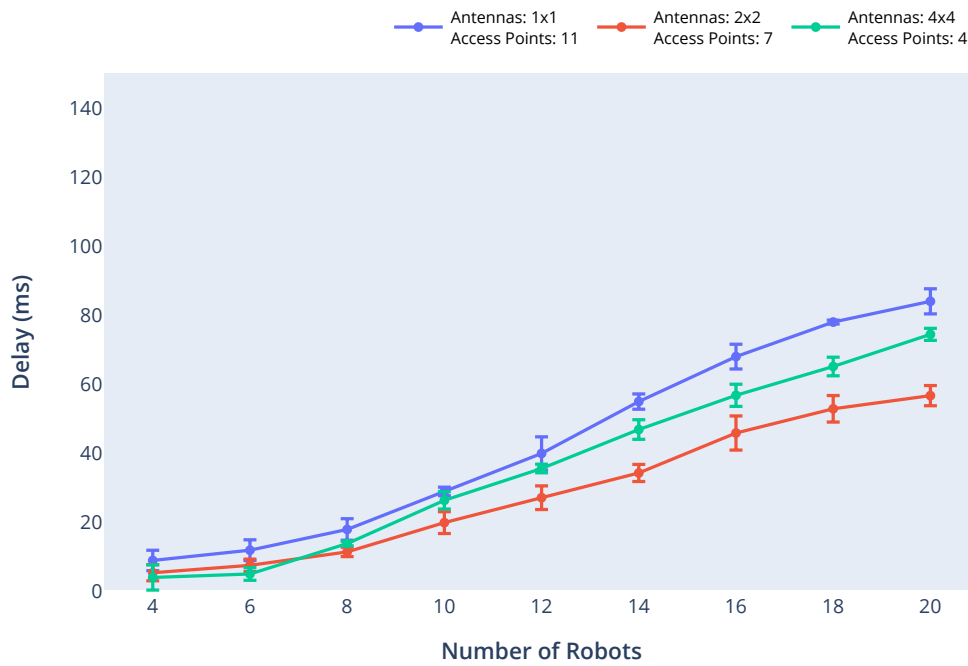


Figure 3.26: Scenario 3 (edge deployment): End-to-End delay for a high quality stream.

theoretical beamforming gain may not be achievable in real-life scenarios due to the hardware manufacturer's design and implementation of the technology.

Note that our application targets Smart Farming, where only a few robots are present in a field with predefined tasks. But if the number of nodes increases, more access points should be added to increase the capacity of the network. Another solution would be to adapt several network metrics to optimize access to the channel.

3.4.2 Empirical Performance Evaluation

Based on the results obtained previously, we derived an empirical model to estimate the performance of the deployment relative to the field size and the number of deployed access points. This study is useful for a similar deployment strategy given a different field size. In what follows, we considered a grid data rate-dependent deployment strategy. We introduce a parameter I which represents the impact of the interference on the performance of the network. It is the ratio of the throughput obtained over the transmission data rate, also known as an offered load. I is given by the following equation:

$$I = \left(1 - \frac{\text{thpt}_{sim}}{D}\right) \quad (3.1)$$

where thpt_{sim} is the throughput obtained through simulation, D is the transmission data rate of the application.

The deployment performance can be estimated using equation 3.1 if we decide to keep the same number of robots and the number of access points intact while only changing the field size. This can be done by dividing the interference impact by the area fraction $Area_{new}/Area_{simulation}$ so that it decreases when the area of the field increases and vice versa.

In reality, we have to change the number of access points when changing the field size, this will directly affect the interference impact. For instance, adding more access points will reduce interference by decreasing the density of robots per access point. To consider this change, we ran multiple simulations with the same field size and number of robots while changing the number of access points, the impact of the change in the results obtained is introduced in the following equation:

$$I_{new} = I_{original} - \alpha \delta N_{APs} \quad (3.2)$$

where I_{new} is the interference impact in the new scenario after changing the number of access points, $I_{original}$ is the interference impact in the original scenario, δN_{APs} is the difference in the number of access points between both scenarios and α is the factor calculated through multiple runs which were found to be related to the number of robots in the field as shown in figure 3.27. As the number of robots increases, the improvement in the performance is less significant when adding additional access points, this can be seen from equation 3.2. As α decreases (when more robots are introduced in the field), more access points are needed to reduce the interference impact.

Using equations 3.1 and 3.2, and the values of α obtained through simulations, we tested our model for fields of areas $2km^2$ and $5km^2$. The estimated performance is compared to simulation results for medium-quality video streaming. Figures 3.28 and 3.29 show the estimation and the simulation results, respectively for the $2km^2$ field. Simulation results include the minimum and maximum throughput values obtained throughout 40 runs. Results show that the estimation using our empirical

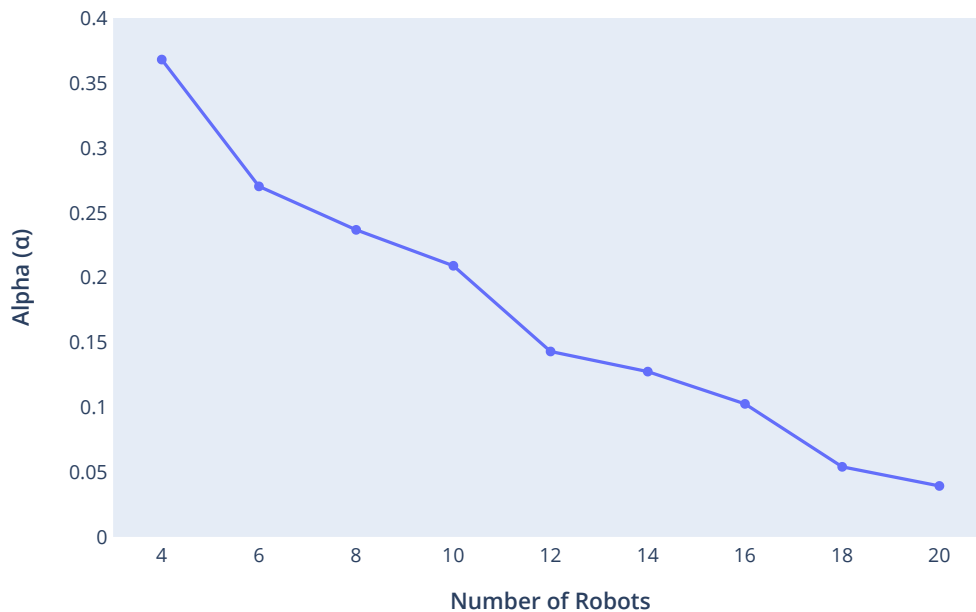


Figure 3.27: Impact of the number of access points on the interference versus the number of robots in the field for a 2x2 system.

model is close to the simulation results and falls within the min-max simulation throughput range.

We extended the area further and tested our model with a $5km^2$ field. Results are shown in Figures 3.30 and 3.31. The main difference between the $5km^2$ and the $2km^2$ fields is the number of access points. The estimation presented in the figures is a result of the calculation of the α factor per number of robots. The graphs show that the impact of additional access points is well addressed in the empirical model. These results make performance evaluation quicker, easier, and closer to the real performance of any similar study with the same deployment strategy and application use cases.

3.4.3 Deployment Feasibility Study

The design phase in Wi-Fi deployment is crucial for ensuring good quality of communication in the deployed network. The planning has to be realistic. In other words, one should consider the complexity of the deployment in terms of the software and the hardware availability and implementation. The deployment technique used in our study does not require any complex processing but solving simple mathematical equations. One can determine the number and the location of the access points needed to satisfy the data rate requirement of the application beforehand. Meaning that the process does not require real-time communication and the optimization could be done offline during the pre-deployment study. Increasing the number of antennas is a good enhancement to the network as it enhances the coverage and the quality of the communication, but it comes at a cost. For example:(1) the beamforming implementation differs from one vendor to another, this impacts the quality of the beamforming in terms of the expected range, the accuracy of the beams, and

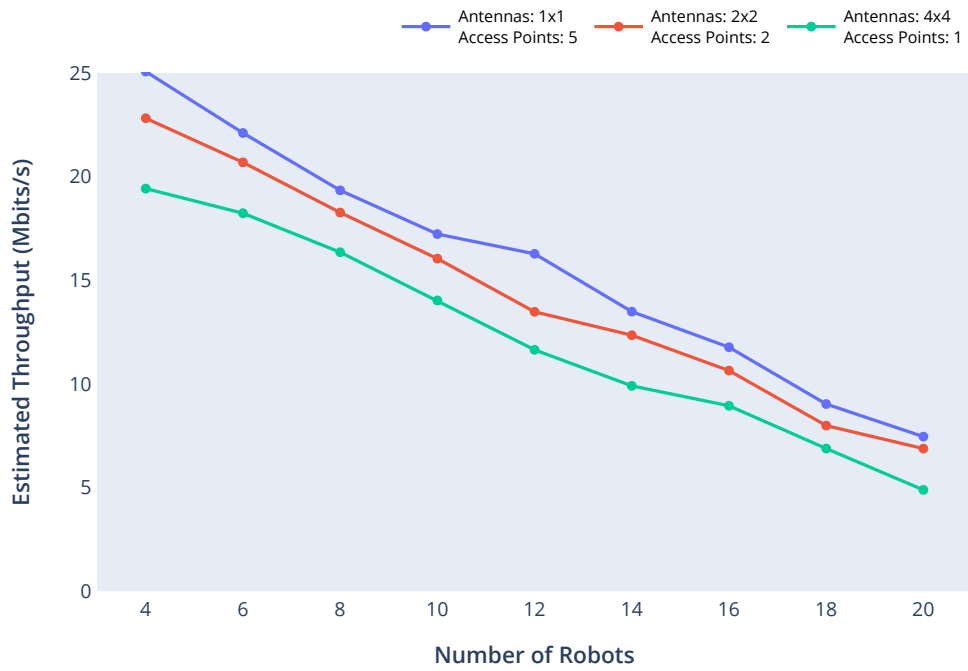


Figure 3.28: Throughput obtained through estimation for a field of area $2km^2$.

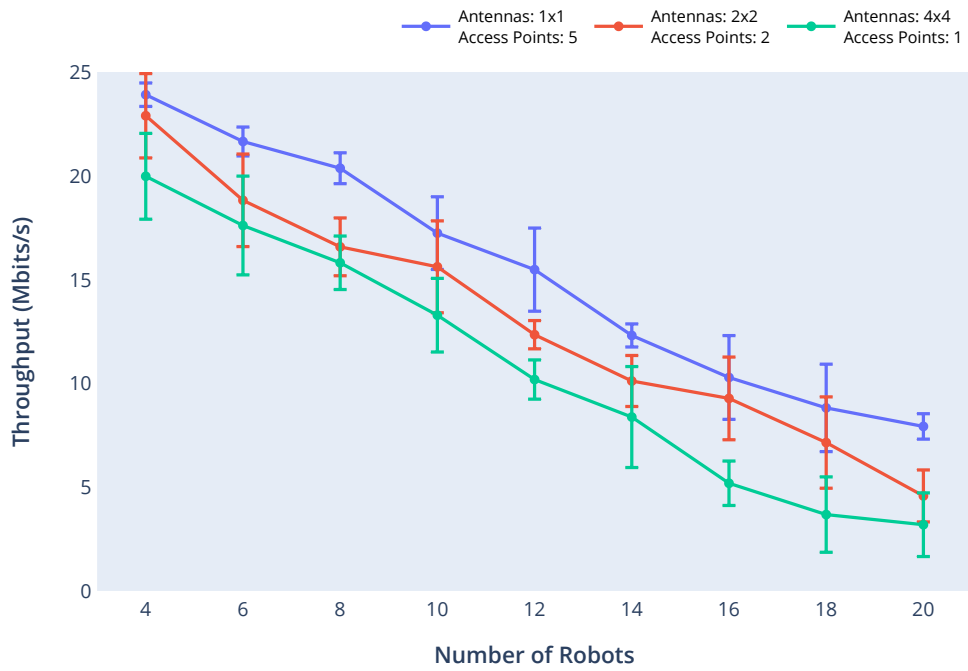


Figure 3.29: Throughput obtained through simulation for a field of area $2km^2$.

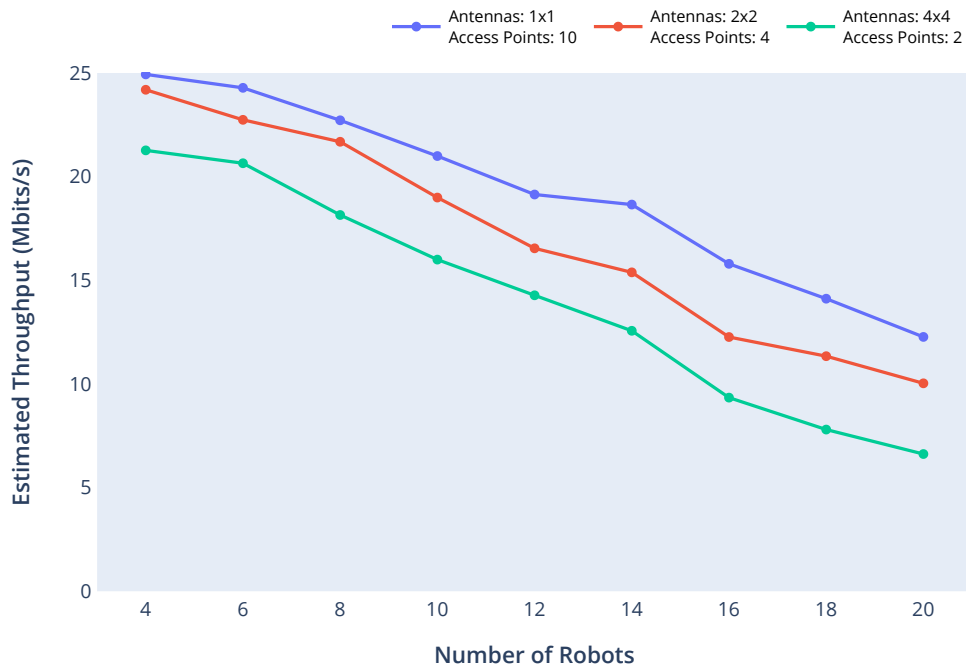


Figure 3.30: Throughput obtained through estimation for a field of area 5km^2 .

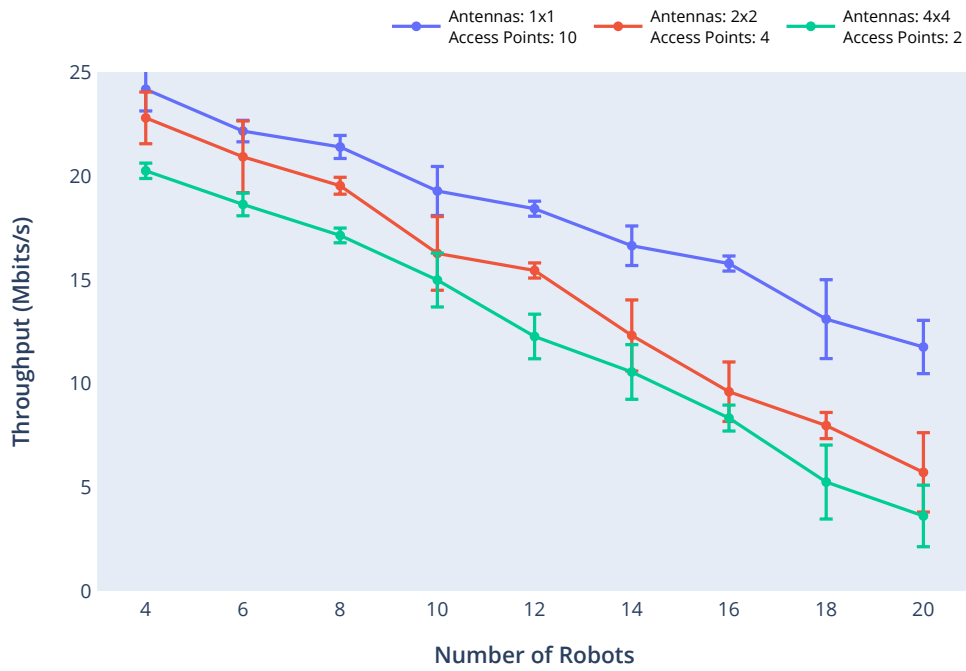


Figure 3.31: Throughput obtained through simulation for a field of area 5km^2 .

the beam generation time. Also, (2) the cost of MIMO access points becomes higher as we increase the number of antennas in the system.

3.5 Conclusion

In this chapter, we have presented our initial contribution. We began by assessing representative rate adaptation algorithms from existing literature in dense networks, focusing on metrics such as throughput and packet loss. Subsequently, we chose the best-performing algorithm and refined it to meet the standard requirements. Rate adaptation algorithms are essential as they adapt data rates over the network and greatly influence network performance. The chosen rate adaptation algorithm (Modified Ideal Rate Adaptation Algorithm) serves as the default RAA for all our contributions.

In the context of autonomous robots for smart farming, we explored deployment strategies that utilize beamforming to enhance communication range and quality. Our strategy involved determining the required number of access points for different MIMO systems across two deployment types (grid and edge deployments). The deployment types took into account the challenges associated with deployment in specific fields. Additionally, we evaluated, through simulation, our deployments for different application requirements to emulate a control station taking control of a robot in the field. After gathering simulation results, we developed an empirical model to obtain results for any field size, and the simulation result demonstrated the accuracy of the model.

Our study highlighted the trade-offs between deployment cost and complexity versus network performance. While deciding on a deployment plan is crucial, there remains potential for improvement across various layers of the Wi-Fi network, targeting specific challenges in Wi-Fi networks as outlined in section 2.2. Tackling these challenges is the objective of our second contribution, which will be introduced in the upcoming chapter.

Chapter 4

Contribution 2 - Reinforcement Learning in Wi-Fi Networks

4.1 Introduction

This chapter details our second contribution to optimizing the performance of Wi-Fi networks. We tackle two aspects of optimisation: (i) adapting the application data rate depending on the state of the network in order to avoid overloading the network, and (ii) adapting the backoff duration of the CSMA/CA algorithm taking into consideration the state of the network. One technique for achieving these adaptations is the employment of machine learning. In this chapter, we employ deep reinforcement learning, a branch of machine learning, to optimize the performance of Wi-Fi network.

In section 4.2, we present the DRL algorithm used in our contribution and some of the technical details of the implementation. In section 4.3, we explore the potential of DRL in enhancing application profile selection within congested Wi-Fi networks, where performance can deteriorate significantly due to saturation. We develop and assess a DRL mechanism to identify the most suitable application profiles for achieving optimal network performance under any given network condition. In section 4.4, we investigate the application of deep reinforcement learning to improve the performance of the CSMA/CA protocol responsible for managing access to the wireless medium in Wi-Fi networks. We develop and evaluate a DRL mechanism through simulation. In section 4.5, we discuss the complexity of the produced DRL models. Finally, we conclude this chapter in 4.6.

4.2 Algorithm and Implementation

This section aims to provide a comprehensive understanding of the DRL algorithm employed in our contribution and the implementation of the communication interface between the DRL module and the network simulator.

4.2.1 Deep Reinforcement Learning Algorithm

Multiple machine learning techniques exist in the literature to tackle various problems in different domains. However, in our context, reinforcement learning has been chosen over other machine learning approaches due to its inherent ability to incorporate an agent that can actively explore and interact with the environment. This interaction enables the agent to learn the most effective actions and strategies based

on feedback from the environment, such as rewards. In contrast, supervised learning requires a large dataset of labeled examples for training, which may not always be readily available or representative of the dynamic nature of Wi-Fi networks. Additionally, supervised learning methods may struggle to generalize well to unseen network conditions, as they primarily focus on minimizing errors in the training dataset [104]. On the other hand, unsupervised learning seeks to find patterns or structures in data without labeled examples. While unsupervised methods can help identify underlying patterns, they might need to be more effective at providing actionable insights or guiding an agent to optimize network performance [104]. As a result, the reinforcement learning approach is better suited for dynamic and complex scenarios often encountered in Wi-Fi networks.

There are various existent DRL algorithms that fall into different categories (on-policy, off-policy, model-free, model-based, etc.) [105]. In our use case, the agent will explore the environment. We have to evaluate the policy that the agent is using for its decision-making. We employ an on-policy approach based on the actor-critic framework, namely, PPO with the clipping surrogate technique [89]. We picked PPO for its stability as it constrains policy updates so the learning does not diverge or fall to a local optimum. Note that we do not claim that PPO is the best DRL approach, various DRL approaches may also be suitable in our context. However, the choice of the algorithm falls into the category of algorithms that have similar behavior in constraining policy updates.

The actor-critic framework consists of two neural networks: the actor-network and the critic network. The actor-network is responsible for the action selection. It takes the state as an input and outputs a probability vector of the possible actions. The critic network is the value function. It takes a state as an input and outputs the expected *return*. The critic network decides if the policy (actor) is improving or deteriorating. In what follows, we explain how PPO works:

In the first step of training, we initialize the hyperparameters and the network weights of the actor and the critic networks, θ and ϕ , respectively. Then, we iterate through multiple episodes of training. At the beginning of each episode, we initialize an empty batch \mathbf{B} that will hold the (*state, action, reward*) tuples. The tuples are used to update the actor and the critic networks at the end of each episode. When the simulation starts, the agent observes the environment by collecting the states. A reward is then calculated based on the collected states. Then, the previous state, the action taken in the previous state, and the newly calculated reward are added to \mathbf{B} . The actor-network is then used to predict an action for the current state.

To check if the algorithm is improving or deteriorating, we calculate the advantage function at the end of each episode shown in (4.1). It indicates how beneficial each action was when using the current policy. It is a comparison between the *return* when taking action in a state and the expected *return* of the state using the previous policy. The advantage function provides insight into the impact of the action of the agent on the *return* of the state.

$$A_t^\pi(s, a) = r_t + \gamma * V^\pi(s_{t+1}) - V^\pi(s_t) \quad (4.1)$$

Where $V^\pi(s_t)$ is the critic network that gives the expected *return* of a state and γ is the discount factor to determine the importance of future rewards in comparison to the immediate reward. The advantage function is then used to update the ϕ parameters of the critic network by performing gradient descent with respect to the loss function (4.2). We update the critic network parameters so that its predictions match the *return* of the policy.

$$L(\phi) = \mathbb{E}[A_t^2] \quad (4.2)$$

The θ parameters of the actor-network are updated by performing gradient ascent with respect to the loss function shown in (4.3) where r_θ is the probability ratio shown in (4.4). r_θ will be greater than 1 when the action is more probable for the current policy than it was for the old policy, it will be between 0 and 1 otherwise. Clipping is done based on ϵ , a hyperparameter in the loss function, used to avoid cases where the actions between policies have a larger probability difference. Thus, it prevents taking big gradient steps when updating the policy. The update of the actor parameters makes the actions that resulted in a better *return* more probable in the new policy.

$$L(\theta) = \mathbb{E}_t[\min(r_\theta * A_t, \text{clip}(r_\theta, 1 - \epsilon, 1 + \epsilon)A_t)] \quad (4.3)$$

$$r_t(\theta) = \frac{\pi_\theta(a_t | s_t)}{\pi_{\theta_{old}}(a_t | s_t)} \quad (4.4)$$

The actor and critic networks are updated until their losses become negligible. The critic loss becomes negligible when the new policies have no more advantage over the old ones. The actor loss becomes negligible when the policies produce almost no difference in the predicted action probabilities for the different states. Thus the training is finished. A DRL model is then produced and ready for evaluation.

4.2.2 Implementation of the Training and Evaluation Scenarios

Implementing the DRL algorithm involves using two main modules: The network simulator NS-3 and Python. NS-3 is used to create the simulation scenarios, while Python is used to develop the DRL model and its training and evaluation logic. However, since the DRL model needs to interact with NS-3, we incorporate an asynchronous messaging library called ZeroMQ (ZMQ) [106]. ZMQ facilitates the exchange of information between independent applications. The relevant metrics from all the robots in NS-3 are collected periodically and encoded in JSON format. This information is transmitted to the DRL model in Python via ZMQ. Upon receiving the JSON formatted object in Python, the necessary metrics are extracted as the state of the DRL model. Then, The DRL model generates the required actions from the extracted metrics, which are then encoded in JSON format and sent back to NS-3 via ZMQ. This process continues until the DRL model completes its training or evaluation. The whole process is depicted in figure 4.1 where normalized throughput and application profile are examples of metrics and decisions being exchanged between the two modules. The implementation of this approach allowed for a seamless exchange of information between NS-3 and the DRL model written in Python, which facilitates the training and evaluation of our DRL models.

4.3 Application Profile Selection Using Deep Reinforcement Learning

In this section, we examine the training and evaluation of our DRL mechanism, which is used to enhance goodput in dense Wi-Fi networks and address performance

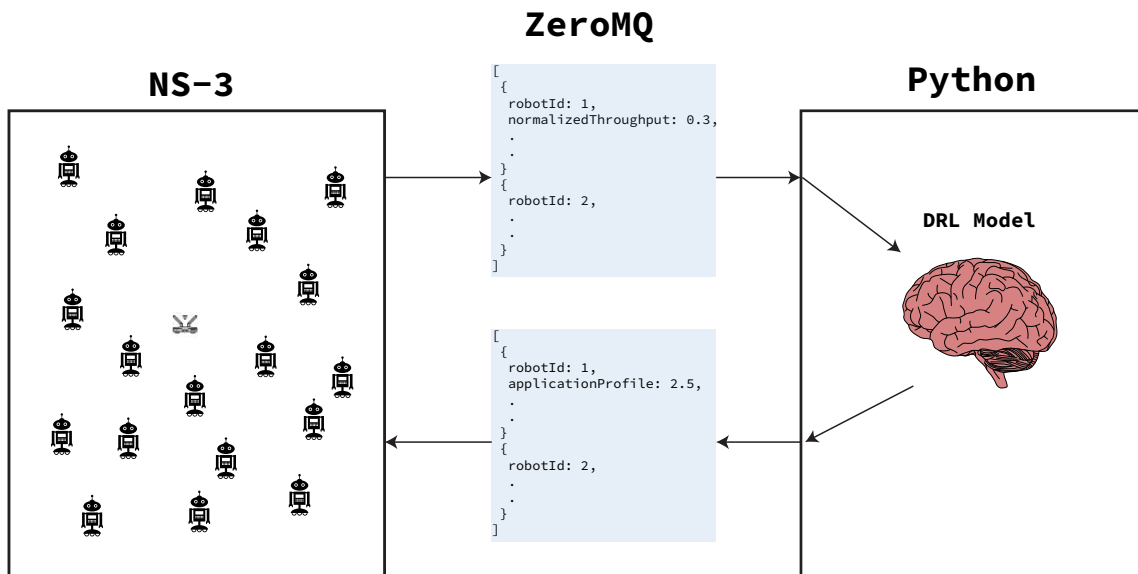


Figure 4.1: Communication between NS-3 and the DRL model

degradation in high offered load situations. We present several scenarios with varying levels of saturation to demonstrate how Wi-Fi performance deteriorates under high offered loads. Subsequently, we demonstrate how our DRL model effectively improves performance in dense scenarios.

4.3.1 Problem Formulation

Modern applications are more and more demanding in terms of network needs. Thus, more bandwidth is required by each Wi-Fi node in order to transmit/receive data while maintaining the satisfaction of users. Studies on Wi-Fi performance have shown a significant decrease in the overall network throughput as the network size increases [107]. The deterioration of the network performance is caused mainly by 2 reasons. First, (i) the increase in the Offered load: when each node increases the amount of data, the overall offered load per second becomes greater than the reception capacity of the access point. This leads to the latter not being able to handle all the transmitted data. Thus, the goodput obtained at each node decreases, reducing the QoE. Second, (ii) the use of a random access protocol (CSMA/CA) in the Medium Access Control (MAC) layer: after transmitting a data packet, a node waits for an acknowledgment from the receiver, if no acknowledgment is received, the node backs off and waits for a random number of time slots before accessing the channel again. At each new attempt to retransmit the packet, the probability of choosing a longer backoff duration is increased. This mechanism causes significant access delays and collisions, especially in dense networks [?].

These two problems occur in dense deployments with high offered loads. Indeed, the probability of collisions increases due to the number of nodes attempting to send at the same time. Furthermore, the amount of data transmitted by each node leads to longer transmission times. This results in longer access delays, reducing the goodput, and increasing the packet loss. In other words, the random access protocol combined with the high offered load causes degradation in the QoE.

In what follows, we propose a DRL mechanism that reduces the offered load per node in an attempt to enhance the QoE. We assume that the application layer offers the possibility to adapt its requirements in terms of offered load. We call

an application profile a specific set of requirements in terms of offered load. For example, in a video streaming application, profiles can be defined based on the frame-per-second requirement or the definition of the generated frame, or a combination of both. Given these profiles, we propose a dynamic mechanism that selects the most suitable profile depending on the current state of the network in terms of observed QoE. For example, under high contention conditions, a node preserves the satisfaction of the user by selecting an application profile with a low data rate requirement. Indeed, the maximum amount of data a node can transmit at a specific time is decided by the current MCS value at the physical layer. Lowering the application profile will result in less channel utilization time. This, in turn, will decrease network congestion and enhance the overall goodput.

4.3.2 Model Design

The optimization problem can be framed as an MDP consisting of (i) an agent exploring the environment, (ii) a set of states S , (iii) a set of actions A , (iv) a reward function R , (v) a transition probability T , and (vi) a reward discount factor $\gamma \in [0,1]$.

The *agent* is an application installed on the Wi-Fi node. The application possesses multiple profiles with different data rate requirements. The agent sends data toward an access point over an interval of time and observes the *state* $s \in S$ of the environment. The state is expressed as the goodput (g) and the packet loss (pl) of the transmissions in the previous interval $[g, pl]$. The agent then takes an *action* $a \in A$ by selecting an application profile based on a policy (π). The policy is a neural network that provides a set of probabilities, each of which corresponds to an action. Based on the policy, the agent samples an action and decides whether to keep using the current application profile or select a new one. The reward r is then calculated as the difference between the goodput and the packet loss, as shown in (4.5).

$$r = \alpha * g - \beta * pl \quad (4.5)$$

Note that the goodput is normalized to the range of $[0,1]$ by dividing it by the maximum expected goodput (based on the maximum application profile). α and β parameters help to fine-tune the reward to be compliant with the requirements of the application. For instance, increasing β will drive the agent to favor learning how to reduce the packet loss more than increasing the goodput. In this case, selecting a lower application profile would be the most probable decision. It leads to less transmission time and less data queued for transmission. Thus, the contention between nodes is decreased, which reduces the probability of collisions but does not necessarily increase the goodput.

The immediate reward may not be sufficient to determine the proper decision in the current state. The decision at any state has an impact on the future series of events. The discount factor $\gamma \in [0,1]$ is used to determine the importance of future rewards in comparison to the immediate reward. This is represented by the cumulative reward equation (4.6). It predicts how much reward is expected in the future after taking action in a certain state. For instance, choosing a low value of γ means favoring short-term rewards. For example, when streaming a live video, we care more about short-term rewards.

$$\bar{r} = r_t + \gamma * r_{t+1} + \gamma^2 * r_{t+2} + \dots + \gamma^T * r_{t+T} \quad (4.6)$$

Our aim is to reach a policy in which, for any observed network conditions, it selects the application profile that maximizes the *return* (cumulative reward) of the agent. Since we will be training our agent through simulation, we can create any scenario with any network conditions. Thus, Our approach is split into two phases: *Offline-training* and *Exploitation*. During the *Offline-training* phase, the agent explores the environment through different simulation scenarios until converging to a policy based on our reward design. At the end of the training, we save the model that contains our final policy to use in the next phase. During the *Exploitation* phase, we deploy the generated model in scenarios where the agent is able to pick with a certain degree of confidence the best application profile given a state. The complete training algorithm is presented in algorithm (1), and figure 4.2 shows the complete structure of the proposed approach.

Algorithm 1 Offline Training

```

1: Set the hyperparameters and initialize the actor and critic network parameters
   ( $\theta_0, \phi_0$ ) respectively
2: for episode:  $1, \dots, E$  do
3:   Initialize an empty batch  $\mathbf{B}$ 
4:   for second:  $1, \dots, S$  do
5:     Get the state of each node:  $s_{t_1}, \dots, s_{t_N}$ ,
       where  $s_{t_n} = [goodput, packetloss]$ 
6:     for node:  $1, \dots, N$  do
7:       Calculate  $r_{t_{node}}$  (4.5)
8:       Add  $(s_{(t-1)_{node}}, a_{t-1_{node}}, r_{t_{node}})$  to  $\mathbf{B}$ 
9:        $a_{t_{node}} = Sample(\pi_{\theta}(s_{t_{node}}))$ 
10:    end for
11:  end for
12:  Calculate the cumulative rewards  $\bar{r}$  (4.6)
13:  Calculate  $A_t$  (4.1) from  $\mathbf{B}$  and the critic predictions
14:  Update  $\phi$  by a gradient method w.r.t  $L(\phi)$  (4.2)
15:  Update  $\theta$  by a gradient method w.r.t  $L(\theta)$  (4.3)
16:  if  $L(\theta)$  is negligible then
17:    Save the model
18:    End training
19:  end if
20: end for

```

4.3.3 Model Training

To start the training process, we prepare a simulation scenario with a varying number of nodes. The total duration of the simulation is set to 180 seconds of NS-3 simulation seconds. We chose to do the training phase on scenarios that do not involve mobility. We start the simulation with 10 nodes, and we split each simulation scenario into 3 phases of 60 seconds. After each phase, we add 10 nodes to the scenario. The number of nodes is chosen in order for the model to explore a variety of states in different network conditions.

Our training reward design of equation 4.5 is set to favor improving the goodput over the packet loss $\alpha > \beta$. Note that the packet loss seen by the application can be caused not only by collisions but also by the buffer overflow. Indeed, in high offered load scenarios, nodes will not be able to access the medium due to channel

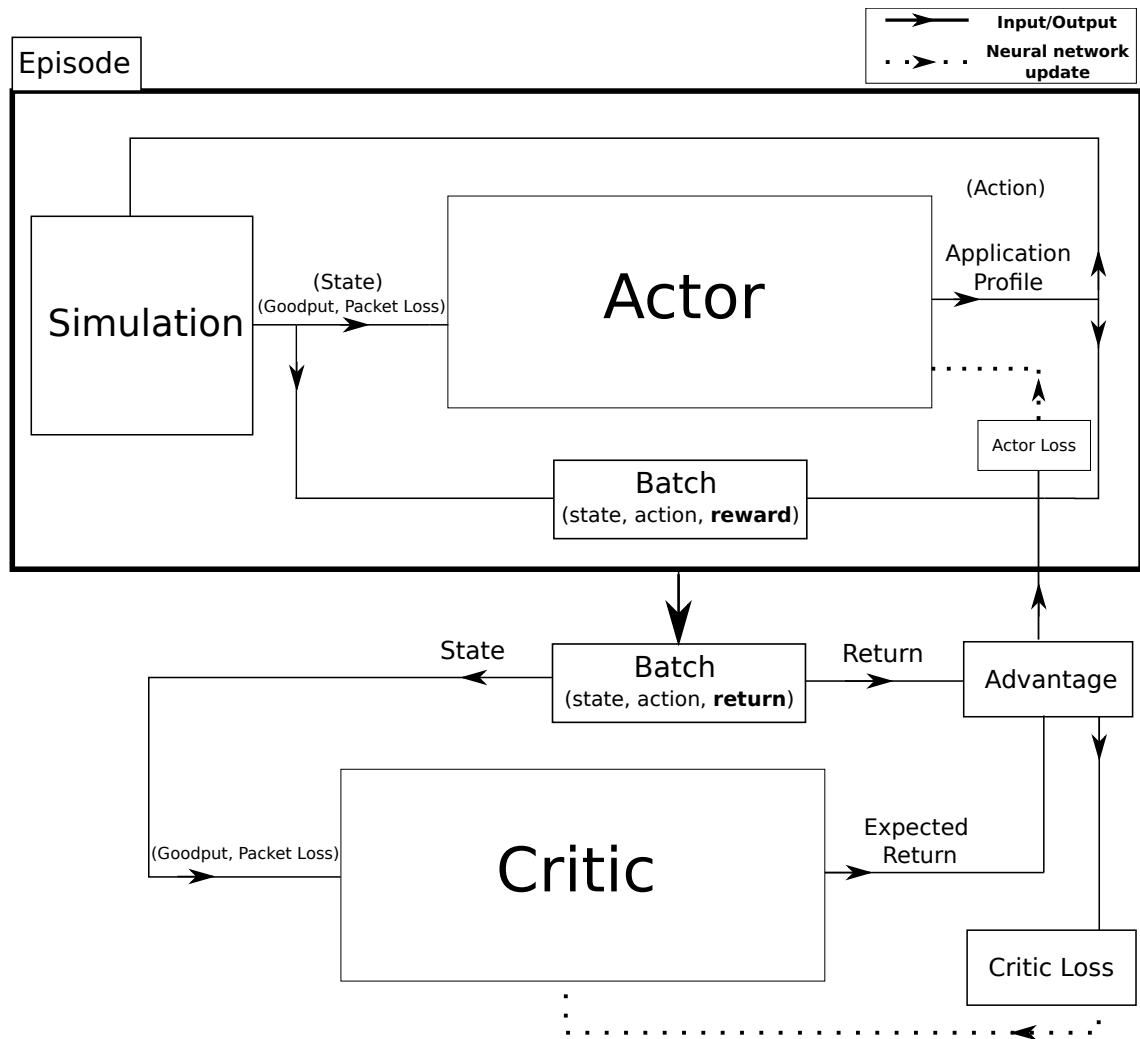


Figure 4.2: The structure of the proposed approach using the PPO algorithm

activity. This will generate an accumulation of traffic in the buffer of the MAC layer. Thus, this will increase the overall packet loss both dSNRue to collisions and buffer overflow. In an attempt to reduce the number of packet losses, we included it in the reward equation.

During training, each NS-3 simulation sends to the DRL module the unique identifier and the state of each node. Then, the DRL module predicts an application profile for each node based on the states and calculates the corresponding rewards. The list of available application profiles are [0.7 Mb/s, 1.5 Mb/s, 2.5 Mb/s, 3.5 Mb/s] corresponding to different video resolutions as used in [108]. In fact, selecting an application profile of 0.7 Mb/s in a dense environment leads to less transmission time and less contention. The predicted application profile for each node is sent back to NS-3. Each node starts transmitting with the corresponding profile for an interval of time which is set to 250 milliseconds. This duration allows the nodes to interact with the environment multiple times, providing a better vision of the environment than a single transmission. Depending on the fault tolerance of the application, a shorter or longer monitoring duration can be used. At the end of the simulation (Episode), the losses of the actor and the critic networks are calculated. The actor and critic networks are then updated accordingly. The training process is marked as done when the losses become stable and the return is not increasing anymore.

Figure 4.3 shows the actor and critic losses during the training process. The value of the actor loss oscillates during the first few episodes marking significant changes in the policy. The critic loss decreases gradually in the first few episodes. When the critic loss reaches its minimum value, this means that new policies have no significant advantage over the old ones. The model is updated at the end of each episode and is used in the next one. After around 35 episodes, the loss values become stable.

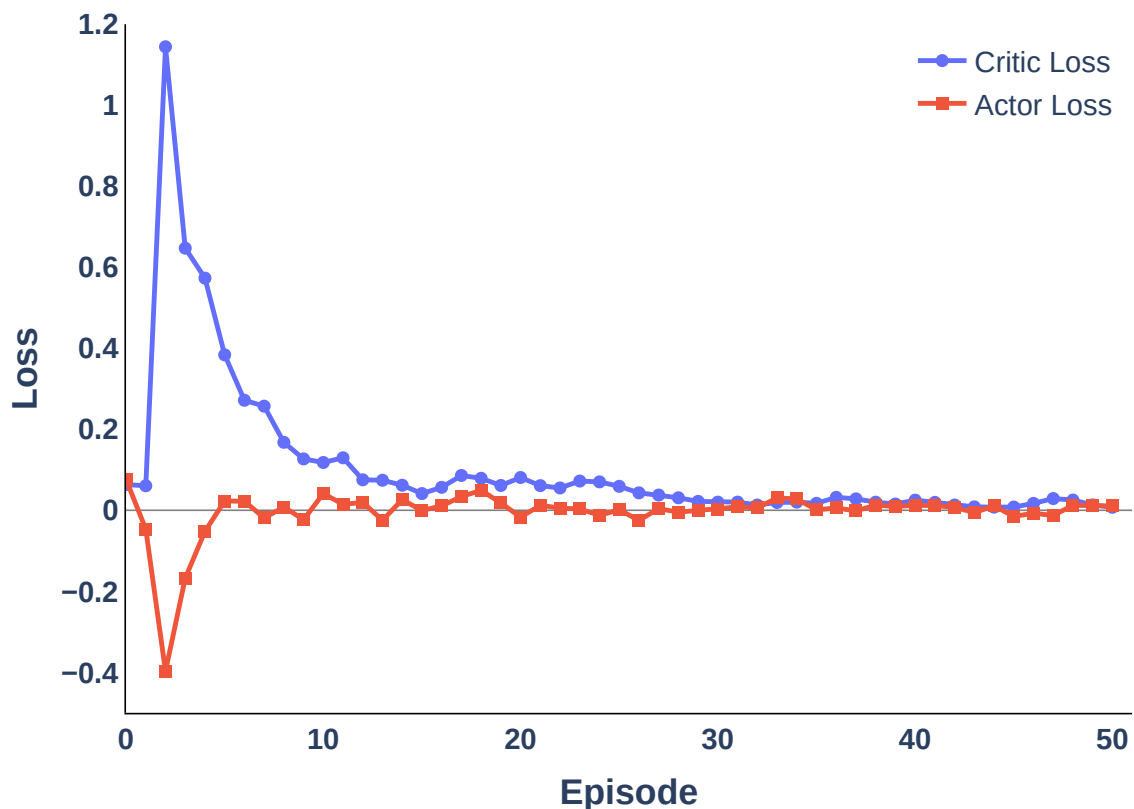


Figure 4.3: Actor and critic losses during the training phase of the DRL model

Figure 4.4 depicts the return values. It shows that the return increases until it reaches a maximum value of almost 1 and stabilizes after 35 episodes. When the return reaches its maximum value and remains stable, we consider that the training is finished.

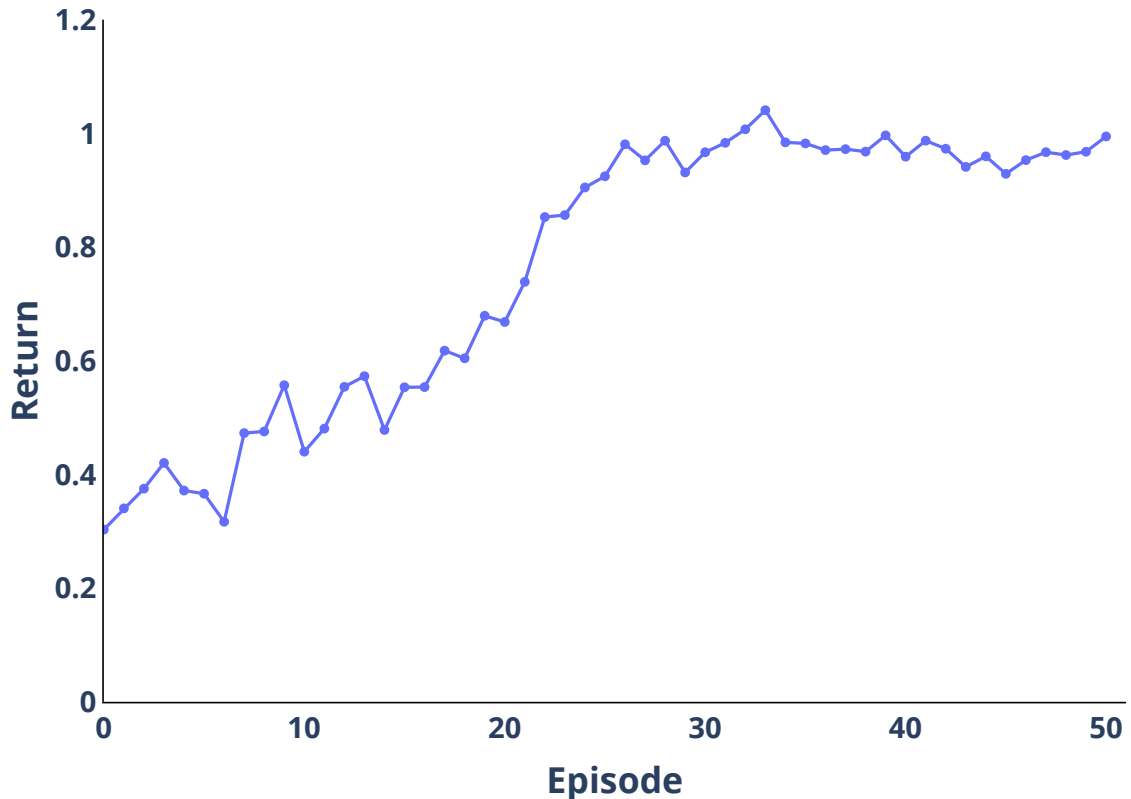


Figure 4.4: Return (cumulative reward)

4.3.4 Model Evaluation

The offline training produces an optimized model that maximizes the return. The goal of our model is to adapt the application profile to enhance the performance of the network. We validate the model, which was trained in a few static scenarios, in a larger space of scenarios, including mobile scenarios. The validation aims at testing the ability of the model to generalize and adapt to scenarios it has not previously encountered.

For the validation phase, we start by evaluating the performance of Wi-Fi using the same application profile for all nodes. This helps us identify the limitations of using a single application profile. We performed simulations by varying the number of nodes from 1 to 100. Each node is transmitting using the same application profile toward the same access point. Note that, in these scenarios, nodes are static in order to eliminate the impact of mobility on the results. The simulation parameters are presented in table 4.1.

Figure 4.5 depicts the aggregated goodput obtained in the network averaged over 50 runs with the standard deviation. During simulation, the network becomes saturated for a certain number of nodes depending on the application profile for each scenario. When more nodes are introduced to the network, the aggregated goodput drops from the saturation value and keeps decreasing. This is mainly due to the probabilistic behavior of the CSMA/CA algorithm used by Wi-Fi, which does

Table 4.1: Simulation parameters table for the application profile optimization.

Parameter	Value
Simulation time (Training)	180 seconds
Simulation time (Validation)	60 seconds
Runs	50
WLAN standard	IEEE 802.11ac
Path loss model	Log-distance
Fading factor	random(0, 2dB)
Traffic	UDP
Channel Width	20 MHz
Packet size	1500 Bytes
Mobility model	Random Walk 2d Mobility Model
Mobility speed	4 m/s
Topology size	Square of boundaries (-30, 30, -30, 30)
α	0.8
β	0.2
Learning rate	0.001

not guarantee access to the medium, especially in high offered load scenarios [109]. Optimizations have been studied to enhance the performance of CSMA/CA [110], but this can only be done by updating the specifications of the 802.11 standard. Hence, our proposal is situated on the application level and can benefit any device without requiring modifications to the standard.

In what follows, we will compare our DRL model with E2E-MAC [108] and the highest application profile. Papers dealing with the same problem as ours often lack details about the proposed model, such as hyperparameters, simulation time, and the environment. This makes producing the same models for comparison almost impossible. Thus we decided to compare our approach with [108]. The algorithm combines throughput measurements with the number of re-transmissions on the MAC layer to improve the QoE of users. In our approach, the DRL model adapts the application profile to improve the overall performance of the network. First, we perform scenarios with mobility to make our simulation more realistic and to test our model in dynamic situations. During the scenarios, the density of nodes is increased from 1 to 100, which covers the decrease in goodput for most application profiles. Nodes adapt their application profiles using the trained model. Then, each node uses the selected application profile to transmit during a time interval. Finally, the model selects a new application profile based on the newly observed state.

Figures 4.6 and 4.7 show the goodput and the ratio of collisions respectively averaged over 50 runs and include the standard deviations. When we only have a few nodes in the network, nodes face relatively low contention during transmission, hence, a low probability of collision. Therefore, goodput achieved by using the DRL model, E2E-MAC, and the 3.5 Mb/s profile is close.

When the number of nodes is increased, nodes using our DRL model and E2E-MAC adapt their application profiles according to the network state. Nodes that are dynamically adapting their offered load using the trained DRL model and E2E-MAC were able to maintain a higher goodput and lower collisions than nodes using the fixed 3.5 Mb/s application profile. Our trained DRL model maintained a higher goodput than E2E-MAC but with more collisions. With the introduction of new nodes, the number of collisions increases due to high contention. Thus, the number

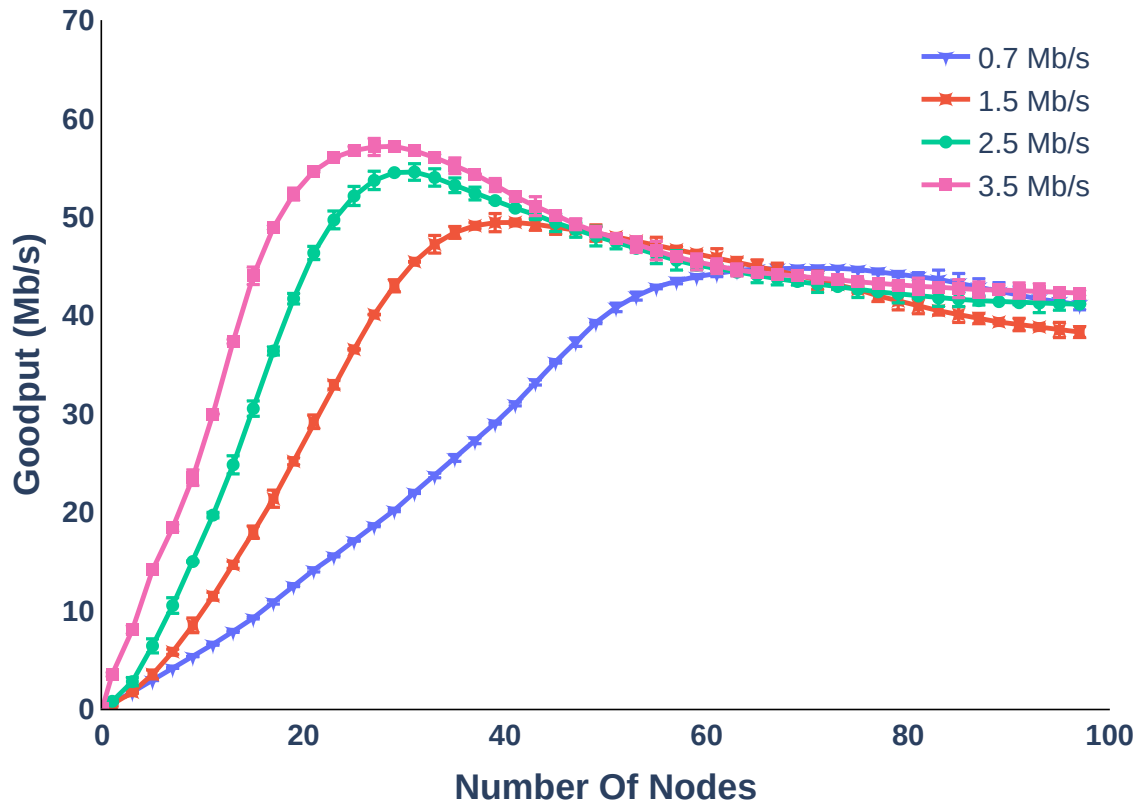


Figure 4.5: The goodput of nodes transmitting using a single application profile.

of re-transmissions increases, causing E2E-MAC to select lower data rates which explains the lower goodput and number of collisions obtained.

In certain situations, the data rate requirement of an application profile can be higher than the physical data rate. In these cases, the DRL model selects application profiles with lower data rate requirements which can be lower than the current physical data rate. This is done without the knowledge of the underlying physical data rate. It is based on local observations for each node. Nodes suffering from bad network performance are more likely to choose lower application profiles. This kind of decision has a double impact: First, they are generating less traffic and thus suffering from less data loss, second they are occupying the channel less often, meaning that other nodes can profit from a lower overall contention and increase their application profile if their performance feedback suggests so.

This contribution was published in a conference paper [23].

4.4 Intelligent CSMA/CA using Deep Reinforcement Learning

In this section, we propose a DRL mechanism, Intelligent CSMA/CA (ICSMA/CA), to dynamically adapt the backoff duration of CSMA/CA algorithm in dense Wi-Fi environments. We train and evaluate our DRL model. Finally, we show the results in terms of the overall throughput of the network, fairness among the users, and channel access delay.

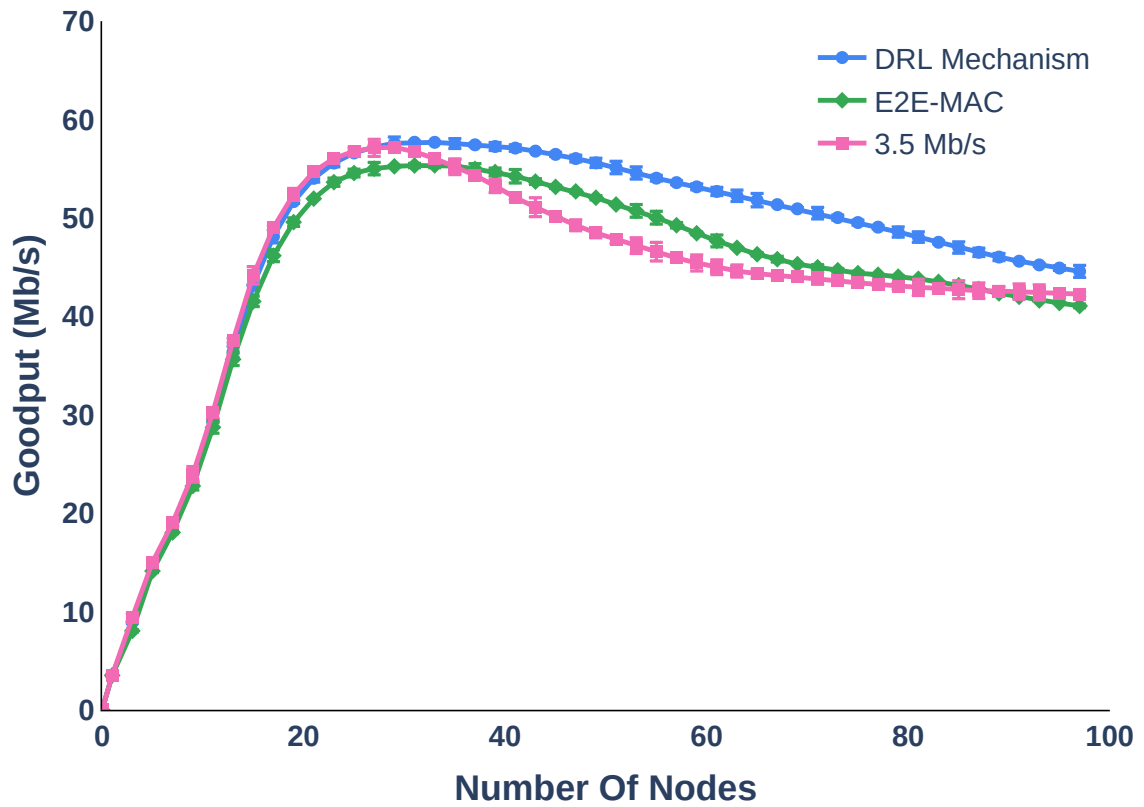


Figure 4.6: Overall goodput obtained in the network using the trained DRL model, E2E-MAC, and single application profile.

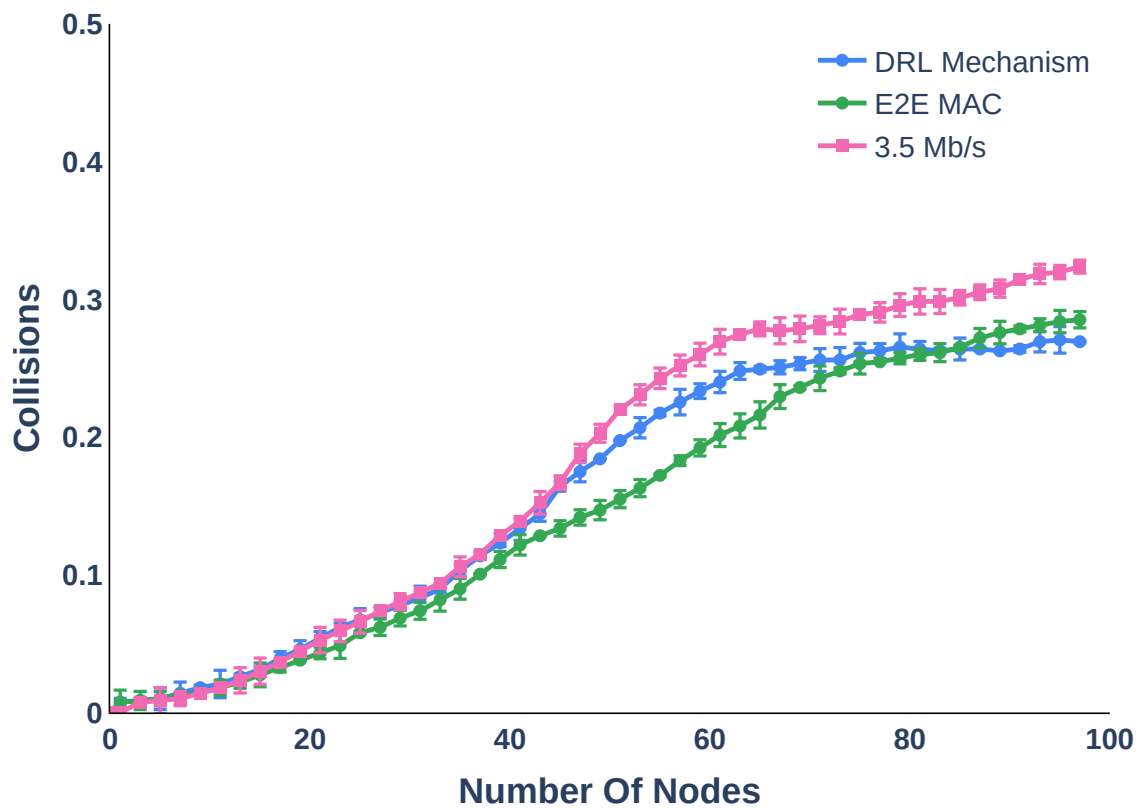


Figure 4.7: Ratio of collisions obtained in the network using the trained DRL model, E2E-MAC, and single application profile.

4.4.1 Problem Formulation

Wi-Fi standards use the CSMA/CA protocol with exponential backoff to control access to the wireless medium. In CSMA/CA, a station monitors the channel before transmission. If the medium is sensed idle for a time equal to DIFS, the station sends the packet. However, if the medium is sensed busy during this interval, a random backoff duration is picked from the interval $[0, CW]$. The station has to wait for the channel to remain idle during the entire backoff before it transmits the packet. Upon receiving a packet, the receiver waits for SIFS before sending an acknowledgment to the transmitter. The absence of an acknowledgment marks the loss of the packet. One of the reasons for packet loss is collisions. Collisions occur when multiple stations attempt to access the channel at the same time after sensing it as idle, and a common neighbor station receives these packets. A station doubles its CW after every packet lost, leading to longer backoff durations.

When the backoff duration is long, stations may wait for more time even after the channel has become idle. Thus, the idle time wasted before sending a packet increases. This can be especially problematic in time-sensitive applications, such as real-time video streaming, where a short delay can ruin the experience. Additionally, Long backoff durations can lead to fairness issues in the network because some stations may have to wait longer than others to access the channel [111]. This can be challenging in high-density networks, where many stations compete for the same channel.

In the following sections, we introduce our DRL mechanism Intelligent Carrier Sense Multiple Access with Collision Avoidance (ICSMA/CA) to improve the CSMA/CA protocol in dense Wi-Fi environments.

4.4.2 Model Design

The optimization problem can be framed as a Markov Decision Process (MDP) consisting of an agent, a set of states S , a set of actions A , a reward function R , a transition probability T , and a reward discount factor $\gamma \in [0,1]$.

The *agent* is installed in a Wi-Fi node. It sends data toward an access point over an interval of time and observes the *state* $s \in S$ of the environment. The state is expressed as the throughput and the backoff duration. Given a state, the agent takes an *action* $a \in A$ by selecting a backoff duration based on a policy (π). The policy is a neural network that provides a set of probabilities, each corresponding to an action. Based on the policy, the agent samples an action corresponding to a new backoff duration. The reward r is then calculated as the overall throughput in the network T_{total} in an attempt to enhance the overall performance of the network.

Throughput is normalized to the range of $[0,1]$ by dividing it by the maximum expected throughput. It is considered the maximum achieved data rate in our configuration.

In our work, we are targeting a generic application. However, for other specific applications, such as video streaming, other metrics can be added to the reward function, such as buffer size or access delay. Note that the fairness issue and energy consumption are out of the scope of our approach, although fairness is just analyzed.

The aim of the model is to select the backoff duration that maximizes the *return* (cumulative reward) of the agent. During the *Offline-training* phase, the agent explores the environment through different simulation scenarios until converging to a policy based on our reward design. At the end of the training, we save the model that contains our final policy to use in the next phase.

During the *Exploitation* phase, we deploy the generated model in new scenarios (different than the ones used in the *Offline-training* phase) where the agent is able to pick with a certain degree of confidence the most suitable backoff duration given a state. Our training method is presented in the next subsection. The technical part is presented in the evaluation subsection.

4.4.3 Model Training

During training, we iterate through multiple episodes. At the beginning of each episode, we initialize an empty batch \mathbf{B} that will hold the (s, a, r) tuples. The tuples are used to update the actor and the critic networks at the end of each episode, where the simulation environment is restarted. When the simulation starts, the agent observes the environment by collecting the state of each Wi-Fi node. A reward is then calculated based on the collected states. Next, the previous state, the action taken in the previous state, and the newly calculated reward are added to \mathbf{B} . The actor-network is then used to predict an action for the current state.

Figure 4.8 shows a detailed representation of the DRL model training in IC-SMA/CA.

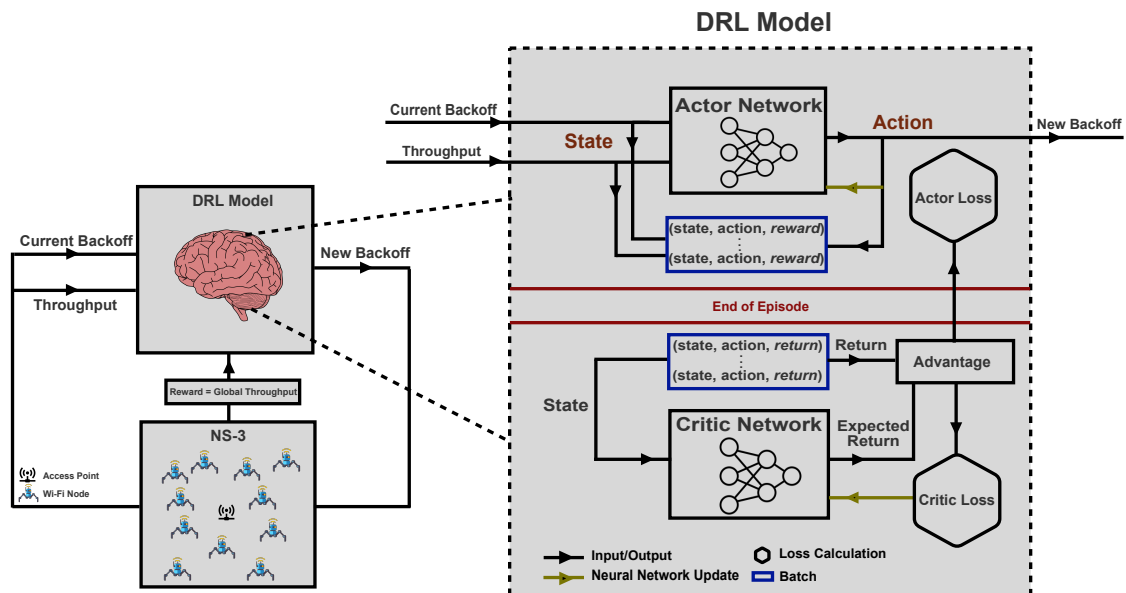


Figure 4.8: A detailed representation of IC-SMA/CA training phase

The DRL model is trained offline over multiple node densities (low, medium, high) to ensure exploring various network conditions. In NS-3, the initial placement and movement directions of the nodes are random. Nodes are placed in a bounded area in which connectivity is guaranteed. Each node sends a flow of 3.5 Mb/s of data packets to a fixed access point. This data rate is sufficient for an HD video transmission [108]. A time interval is set to collect metrics based on multiple transmissions from multiple nodes for two reasons: (1) It allows the calculation of the overall throughput used in our DRL model training. (2) Averaging multiple transmission metrics provides a broader understanding of the performance of the current backoff duration setting. At the end of each interval, the overall throughput is calculated in the simulator based on the acknowledged packets at each node. Then, the overall and node throughput are sent to the DRL module. The current backoff duration and the calculated node throughput of each node are fed to the

neural network of the DRL model as input. The overall throughput is used as the reward. Then, the model sends NS-3 a new backoff duration corresponding to each node. This DRL model is made possible in a simulated network where all data is available to the model. Note that during the exploitation phase, the model only relies on local information for backoff selection.

During the *Offline-training* phase, we rely on global information obtained from the simulation tool. Indeed, this is the main advantage of training in the simulated environment, where we can access all data we need to describe the state of the system. Namely, in our DRL model, we used the global throughput of the network as input data for our model. Nevertheless, during the *Exploitation* phase, which is supposed to be emulating a real-life test, we only use locally available data on each node, namely backoff values and local throughput.

The actor and critic losses are calculated at the end of each episode. The model finishes training when the losses become stable. Figure 4.9 shows the actor and critic losses during the training of the model.

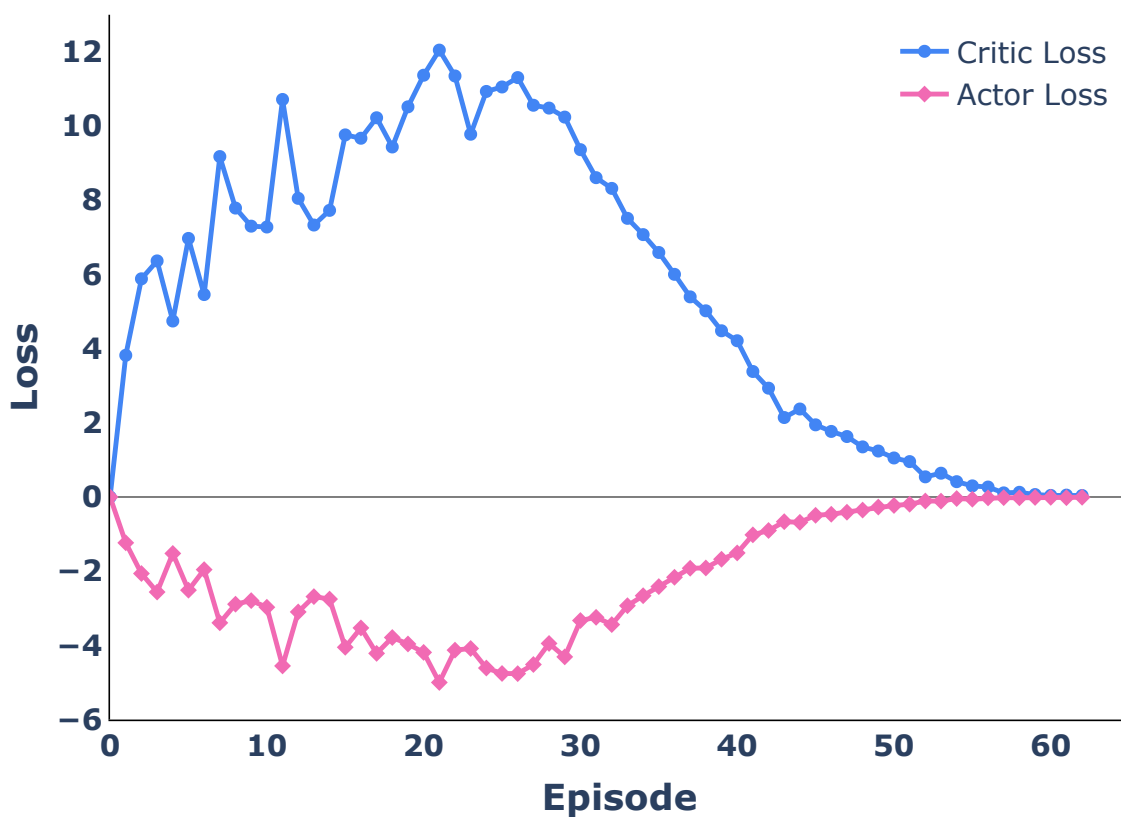


Figure 4.9: Actor and critic losses during the training phase of the ICSMA/CA

4.4.4 Model Evaluation

We validate the trained DRL model in multiple scenarios, each corresponding to a different number of nodes. In each scenario, we fix the number of nodes and conduct 50 rounds of 60-second simulations. In each round, We set a different run number in NS-3 to cover a wide variety of network conditions. The nodes are placed at random locations at the beginning of each round. The results are collected and averaged over all rounds. Finally, we compare the results obtained using the ICSMA/CA model to the standard baseline using the default CSMA/CA access mechanism. Results in terms of network capacity, ratio of collisions, access delay (time needed for a

node to be able to send the frame once it is de-queued at the Medium Access Control sublayer), and fairness are shown. Note that for each result, we varied the number of nodes from 1 to 100 with a step size of 5 nodes. In the graphs, the x axis represents the total amount of traffic generated by all nodes in the network. The simulation parameters are shown in table 4.2.

Table 4.2: Simulation parameters table for the CSMA/CA optimization.

Parameter	Value
Simulation time (Validation)	60 seconds
Runs	50
WLAN standard	IEEE 802.11ac
Data Rate	3.5 Mb/s
Path loss model	Log-distance
Fading factor	random(0, 2dB)
Traffic	UDP
Time interval	400 ms
Channel Width	20 MHz
Packet size	1500 Bytes
Mobility model	Random Walk 2d Mobility Model
Mobility speed	4 m/s
Topology size	Square of boundaries (-30, 30, -30, 30)
α	0.8
β	0.2
Learning rate	0.001

Figure 4.10 shows the normalized throughput in the network in terms of offered load. Normalized throughput is the ratio of the overall throughput divided by the maximum achievable data rate in the 802.11ac for an MCS (Modulation and Coding Scheme) index 8 with 20 MHz channel width: 86.7 Mb/s [1]. The overall throughput is calculated based on what the access point has correctly received. Note that this is also applicable to Wi-Fi 6 and any other version of Wi-Fi using CSMA/CA. Both the baseline and the ICSMA/CA achieve close results before the network saturation. However, ICSMA/CA achieves higher capacity utilization as the offered load increases. The standard CSMA/CA saturation point is reached at around 100 Mb/s achieving 65% of the network's capacity. At the same offered load value, ICSMA / CA reaches 80% of network capacity. As we continue increasing the offered load, standard CSMA/CA experiences degradation, and its total capacity reaches around 50% for an offered load of 250 Mb/s, whereas our ICSMA/CA maintains its 80% capacity results for offered loads between 100 and 350 Mb/s.

In the standard CSMA/CA mechanism, the contention window doubles after each collision, then a backoff duration is selected randomly. However, the ICSMA/CA is trained to directly select a backoff duration that maximizes the overall throughput of the network. The improvement in capacity utilization over standard CSMA/CA is due to the ability of ICSMA/CA to reduce collisions and access delay, making access to the channel more efficient. These results are shown in Figures 4.11 and 4.12, respectively.

The collision rate of ICSMA/CA remains lower than that of the baseline for all tested scenarios. The access delay is directly related to the number of repetitions that each frame has to undergo before the access point receives it correctly.

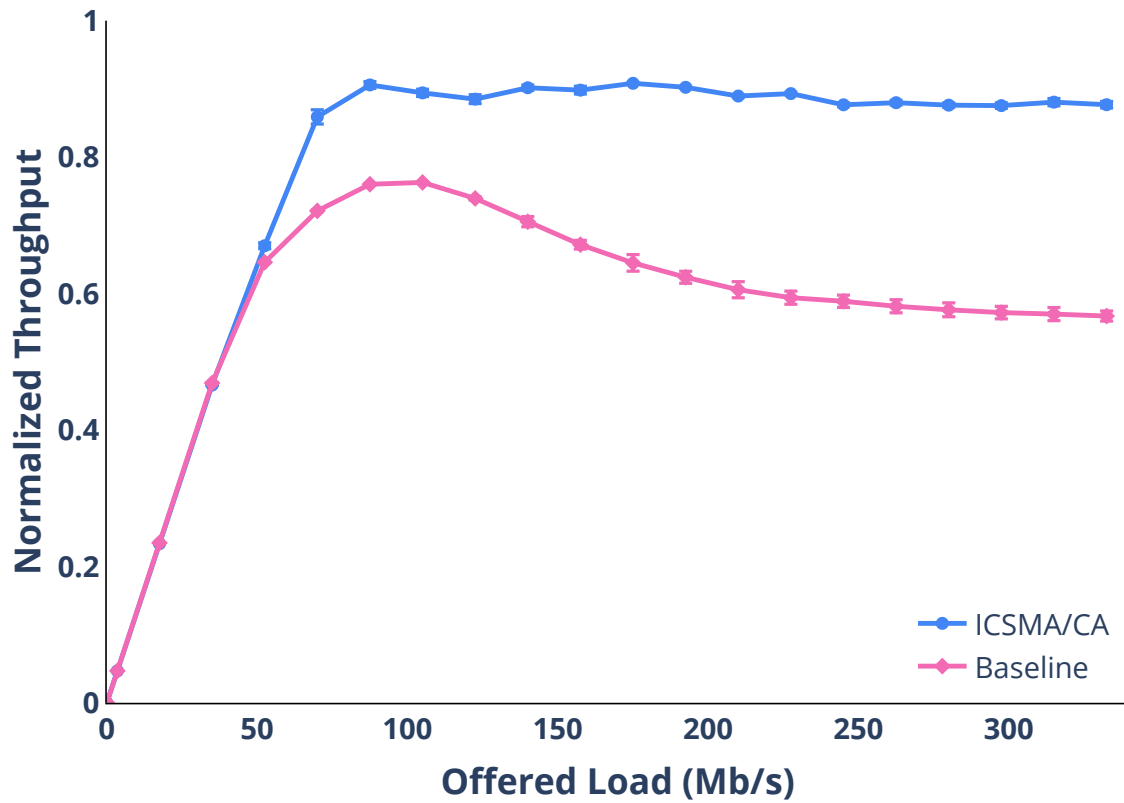


Figure 4.10: Normalized throughput in the network in terms of the offered load in Mb/s.

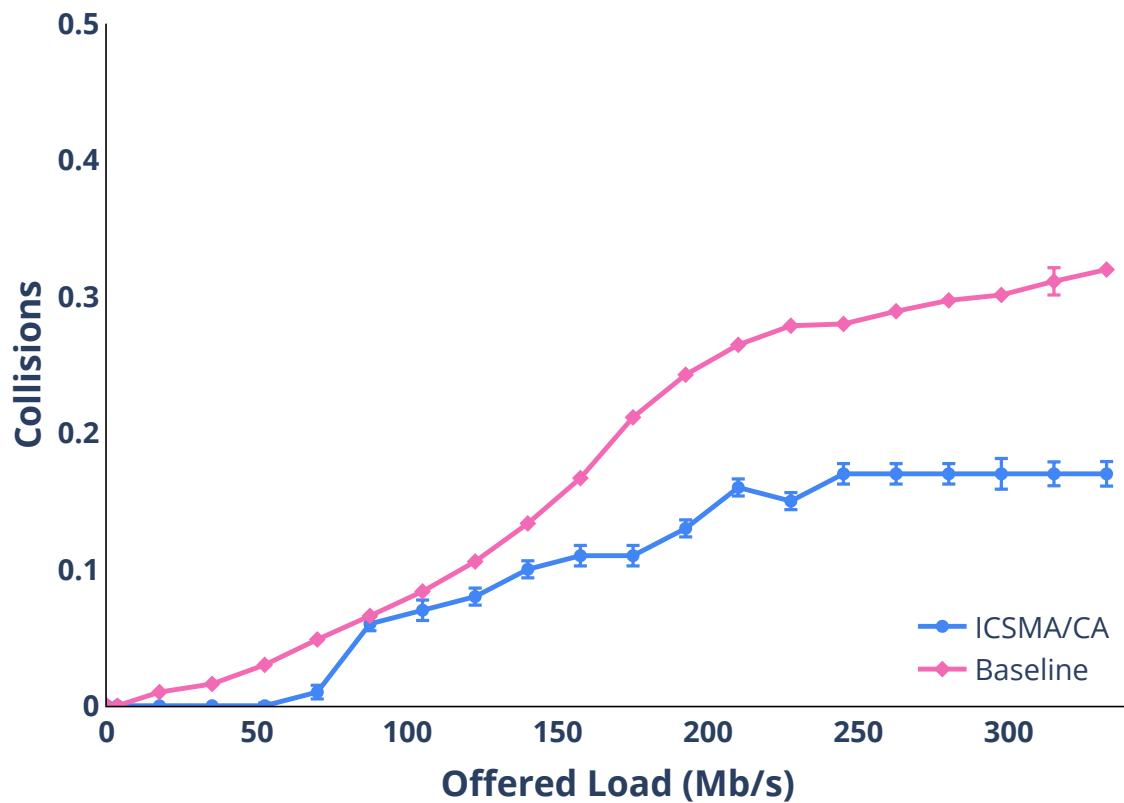


Figure 4.11: Collisions in the network in terms of the offered load in Mb/s.

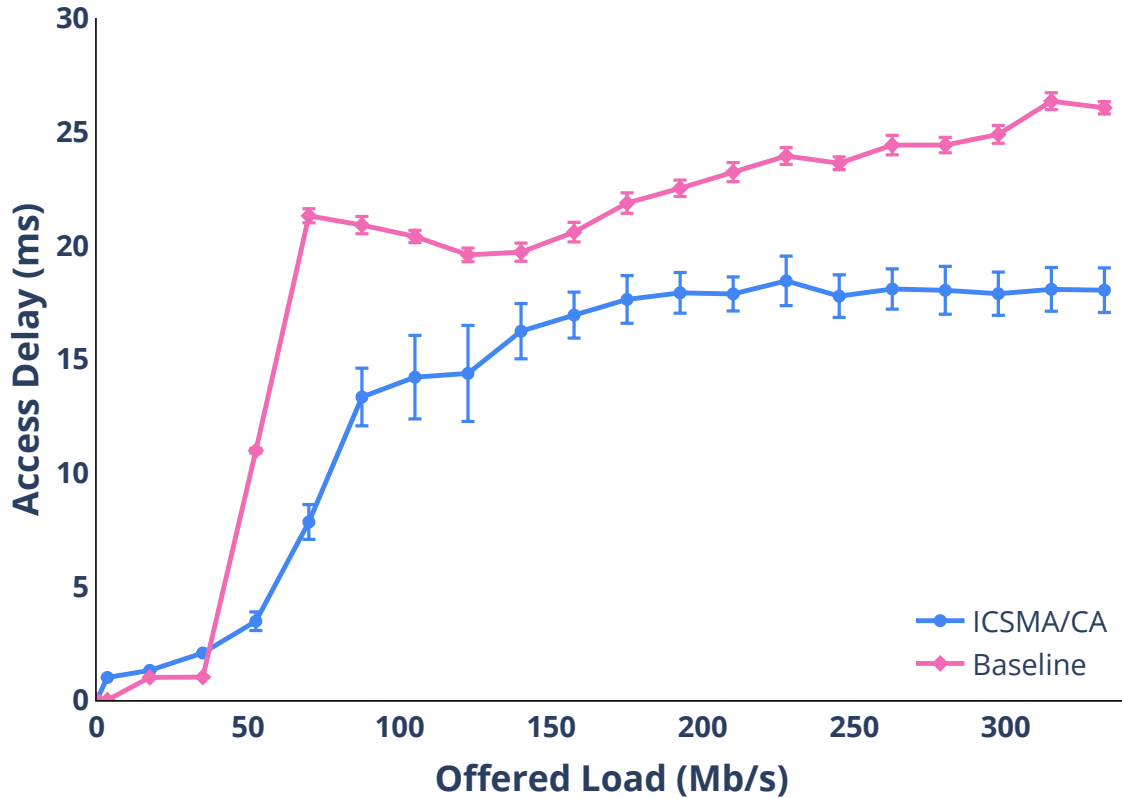


Figure 4.12: Access Delay (ms) in the network in terms of the offered load in Mb/s.

Fewer collisions lead to fewer repetitions which leads to less access delay. It can be seen from both figures that the number of collisions and the access delay are very correlated.

Access delay increases whenever we introduce new nodes to the network. Upon reaching around 200 Mb/s of offered load, the access delay does not change due to the equilibrium reached in the network.

Figure 4.13 shows the fairness in channel utilization among the nodes. Fairness is calculated using Jain’s fairness index [112]. Although fairness is not the aim of our model, using ICSMA/CA improved fairness among the users compared to that of the baseline. The ability of our model to refrain from selecting random large backoff durations and to reduce the access delay optimized the channel utilization among the nodes.

4.5 Complexity

Our DRL models are trained completely offline inside the simulator. The time complexity of training can vary depending on the specifications of the device used for simulation. Since the resulting model may be deployed on end devices with limited computational power and memory, we are interested in both the time complexity and memory footprint of the DRL model. Thus, we measured the Random Access Memory (RAM) usage of the model during inference, which was found to be less than 1 Kilobyte. Additionally, the average time consumed during inference was found to be 8 microseconds per step, which is less than the slot duration in 802.11ac (9 microseconds) [37].

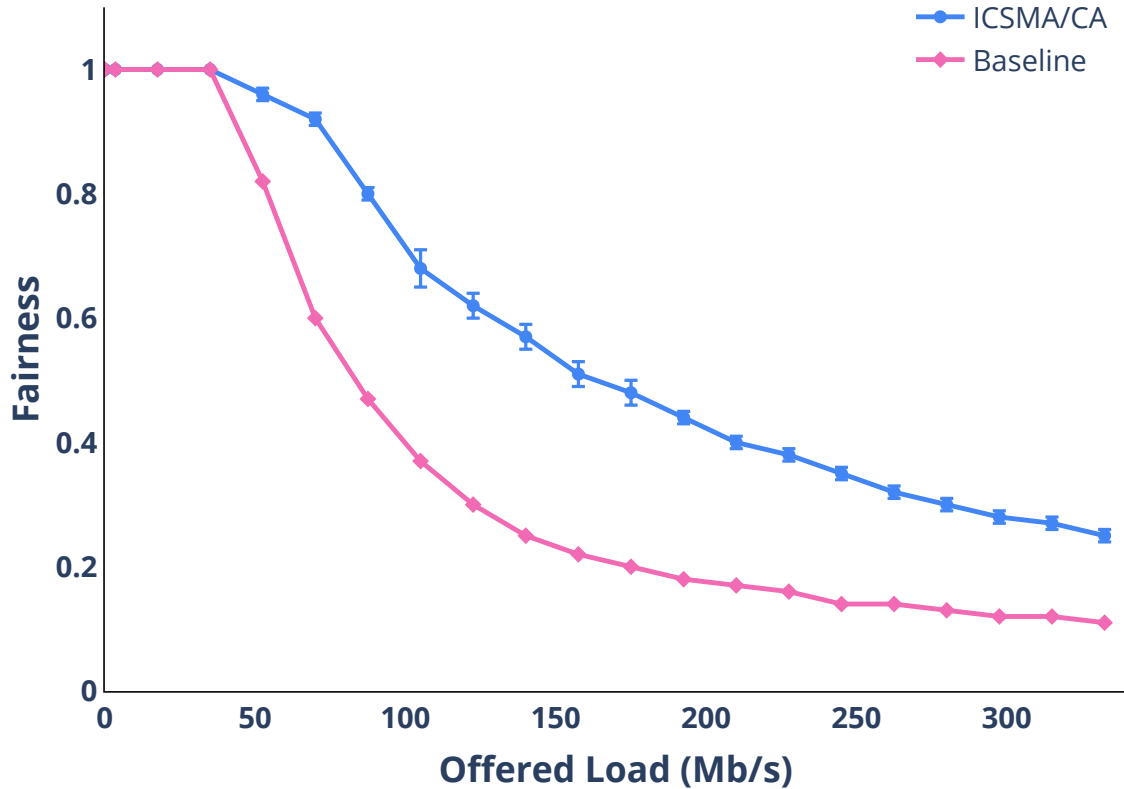


Figure 4.13: Fairness in the network in terms of the offered load in Mb/s

4.6 Conclusion

In this chapter, we proposed two DRL mechanisms to improve the performance of Wi-Fi networks under high offered load situations. First, we proposed a DRL mechanism to adapt application-level data rates based on the locally observed network state. The proposed mechanism learns to choose the most suitable data rate dynamically and is trained offline using network simulation. We evaluated the produced DRL model through simulation. Results showed that the proposed mechanism achieved better goodput performance during network saturation compared to a similar approach from the literature and the baseline.

In the second part, we proposed another DRL mechanism to enhance the multiple access method CSMA/CA by selecting the most suitable backoff duration based on the state of the network. We evaluated the DRL model produced by the proposed mechanism (ICSMA/CA). Results showed that the proposed mechanism improved the overall throughput of the network, reduced access delay, and increased fairness among the nodes.

Chapter 5

Conclusions and Future Directions

5.1 Conclusions

In this thesis we provided a comprehensive study of Wi-Fi networks in the context of smart farming, addressing their underlying technologies, challenges, and limitations and offering novel solutions by leveraging machine learning techniques, specifically deep reinforcement learning. The primary contributions of this thesis are summarized in this section.

5.1.1 Evaluating and Selecting the Best Rate Adaptation Algorithm

Rate adaptation algorithms help wireless networks achieve better performance by selecting the most suitable MCS value, depending on the channel state. These algorithms are rarely specified in wireless standards, and they are left for the constructors to implement. In our work, we have presented several rate adaptation algorithms and grouped them into different categories according to how they function. We also provided an analysis of these algorithms highlighting their pros and cons. We implemented algorithms of different categories in the NS-3 simulator and performed a comparison study for Ad hoc and infrastructure modes to select the best-performing one.

This study provided insights into how the assessment technique of the quality of the wireless channel impacts the performance of Wi-Fi networks and how to improve the data transmission rate to enhance throughput and reliability. Once an appropriate rate adaptation algorithm is selected based on the performance evaluation, it can be deployed in Wi-Fi access points.

5.1.2 A Study on Capacity-Aware Deployment

When it comes to guaranteeing a certain level of QoS in a Wi-Fi network, deployment optimization should take into account the requirements and constraints of the applications. The deployment study involves using the right number of access points and their positioning to reach the needed network performances. Beamforming is one of the most efficient techniques for enhancing performance in a Wi-Fi network. It allows increasing coverage and data rate.

We investigated the impact of beamforming on the performance of Wi-Fi networks that monitor and control mobile robots in Smart Farming applications. We showed how the coverage area of an access point is calculated and the enhancement

that beamforming brings to the coverage and capacity of a Wi-Fi network. A deployment strategy inspired by cellular networks grid deployment was used, and multiple simulation scenarios were tested using the NS-3 simulator, taking into consideration different application data rate requirements. We also presented a deployment strategy that would be less complex by placing access points on the edge of the area of interest. Based on a scenario inspired by the use of autonomous mobile robots in Smart Farming, we conducted a comparison between different deployment strategies.

In our study, we found that a balance between deployment cost, complexity, and network performance must be achieved. While it is crucial to decide on a deployment plan, we also recognized the potential for improving various layers of the Wi-Fi network to address additional challenges that can affect Wi-Fi network performance.

5.1.3 Wi-Fi Performance Optimization using Reinforcement Learning

Wi-Fi networks suffer from performance degradation under high offered load situations. We explored different saturation situations, showing how Wi-Fi achieves low goodput under high offered loads. Then, we proposed two DRL mechanisms to tackle this issue on the application and MAC layers. The DRL models were trained and evaluated in NS-3, a cost-free approach that can be done before real deployments. The scenarios included mobility and fading to simulate a real wireless environment.

On the application layer, we investigated application-level data rate adaptation using DRL, which allows for adapting the offered load depending on the locally observed network state. The proposed DRL mechanism considers the current performance of the network to choose the most suitable application data rate dynamically. We demonstrated how the training achieved better results in terms of goodput under network congestion compared to a similar learning approach and baseline cases.

On the MAC layer, we tackled the degradation of the performance in dense networks caused by the random CSMA/CA protocol. We proposed a DRL mechanism to enhance CSMA/CA in Wi-Fi. Standard CSMA/CA does not have a way to choose the best backoff duration for a particular network condition. Instead, the backoff duration is randomly picked from a contention window and doubled after each collision. Our proposed DRL model enhances CSMA/CA by selecting the most suitable backoff value depending on the state of the network. The DRL model was evaluated in different situations with varying node densities and compared to the standard CSMA/CA.

While our results were promising, there is still significant potential and considerable scope for improving the performance of Wi-Fi networks.

5.2 Future Directions

The contributions of this thesis can be extended in several directions, which are listed in this section.

5.2.1 Potential Contributions to Rate Adaptation Algorithms

We primarily focused on evaluating existing rate adaptation algorithms rather than developing our own. Our objective was to identify the most effective algorithm to incorporate into our primary thesis target, given that rate adaptation plays a critical role at the bottom of the network stack and significantly impacts network performance. Our literature review found that there is still scope for enhancing rate adaptation algorithms. Therefore, we have identified several areas for improvement that can be explored in future research.

Wi-Fi channel characteristics differ for the uplink and downlink directions, which poses a significant challenge for designing effective rate adaptation algorithms. For instance, calculating the SINR of the ACK frames from the transmitter side does not necessarily reflect SINR as seen from the receiver side. In order to address this issue, it might be necessary to combine metrics from both the transmitter and receiver sides to obtain a more comprehensive view of the network conditions. Metrics from the transmitter side, such as packet loss and throughput, can provide information about the channel from the point of view of the transmitter. In contrast, metrics from the receiver side, such as SINR, can provide information about the quality of the received signal. Combining these metrics can make it possible to develop more robust rate adaptation algorithms that consider metrics as seen from both the transmitter and receiver. These algorithms may not necessarily only use traditional statistical methods to analyze the metrics and make decisions.

Incorporating machine learning can help identify the optimal combination of metrics for a novel rate adaptation algorithm. This involves, for instance, employing reinforcement learning to dynamically adjust various network metrics such as MCS values, channel widths, and guard intervals in real time based on changing network conditions. Such an approach can result in more efficient and effective rate adaptation decisions, enhancing network performance. Furthermore, future research may consider factors such as the type of data being transmitted, its priority level, and the application data rate requirements to improve the algorithm further.

These techniques would lead to further improvement in the performance of Wi-Fi devices, but it is also important to consider energy consumption, especially in domains where communication nodes are battery-powered. Rate adaptation algorithms that prioritize performance over energy consumption may lead to faster battery drain and reduced battery life of the Wi-Fi device. This requires a trade-off between performance and energy consumption. Such rate adaptation algorithms must consider the current battery level and transmission power to balance performance and energy consumption. One significant option in this context could be to deploy RAAs and collect information about the difference in battery consumption of Wi-Fi devices between different RAA solutions from real deployments.

Furthermore, real hardware testing can investigate the effectiveness of rate adaptation algorithms in real-world scenarios. While simulations are useful for understanding the behavior of rate adaptation algorithms, they do not always accurately reflect the complexity and dynamic nature of Wi-Fi network conditions. Real hardware testing allows the evaluation of the performance of rate adaptation algorithms under real-world conditions and identifies any potential issues or defects in simulations. Note that experimental evaluations requires hardware that gives access to the metrics involved in the rate decision such as the SINR which is available in the ath10k wireless driver [113], but not in other devices such as the raspberry pi and Esp32.

5.2.2 Possible Enhancements to the Capacity-Aware Deployment

Our work focused on the deployment of Wi-Fi networks for the use-case of monitoring and remote control of autonomous mobile robots in Smart Farming. We evaluated the usage of beamforming in various deployment techniques. Although the benefits of beamforming are promising, we assumed that beamforming is perfectly implemented and optimal on both sides of the transmission link. In real Wi-Fi deployments, the beamforming gain may not reach the theoretical gain shown in our work due to hardware limitations and the environmental impact, this gain will be less in terms of dBs. We have identified several areas for improvement that can be explored in future research:

To improve the accuracy of beamforming simulations, a more realistic model could be developed in the simulator that better reflects the functionality of beamforming in real-world deployments. This would involve mathematical modeling, the collection of deployment coverage graphs for reference, and the accurate implementation of the model inside the simulator. Furthermore, The enhancement of the performance of Wi-Fi devices is not only limited to leveraging only the beamforming technology.

The employment of spatial multiplexing to improve network throughput and reduce collisions could be investigated. Spatial multiplexing can significantly improve network throughput by allowing the transmission of multiple data streams simultaneously. However, using spatial multiplexing in dense scenarios can be complex, as the number of antennas required can increase significantly, and interference can be a significant challenge. Therefore, The trade-offs between the benefits of spatial multiplexing and the complexity of its implementation in dense scenarios could be explored. Also, the combination of beamforming and spatial multiplexing could be looked into. One could also possibly attempt to reduce the cost of deployment of complex access points with the ability to combine spatial multiplexing and beamforming by using multi-hop networks.

Multi-hop networks can reduce the number of access points required to cover a given area, resulting in cost savings and simplifying network management. Additionally, robots can be used as relay nodes to extend the coverage area and provide additional connectivity in areas that are difficult to reach. However, robots in multi-hop networks may bottleneck the throughput due to various factors, such as distance and obstacles. Therefore, trade-offs between the benefits of multi-hop networks and the drawbacks that come with this approach could be explored. Studying the optimal placement of access points and the work area of robots are equally important considerations in this context.

In our work, we used fixed configurations for deployment (edge and grid deployments). However, using optimization techniques such as genetic algorithms to optimize the placement of access points can be looked into in the future. These techniques automate selecting the optimal placement of access points by evaluating various combinations of access point locations and configurations depending on the size and geometry of the field. By using genetic algorithms, one can develop more effective deployment strategies that are optimized for the specific needs and requirements of Wi-Fi networks. Additionally, this approach can help identify the most suitable type of access point for each location and the best possible configuration settings, providing optimal performance, coverage, and user experience while minimizing the costs of deployment.

The deployment optimization study can be extended to include other perfor-

mance criteria in the objective function. While our current study focused on optimizing throughput based on the application requirements as the primary target, other metric constraints, such as latency, greatly impact the QoE in certain applications. By incorporating these metrics into the objective function of deployment optimization, more effective deployment strategies could be developed.

5.2.3 Enhancement of Reinforcement Learning Approaches for Wi-Fi Optimization

While the proposed reinforcement learning mechanisms in this thesis have significantly enhanced the performance of Wi-Fi networks, there is still considerable room for improvement, along with several issues that must be addressed. These issues include real-world applications and testing, additional design objectives and metrics, and cross-layer mechanisms. In what follows, we outline some potential future directions to address these challenges:

Our work focused on certain metrics, including goodput and packet loss on the application layer and throughput on the physical layer. However, the analysis could be extended to include additional metrics that are also related to the tackled problem, such as access delay. Furthermore, using cross-layer metrics could provide a more comprehensive view of network conditions and performance across multiple layers of the network stack. Additionally, using more realistic applications would be an added value to this study. For instance, modern applications, such as video streaming, involve more complex processes than adjusting the data rate. For example, Dynamic Adaptive Streaming over HTTP (DASH) [114] incorporates buffer management and network congestion control, as well as adaptation algorithms, on top of the Wi-Fi network stack. To more accurately simulate video streaming and evaluate its performance, video streaming protocols such as DASH could be integrated into the study.

In addition to adding more metrics to the study, more objectives could be added to the RL mechanism. The quality of the RL model heavily relies on the reward function, which defines the objective of the model. Typically, RL approaches define a reward function based on the application objective. However, a multi-objective reward design could be implemented to account for additional objectives, such as fairness. This approach would enable the investigation of the trade-off between maximizing metrics, such as throughput, while maintaining an acceptable level of user fairness. Furthermore, creating a good reward design is not enough, as RL algorithms usually need different hyperparameters that should be set differently for distinct applications and use cases.

Hyperparameter tuning has an enormous impact on the accuracy of the trained models. Tuning these parameters can make RL models more reliable for wireless environments, which are dynamic by nature. While automatic tuning of hyperparameters was not considered in our work due to the time consumed, it can be a useful approach for optimizing RL models. However, NS-3 simulations with a large number of nodes can be time-consuming, particularly with multiple runs. Therefore, hyperparameter tuning could be initially considered for a smaller number of nodes to generate potentially better results while minimizing the time and computational resources required.

In this study, we evaluated a DRL algorithm from the actor-critic family. However, conducting a comparative study between multiple RL algorithms from different categories could provide a more comprehensive understanding of their respective

advantages and disadvantages for Wi-Fi optimization. This approach would allow the identification of the most suitable RL algorithm for a given Wi-Fi optimization problem based on factors such as performance, complexity, and computational requirements.

Testing DRL models only in simulations has limitations, and conducting real-world testing provides several benefits. Real-world testing offers an opportunity to validate the performance of DRL models under real conditions, which may differ from simulation scenarios, and provide a more accurate assessment of their performance. Additionally, real-world testing can help evaluate the robustness of the DRL model and identify areas for optimization and improvement that may not have been encountered in simulations.

Publications

Certain contributions of this thesis have been published in international journal articles and in an international peer-reviewed conference:

- Sammour I, Chalhoub G. Evaluation of Rate Adaptation Algorithms in IEEE 802.11 Networks. *Electronics*. 2020; 9(9):1436. [22]
- Sammour I, Chalhoub G., Application-Level Data Rate Adaptation in Wi-Fi Networks Using Deep Reinforcement Learning. 2022 IEEE 96th Vehicular Technology Conference (VTC2022-Fall), London, United Kingdom, 2022, pp. 1-7. [23]
- André, G., Bachelet, B., Battistoni, P. et al. LambdaAgrIoT: a new architecture for agricultural autonomous robots' scheduling: from design to experiments. *Cluster Comput* (2022). [115]

Other contributions were submitted to international journals, and we are currently awaiting the outcome of the review process:

- Sammour I., Chalhoub G., De Sousa G. Capacity Aware Wi-Fi Networks Deployment.
- Sammour I., Chalhoub G. Intelligent CSMA/CA for Wi-Fi networks.
- Ergun S., Sammour I., Chalhoub G. A Survey on How Network Simulators Serve Reinforcement Learning in Wireless Networks.

Bibliography

- [1] <https://wlanprofessionals.com/mcs-table-and-how-to-use-it/>. Accessed: 2023-02-28.
- [2] Sushma. U. Bhoover, Anusha Tugashetti, and Pratiksha Rashinkar. V2x communication protocol in vanet for co-operative intelligent transportation system. *2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, pages 602–607, 2017.
- [3] Wenli Yang, Xiaojing Wang, Xianghui Song, Yun Yang, and Srikanta Patnaik. Design of intelligent transportation system supported by new generation wireless communication technology. *International Journal of Ambient Computing and Intelligence*, 9:78–94, 01 2018.
- [4] Haider Mahmood Jawad, Rosdiadee Nordin, Sadik Kamel Gharghan, Aqeel Mahmood Jawad, and Mahamod Ismail. Energy-efficient wireless sensor networks for precision agriculture: A review. *Sensors*, 17(8), 2017.
- [5] Chamil Kulatunga, Laurence Shalloo, William Donnelly, Eric Robson, and Stepan Ivanov. Opportunistic wireless networking for smart dairy farming. *IT Professional*, 19(2):16–23, 2017.
- [6] Dangi Ramraj, Lalwani Praveen, Choudhary Gaurav, You Ilsun, and Pau Giovanni. Study and investigation on 5g technology: A systematic review. *Sensors*, 2022.
- [7] Chilakala Sudhamani, Mardeni Roslee, Jun Jiat Tiang, and Aziz Ur Rehman. A survey on 5g coverage improvement techniques: Issues and future challenges. *Sensors*, 2023.
- [8] Kaveh Pahlavan and Prashant Krishnamurthy. Evolution and impact of wi-fi technology and applications: A historical perspective. *International Journal of Wireless Information Networks*, 2020.
- [9] Zeadally Sherali, Siddiqui Farhan, and Baig Zubair. 25 years of bluetooth technology. *Future Internet*, 2019.
- [10] Mario Collotta, Giovanni Pau, Timothy Talty, and O.K. Tonguz. Bluetooth 5: A concrete step forward toward the iot. *IEEE Communications Magazine*, 56, 07 2018.
- [11] Mustafa Ergen. *WiMAX Network Layer*, pages 341–378. Springer US, Boston, MA, 2009.
- [12] Haider AH Alobaidy, JS Mandeep, Rosdiadee Nordin, and Nor Fadzilah Abdullah. A review on zigbee based wsns: concepts, infrastructure, applications, and challenges. *Int. J. Electr. Electron. Eng. Telecommun*, 9(3):189–198, 2020.

- [13] Bharat S Chaudhari, Marco Zennaro, and Suresh Borkar. Lpwan technologies: Emerging application characteristics, requirements, and design considerations. *Future Internet*, 12(3):46, 2020.
- [14] Oltjon Kodheli, Eva Lagunas, Nicola Maturo, Shree Krishna Sharma, Bhavani Shankar, Jesus Fabian Mendoza Montoya, Juan Carlos Merlano Duncan, Danilo Spano, Symeon Chatzinotas, Steven Kisseleff, et al. Satellite communications in the new space era: A survey and future challenges. *IEEE Communications Surveys & Tutorials*, 23(1):70–109, 2020.
- [15] Kumar Sachin, Tiwari Prayag, and Zymbler Mikhail. Internet of things is a revolutionary approach for future technology enhancement: a review. *Journal of Big Data*, 2019.
- [16] Charlie Wilson, Tom Hargreaves, and Richard Hauxwell-Baldwin. Benefits and risks of smart home technologies. *Energy Policy*, 2017.
- [17] Elsayed Said Mohamed, AA. Belal, Sameh Kotb Abd-Elmabod, Mohammed A El-Shirbeny, A. Gad, and Mohamed B Zahran. Smart farming for improving agricultural management. *The Egyptian Journal of Remote Sensing and Space Science*, 2021.
- [18] Li Shuo, Edwards Simon, Isik Memduh Ozer, Zhang Yanghanzi, and Blythe Philip T. Qualitative examination of cooperative-intelligent transportation systems in cities to facilitate large-scale future deployment. *Sensors*, 2022.
- [19] Laura García, Jose Jimenez, Miran Taha, and Jaime Lloret. Wireless technologies for iot in smart cities. *Network Protocols and Algorithms*, 10:23, 04 2018.
- [20] Martin Wollschlaeger, Thilo Sauter, and Juergen Jasperneite. The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0. *IEEE Industrial Electronics Magazine*, 11(1):17–27, 2017.
- [21] Nadege Levallet, Norm O’Reilly, Elizabeth Wanless, Michael Naraine, Ethan Alkon, and Wade Longmire. Enhancing the fan experience at live sporting events: The case of stadium wi-fi. *Case Studies In Sport Management*, 8:6–12, 01 2019.
- [22] Ibrahim Sammour and Gerard Chalhoub. Evaluation of rate adaptation algorithms in iee 802.11 networks. *Electronics*, 2020.
- [23] Ibrahim Sammour and Gerard Chalhoub. Application-level data rate adaptation in wi-fi networks using deep reinforcement learning. *2022 IEEE 96th Vehicular Technology Conference (VTC2022-Fall)*, 2022.
- [24] Vikram Singh and Lalit Awasthi. A review paper on iee 802.11 wlan. 2014.
- [25] IEEE standard for information technology - telecommunications and information exchange between systems - local and metropolitan networks - specific requirements - part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications: Higher speed physical layer (phy) extension in the 2.4 ghz band. *IEEE Std 802.11b-1999*, pages 1–96, 2000.

- [26] Ian Sherlock. Wi-fi alliance: Connecting everyone and everything, everywhere. *IEEE Communications Standards Magazine*, 2017.
- [27] IEEE standard for telecommunications and information exchange between systems - lan/man specific requirements - part 11: Wireless medium access control (mac) and physical layer (phy) specifications: High speed physical layer in the 5 ghz band. *IEEE Std 802.11a-1999*, pages 1–102, 1999.
- [28] IEEE standard for information technology– local and metropolitan area networks– specific requirements– part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications: Further higher data rate extension in the 2.4 ghz band. *IEEE Std 802.11g-2003 (Amendment to IEEE Std 802.11, 1999 Edn. (Reaff 2003) as amended by IEEE Stds 802.11a-1999, 802.11b-1999, 802.11b-1999/Cor 1-2001, and 802.11d-2001)*, pages 1–104, 2003.
- [29] IEEE standard for information technology– local and metropolitan area networks– specific requirements– part 11: Wireless lan medium access control (mac)and physical layer (phy) specifications amendment 5: Enhancements for higher throughput. *IEEE Std 802.11n-2009 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, and IEEE Std 802.11w-2009)*, pages 1–565, 2009.
- [30] IEEE standard for information technology– telecommunications and information exchange between systemslocal and metropolitan area networks– specific requirements–part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications–amendment 4: Enhancements for very high throughput for operation in bands below 6 ghz. *IEEE Std 802.11ac-2013 (Amendment to IEEE Std 802.11-2012, as amended by IEEE Std 802.11ae-2012, IEEE Std 802.11aa-2012, and IEEE Std 802.11ad-2012)*, pages 1–425, 2013.
- [31] IEEE standard for information technology–telecommunications and information exchange between systems local and metropolitan area networks–specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 1: Enhancements for high-efficiency wlan. *IEEE Std 802.11ax-2021 (Amendment to IEEE Std 802.11-2020)*, pages 1–767, 2021.
- [32] IEEE draft standard for information technology–telecommunications and information exchange between systems local and metropolitan area networks–specific requirements - part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment: Enhancements for extremely high throughput (eht). *IEEE P802.11be/D3.0, January 2023*, pages 1–999, 2023.
- [33] Renato Vicente, David Saad, and Yoshiyuki Kabashima. Low-density parity-check codes—a statistical physics perspective. 2003.
- [34] K Vlvcek, D MIKLIK, J KOVALSKY, and J MITRYCH. Turbo coding performance and implementation. *Proc. of the IEEE DDECS 2000 Workshop*, 2000.

- [35] Eustathia Ziouva and Theodore Antonakopoulos. Cdma/ca performance under high traffic conditions: throughput and delay analysis. *Computer communications*, 25(3):313–321, 2002.
- [36] Barry D Van Veen and Kevin M Buckley. Beamforming: A versatile approach to spatial filtering. *IEEE assp magazine*, 5(2):4–24, 1988.
- [37] Matthew S. Gast. *802.11ac: A Survival Guide*. ”O’Reilly Media, Inc.”, 2013.
- [38] https://www.itu.int/dms_pub/itu-r/opb/rep/R-REP-SM.2153-5-2015-PDF-E.pdf. Accessed: 2023-03-15.
- [39] Saad Biaz and Shaoen Wu. Rate adaptation algorithms for ieee 802.11 networks: A survey and comparison. pages 130–136, 2008.
- [40] Wei Yin, Peizhao Hu, Jadwiga Indulska, Marius Portmann, and Ying Mao. Mac-layer rate control for 802.11 networks: a survey. *Wireless Networks*, 2020.
- [41] James Gross, Florian Schmidt, Oscar Punyal, Anwar Hithnawi, and Klaus Wehrle. A receiver based 802.11 rate adaptation scheme with on-demand feedback. *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2012.
- [42] Okhwan Lee, Jihoon Kim, Jong tae Lim, and Sunghyun Choi. Sira: Snr-aware intra-frame rate adaptation. *IEEE Communications Letters*, 2015.
- [43] Seongho Byeon, Kangjin Yoon, Changmok Yang, and Sunghyun Choi. Strale: Mobility-aware phy rate and frame aggregation length adaptation in wlans. pages 1–9, 2017.
- [44] Teuku Arif, Rizal Munadi, and Fardian Fardian. Evaluation of the minstrelht rate adaptation algorithm in ieee 802.11n wlans. *International Journal of Simulation: Systems, Science and Technology*, 2017.
- [45] Ioannis Pefkianakis, Suk-Bok Lee, and Songwu Lu. Towards mimo-aware 802.11n rate adaptation. *IEEE/ACM Transactions on Networking*, 2013.
- [46] Duy Nguyen and J. J. Garcia-Luna-Aceves. A practical approach to rate adaptation for multi-antenna systems. page 331–340, 2011.
- [47] Lito Kriara and Mahesh K. Marina. Samplelite: A hybrid approach to 802.11n link adaptation. *SIGCOMM Comput. Commun. Rev.*, 2015.
- [48] Ioannis Selinis, Konstantinos Katsaros, Seiamak Vahid, and Rahim Tafazolli. Damysus: A practical ieee 802.11ax bss color aware rate control algorithm. *International Journal of Wireless Information Networks*, 2019.
- [49] Wei Yin, Peizhao Hu, Jadwiga Indulska, and Konstanty S. Bialkowski. Performance of mac80211 rate control mechanisms. *International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2011.
- [50] S. Khan, S. A. Mahmud, H. Nouredine, and H. S. Al-Raweshidy. Rate-adaptation for multi-rate ieee 802.11 wlans using mutual feedback between transmitter and receiver. *21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2010.

- [51] Tingpei Huang, Haiming Chen, Zhaoliang Zhang, and li Cui. Easira: A hybrid rate adaptation scheme for 802.11 mobile wireless access networks. *IEEE Wireless Communications and Networking Conference, WCNC*, 2012.
- [52] Theodore S. Rappaport. *Wireless communications - principles and practice*. Prentice Hall, 1996.
- [53] C.E. Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 1949.
- [54] N. Chiurtu, B. Rimoldi, and E. Telatar. *On the capacity of multi-antenna Gaussian channels*. 2001.
- [55] Herbert Taub and Donald L. Schilling. *Principles of Communication Systems*. McGraw-Hill Higher Education, 2nd edition, 1986.
- [56] Ruiguang Tang, Xiao Zhou, and Chengyou Wang. Singular value decomposition channel estimation in stbc mimo-ofdm system. *Applied Sciences*, 2019.
- [57] R. G. Gallager. *Information Theory and Reliable Communication*. John Wiley Sons, Inc., 1968.
- [58] Mohammad-Ali Khalighi, Kosai Raouf, and Geneviève Jourdain. Capacity of wireless communication systems employing antenna arrays, a tutorial study. *Wireless Personal Communications*, 2002.
- [59] Maan Muataz Abdulwahid, M Al-Hakeem, Mahmood Mosleh, and Raed Abd-Alhameed. Investigation and optimization method for wireless ap deployment based indoor network. *IOP Conference Series: Materials Science and Engineering*, 2020.
- [60] REMCOM Inc. The wireless insite user's manual, 2009.
- [61] Mevlüt Ersoy, Tuncay Yigit, and Asım Yüksel. A decision support tool for indoor 801.11ac wlan modeling using optimization techniques. *El-Cezeri Fen ve Mühendislik Dergisi*, 2020.
- [62] Tomás de Jesús Mateo Sanguino and Jhon Carlos Mendoza Betancourt. Optimal modeling of wireless lans. *Complex.*, 2018.
- [63] Yuan Zhou, Zezhou Luo, and Hongcheng Zhuang. Sensor-assisted coverage self-optimization for wireless local area network. 2013.
- [64] Katia Jaffrès-Runser, Jean-Marie Gorce, and Stéphane Ubéda. Mono and multiobjective formulations for the indoor wireless lan planning problem. *Computers & Operations Research*, 2008.
- [65] S. Kouhbor, J. Ugon, A. Kruger, and A. Rubinov. Optimal placement of access point in wlan based on a new algorithm. 2005.
- [66] Bushra AL Maamari. Measurement and optimisation of ap placement and channel assignment in wireless lan. *International Journal of Information Technology, Research and Applications*, 2022.
- [67] Farhana Binte Kamrul Easha, Robert Abbas, and Matthew Daley. Campus wi-fi coverage mapping and analysis. *ArXiv*, 2020.

- [68] Nitin Tripathi and Bayu Primawan. The study of access point outdoor coverage deployment for wireless digital campus network. *International Journal of Information and Communication Technology*, 2018.
- [69] Stefan Wendt, Alain Chicot, and Michel Skrok. On beamforming performance in wi-fi outdoor networks. 2014.
- [70] Jane-Hwa Huang, Li-Chun Wang, and Chung-Ju Chang. Deployment strategies of access points for outdoor wireless local area networks. *2005 IEEE 61st Vehicular Technology Conference*, 2005.
- [71] Xianghui Cao, Rui Ma, Lu Liu, Hongbao Shi, Yu Cheng, and Changyin Sun. A machine learning-based algorithm for joint scheduling and power control in wireless networks. *IEEE Internet of Things Journal*, 2018.
- [72] Satish Jondhale and Raed Shubair. Application of supervised learning approach for target localization in wireless sensor network. 2020.
- [73] He Fang, Xianbin Wang, and Stefano Tomasin. Machine learning for intelligent authentication in 5g and beyond wireless networks. *IEEE Wireless Communications*, 2019.
- [74] Marcelo E. Pellenz, Rosana Lachowski, Edgard Jamhour, Glauber Brante, Guilherme Luiz Moritz, and Richard Demo Souza. In-network data aggregation for information-centric wsns using unsupervised machine learning techniques. *2021 IEEE Symposium on Computers and Communications (ISCC)*, 2021.
- [75] Haesik Kim. Performance analysis of k means clustering algorithms for mmte systems. *2020 International Conference on Information and Communication Technology Convergence (ICTC)*, 2020.
- [76] Sathya Racharla. Data representation and clustering in sensor networks using unsupervised learning algorithm. *Journal of Information and Computational Science*, 2020.
- [77] Jing Ran, Yidong Ji, and Bihua Tang. A semi-supervised learning approach to ieee 802.11 network anomaly detection. *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, 2019.
- [78] Yihong Dong, Xiaohan Jiang, Lei Cheng, and Qingjiang Shi. Ssrcnn: A semi-supervised learning framework for signal recognition. *IEEE Transactions on Cognitive Communications and Networking*, 2021.
- [79] Zhanyi Ren, Pinyi Ren, and Tiantian Zhang. Deep rf device fingerprinting by semi-supervised learning with meta pseudo time-frequency labels. *2022 IEEE Wireless Communications and Networking Conference (WCNC)*, 2022.
- [80] Abdulmalik Alwarafy, Mohamed Abdallah, Bekir Sait Çiftler, Ala Al-Fuqaha, and Mounir Hamdi. The frontiers of deep reinforcement learning for resource management in future wireless hetnets: Techniques, challenges, and research directions. *IEEE Open Journal of the Communications Society*, 2022.
- [81] Soohyun Cho. Rate adaptation with q-learning in csma/ca wireless networks. *Journal of Information Processing Systems*, 2020.

- [82] Ziyang Guo, Zhenyu Chen, Peng Liu, Jianjun Luo, Xun Yang, and Xinghua Sun. Multi-agent reinforcement learning-based distributed channel access for next generation wireless networks. *IEEE Journal on Selected Areas in Communications*, 2022.
- [83] Mohamed Said Frikha, Sonia Mettali Gammar, Abdelkader Lahmadi, and Laurent Andrey. Reinforcement and deep reinforcement learning for wireless internet of things: A survey. *Computer Communications*, 178:98–113, 2021.
- [84] Luíza Caetano Garaffa, Maik Basso, Andréa Aparecida Konzen, and Edison Pignaton de Freitas. Reinforcement learning for mobile robotics exploration: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.
- [85] Michel Tokic and Gunther Palm. Value-difference based exploration: Adaptive control between epsilon-greedy and softmax. *Annual Conference on Artificial Intelligence*, 2011.
- [86] Björn Lütjens, Michael Everett, and Jonathan P. How. Safe reinforcement learning with model uncertainty estimates. *2019 International Conference on Robotics and Automation (ICRA)*, 2019.
- [87] Arthur Aubret, Laetitia Matignon, and Salima Hassas. A survey on intrinsic motivation in reinforcement learning. *arXiv*, 2019.
- [88] Devarani Devi Ningombam, Byunghyun Yoo, Hyun Woo Kim, Hwa Jeon Song, and Sungwon Yi. Cumarl: Curiosity-based learning in multiagent reinforcement learning. *IEEE Access*, 2022.
- [89] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. 2017.
- [90] John Schulman, Sergey Levine, Pieter Abbeel, Michael I Jordan, and Philipp Moritz. Trust region policy optimization. pages 1889–1897, 2015.
- [91] Ziad El Jamous, Kemal Davaslioglu, and Yalin E Sagduyu. Deep reinforcement learning for power control in next-generation wifi network systems. *MILCOM 2022-2022 IEEE Military Communications Conference (MILCOM)*, 2022.
- [92] Shiyao Chen, Qinghua Zeng, and Jinsong Wu. A reinforcement learning-based admission control algorithm for multi-service hetnets. *IEEE Access*, 2021.
- [93] Rashid Ali, Nurullah Shahin, Yousaf Bin Zikria, Byung-Seo Kim, and Sung Won Kim. Deep reinforcement learning paradigm for performance optimization of channel observation-based mac protocols in dense wlans. *IEEE Access*, 2019.
- [94] Shi Su, Wai-Tian Tan, Xiaoqing Zhu, and Rob Liston. Client pre-screening for mu-mimo in commodity 802.11ac networks via online learning. *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, 2019.
- [95] Mohamed Amine Kafi, Alexandre Mouradian, and Véronique Vèque. On-line client association scheme based on reinforcement learning for wlan networks. *2019 IEEE Wireless Communications and Networking Conference (WCNC)*, 2019.

- [96] Ahmad Musa and Irfan Awan. Functional and performance analysis of discrete event network simulation tools. *Simulation Modelling Practice and Theory*, 2021.
- [97] Francesc Wilhelmi, Marc Carrascosa, Cristina Cano, Anders Jonsson, Vishnu Ram, and Boris Bellalta. Usage of network simulators in machine-learning-assisted 5g/6g networks. *IEEE Wireless Communications*, 2021.
- [98] Piotr Gawłowicz and Anatolij Zubow. ns3-gym: Extending openai gym for networking research, 2018.
- [99] et al Martin Abadi, Paul Barham. Tensorflow: A system for large-scale machine learning. USENIX Symposium on Operating Systems Design and Implementation, 2016.
- [100] Ieee standard for information technology–telecommunications and information exchange between systems local and metropolitan area networks–specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications. *IEEE Std 802.11-2012 (Revision of IEEE Std 802.11-2007)*, 2012.
- [101] Williamjeet Singh and Jyotsna Sengupta. An efficient algorithm for optimizing base station site selection to cover a convex square region in cell planning. *Wirel. Pers. Commun.*, 72(2), 2013.
- [102] <https://www.cctvcalculator.net/en/calculations/bandwidth-calculator/>. Accessed: 2023-02-28.
- [103] John K Haas. A history of the unity game engine. 2014.
- [104] E. Alpaydin. *Introduction to Machine Learning, fourth edition*. Adaptive Computation and Machine Learning series. MIT Press, 2020.
- [105] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. A Bradford Book, Cambridge, MA, USA, 2018.
- [106] Akgul and Faruk. *ZeroMQ*. Packt Publishing, 2013.
- [107] Anirudh Ganji, Griffin Page, and Muhammad Shahzad. Characterizing the performance of wifi in dense iot deployments. *2019 28th International Conference on Computer Communication and Networks (ICCCN)*, pages 1–9, 2019.
- [108] Abdallah S. Abdallah and Allen B. MacKenzie. A cross-layer controller for adaptive video streaming over ieee 802.11 networks. *2015 IEEE International Conference on Communications (ICC)*, 2015.
- [109] K. Medepalli and F. A. Tobagi. Towards performance modeling of ieee 802.11 based wireless networks: A unified framework and its applications. *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*, 2006.
- [110] Yalda Edalat and Katia Obraczka. Dynamically tuning ieee 802.11’s contention window using machine learning. *Proceedings of the 22nd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2019.

- [111] Shejie Lu, Siping Hu, Yuntao Duan, Jinfeng Gao, and Fang Liu. Contention window adaptive adjustment strategy for fairness in multi-rate ieee 802.11 network. In *2022 6th International Conference on Communication and Information Systems (ICCIS)*, pages 82–86, 2022.
- [112] R. Jain, D. Chiu, and W. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared computer systems, 1998.
- [113] Kernel.org. ath10k - linux wireless. Online, Accessed 2023.
- [114] Demostenes Zegarra Rodriguez, Renata Rosa, Eduardo Costa Alfaia, Júlia Abrahão, and Graca Bressan. Video quality metric for streaming service using dash standard. *IEEE Transactions on Broadcasting*, 62, 2016.
- [115] Géraldine Andre, Bruno Bachelet, Pietro Battistoni, Amina Belhassena, Sandro Bimonte, Christophe Cariou, Frederic Chabot, Gérard Chalhoub, Adrian Couvent, Georgia Garani, Laneurit Jean, Rim Moussa, Konstantinos Oikonomou, Ibrahim Sammour, Monica Sebillio, Mateus Souza, Tricot Nicolas, and Robert Wrembel. Lambdagriot: a new architecture for agricultural autonomous robots' scheduling: from design to experiments. *Cluster Computing*, pages 1–23, 10 2022.