Intelligent Framework for Radio Resource Management for Future Heterogeneous Wireless Networks

by

Nahed BELHADJ MOHAMED

MANUSCRIPT-BASED THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE IN PARTIAL FULFILLMENT FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
Ph.D.

MONTREAL, APRIL 26, 2026

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Professor Georges Kaddoum, Thesis supervisor
Department of Electrical Engineering, École de Technologie Supérieure

Professor Coulombe Stéphane, Chair, Board of Examiners
Department of Software Engineering and IT, École de Technologie Supérieure

Professor Eric Granger, Member of the jury
Department of Systems Engineering, École de Technologie Supérieure

Professor Jamal Bentahar, External Independent Examiner
Concordia Institute for Information Systems Engineering, Concordia University

THIS THESIS WAS PRESENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC

ON APRIL 17, 2026

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

FOREWORD

The research presented in this thesis encompasses the outcomes achieved during my doctoral studies under the supervision of Prof. Georges Kaddoum. This work was supported in part by the Canada Research Chair Program Tier II, titled "*Toward a Novel and Intelligent Framework for the Next Generations of IoT Networks*."

This thesis primarily focuses on resource allocation in next-generation wireless networks. During my Ph.D. studies, I authored three published journal papers, one journal paper under revision, two accepted conference papers, and one workshop accepted for publication. In addition, I contributed as a co-author to one journal paper under review.

The structure of this thesis is as follows. The first two chapters provide an introduction, along with the technical background, and a comprehensive literature review. The subsequent four chapters are based on the research publications produced during my doctorate. Finally, the concluding chapter summarizes the key findings and outlines potential directions for future research.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my supervisor, Prof. Georges Kaddoum, for his invaluable guidance, continuous support, and encouragement throughout my Ph.D. journey. His insightful advice, constructive feedback, and unwavering belief in my potential have been instrumental in shaping the direction and quality of this research. His mentorship extended far beyond academic matters, providing me with strong motivation and confidence during both challenging and rewarding moments. This work would not have been possible without his exceptional leadership and vision.

I am also profoundly thankful to Prof. Md. Zoheb Hassan for his dedicated mentorship, unwavering support, and expert guidance. I am particularly grateful for his constant availability, patience, and the considerable time he devoted to helping me refine my ideas, address challenges, and elevate the quality of my work. His insightful feedback and steadfast encouragement have played a pivotal role in my academic and professional development.

I would like to sincerely thank all members of my PhD committee, Professor Stéphane Coulombe and Professor Eric Granger, for reviewing my thesis and providing invaluable feedback. I would also like to thank Professor Jamal Bentaha for her time and for serving as the external examiner for my PhD defense.

A heartfelt thank you goes to my friends and colleagues in the lab—Aida Meftah, Hamda Bouzabia, and Imene Romdhane—for their camaraderie, motivation, and all memorable moments we shared during this journey.

I would like to extend my deepest appreciation to my beloved husband, Khaled, for his unconditional love, endless patience, and unwavering support. Your belief in me has given me the strength to overcome every challenge. To my little sunshine, Elias—your smiles, laughter, and love bring meaning and joy to everything I do. You are my greatest source of inspiration.

A special thanks goes to my parents, Lakdhar and Mabrouka, for their love, sacrifices, and unwavering faith in me. To my sisters, Neila and Nawal; my brothers, Nabil, Nader, and

Nasr; their children; and my sister-in-law, Naima and Marwa—thank you for your constant encouragement and for always being there with your kind words and heartfelt support. Your presence has been a pillar of strength throughout this journey.

Cadre intelligent pour la gestion des ressources radio dans les futurs réseaux sans fil hétérogènes

Nahed BELHADJ MOHAMED

RÉSUMÉ

Avec la croissance rapide du nombre d'appareils liés à l'Internet des objets (IoT), l'allocation efficace des ressources devient de plus en plus cruciale afin de garantir une communication fiable, évolutive et écoénergétique. Cependant, l'hétérogénéité des dispositifs IoT, les déploiements de réseaux denses et la présence d'imperfections matérielles (HWIs) posent d'importants défis aux stratégies d'allocation de ressources conventionnelles. Les réseaux IoT nécessitent une plateforme distribuée et intelligente, capable de s'adapter aux conditions dynamiques du réseau tout en assurant l'exécution de tâches complexes selon les exigences variées des applications.

Dans cette thèse, nous faisons progresser l'état de l'art en matière d'allocation de ressources radio pour les réseaux IoT en abordant l'impact combiné de l'interférence et des HWIs. Nous proposons de nouvelles plateformes d'optimisation visant à améliorer la gestion des ressources dans les réseaux IoT, en tenant compte de la variabilité temporelle des canaux et des altérations matérielles induites par des dispositifs non idéaux. Les approches d'optimisation déterministes classiques disponibles à ce jour ne parviennent pas à fournir des solutions optimales en présence de telles altérations stochastiques et inconnues, qui sont difficiles à modéliser et à prévoir. Les techniques classiques d'apprentissage automatique présentent également des limites, car elles reposent généralement sur un apprentissage supervisé et des ensembles de données statiques, ce qui les rend inadaptées à la prise de décision séquentielle dans des environnements dynamiques. L'apprentissage par renforcement (RL) permet de surmonter cette limitation en permettant aux agents d'apprendre des politiques optimales par interaction avec l'environnement, sans nécessiter de modèles explicites. Cependant, les méthodes RL standards ne passent pas bien à l'échelle lorsqu'elles sont confrontées à des espaces d'états et d'actions de grande dimension, comme ceux rencontrés dans les réseaux IoT à grande échelle. Pour répondre à ce défi, nous adoptons l'apprentissage par renforcement profond (DRL), qui exploite des réseaux de neurones profonds comme approximateurs de fonctions afin de capturer des relations complexes et non linéaires et de permettre une prise de décision scalable. Cela rend le DRL particulièrement adapté pour apprendre des politiques d'allocation de ressources à la fois adaptatives et efficaces dans des scénarios IoT réalistes.

Le Chapitre 2 aborde le problème de l'allocation de puissance en liaison descendante dans les réseaux d'accès radio en brouillard intégrés aux réseaux IoT, en tenant compte à la fois des HWIs et des interférences co-canal. Plus précisément, nous proposons une plateforme d'allocation de ressources distribuée dans laquelle chaque point d'accès en brouillard fonctionne comme un agent DRL. Ces agents ajustent dynamiquement la puissance d'émission des dispositifs associés en fonction des états observés du réseau, dans le but de maximiser l'efficacité spectrale globale. Pour améliorer davantage la performance de l'apprentissage et la précision des décisions, nous intégrons une stratégie d'apprentissage par ensemble permettant de sélectionner le modèle DRL

le plus performant parmi un ensemble de politiques entraînées. Cette intégration améliore significativement la vitesse de convergence et la robustesse dans des environnements réseau dynamiques.

Cependant, bien que le Chapitre 2 se concentre sur la maximisation de la capacité réseau des réseaux IoT en supposant la transmission de codes longs (de longueur théoriquement infinie), cette hypothèse devient impraticable dans les réseaux IoT soumis à des contraintes de délai où les messages sont généralement de courte longueur. En conséquence, les Chapitres 3 et 4 s'attaquent au problème conjoint de regroupement et d'allocation de puissance dans les réseaux industriels de l'Internet des objets, en tenant compte des contraintes de longueur de bloc finie, des HWIs et des interférences co-canal. À cet effet, nous proposons une plateforme distribuée en deux étapes intégrant le regroupement et la gestion de puissance. Dans un premier temps, un algorithme de regroupement glouton est introduit pour organiser les dispositifs en plusieurs groupes. Le clustering glouton est adopté en raison de sa faible complexité computationnelle, de sa scalabilité et de sa capacité à fonctionner avec des informations locales limitées. Pour résoudre le problème d'allocation de puissance, nous développons ensuite un algorithme basé sur l'apprentissage par renforcement profond multi-agent (MADRL) dans lequel chaque groupe agit en tant qu'agent autonome. Cette approche décentralisée améliore l'évolutivité et les performances du réseau tout en assurant une adaptation aux environnements sans fil dynamiques et hétérogènes.

Enfin, le Chapitre 5 explore l'allocation de ressources dans les réseaux IoT à technologies d'accès radio multiples, en traitant à la fois les interférences de canaux adjacents et les HWIs. Motivés par l'intégration décentralisée de l'intelligence artificielle dans les systèmes IoT de sixième génération, nous développons une plateforme MADRL dans laquelle chaque dispositif IoT agit comme un agent autonome. Ces agents apprennent continuellement des politiques optimales d'allocation de ressources — y compris la modulation, la puissance d'émission et le taux de codage — à partir d'observations locales. Pour permettre un apprentissage sécurisé et efficace, nous utilisons un réseau jumeau numérique qui permet aux agents d'apprendre dans un environnement simulé et sans risque. Contrairement aux approches existantes qui considèrent les points d'accès ou les groupes comme agents, notre plateforme modélise chaque dispositif comme un apprenant indépendant, renforçant ainsi la flexibilité, l'évolutivité et la capacité d'adaptation. Cette contribution soutient l'évolution vers des systèmes IoT intelligents et auto-organisés, et fait progresser l'application des algorithmes d'apprentissage distribué dans des environnements sans fil complexes et hétérogènes.

Mots-clés: apprentissage par renforcement profond, imperfections matérielles, interférences, réseaux IoT, gestion des ressources.

Intelligent Framework for Radio Resource Management for Future Heterogeneous Wireless Networks

Nahed BELHADJ MOHAMED

ABSTRACT

With the rapid growth of the number of Internet of Things (IoT) devices, efficient resource allocation becomes increasingly critical to ensure reliable, scalable, and energy-efficient communication. However, heterogeneity of IoT devices, dense network deployments, and the presence of hardware impairments (HWIs) pose significant challenges to conventional resource allocation strategies. IoT networks require a distributed and intelligent platform capable of adapting to dynamic network conditions while managing execution of complex tasks based on diverse application requirements.

In this thesis, we advance the state-of-the-art in radio resource allocation for IoT networks by addressing the combined impact of interference and HWIs. We propose novel optimization frameworks that enhance resource management in IoT networks under time-varying channel conditions and HWI-induced distortions caused by non-ideal devices. Conventional deterministic optimization approaches available to date fail to provide optimal solutions in the presence of such stochastic and unknown impairments that are hard to model and predict. Classical machine learning techniques are also limited, as they typically rely on supervised learning and static datasets, making them unsuitable for sequential decision-making in dynamic environments. Reinforcement learning (RL) overcomes this limitation by enabling agents to learn optimal policies through interaction with the environment without requiring explicit models. However, standard RL methods do not scale well to high-dimensional state and action spaces encountered in large-scale IoT networks. To address this, we adopt deep RL (DRL), which leverages deep neural networks as function approximators to capture complex, non-linear relationships and enable scalable decision-making. This makes DRL essential for learning adaptive and efficient resource allocation policies in realistic IoT scenarios.

In Chapter 2, we address the power allocation problem in downlink fog radio access networks-enabled IoT networks, taking into account both HWIs and co-channel interference. Specifically, we propose a distributed resource allocation framework in which each fog access point operates as a DRL agent. These agents dynamically adjust the transmit power of associated devices based on observed network states, with the objective of maximizing overall spectral efficiency. To further enhance learning performance and decision making accuracy, we integrate an ensemble learning strategy enabling the selection of the best-performing model among a set of trained DRL policies. This integration significantly improves convergence speed and robustness in dynamic network environments.

However, while Chapter 2 focuses on maximizing the network capacity of IoT networks under the assumption of transmitting long (theoretically infinite-length) codewords, in delay-constrained IoT networks, the message lengths are typically short, making this assumption impractical. Accordingly, Chapters 3 and 4 focus on the joint clustering and power allocation problem in

Industrial Internet of Things networks, accounting for finite blocklength constraints, HWIs, and co-channel interference. To this end, we propose a two-step distributed framework that integrates clustering and power management. In the first step, a greedy clustering algorithm is introduced to group devices into multiple clusters. Greedy clustering is adopted due to its low computational complexity, scalability, and ability to operate with limited local information, making it well-suited for dynamic IoT environments. To address the power allocation problem, we then develop a multi-agent DRL (MADRL)-based algorithm where each cluster operates as an independent agent. This decentralized approach is found to enhance network scalability and performance while ensuring adaptability to dynamic and heterogeneous wireless environments.

Finally, Chapter 5 explores resource allocation in multi-radio access technology IoT networks, addressing both adjacent channel interference and HWIs. Motivated by the decentralized integration of artificial intelligence in sixth-generation IoT systems, we develop a MADRL framework where each IoT device acts as an autonomous agent. These agents continuously learn optimal resource allocation policies—including modulation, transmit power, and coding rate—based on local observations. To facilitate safe and efficient training, we employ a digital twin network enabling the agents to learn in a simulated, risk-free environment. Unlike existing approaches that treat access points or clusters as agents, our framework models each device as an independent learner, thereby enhancing flexibility, scalability, and adaptability. This contribution supports the evolution of intelligent, self-organizing IoT systems and advances the application of distributed learning algorithms in complex, heterogeneous wireless environments.

Keywords: deep reinforcement learning, hardware impairments, interference, IoT networks, resource management.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

LIST OF ALGORITHMS

LIST OF ABBREVIATIONS

4G	Fourth Generation
5G	Fifth Generation
6G	Sixth Generation
ACI	Adjacent Channel Interference
AI	Artificial Intelligence
AM	Adaptive Modulation
AMC	Adaptive Modulation And Coding
AO	Alternating Optimization
AP	Access Point
AWGN	Additive White Gaussian Noise
B5G	Beyond-5G
BS	Base Station
CBS	Cloud Base Station
CQI	Channel Quality Indicator
C-RAN	Cloud Radio Access Network
CR	Clipping Ratio
CS	Coalition Structure
CSI	Channel State Information
CSS	Chirp Spread Spectrum

CTDE	Centralized Training And Decentralized Execution
D2P	Digital-To-Physical
DCPM	Distributed Clustering And Power Management
DDPA	Distributed DRL-Based Power Allocation
DL	Deep Learning
DNN	Deep Neural Network
DQL	Deep Q-Learning
DQN	Deep Q-Network
DRL	Deep Reinforcement Learning
DT	Digital Twin
DTN	Digital Twin Network
eDQN	Ensemble DQN
EE	Energy Efficiency
EL	Ensemble Learning
eMBB	Enhanced Mobile Broadband
ESA	Exhaustive Search Algorithm
FBL	Finite Blocklength
FDRL	Federated DRL
FIFO	First-Input-First-Output
FL	Federated Learning

FP	Fractional Programming
F-RAN	Fog Radio Access Network
F-AP	Fog Access Point
GAP	Generalized Assignment Problem
HetNets	Heterogeneous Networks
HWIs	Hardware Impairments
IID	Independent and Identically Distributed
IIoT	Industrial Internet of Things
IoE	Internet of Everything
IoT	Internet of Things
I/Q	In-Phase and Quadrature
IRS	Intelligent Reflecting Surface
KKT	Karush-Kuhn-Tucker
KPM	Key Performance Measurement
LoRa	Long Range
LoS	Line-of-Sight
LPWAN	Low Power Wide Area Network
LTE	Long-Term Evolution
MADRL	Multi-Agent DRL
MA	Multiple Access

MARL	Multi-Agent Reinforcement Learning
MaxP	Maximum Power
MCS	Modulation And Coding Scheme
MDP	Markov Decision Process
MINLP	Mixed-Integer Non-Linear Programming
MIMO	Multiple Input Multiple Output
ML	Machine Learning
NB	NarrowBand
NLoS	Non-Line-of-Sight
NE	Nash Equilibrium
NOMA	Non-Orthogonal Multiple Access
NR	New Radio
OCS	Optimal Coalition Structure
OFDMA	Orthogonal Frequency Division Multiple Access
OFDM	Orthogonal Frequency Division Multiplexing
P2D	Physical-To-Digital
PER	Packet Error Rate
PF	Proportional Fairness
PMF	Probability Mass Function
POMDPS	Partially Observable Markov Decision Processes

PSD	Power Spectral Density
PSK	Phase-Shift Keying
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
RATS	Radio Access Technologies
ReLU	Rectified Linear Unit
RF	Radio Frequency
RL	Reinforcement Learning
RP	Random Power
RR	Round Robin
RRB	Radio Resource Block
RRBGS	RRB Groups
RRM	Radio Resource Management
RSMA	Rate-Splitting Multiple Access
SADRL	Single-Agent DRL
SARL	Single-Agent Reinforcement Learning
SC	Superposition Coding
SE	Spectral Efficiency
SER	Symbol Error Rate
SIC	Successive Interference Cancellation

SINR	Signal To Interference Plus Noise Ratio
TANH	Hyperbolic Tangent
TIN	Treat Interference As Noise
TS	Time Slot
UAV	Unmanned Aerial Vehicle
UMI	Urban Micro
UNB	Ultra-NarrowBand
URLLC	Ultra-Reliable Low-Latency Communication
WMMSE	Weighted Minimum Mean Squared Error

INTRODUCTION

Motivation

With the growth of the number of connected devices, Internet of Things (IoT) technology is gaining widespread adoption across various domains, such as transportation, healthcare, industry, agriculture, smart cities, and environmental monitoring. This rapid expansion is fueled by advancements in diverse communication technologies, including, but not limited to, wireless neighborhood area networks, cellular IoT, satellite networks, and low-power wide-area networks (LPWANs) (Hartung, 2024). These technologies enable seamless connectivity for a wide range of IoT applications, from real-time patient monitoring in healthcare to intelligent transportation systems that improve traffic management and road safety.

Moreover, with the widespread deployment of fifth-generation (5G) and the ongoing development of sixth-generation (6G) wireless networks, which aim to support connectivity for an unprecedented number of devices with diverse performance requirements (Qadir, Le, Saeed & Munawar, 2023), this trend is expected to accelerate. As argued by (Hartung, 2024), by 2030, over 40 billion IoT devices are projected to be connected to the internet. This exponential growth not only underscores the increasing reliance on IoT technologies, but also introduces significant challenges in managing this massive ecosystem of connected devices. Furthermore, with billions of IoT devices in operation, an enormous volume of data is being continuously generated (Yan, Cheng, Chen, Vucetic & Li, 2021). Several previous studies acknowledged that the rapid proliferation of IoT devices—along with the vast amount of data they produce—poses major challenges for resource management in IoT networks, necessitating the development of innovative, scalable, and efficient solutions to ensure reliable network performance and effective data processing (Wang, Sheng, Yang & Leung, 2016; Cheng *et al.*, 2023).

In this context, resource allocation remains one of the most pressing challenges in IoT networks, encompassing spectrum assignment (Salameh, Al-Masri, Benkhelifa & Lloret, 2019; Bany Salameh, Almajali, Ayyash & Elgala, 2018), power allocation (Xu *et al.*, 2022a; Masood *et al.*, 2021), user association (Sun, Xu & Cui, 2022; Zhao *et al.*, 2019), and so on. Efficient management of these resources is vital to ensuring network performance and avoiding severe degradation in increasingly dense and heterogeneous IoT environments. However, these resource allocation problems are inherently complex, as they frequently take the form of static, non-convex, and non-linear combinatorial optimization problems that must be solved at each time slot (TS). Such problems involve a large number of interdependent variables, making them computationally intractable. As a result, they are classified as NP-hard (Salameh *et al.*, 2019; Bany Salameh *et al.*, 2018)–that is, no known algorithm can optimally solve them in polynomial time. Instead, researchers tend to rely on low-complexity heuristic algorithms to provide feasible approximations (Ghanem, Jamali, Sun & Schober, 2019). However, a major limitation of heuristic-based approaches is the difficulty in quantifying their performance loss relative to the optimal solution, which results in uncertainties in their efficiency and reliability in dynamic IoT environments.

Considering the dynamic nature of communication channels and network conditions, conventional optimization-based methods are becoming increasingly impractical for real-time resource allocation in large-scale IoT networks (Hussain, Hassan, Hussain & Hossain, 2020). Traditional solutions experience difficulties in adapting to time-varying environments and require extensive prior knowledge of network parameters, which makes them unsuitable for highly dynamic systems. In addition, conventional machine learning (ML) approaches, such as supervised learning, also face limitations in this context, as they rely on labeled datasets and static input-output mappings. In dynamic wireless environments, generating labeled optimal decisions is computationally expensive and often infeasible, particularly when the system model is unknown or changes over time. Moreover, such models lack the ability to adapt online to evolving network

conditions. To overcome these challenges, it is essential to leverage adaptive, self-learning algorithms capable of dynamically optimizing resource allocation in response to changing network conditions. In this context, a promising paradigm for online resource allocation is deep reinforcement learning (DRL), which can learn optimal policies through continuous interaction with the environment (Naderializadeh, Sydir, Simsek & Nikopour, 2021). Unlike conventional optimization techniques, DRL-based approaches can autonomously adapt to non-stationary environment, generalize across diverse scenarios, and make near-optimal decisions in real time without relying on predefined mathematical models. In addition, DRL's ability to learn long-term decision making policies enables it to exhibit robustness to a variety of system uncertainties and imperfections. By optimizing cumulative rewards over time, DRL can effectively smooth out the impact of transient disturbances—such as time-varying interference, environmental noise, channel fluctuations, and hardware-induced anomalies—and promote decisions ensuring stable, reliable performance in non-ideal operating conditions.

Among various sources of imperfections that challenge the performance of IoT systems, hardware impairments (HWIs) stand out as particularly critical and pervasive. However, HWIs remain insufficiently addressed in most existing research. Due to cost and design constraints, IoT devices are highly susceptible to HWIs, including phase noise, power amplifier non-linearity, in-phase/quadrature (I/Q) imbalance, and quantization errors. These impairments can considerably deteriorate signal quality and system efficiency. Despite the growing body of literature exploring DRL for resource allocation, most previous studies assumed ideal hardware conditions (Mao, Alizadeh, Menache & Kandula, 2016; Ren, Pan, Deng, Elkashlan & Nallanathan, 2019a, 2020), thereby overlooking the practical limitations faced in real-world deployments. This simplification not only limits the realism of the proposed models, but also risks overstating their performance and applicability. To fully exploit the potential of DRL in realistic environments, it is necessary to incorporate the impact of HWIs into the training and decision making processes. Doing so

will lead to the development of more robust, resilient, and deployment-ready resource allocation strategies tailored to the practical constraints of IoT systems.

Challenges in IoT Networks

The fact that IoT networks scale to support billions of concurrently communicating devices considerably increases the demand for communication channels and radio resource blocks (RRBs). However, due to the finite spectrum and limited availability of RRBs, traditional orthogonal resource allocation schemes become infeasible in dense network environments (Rezwan & Choi, 2021). This resource scarcity necessitates adopting more spectrum-efficient access techniques. In this context, rate-splitting multiple access (RSMA) has emerged as a promising state-of-the-art solution, enabling multiple devices to share the same RRB by splitting messages into common and private parts (Hassan, Kaddoum & Akhrif, 2022). However, although RSMA significantly improves RRB utilization, it also introduces intra-cell interference among devices sharing the same RRB. Therefore, effective radio resource optimization should jointly account for this co-channel interference to fully exploit the benefits of RSMA in dense IoT settings.

Along with co-channel interference, the coexistence of multiple radio access technologies (RATs) within the same network—such as long-term evolution (LTE) and 5G new radio (NR)—can cause considerable adjacent channel interference (ACI), particularly when communication links operate on closely spaced frequencies. While existing interference management and resource allocation techniques attempt to mitigate such effects, their performance is often limited by simplifying assumptions and static network configurations, which reduces their effectiveness in heterogeneous and highly dynamic IoT environments.

Furthermore, a branch of IoT networks known as the Industrial IoT (IIoT) has recently emerged as a transformative force, driving the automation and digitalization of industrial processes.

However, realizing the full potential of IIoT requires addressing several significant challenges, particularly in meeting stringent quality-of-service (QoS) requirements under time-varying fading channels. Many IIoT applications—such as real-time control, predictive maintenance, and factory automation—require ultra-reliable and low-latency communication. These requirements impose strict delay constraints, resulting in the use of short packet blocklengths that render the classical Shannon capacity formula inaccurate for practical rate estimation (Ranjha, Kaddoum & Dev, 2022). To address this limitation, finite blocklength (FBL) theory has been introduced in the literature as a more accurate framework for performance analysis in short-packet communications.

Research Questions

To address the core challenges outlined above, the present thesis investigates the following interrelated research questions:

- How can interference be effectively mitigated in the presence of unknown HWIs to enable robust and efficient resource allocation in IoT networks?
- How can QoS requirements be reliably met in large-scale IoT networks, while efficiently accounting for FBL effects in a scalable and computationally practical way?
- How can the coexistence of heterogeneous RATs be effectively managed to mitigate ACI and enhance spectrum utilization in dense, multi-RAT IoT environments?

Research Objectives

This thesis aims to advance resource allocation techniques in downlink IoT networks by addressing key limitations identified in the existing literature, such as interference and the presence of HWIs. Many IoT applications—such as over-the-air firmware updates (Bauwens, Ruckebusch, Giannoulis, Moerman & Poorter, 2020a), remote driving commands (Kakkavas *et al.*, 2022), and internet of multimedia things streaming (Nguyen *et al.*, 2024)—largely rely on high-quality downlink services, as they involve data-intensive communications from the

network to the devices. Moreover, in large-scale deployments, base stations (BSs) serve a massive number of devices simultaneously, thereby considerably increasing the susceptibility of downlink transmissions to various forms of interference, including co-channel and ACI. These challenges make the downlink a critical performance bottleneck in dense IoT environments. Consequently, it becomes vital to optimize downlink resource allocation to ensure network reliability, efficiency, and scalability. Accordingly, the overarching goal of this thesis is to develop resource management solutions that are not only efficient and scalable, but also robust against interference and HWIs.

To this end, we formulate the following specific research objectives:

- To develop a novel power allocation framework for downlink fog radio access networks (F-RANs) that would maximize spectral efficiency (SE) while explicitly accounting for both HWIs and co-channel interference;
- To design a joint clustering and power allocation framework for downlink IIoT networks that would consider the impact of FBL-coded transmission, HWIs, and co-channel interference;
- To propose a radio resource optimization scheme that would dynamically adjust link adaptation parameters—such as transmit power, modulation, and coding rate—to maximize network throughput while effectively mitigating ACI and HWIs in downlink multiple RATs IoT systems.

Each of these objectives is addressed through a dedicated study presented in the subsequent chapters of this dissertation. The results collectively advance the state of the art in intelligent and robust resource allocation for IoT networks, particularly under practical constraints such as HWIs, dynamic interference conditions, and heterogeneous system architectures. In what follows, we provide a chapter-by-chapter overview of the dissertation and highlight the specific contributions made in each.

Contributions and Outline

This dissertation, which includes five chapters, is organized as follows.

Chapter 1 provides the technical background and a comprehensive literature review that form the foundation of this research. Here, an overview of representative IoT communication schemes and key multiple access (MA) techniques used in cellular IoT networks is followed by a discussion of interference and HWIs in wireless communication systems. Chapter 1 then introduces AI-based resource management strategies in IoT, with particular focus on learning approaches such as reinforcement learning (RL), Markov decision processes (MDPs), deep Q-learning (DQL), and multi-agent reinforcement learning (MARL). Chapter 1 concludes with a detailed review of the relevant literature.

Next, Chapter 2 presents the first article, which investigates the power allocation problem in downlink F-RAN. Taking into account the limited availability of RRBs, we assume that each fog access point (F-AP) serves its associated devices over the same RRB, leading to significant co-channel interference. To address this issue, we formulate an optimization problem aimed at maximizing the network's SE, explicitly accounting for both co-channel interference and HWIs. Due to the NP-hard nature of the problem, we propose a distributed DRL-based power allocation scheme, where each F-AP operates as an independent DRL agent, dynamically adjusting its transmit power based on real-time network conditions. Furthermore, to enhance the learning efficiency and robustness of the proposed DRL framework, we integrate an ensemble learning (EL) approach, which uses multiple models to improve decision making and convergence stability. Numerical results are presented to demonstrate the advantages of the proposed scheme as compared to other benchmark schemes.

Chapter 3 presents the second article, which introduces an interference management scheme tailored for IIoT networks. Specifically, we address the challenge of interference mitigation in

densely deployed IIoT environments by considering FBL coded transmissions, HWI-induced signal distortions, and co-channel interference. To this end, we formulate a joint device clustering and power allocation problem so as to maximize overall network capacity. To this end, we propose a two-step distributed clustering and power management (DCPM) framework. In the first step, aiming to maximize the signal-to-interference-plus-noise ratio (SINR) within each cluster, a greedy clustering algorithm is employed to group devices for each access point (AP). Devices within a cluster are served using RSMA, which improves both capacity and RRB utilization. In the second step, a multi-agent DRL (MADRL) approach is applied to optimize transmit power allocation, whereby each cluster operates as an independent agent. Simulation results demonstrate that the proposed DCPM framework achieves superior performance across diverse channel conditions as compared to baseline methods.

Furthermore, Chapter 4 presents an extension of the second article, with a particular focus on energy efficiency (EE) optimization in densely deployed IoT networks. Unlike the previous chapter, where clustering is based on SINR, here we consider clustering IoT devices into non-overlapping groups based on their channel state information (CSI). The proposed framework aims to improve the EE of downlink RSMA-based IoT networks by optimizing transmit power allocation across clustered devices. Our simulation results demonstrate that the proposed scheme significantly outperforms state-of-the-art power allocation methods in terms of EE under various network conditions.

Finally, Chapter 5 presents the third article, which investigates resource allocation in multi-RAT downlink IoT networks operating in the presence of ACI and HWI. Here, we propose a radio resource optimization framework that dynamically adapts transmission parameters—including power, modulation, and coding rate—aiming to maximize network throughput while accounting for both ACI and HWI. Given the NP-hard nature of the problem and the impracticality of centralized solutions in dynamic, large-scale settings, we reformulate it as a Markov game

and propose a MADRL approach. Specifically, in this framework, each IoT link is modeled as an independent agent that learns optimal resource allocation policies based solely on local observations. To facilitate efficient and safe training, we introduce a digital twin (DT)-enabled continual learning scheme. The DT replicates the multi-RAT IoT environment virtually, thus enabling the MADRL agents to continuously refine their policies with minimal real-world interaction and communication overhead. Our simulation results confirm the scalability, adaptability, and performance gains of the proposed DT-enhanced MADRL framework as compared to state-of-the-art benchmarks.

The thesis concludes with a summary of the key findings from the main chapters and discusses potential directions for future research.

Related Publications

The author's Ph.D. research contributed to the following published and submitted journal articles:

Chapter 2: N. B. Mohamed, M. Z. Hassan, and G. Kaddoum, "Spectral Efficiency Improvement in Downlink Fog Radio Access Network With Deep-Reinforcement-Learning-Enabled Power Control," *IEEE Internet of Things Journal*, vol. 10, no. 17, pp. 15044-15059, Sept 1, 2023, doi: 10.1109/JIOT.2023.3263756.

Chapter 3: N. B. Mohamed, M. Z. Hassan, and G. Kaddoum, "RSMA-Enabled Interference Management for Industrial Internet of Things Networks With Finite Blocklength Coding and Hardware Impairments," *IEEE Transactions on Machine Learning in Communications and Networking*, vol. 2, pp. 1319-1340, 2024, doi: 10.1109/TMLCN.2024.3455268.

Chapter 4: N. B. Mohamed, M. Z. Hassan, and G. Kaddoum, "Energy-Efficient Clustering and Power Allocation in RSMA-Enabled IoT Networks with Finite Blocklength Coding and Hardware Impairments," *2024 IEEE 10th World Forum on Internet of Things (WF-IoT), Ottawa,*

ON, Canada, 2024, pp. 438-443, doi: 10.1109/WF-IoT62078.2024.10811339. **Awarded best local paper award.**

Chapter 5: N. B. Mohamed, G. Kaddoum, and M. Z. Hassan, "Digital Twin-Driven Continual Deep Reinforcement Learning for Coexistence of Multiple Radio Access Technology IoT Links with Nonlinear Receivers," *IEEE Transactions on Machine Learning in Communications and Networking*,vol. 4, pp. 591-611, 2026, doi: 10.1109/TMLCN.2026.3669837.

Along with the aforementioned articles that contribute to the main contents of this dissertation, other publications that the author was involved in and which are not included in this dissertation are as follows:

N. B. Mohamed, M. Z. Hassan, and G. Kaddoum, "Open RAN Internet of Drones: Digital Twin-Driven Multi-Agent RL for Interference Management," *Wireless Communications Letters*, 2026, under review.

I. Romdhane, Z. Rahman, **N. B. Mohamed**, M. Z. Hassan, and G. Kaddoum, "Dynamic Resource Optimization for a Joint Max-Min Fairness and Energy-Efficiency Problem in NOMA-Aided Underwater Optical Wireless Systems," *under revision at IEEE Open Journal of the Communications Society*.

N. B. Mohamed, M. Z. Hassan, and G. Kaddoum, "Deep Reinforcement Learning-Enabled Resilient Radio Resource Allocation for Internet-of-Things Networks with Receiver Non-Linearity," *2024 7th Conference on Cloud and Internet of Things (CIoT), Montreal, QC, Canada*, 2024, pp. 1-5, doi: 10.1109/CIoT63799.2024.10757146.

N. B. Mohamed, M. Z. Hassan, and G. Kaddoum, "Improving Energy Sustainability of Multi-RAT IoT Coexistence Networks in the Presence of Adjacent Channel Interference and

Hardware Impairments," *accepted for publication at 2026 IEEE International Conference on Communications Workshops*.

CHAPTER 1

BACKGROUND AND LITERATURE REVIEW

1.1 IoT Schemes

The IoT ecosystem comprises a wide range of wireless communication technologies, each tailored to support specific application requirements. These technologies considerably vary in terms of spectrum access (licensed versus unlicensed), coverage range, energy consumption, data rates, scalability, and access control mechanisms. In this section, we provide an overview of the following three representative IoT schemes: Cellular IoT, Sigfox, and LoRa.

Cellular IoT

Cellular IoT, built upon 3GPP-standardized technologies such as LTE-M, narrowband (NB)-IoT, and 5G NR, has recently emerged as a leading solution for secure, scalable, and wide-area IoT deployments (Varsier, Dufrène, Dumay, Lampin & Schwoerer, 2021). Operating over licensed spectrum and using existing mobile network infrastructure, cellular IoT enables massive device connectivity, robust security, ultra-low latency, and seamless integration with core cellular systems (Moges, Lakew, Nguyen, Dao & Cho, 2023). These capabilities make cellular IoT particularly well-suited for mission-critical and real-time applications, including intelligent transportation systems (Benhiba, Madi & Addaim, 2018), industrial automation (Ahmadzadeh, Parr & Zhao, 2021), and connected healthcare (Moges *et al.*, 2023). To further broaden the applicability of cellular IoT, 3GPP Release 17 introduced a new class of devices known as reduced capability (RedCap) (Ratasuk, Mangalvedhe, Lee & Bhatoolaul, 2021). RedCap is specifically designed to support mid-tier IoT use cases by lowering device cost, complexity, and power consumption—while still benefiting from reliability and wide-area coverage of 5G networks.

A defining feature of cellular IoT is its use of grant-based scheduling, whereby devices request transmission resources and are assigned uplink/downlink slots by the BS. This centralized and coordinated access mechanism facilitates efficient spectrum use, mitigates interference,

and enables QoS provisioning, particularly in dense IoT deployments. In this dissertation, we adopt cellular IoT as the foundational connectivity model. Under this architecture, efficient and adaptive resource allocation becomes critical to supporting large-scale deployments with diverse device requirements and fluctuating network conditions. To achieve such adaptability and scalability, contemporary cellular standards incorporate advanced MA techniques, including orthogonal frequency-division MA (OFDMA), non-orthogonal MA (NOMA), and RSMA. While the design of these MA schemes is beyond the scope of the present work, our proposed framework builds upon them to address key resource allocation challenges in dynamic IoT environments. In what follows, we provide a focused discussion of these MA techniques.

Sigfox

In contrast to cellular IoT—which operates over licensed spectrum and relies on centralized, grant-based scheduling—Sigfox is a proprietary LPWAN technology that uses unlicensed sub-GHz industrial, scientific, and medical (ISM) bands. Adopting ultra-narrowband (UNB) modulation and a simple ALOHA-based, grant-free access scheme, Sigfox enables devices to autonomously transmit without prior coordination or resource reservation. Sigfox is specifically designed for ultra-low-power, infrequent uplink transmissions with minimal payloads (typically up to 12 bytes) and extremely low data rates (around 100 bps) (Khalifeh, Aldahdouh, Darabkh & Al-Sit, 2019; Mekki, Bajic, Chaxel & Meyer, 2018). In addition, its long-range capabilities—reaching up to 40 km in rural areas—make it well-suited for delay-tolerant applications such as smart metering, environmental monitoring, and asset tracking (Saavedra, del Campo & Santamaria, 2020; Joris *et al.*, 2019). However, the lack of scheduling mechanisms and limited downlink capacity considerably constrain its scalability in dense deployments and its suitability for latency-sensitive or high-throughput IoT use cases.

LoRa

Similarly to Sigfox, LoRa (Long Range) is a leading LPWAN technology operating in the unlicensed sub-GHz ISM bands (e.g., 868 MHz in Europe, 915 MHz in USA, and 433 MHz in Asia (Khalifeh *et al.*, 2019)). LoRa uses chirp spread spectrum (CSS) modulation, which

offers high interference resilience and allows for communication over distances of up to 20 km in rural areas (Mekki *et al.*, 2018). Like Sigfox, LoRa adopts a grant-free, ALOHA-based random access mechanism where devices transmit data asynchronously without prior scheduling or coordination. This design simplifies device operation and supports extended battery life, making LoRa well-suited for low-data-rate, delay-tolerant applications such as smart farming, infrastructure monitoring, environmental sensing, and smart cities (Rawat *et al.*, 2023; Mekki *et al.*, 2018). Yet, the lack of centralized scheduling and guaranteed access leads to scalability challenges in dense deployments, such as the increased risk of collisions and the absence of QoS support. As with Sigfox, these limitations make LoRa less appropriate for time-critical or reliability-sensitive IoT applications, which are better addressed through grant-based access mechanisms such as those employed in cellular IoT.

1.2 Radio Access Technologies in Cellular IoT Networks

With the scaling of IoT networks, efficient MA techniques become critical for managing the massive number of heterogeneous connected devices and meeting diverse QoS requirements. Many IoT systems, particularly those based on cellular and Wi-Fi standards (e.g., LTE-M, NB-IoT, 5G NR, and Wi-Fi 6) use OFDMA to ensure high device density, SE, and reliable connectivity. To accommodate a wide range of IoT services, modern networks evolve to support the following three main communication paradigms: (i) ultra-reliable low-latency communication (URLLC) for mission-critical applications such as industrial automation and autonomous systems; (ii) massive machine-type communication (mMTC) for large-scale deployments of low-power, sporadically active devices; and (iii) enhanced mobile broadband (eMBB) for bandwidth-intensive use cases. To meet the unique demands of these application classes, more sophisticated MA techniques have emerged. Two prominent examples enhancing SE and user fairness by enabling concurrent transmissions over shared resources while effectively managing interference are NOMA and RSMA.

In the next sections, we provide an overview of these MA techniques, highlighting their principles, strengths, and limitations in the context of IoT networks.

1.2.1 Orthogonal Frequency Division Multiple Access (OFDMA)

OFDMA, an extension of the orthogonal frequency division multiplexing (OFDM), has been widely implemented in wireless systems that support IoT connectivity, such as LTE-M, NB-IoT, 5G NR (Zhang, Ijaz, Xiao, Molu & Tafazolli, 2018a). In OFDMA, available bandwidth is partitioned into multiple orthogonal subcarriers, which are further grouped into resource blocks. These resource blocks are dynamically allocated to users based on their instantaneous channel conditions. The allocation of resource blocks is typically managed through scheduling algorithms designed to balance system throughput and user fairness. Common techniques include round robin (RR), which cyclically assigns resource blocks to users regardless of channel conditions, and proportional fairness (PF), which aims to maximize overall throughput while ensuring fair access by considering both current data rates and average user throughput (Fitriasari, Lestari, Luhurkinanti & Sari, 2021).

1.2.2 Non-Orthogonal Multiple Access (NOMA)

NOMA, a promising MA technique for 5G communications, offers significant improvements in SE and user fairness (Dai *et al.*, 2015). Unlike OFDMA, which assigns separate frequency resources to different users, NOMA enables multiple users to share the same time-frequency resources by using power-domain multiplexing. This is achieved through superposition coding (SC) and successive interference cancellation (SIC), enabling users with significantly different channel conditions to coexist within the same resource block (Huang *et al.*, 2018). In this scheme, users with stronger channel conditions are allocated lower power, while those with weaker channels receive higher power. This power allocation facilitates the simultaneous transmission of multiple user signals over a shared resource block.

1.2.3 Rate-Splitting Multiple Access (RSMA)

RSMA is a flexible and generalized MA technique that optimally balances interference management and resource allocation. In RSMA, the transmitter partitions each device's

message into the following two components: a common part, intended for all devices, and a private part, dedicated to the intended recipient (Hassan, Hossain, Cheng & Leung, 2021a). Before transmission, the common and private streams are linearly precoded and transmitted together over the same RRB. Upon reception, each device applies SIC to first decode and subtract the common stream, which contains both the common portions of other users' messages and its own. Once the common stream is successfully removed, the device proceeds to decode its private stream. By combining the decoded common and private parts, each device reconstructs its intended message (Abanto-Leon *et al.*, 2024).

Table 1.1 Comparison of MA techniques across multiple key features

Feature	**OFDMA**	**NOMA**	**RSMA**
Multiple Access Mechanism	Orthogonal subcarriers allocation	Power-domain multiplexing	Rate splitting and linear precoding
SE	Efficient, but limited in high-user-density scenarios	Higher efficiency by allowing multiple users per resource block	Balances interference management and SE
Interference Management	Avoids interference through orthogonality	Requires SIC for decoding weaker signals	Uses linear precoding and SIC for interference decoding
Complexity	Low complexity due to orthogonal allocation	High complexity due to power allocation and SIC	Moderate complexity due to message splitting and SIC
Performance in High-User Environments	Suffers in large-scale networks due to resource limitations	Performs well by multiplexing users over the same resources	Adapts dynamically, outperforming NOMA and OFDMA in many scenarios
Flexibility	Rigid resource allocation	Limited flexibility due to fixed power domain allocation	Highly flexible and adaptable to dynamic network conditions

1.2.4 Comparison of OFDMA, NOMA, and RSMA

RSMA has emerged as a promising MA technique demonstrating superior performance as compared to that of NOMA and OFDMA, particularly in managing multiuser interference (Clerckx *et al.*, 2023; Şahin, Dizdar, Clerckx & Arslan, 2024). Table 1.1 shows the results of a comparative analysis of OFDMA, NOMA, and RSMA across multiple key features. OFDMA, a conventional and widely adopted MA scheme, ensures interference-free communication by assigning orthogonal subcarriers to users (Makki, Chitti, Behravan & Alouini, 2020). However, due to rigid resource allocation, its SE diminishes in high-user-density scenarios. Using power-domain multiplexing and SIC for signal decoding, NOMA improves SE by enabling multiple users to share the same frequency-time resources. Despite its advantages, NOMA introduces increased complexity due to power allocation and SIC requirements (Dai *et al.*, 2018). In contrast, RSMA introduces message splitting, thus offering a more balanced approach to SE, interference management (Şahin *et al.*, 2024), and flexibility. By decomposing messages into common and private components, RSMA dynamically adapts to varying network conditions, efficiently mitigating interference while optimizing resource utilization (Şahin *et al.*, 2024). This adaptability allows RSMA to outperform both NOMA and OFDMA across diverse network scenarios. As a result, RSMA stands out as a strong contender for next-generation wireless communication systems, demonstrating significant potential to address the growing connectivity demands of future networks (Abanto-Leon *et al.*, 2024; Clerckx *et al.*, 2023).

1.3 Interference and Hardware Impairments in Wireless Networks

As discussed in Section 1.2, MA techniques, such as OFDMA, NOMA, and RSMA, adopt distinct mechanisms to manage resource sharing and mitigate multi-user interference. However, interference remains a fundamental challenge in dense IoT networks, primarily due to the high number of simultaneously active devices and the limited availability of spectral resources.

In OFDMA systems, ACI may result from the non-linear behavior of the radio frequency (RF) front end, particularly affecting closely spaced subcarriers and limiting SE in densely

deployed scenarios. In NOMA-based systems, performance largely depends on accurate power allocation and ideal SIC. However, in practical IoT environments, imperfections in SIC and channel estimation frequently result in residual interference, thus disproportionately degrading the performance of users with weaker channel conditions or lower decoding priority. In RSMA-based systems, co-channel interference arises when multiple devices are multiplexed over the same RRB. While RSMA reduces interference through message splitting and SIC, overlapping transmissions can still lead to performance degradation, especially in dense scenarios.

Beyond these conventional sources of interference, a less explored, but equally critical contributor to interference is HWIs. IoT devices are commonly designed with low-cost and low-complexity RF front ends, which inherently introduce HWIs that can severely degrade system performance. These impairments stem from several non-idealities, such as phase noise, quantization errors, amplifier non-linearity, I/Q imbalance, and oscillator drift (Chu *et al.*, 2022). The impact of these HWIs is particularly significant when cost constraints necessitate the use of inexpensive components, resulting in higher levels of distortion and reduced signal fidelity (Shahiri, Kuhestani & Hanzo, 2022). Despite their practical importance, HWIs are frequently neglected or idealized in the literature. For example, in (Cao, Zhu, Jiang, Liu & Zheng, 2020), the authors investigated throughput maximization in an IIoT network while assuming ideal hardware conditions. Similarly, in (Ren *et al.*, 2020), the authors addressed resource allocation for uplink massive multiple-input multiple-output (MIMO) systems without accounting for device-level impairments. Other representative works overlooking HWIs in the context of IoT resource management can be found in (Omidkar, Khalili, Nguyen & Shafiei, 2022; Wang *et al.*, 2020a; Cao, Zhang & Liang, 2019; Munaye, Juang, Lin, Tarekegn & Lin, 2021). This widespread assumption of ideal hardware leads to overly optimistic performance evaluations and limits applicability of proposed solutions in real-world deployments.

By contrast, some recent works acknowledged that HWIs—particularly in the RF front end—not only degrade signal quality, but also amplify interference and negatively impact both SE and EE (Pandey, Gurjar, Yadav & Solanki, 2023; Li, Zheng, Zeng, Liu & Dobre, 2023). As such, accurately modeling HWIs is essential for realistic performance evaluation and the design of

effective mitigation strategies, especially in IoT networks where cost and power limitations amplify adverse effects of non-ideal hardware. For instance, in (Liu, Garg & Ratnarajah, 2024a), the authors investigated EE-oriented power allocation strategies in massive MIMO systems while explicitly considering HWIs. Their results underscore the risk of performance overestimation when HWIs are neglected in system design. Similarly, in (Boshkovska, Ng, Dai & Schober, 2018), the authors proposed a robust resource allocation scheme for wireless-powered communication networks, integrating both a non-linear energy harvesting model and residual HWIs. The authors emphasized that assuming ideal hardware leads to overly optimistic system models, which are unsuitable for real-world deployments. Moreover, in (Zhu, Ng, Wang, Schober & Bhargava, 2017), the authors examined the impact of HWIs on downlink massive MIMO systems in the presence of a passive eavesdropper. The study highlighted that large-scale antenna systems—typically reliant on low-cost components—are particularly susceptible to HWIs. This makes it imperative to incorporate HWIs into both system analysis and optimization processes.

In summary, the growing body of literature clearly underscores the need to integrate HWIs into wireless system design. The reviewed studies lay down the foundation for the approach adopted in this thesis, which seeks to develop practical and robust resource allocation strategies under realistic hardware assumptions.

1.4 AI-based Resource Management in IoT Networks

1.4.1 Learning Approaches for Intelligent Systems

In the context of the increasing complexity and dynamic nature of IoT networks, efficient management of resources requires intelligent decision making mechanisms that go beyond static or offline optimization. Traditional approaches—such as optimization-based, heuristic, and classical ML methods, including supervised learning—made important contributions in modeling and solving resource allocation problems. However, these methods frequently rely on strong assumptions about network conditions, require centralized control, or depend on large

amounts of labeled data that may not reflect real-time variations or imperfections in practical deployments.

In addition, metaheuristic optimization techniques—such as particle swarm optimization, genetic algorithms, and other evolutionary methods—have also been widely applied to solve complex and non-convex resource allocation problems. While these approaches are effective in finding high-quality solutions, they typically rely on iterative search procedures that must be executed for each network realization, resulting in high computational overhead. Moreover, they do not inherently support online adaptation or policy generalization across different network states, which limits their applicability in highly dynamic and real-time IoT environments.

By contrast, RL has recently emerged as a powerful framework for adaptive, real-time decision-making through direct interaction with the environment. In RL, an agent learns an optimal policy by sequentially observing the system state, taking actions, and receiving feedback in the form of rewards, without requiring explicit system models or labeled data. In this thesis, the system is modeled such that each AP, cluster, or IoT link—depending on the considered scenario—acts as an RL agent. The state typically includes relevant network information such as CSI, interference levels, and HWI effects. The action corresponds to resource allocation decisions, such as transmit power selection or other transmission parameters, while the reward function is designed to capture system objectives such as SE or EE, potentially with penalties for interference or power consumption. This formulation enables distributed and adaptive control in highly dynamic environments.

This makes RL particularly suitable for multi-objective optimization, decentralized control, and online learning under partial or imperfect information, all of which are essential for robust operation in practical IoT systems. In the context of this work, the ability of RL to explicitly incorporate HWIs, adapt to time-varying interference, and learn long-term policies makes it a compelling choice for resource allocation in next-generation IoT networks. In what follows, we provide the background necessary to understand the learning algorithms used in this thesis.

1.4.1.1 Reinforcement Learning

RL is a learning paradigm where an agent interacts with an environment by taking actions and receiving feedback in the form of rewards or penalties. The agent's goal is to learn an optimal policy that would maximize cumulative long-term rewards through trial-and-error interactions. Unlike traditional ML approaches, such as supervised and unsupervised learning, RL excels in scenarios where the optimal solution is not explicitly provided (Hussain *et al.*, 2020). A key advantage of RL is its ability to learn dynamically through interaction with the environment (Park, Abuzainab & Saad, 2016), thereby enabling the agent to adapt its behavior based on past interactions and progressively improve its policy. RL relies on the following elements:

- **States:** Every scenario an agent encounters in an environment is formally called a state. This scenario presents the agent's relation with its environment, which is everything the agent can directly or indirectly interact with.
- **Actions:** An action is a choice the agent can make in an environment.
- **Policy:** Policy $\pi(\cdot)$ is a mapping between the state and the action to be performed by the agent. The RL algorithm aims to find the optimal policy that maximizes the cumulative reward.
- **Reward:** The reward is a scalar function defining the main objective in an RL problem. It is achieved after the agent performs an action, i.e., moves from state s to the next state, s'. For instance, in a communication system, the reward function may be defined as the network's overall capacity, throughput, average sum rate, etc.
- **State-action value function**: The state-action value function quantifies the expected cumulative discounted reward when starting in state s_t and taking a particular action a_t under given policy $\pi(s/a)$.
- **Model**: The model is a representation of the environment. In fact, in RL, we have the following two types of algorithms: model-free RL and model-based RL. The former is offline learning, where the agent learns the best policy through trial and error such as temporal difference methods, Q-learning, SARSA, and actor-critic. However, for the latter (online

learning), the agent plans actions before any interactions with the environment since it knows the model of the environment; here, we find value iteration and policy iteration.

1.4.1.2 Markov Decision Process (MDP)

MDP is an approach in RL to make decisions in a stochastic environment. MDP models the environment as a discrete-time, probabilistic control process, where an agent interacts with the system by selecting actions that influence state transitions. At each time step, given a particular state s_t, the agent chooses an action a_t from a set of possible actions, resulting in a transition to a new state s_{t+1} and the reception of a corresponding reward r_{t+1}. The MDPs approach obeys the *Markov property*, which assumes that *the future is independent of the past given the present*, as shown below:

$$P(S_{t+1}|s_t) = P(s_{t+1}|s_1, ..., s_t). \tag{1.1}$$

An MDP is typically defined by the tuple (S, A, P, R), where S is the set of possible states, A is a set of actions, P is the state transition probability, and R is the reward received by the agent upon performing some action(s) at some state(s) in the environment. The MDP's main objective is to find a function, called a policy, which specifies which action to take in each state to maximize the cumulative reward.

In IoT networks, MDPs provide a principled approach to modeling and solving long-term resource allocation problems, such as power control, spectrum assignment, and user association. Specifically, MDPs are particularly suitable when the objective is to maximize the average reward over an infinite time horizon. This formulation is known as the *infinite-horizon average reward* setting, where the goal is to find a stationary policy π that maximizes:

$$\rho^{\pi} = \lim_{T\to\infty} \frac{1}{T} \mathbb{E}_{\pi} \left[\sum_{t=0}^{T-1} R(s_t, a_t) \right], \tag{1.2}$$

where T denotes the number of time steps.

Under the assumption that the state, action, and reward spaces are known, the environment's dynamics satisfy the Markov property, and the state distribution converges to a steady state, MDPs enable the derivation of optimal or near-optimal policies ensuring efficient and adaptive decision making over time. Accordingly, in this thesis, the resource allocation problem is modeled as an MDP, where the optimization objective is to maximize the infinite-horizon average reward. This enables the system to learn policies that not only react to the current state, but also consider long-term performance, making MDPs an ideal foundation for developing learning-based algorithms in dynamic IoT environments.

While the MDP framework provides a solid foundation for formulating sequential decision making problems in IoT networks, solving an MDP requires thorough knowledge of the environment's dynamics—namely, the state transition probabilities and the reward function. However, due to randomness, HWIs, and the dynamic behavior of multiple interacting agents, such information is frequently unavailable or difficult to model explicitly. To address this limitation, in this thesis, we consider model-free RL techniques, such as Q-learning. These approaches allow an agent to learn optimal policies directly from experience by interacting with the environment, without requiring prior knowledge of transition models. In what follows, we provide an overview of Q-learning, followed by its deep learning (DL) extension—namely, deep Q-networks (DQN)—which enables effective policy learning in high-dimensional state spaces commonly encountered in complex IoT scenarios.

1.4.2 Deep Q-Learning Overview

Q-learning is one of the most popular RL algorithms dealing with MDP problems (Mnih *et al.*, 2015). Conceptually, Q-learning aims to find an optimal policy telling the agent which action to take under what circumstances. It is a model-free RL that does not require a model or knowledge of the environment and where the transition and reward values are stochastic (Sutton & Barto, 2018).

Let S denote a set of possible states, while A denotes a set of discrete actions. At TS t, the RL agent observes the state of its environment ($s_t \in S$) and takes an action ($a_t \in A$) based on a certain policy $\pi(s/a)$. Once the agent takes an action, the environment moves to the next state, s_{t+1}. The agent then receives a reward r_{t+1} describing how beneficial the action taken at state s_t was. The tuple $(s_t, a_t, r_{t+1}, s_{t+1})$ forms an experience that describes the agent's interaction with the environment. The agent's main goal is to maximize the discounted accumulated reward, which is expressed as $R_t = \sum_{\tau=1}^{\infty} \gamma^{\tau}\, r_{t+\tau}$, where γ is the discount factor, and $r_{t+\tau}$ is the value of the reward at TS $t + \tau$. The agent needs to learn an optimal policy, $\pi^*(s, a)$, that would map the state space to the action space. Q-learning is based on the action value function, which represents the return expected for taking action a in state s using policy π. It is defined as $Q_\pi(s, a) = \mathbb{E}(R_t|s_t = s, a_t = a, \pi)$. The optimal action value function satisfies the Bellman optimality equation as follows (see Eq. (1.3)):

$$Q^*(s, a) = \mathbb{E}(r_{t+1} + \gamma \max_{a'} Q^*(s_{t+1}, a')|s_t = s, a_t = a), \tag{1.3}$$

where $Q^*(s, a) = \max_\pi Q_\pi(s, a)$ represents the maximum action value obtained by following any policy. Clearly, the optimal action selection policy in a given state s is defined as shown in Eq. (1.4):

$$\pi^*(s, a) = \begin{cases} 1, a = \arg\max_{a \in A} Q^*(s, a) \\ 0, \text{otherwise.} \end{cases} \tag{1.4}$$

Q-learning based on Eq. (1.3) is guaranteed to converge under the Markov property and sufficient exploration. However, in environments with large or continuous state spaces, its practical applicability becomes limited due to the need to explicitly store and update a Q-table for every state–action pair, as well as the requirement to sufficiently visit all states, which is infeasible in high-dimensional systems. As a result, despite its theoretical convergence guarantees, classical Q-learning does not scale to complex wireless environments.

To address this limitation, DQN replaces the tabular representation of the action-value function with a deep neural network (DNN), parameterized by θ, which approximates the Q-function as

$q(s, a; \theta)$ (Mao *et al.*, 2016; Jang & Yang, 2020). Instead of storing values for each state-action pair, the DNN generalizes across similar states and directly maps high-dimensional state inputs to Q-values for all possible actions. This function approximation capability enables DQN to handle continuous or very large state spaces that are common in IoT and wireless systems.

During training, the agent interacts with the environment using an ϵ-greedy policy, which balances exploration and exploitation: random actions are selected with probability ϵ to explore the environment, while the current learned policy is followed with probability $1 - \epsilon$ to exploit accumulated knowledge (Sutton & Barto, 2018). This mechanism allows the agent to gradually refine its policy while still discovering potentially better actions. Similarly to the Q-learning algorithm, the Q-function's values are obtained through trial and error and are updated as shown in Eq. (1.5):

$$q(s_t, a_t; \theta) \leftarrow (1 - \alpha) q(s_t, a_t; \theta) + \alpha [r_{t+1} + \gamma \max_{a'} q(s_{t+1}, a'; \theta)], \tag{1.5}$$

where α represents the learning rate. DQN is an off-policy RL algorithm storing the previous experience in a replay memory D. Then, a mini-batch from this memory is sampled to train the DQN by minimizing the mean square loss function. If the same DQN network is used to provide the ground truth for the loss function and update parameters, then the training of DQN becomes highly unstable. We address this concern by using a quasi-static target Q-network with parameter θ^- to predict the Q-function's target value (i.e., the ground truth). The loss function for training the DQN is constructed as shown in Eq. (1.6):

$$\mathcal{L}oss = \frac{1}{2} \sum_{(s,a,r,s') \in D} (r' - q(s, a; \theta))^2, \tag{1.6}$$

where $r' = r + \gamma \max_{a'} q(s', a'; \theta^-)$ is the Q-function's target value. To minimize Eq. (1.6), we iteratively update θ by sampling a mini-batch of experiences from D and using stochastic gradient descent (see Eq. (1.7)):

$$\theta \leftarrow \theta - [r' - q(s, a; \theta)] \bigtriangledown q(s, a; \theta). \tag{1.7}$$

The target Q-network is periodically updated by copying parameters from the trained DQN. The DQN rapidly converges to a suitable parameterized policy for a MDP with a continuous state space (Nasir & Guo, 2019).

Despite these improvements, DQN and DRL methods in general still face several challenges, including training instability due to function approximation and non-stationary data distributions, high sample complexity requiring extensive environment interactions, sensitivity to reward design, and limited interpretability, which makes it difficult to fully characterize or guarantee their behavior in critical systems. These limitations highlight that, although DRL significantly extends classical RL to complex environments, careful algorithm design is still required when applying it to practical wireless communication problems. These challenges are explicitly addressed in this thesis through appropriate design choices in the formulation of the state space, action space, and reward function, as well as through the use of distributed learning architectures, stable training strategies, and structured simulation environments.

1.4.3 Multi-Agent Reinforcement Learning

RL has gained considerable attention as a promising approach for enabling intelligent agents to operate effectively in complex and uncertain environments. Using trial-and-error interactions, RL allows a single agent to learn optimal behaviors without requiring prior knowledge of the environment. While single-agent reinforcement learning (SARL) was successfully applied in wireless networks, its centralized design and reliance on global state information limit its scalability and adaptability in large-scale, dynamic, and multi-user scenarios (Zhang *et al.*, 2025; Zhao *et al.*, 2019). By contrast, distributed environments such as IoT networks demand coordinating multiple autonomous agents that must make sequential decisions with a shared, frequently partially observable environment. In these settings, MARL emerges as a powerful framework. MARL enables agents to learn optimal strategies not only through their own interactions with the environment, but also by accounting for the actions and behaviors of other agents (Liu *et al.*, 2024b). This makes MARL particularly well-suited for decentralized and

dynamic systems characterized by uncertainty, limited observability, and potentially conflicting objectives.

When extending the traditional RL framework to multi-agent systems, the decision making process is typically modeled as a Markov game, also referred to as a stochastic game. This formulation, initially introduced by (Busoniu, Babuska & De Schutter, 2008), generalizes the MDP to account for scenarios involving multiple interacting agents. In a Markov game, each agent aims to maximize its own expected cumulative reward, while explicitly considering the presence, actions, and potential strategies of other agents.

A stochastic resource allocation game can be formally defined as a tuple $\mathcal{G} = (\mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$, where $\mathcal{N}$ denotes the set of agents, $\mathcal{S}$ is the global state space, and $\mathcal{A} = \{A_1, A_2, \ldots, A_N\}$ is the joint action space, where A_n denotes the action space of the n-th agent. Furthermore, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the state transition probability function modeling the stochastic dynamics of the environment. $\mathcal{R} = \{R_1, R_2, \ldots, R_N\}$ defines the set of reward functions, where $R_n : S_n \times A_n \to \mathbb{R}$ specifies the immediate reward received by the n-th agent for a given state and action. This formulation captures the decentralized and interactive nature of multi-agent decision making in stochastic environments, which is well-suited for modeling complex IoT resource allocation problems.

Despite its strong modeling capabilities, MARL introduces several practical challenges. First, the non-stationarity of the environment arises as multiple agents simultaneously adapt their policies while interacting, leading to a continuously evolving learning dynamics (Kapoor, 2018). Second, scalability becomes a critical issue in large-scale IoT systems, where the growing number of agents and the high-dimensional joint state-action space significantly increase computational complexity and training difficulty. Third, coordination among agents is non-trivial in decentralized settings with partial observability, as decisions must be made based on local information while accounting for the behavior of others. Finally, MARL methods often suffer from high sample complexity and training instability, which can hinder their practical deployment in wireless networks (Zhou, Liu & Tang, 2024).

To address these challenges, this thesis adopts distributed learning architectures and agent-level decision-making to improve scalability and reduce problem dimensionality. In addition, simulation-based training environments are employed to enhance training stability and mitigate sample inefficiency, enabling more robust and practical MARL-based resource allocation strategies for IoT networks. Having introduced learning-based approaches and their extension to multi-agent settings, the following sections review their application to key resource allocation problems in IoT networks. In particular, we first examine optimization-, heuristic-, and game-theoretical methods as classical baselines, followed by ML and RL-based approaches that enable adaptive decision making in dynamic environments. The review then focuses on more specific communication scenarios, including FBL-coded industrial IoT systems, MA techniques, device clustering strategies, and emerging DT-enabled wireless networks.

1.5 Resource Allocation

1.5.1 Optimization and Heuristic Approaches

To date, various model-driven algorithms to deal with resource allocation using optimization and heuristic techniques have been proposed. In (Shi, Razaviyayn, Luo & He, 2011), the authors proposed a weighted minimum mean squared error (WMMSE) algorithm, which can achieve optimal power allocation. Similarly, an iterative algorithm based on fractional programming (FP) was proposed in (Shen & Yu, 2018); this algorithm performs comparably to the WMMSE algorithm. Finally, in (Kim & Ko, 2015), the authors proposed a resource allocation technique that uses a genetic algorithm. They converted the allocation problem into a degree-constrained minimum spanning tree problem and used a genetic algorithm to reduce the time needed to find near-optimal solutions.

1.5.2 Game-Theoretical Approaches

To date, game theory, characterized by the ability to capture the competitive or cooperative interactions among rational agents, has been widely applied to model and solve resource

allocation problems in wireless networks. In (Zhou, Dong, Ota, Wang & Yang, 2016), the authors proposed a non-cooperative game-based resource allocation strategy for LTE-based IoT networks, focusing on minimizing interference and introducing an efficient power control mechanism. Similarly, in (Guruacharya, Niyato, Hossain & Kim, 2010), a Stackelberg game framework was adopted to address downlink power allocation in two-tier cellular networks, where macro and femto BSs act as leader and follower, respectively, aiming to maximize their own transmission capacities under power constraints. Another Stackelberg game-based approach was presented in (Wang, Song, Han, Zhao & Wang, 2013), tackling joint power control, channel assignment, and device-to-device communication scheduling.

However, despite their analytical elegance, game-theoretical models frequently rely on strong assumptions—such as complete information about all players' strategies, utility functions, and perfect channel knowledge—that limit their practicality in real-world scenarios. These assumptions become particularly restrictive in dynamic and uncertain wireless environments with prevalent HWIs, time-varying channels, and incomplete or delayed observations. Moreover, classical game-theoretical approaches typically struggle with scalability, as the number of possible strategy combinations grows exponentially with the number of devices. This makes real-time adaptation in dense IoT networks with stringent latency and energy constraints infeasible. These limitations highlight the need for more flexible and data-driven methods. ML and, more specifically, RL offer promising alternatives by enabling agents to learn optimal or near-optimal strategies through interaction with the environment, without requiring prior knowledge of system models or other agents' utilities.

1.5.3 ML-Based Approaches

Considering the scalability of ML algorithms to massive numbers of devices, numerous resource allocation solutions have been developed using ML algorithms. ML was previously reported to be able to overcome the complex issues associated with optimization-based methods and reduce the need for unrealistic assumptions to simplify non-convex optimization problems. Moreover, ML models can derive actions from run-time context data and re-train themselves following

changes in the environment (Hussain *et al.*, 2020). For instance, in (Fan, Gu, Nie & Chen, 2017), the authors proposed distributed Q-learning and classification and regression trees algorithms for power management in device-to-device communication networks. In (Li *et al.*, 2019), the authors used K-means-enabled unsupervised learning techniques to group the users in an area and allocate power to each of them in a way minimizing the overall interference in the network. The aforementioned studies confirmed that ML techniques can improve the SE and EE of wireless networks while reducing the time complexity of obtaining optimal solutions.

Recently, RL has emerged as a promising technique to solve complex optimization problems (Nguyen, Nguyen & Nahavandi, 2020). In RL, the agent learns the optimal policy while interacting with the environment. Owing to its ability to interact with an unknown environment through exploitation and exploration, RL was reported to be an efficient technique to solve radio resource optimization. In (Mota, Araujo, Costa Neto, de Almeida & Cavalcanti, 2019), the authors proposed an RL-based modulation and coding scheme (MCS) to maximize the SE of 5G networks. Similarly, in (Bruno, Masaracchia & Passarella, 2014), the authors introduced a Q-learning algorithm to solve the adaptive modulation and coding (AMC) problem in fourth-generation (4G) LTE networks. In another relevant study (Vu *et al.*, 2022), the authors proposed a MARL approach for joint channel assignment and power allocation in platoon-based cellular vehicle-to-everything systems. In (Bennis & Niyato, 2010), the authors elaborated a classical Q-learning algorithm to solve the power allocation problem for macro-femtocell coexistence networks.

However, since the practical wireless environment is characterized by vast and continuous state and action spaces, it remains challenging to find a suitable RL policy capable of learning the optimal mapping between environmental states and actions. DRL, which combines DL and RL, has gained a lot of scholarly attention in solving complicated control problems with highly-dimensional data (Xu, Wang, Tang, Wang & Gursoy, 2017). In the DRL framework, the agent follows a certain policy to maximize the discounted cumulative reward notion through a series of actions (Ahmed & Hossain, 2019). The agent learns the optimal policy to maximize the long-term reward by continuously interacting with the environment. For instance, an AP can

use DRL to allocate computation resources to users in real-time in accordance with a certain policy aimed at maximizing the network's overall SE. Importantly, the decision making process in IoT environments is frequently non-linear and may involve complex interactions. However, these complexities can be captured by DRL, with its capacity to model non-linear relationships through neural networks. To date, DRL has been widely applied in numerous resource allocation problems in various wireless communication environments, such as vehicle-to-vehicle networks (Ye, Li & Juang, 2019), heterogeneous networks (HetNets) (Zhao *et al.*, 2019; Wang *et al.*, 2020a), IoT networks (Omidkar *et al.*, 2022; Munaye *et al.*, 2021), and cognitive radio networks (Kaur & Kumar, 2020).

Furthermore, several studies investigated DRL-enabled resource management for wireless networks. In (Mao *et al.*, 2016) and (Meng, Chen, Wu & Cheng, 2020), the authors proposed a DRL-based downlink power allocation scheme for multi-cell networks. In another relevant study (Zhang, Kang, Ma, Teng & Guo, 2018b), the authors developed a DRL-based power allocation to maximize the overall capacity of an ultra-dense network. In (Nasir & Guo, 2019), DRL was used to control the transmit power of a wireless network with partial information about the environment and a non-negligible delay in obtaining CSI. In addition, in (Jang & Yang, 2020), the authors proposed a multi-agent DQN-based algorithm for resource allocation and power control in small cell wireless networks with limited information exchange between small cell BSs. In (Ahmed & Hossain, 2019), the authors suggested using a novel centralized DRL-based downlink power allocation scheme to maximize the total network throughput in a multi-cell system.

Furthermore, in (Zhang, Tan, Liang, Feng & Niyato, 2019a), the authors proposed an intelligent DRL-based MCS selection algorithm in a cognitive HetNets. Similarly, in (Naparstek & Cohen, 2019), a distributed DRL-based spectrum access approach capable of maximizing the network utility was outlined. In addition, a DRL-based energy-efficient link adaptation algorithm was proposed to jointly determine the transmitted power level and MCS (Parsa, Moghim & Salavati, 2022). Likewise, a MADRL framework was introduced for joint power allocation and MCS

selection to maximize throughput in a downlink cellular network (Jamshidiha, Pourahmadi, Mohammadi & Bennis, 2020).

DRL was also extensively applied in wireless body sensor networks. In (Xiao, Hong, Xu, Yang & Ji, 2022), the authors proposed an intelligent reflecting surface (IRS)-aided energy-efficient RL-based secure WBAN transmission scheme enabling the coordinator to jointly optimize the sensor encryption key and the transmit power. Up-to-date surveys about the application of DRL in wireless communication and networking can also be found in (Luong *et al.*, 2019) and (Alwarafy, Abdallah, Ciftler, Al-Fuqaha & Hamdi, 2022). To further narrow the focus to practical wireless communication constraints, the following section reviews resource allocation approaches in FBL-coded IIoT systems, which are essential for supporting URLLC in time-critical applications.

1.6 Power Allocation in FBL-Coded IIoT Systems

Several resource allocation schemes for FBL-coded IIoT systems were proposed in the state-of-the-art literature. In (Sun *et al.*, 2019), joint power and bandwidth allocation and active BS antenna selection to maximize the short blocklength regime's EE in URLLCs were outlined. However, the considered URLLC rate expression was not appropriate. Furthermore, the authors of (Ghanem *et al.*, 2019) proposed using beamforming optimization to maximize the weighted sum rate in multiple-input single-output and multi-user downlink URLLC networks. In (Nasir, Tuan, Nguyen, Debbah & Poor, 2021), joint bandwidth and power allocation to ensure max-min fairness in a multi-user URLLC network was outlined. In (Cao *et al.*, 2020), the authors investigated the joint optimization of FBL and shared-pilot length for a multi-device downlink IIoT network.

While the aforementioned works focused on downlink systems, for IIoT networks, uplink resource allocation is equally important. The authors of (Ranjha & Kaddoum, 2021) investigated the possibility of jointly optimizing the blocklength and a drone-mounted flying BS's position, height, and beamwidth to minimize the power consumption of uplink URLLC devices. In

another relevant study, (Ren *et al.*, 2019a), the authors investigated how transmit power allocation can be used to minimize the decoding error of relay-assisted URLLC IoT systems. In (Ren *et al.*, 2020), a joint pilot and payload power allocation scheme for massive MIMO-enabled IIoT networks was proposed.

However, the aforementioned studies either ignored interference in the network by using orthogonal RRB allocation (Sun *et al.*, 2019; Nasir *et al.*, 2021; Cao *et al.*, 2020; Ren *et al.*, 2019a, 2020) or adopted complex iterative optimization methods (Ghanem *et al.*, 2019). Importantly, in an FBL-coded system, the achievable rate was previously reported to be a highly evolved function of the transmit power, blocklength, and decoding error probability (Polyanskiy, Poor & Verdu, 2010), thus leading to computationally challenging non-convex resource allocation problems. The concurrent presence of multi-user interference and HWI-induced signal distortions makes the resource allocation problems even more complicated. Standard iterative optimization methods, such as successive convex approximation and difference-of-convex optimization, require many iterations and a significant amount of memory to converge, which makes them inefficient for dynamic resource optimization in large-scale IIoT networks. These limitations motivate the adoption of DRL, which has recently emerged as a promising solution for addressing complex, non-convex optimization problems. More specifically, the fact that DRL can learn a suitable policy by adapting to dynamic and complex environments makes it an ideal candidate for resource allocation in IIoT networks, particularly in the presence of co-channel interference and unknown HWI-induced signal distortions.

1.7 Multiple Access Systems

1.7.1 Multiple Access and Interference Management

Designing optimal MA schemes is central for interference mitigation and improving resource use in multi-user networks. Although OFDMA, which the LTE-A and 5G NR standards rely on, can avoid multi-user interference by allocating each user dedicated RRBs, it is inherently resource-inefficient for dense networks. In recent years, RSMA has emerged as a novel MA scheme to

simultaneously schedule multiple users over the same RRB. RSMA manages interference in multi-antenna systems by splitting messages into the power and spatial domains and providing flexibility between complete and partial interference cancellation at user devices (Mishra, Mao, Dizdar & Clerckx, 2022). Furthermore, available evidence suggests that RSMA requires fewer SIC operations at devices than NOMA (Clerckx *et al.*, 2023), which makes RSMA more suitable for low-complexity IIoT devices. In addition, even in single-antenna multi-user networks, in the presence of erroneous CSI at the transmitter or practical SIC constraints, RSMA can achieve a higher sum rate than NOMA (Hassan *et al.*, 2021a), (Yang, Chen, Saad & Shikh-Bahaei, 2021). When it comes to improving the EE of single-antenna multi-user networks, RSMA's effectiveness was also demonstrated in (Hassan, Hossain, Cheng & Leung, 2021b). Yet, while these studies reported RSMA's advantages for infinite blocklength-coded systems, RSMA was also shown to be effective at dealing with interference in FBL-coded systems. For instance, in (Xu, Dizdar & Clerckx, 2023), the authors optimized the RSMA scheme for a two-user FBL-coded uplink communication system, while (Xu, Mao, Dizdar & Clerckx, 2022b) proposed optimizing the precoding matrix in order to maximize the sum rate in multi-antenna FBL downlink communications that require low latency. However, the HWI-induced signal distortions resulting from the FBL-coded devices' low-complexity RF front ends were ignored in these studies. In addition, both (Xu *et al.*, 2023) and (Xu *et al.*, 2022b) rely on time-consuming iterative optimization methods to find the near-optimal solution for each channel fading state, which is challenging to implement in delay-constrained networks.

1.7.2 Device Clustering in Multiple Access

The clustering of devices in accordance with specific criteria was extensively studied in combination with various access technologies, including NOMA (Zhang *et al.*, 2020; Ren, Wang, Xu, Fang & Ding, 2019b), OFDMA (Abdelnasser, Hossain & Kim, 2014; Hatoum, Langar, Aitsaadi, Boutaba & Pujolle, 2014) and RSMA. Specifically, in the context of RSMA technologies, available research delved into employing the RSMA strategy within each cluster. This strategy aims to schedule the devices in an RRB in such a way as to potentially enhance

network performance and resource utilization. In (Hassan *et al.*, 2021b), the authors proposed using a clustering algorithm to organize devices into non-overlapping clusters according to their respective locations. The authors also applied an RSMA strategy to enable devices to receive data from a suitable F-AP and a cloud BS (CBS) over the same RRB. Similarly, in (Hassan *et al.*, 2021a), the authors extended the application of the RSMA strategy to each cluster. However, to this end, the authors used a distinct approach for clustering user devices in this context. Specifically, they used a MARL technique to maximize the user device's long-term achievable data rate. In (Katwe, Singh, Clerckx & Li, 2022), the authors proposed a low-complexity k-means clustering algorithm that dynamically divides multiple users into clusters based on their respective locations.

Taken together, the aforementioned works demonstrate that clustering can improve the RSMA strategy's performance. However, in the literature, it remains largely unexplored how clustering can be exploited to enhance the performance of RSMA-aided networks in the presence of FBL-coded data transmission and HWI-induced signal distortions. In this context, this thesis adopts a greedy clustering strategy due to its low computational complexity, scalability, and suitability for real-time implementation in large-scale IoT networks with limited local information. This research gap is addressed in Chapters 3 and 4 of the present thesis.

1.8 Digital Twin Network (DTN)

While previous DRL-based studies demonstrated the potential of DRL for addressing resource allocation challenges, there remain several barriers in deploying DRL solutions in real-world environments. A key issue arises during the training phase, where DRL algorithms explore the environment to optimize their decision making. However, incorrect or suboptimal decisions during this exploration can lead to unintended consequences, such as system inefficiencies or failures, when training directly within the physical network. DTNs provide a vital solution to the challenges of deploying DRL in real-world IoT systems. By creating a virtual replica of the physical network, DTNs enable simulating real-world scenarios, thereby allowing algorithms to be tested and optimized without impacting the actual operational network (Zhang *et al.*,

2024). This approach offers a controlled, risk-free environment where DRL models can explore and refine their decision making processes, thus ensuring safe, effective performance before real-world deployment. In addition, DTNs can generate realistic, real-time data, which is crucial for training artificial intelligence (AI) and ML models, ensuring that resource allocation strategies are not only efficient, but also adaptable to dynamic network conditions. By integrating DTNs, IoT networks can use the power of advanced AI-driven solutions like DRL while safeguarding system integrity and performance in a manageable and secure manner.

For instance, in (Zhang *et al.*, 2024), the authors proposed DT-enhanced DRL to optimize resources in network slicing. In (Cui, Lv, Ni & Jamalipour, 2023), the authors elaborated a DT-aided learning framework to maximize the sum rate of the new reconfigurable intelligent surface-assisted uplink, user-centric cell-free system. Furthermore, in (Elloumi, Hassan & Kaddoum, 2025a) and (Elloumi, Kaddoum, Hassan & Selim, 2025b), high-fidelity DTNs for dynamic internet-of-vehicle networks were developed using NVIDIA Sionna and Microsoft Azure DT tools, respectively, thus enabling the design of proactive interference management algorithms. In another relevant study (Haider *et al.*, 2025), the authors presented a high-fidelity RF channel twin, demonstrating that a DTN leveraging off-the-shelf ray-tracing and AI tools can reliably replicate complex channel conditions in dynamic wireless networks.

CHAPTER 2

SPECTRAL EFFICIENCY IMPROVEMENT IN DOWNLINK FOG RADIO ACCESS NETWORK WITH DEEP REINFORCEMENT LEARNING-ENABLED POWER CONTROL

Nahed Belhadj Mohamed[1] , Md. Zoheb Hassan[2], Georges Kaddoum[1]

[1] Electrical Engineering Department, École de technologie supérieure (ETS), 1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3
[2] Electrical and Computer Engineering Department, Université Laval, 2325 Rue de l'Université, Québec, QC G1V 0A6

Paper published in *IEEE Internet of Things Journal*, vol. 10, no. 17, pp. 15044-15059, September 2023.

2.1 Abstract

F-RAN is a promising architecture that leverages edge computing and caching to improve devices' latency and QoS. However, interference, which arises when multiple devices are concurrently scheduled on the same RRB, limits the performance of a dense F-RAN. This paper considers a multi-cell F-RAN in which the devices of each small cell receive data from the associated F-AP over the same RRB(s). The F-APs transmit data to the associated devices using RSMA schemes to manage co-channel interference within the small cells efficiently. A transmit power control scheme is proposed to maximize the network's SE while considering the devices' HWIs. The considered transmit power control scheme is an NP-hard problem, which is highly challenging to solve using the legacy optimization approach. To address this challenge, we propose a distributed DRL-based power allocation (DDPA) scheme that takes the time-varying dynamics of the network and the HWIs of devices into account. Each F-AP in the proposed framework is equipped with a DRL agent that collects SINR and CSI from connected devices and adapts the transmit power allocation in each scheduling interval. In addition, the EL framework is exploited to further improve the proposed DDPA scheme's performance. We use extensive simulations to demonstrate that the DDPA scheme achieves greater SE than contemporary transmit power control schemes. In particular, the proposed DDPA scheme is especially suited to scenarios with non-negligible HWI-induced distortion.

2.2 Introduction

IoT is expected to revolutionize the world through seamlessly connected heterogeneous smart devices and enhance living standards by providing solutions in various domains, including healthcare, smart cities, and transportation (Shafique, Khawaja, Sabir, Qazi & Mustaqim, 2020). There are expected to be around 500 billion IoT devices in use by the end of 2030 (Cisco, 2016), which is about 59 times the expected world population at that time (8.5 billion (News, 2015)). These IoT devices will have a wide variety of QoS requirements. Therefore, resource optimization will play a critical role in enhancing the IoT network's capacity and reducing its energy consumption. On the other hand, resource allocation using conventional optimization approaches usually takes a long time to converge and is complex. As a result, conventional optimization approaches will not be suitable to perform optimal resource allocation for large IoT networks.

The F-RAN architecture, which leverages emerging computational intelligence at the network edge, is proposed as a promising paradigm to provide high SE and EE for future generation wireless communication systems, such as 5G and beyond-5G (B5G) networks (Peng, Yan, Zhang & Wang, 2016). F-RANs consist of multiple F-APs equipped with cloud computing and caching capabilities. The F-APs process users' requests in real-time and transmit cached content directly to users without any intervention from the centralized cloud server. Thus, F-RAN makes it possible to decentralize resource management and signal processing, which can considerably improve end-to-end latency compared to the contemporary cloud radio access network (C-RAN) (Dinh, Kaneko, Fukuda & Boukhatem, 2021). However, due to the scarcity of the RRBs compared to the numbers of IoT devices, multiple IoT devices have to simultaneously share the available RRBs. Such a fact introduces interference in the system, and inevitably impedes the SE and EE of the F-RAN (Hassan *et al.*, 2021b). This fact motivates our research on developing a reliable power management approach to overcome interference in F-RANs.

Lately, the RL framework has gained considerable attention for scalable optimization in wireless communications. RL is a category of ML in which an agent learns how to interact with the

environment by performing actions and observing the results. In particular, the agent learns an optimized policy to perform actions so that the long-term reward is maximized. RL techniques are effective in decision making scenarios without any prior knowledge of the environment. Consequently, RL is suitable for resource allocation problems in F-RANs with incomplete or no user parameter information. In particular, the RL framework is applied in HetNets (Wang *et al.*, 2020a), cognitive radio networks (Kaur & Kumar, 2020), and unmanned aerial vehicle (UAV) networks (Cao *et al.*, 2019). A promising technique called DRL was designed to leverage the considerable advancements of DL. DRL is a combination of RL and DL that deals with the decision making problems of very large state and action spaces. DRL is a suitable technique to manage interference in F-RANs because it can near-optimally solve complex and highly-dimensional optimization problems (Naderializadeh *et al.*, 2021).

When multiple users simultaneously transmit over the same RRBs, MA technology plays a vital role in improving SE in the network. Recently, RSMA has received considerable interest to manage inter-user interference in both single-antenna and multi-antenna multi-user communication networks (Hassan *et al.*, 2021a; Park, Choi, Lee, Shin & Poor, 2023). RSMA makes it possible to effectively transmit data to multiple users over the same RRBs. To this end, the users' messages are encoded into a common message and a set of private messages at the transmitter end. The common message is intended for a set of scheduled users, whereas each private message is intended for a specific user. These messages are transmitted simultaneously using the superposition technique (Li, Chai, Li & Li, 2020). Each receiver first decodes the common message, removes its interference by applying SIC, and subsequently decodes its private message. RSMA can not only improve the SE of multi-cell networks (Dahrouj & Yu, 2011) but also outperform contemporary MA techniques by effectively managing co-channel interference in the network (Mao, Clerckx & Li, 2019).

This article proposes a DDPA scheme based on DRL to maximize the network SE of the RSMA-enabled F-RAN architecture. The proposed DDPA scheme enables all F-APs to compute their power profiles in response to a variation in channel characteristics and network conditions.

2.2.1 Motivation

Despite the remarkable progress that has been made in optimization, game theory, and RL-based transmit power control solutions, the state-of-the-art literature has certain limitations. Generally speaking, algorithms that are based on optimization and heuristic techniques experience severe obstacles in practical deployment since they rely on a clearly defined convex optimization approach. However, power allocation is often non-convex in practice due to the presence of HWIs in the transmitters and receivers. Existing optimization-based power control approaches are not efficient in the presence of HWIs since an accurate model of the HWIs is usually intractable. To overcome the non-convexity of the problem, the solutions proposed in (Shi *et al.*, 2011; Shen & Yu, 2018; Kim & Ko, 2015) adopt iterative and heuristic approaches, which require significant convergence time and memory. They are unsuitable when rapid power control decisions need to be made. Consequently, optimization-based methods are not appropriate for transmit power control in future communication networks, especially in large-scale and complex F-RANs.

On the other hand, game theory-based transmit power control is interesting for F-RANs because of the decentralization aspect of implementation. Note that the game theory-based framework requires that the interaction between the players and the environment be accurately modeled to achieve stable solutions. However, it is challenging to accurately model that interaction with incomplete information, especially when the number of players increases considerably. Consequently, the effectiveness of game theory-based transmit power control is limited for large-scale F-RANs.

Finally, compared with optimization- and game theory-empowered solutions, DRL-empowered solutions are more appropriate for future wireless networks due to the DRL framework's ability to learn optimal solutions by interacting with the environment. However, DRL's effectiveness at managing interference in an F-RAN in the presence of HWIs is not evident from the state-of-the-art literature. Most of the existing works on radio resource allocation usually assume ideal hardware, while in the practical environment, the transmitter and the receiver devices suffer from

different types of distortions that affect the communication systems' performance. This work is motivated by the aforementioned facts and applies DRL techniques to manage power allocation in an RSMA-based F-RAN while considering both co-channel interference and HWI-induced signal distortion.

2.2.2 Contributions

This paper investigates the transmit power allocation problem for a downlink RSMA-based F-RAN architecture composed of several F-APs each transmitting data to its associated users over orthogonal RRBs to avoid inter-cell interference. However, we consider that each F-AP serves multiple users simultaneously over the same RRB using the RSMA technique. As a result, the SE of the network depends heavily on the effective mitigation of intra-cell interference. Hence, an optimal transmit power allocation scheme is proposed to maximize overall network capacity and mitigate intra-cell interference. The specific contributions of this work are summarized as follows.

1. Motivated by the fact that future wireless networks will deploy intelligent APs to support a massive number of IoT devices, we use a MARL technique to address the practical aspects of wireless resource management. In our proposed system, each F-AP represents an RL agent, and its role is to make suitable transmit power allocation decisions in a distributed and dynamic way. Each agent makes its own decision at each scheduling interval based only on the observations it makes about the environment.
2. DDPA algorithm is proposed by leveraging the DQN, which is model-free and holds up to unpredictable changes in the wireless network. To further improve the DQN-based DDPA algorithm's performance, an ensemble DQN (eDQN) learning algorithm is proposed by integrating the DQN-based power control algorithm with EL techniques. The proposed eDQN-based DDPA algorithm creates K-independent copies of the DQN-based DDPA scheme trained with different wireless network configurations and by varying the discount factor from model to model. It then selects the model whose resultant transmit power profile provides the largest sum rate. Simulations show that the eDQN-based DDPA algorithm holds

up against the local optimality of the training data sets and can considerably outperform the DQN-based DDPA algorithm.

3. We emphasize that our proposed algorithms (eDQN- and DQN-based DDPA) are resilient to intra-cell interference and non-negligible transceivers' HWIs. The proposed algorithms do not need to know the exact model of HWIs and do not suffer from sub-optimality like all the well-known transmit power allocation schemes do. To the best of our knowledge, this is the first work investigating optimal power allocation for RSMA-enabled F-RAN systems while considering both co-channel interference and HWI-induced distortion.
4. Extensive simulations have been conducted to compare the proposed algorithms with state-of-the-art transmit power allocation algorithms. The simulation results demonstrate that our proposed algorithms achieve better achievable SE than the existing transmit power allocation algorithms for various user deployments and distortion scenarios.

The remainder of this article is organized as follows. In Section 2.3, the system model and the problem formulation are presented. The proposed transmit power allocation algorithms are explained in Section 2.4. The simulation results and discussion are presented in Section 2.5. Finally, some concluding remarks are provided in Section 2.6.

2.3 System Model and Problem Formulation

2.3.1 System Overview

We consider a downlink F-RAN architecture that contains a CBS and M small cells, which are randomly dropped within the coverage area of the macro cell. A single-antenna F-AP is located at the center of each small cell, and N devices[1] are randomly located within the area of each small cell according to a uniform distribution. The F-APs are connected to the CBS via high-capacity fronthaul links. Fig. 2.1 provides an example of such a network. The set of F-APs is denoted by $\mathcal{M} = \{1, .., M\}$, the set of user devices in each small cell is represented by $\mathcal{N} = \{1, .., N\}$, and the set of RRBs is denoted by $\mathcal{K} = \{1, .., K\}$.

[1] Note that the terms "user", "user device", and "device" are used interchangeably throughout the paper.

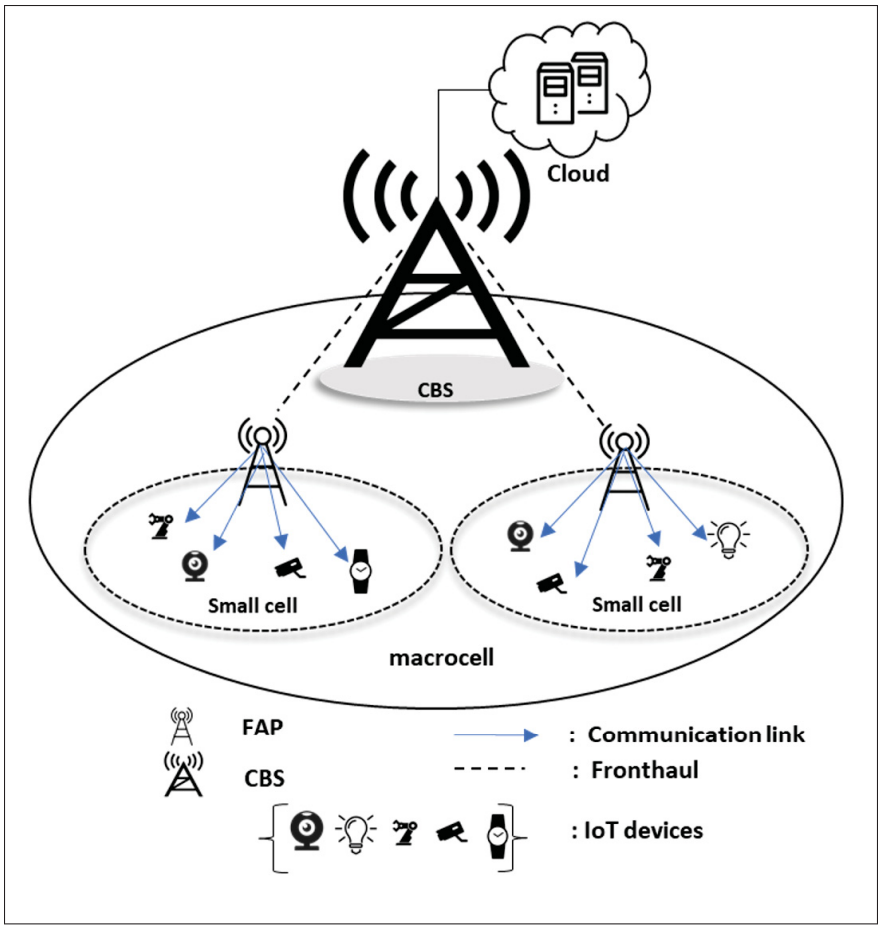


Figure 2.1 A downlink F-RAN with a CBS, two F-APs, and four IoT devices in each small cell

The F-APs have limited cloud computing capabilities to execute radio resource management (RRM) tasks (Dinh *et al.*, 2021). The F-APs can transmit data directly to the cache hit devices without delivering data from the CBS over the fronthaul links. We assume that the contents required by devices are available in their serving F-APs, hence, devices obtain data by accessing the local cache directly. For simplicity, in this work, we consider the devices to be served only by their nearest F-AP. An extension to this, in which devices can receive data from both their nearest F-AP and the CBS, is left for future work.

The devices in each small cell receive data from the associated F-AP over the same RRB. Therefore, a set of orthogonal RRBs is deployed to mitigate interference among different F-APs.

Moreover, to combat intra-cell interference in each small cell, a one-layer RSMA strategy is used for the devices that are served over the same RRB.

For analytical tractability, we make the following assumptions.

A1: Each F-AP is aware of the local CSI of its assigned devices. At TS t, the downlink channel gain from the m-th F-AP to the n-th device over the k-th RRB is modeled the same as in article (Nasir & Guo, 2019), $g^k_{m,n}(t) = |h^k_{m,n}(t)|^2 \beta_{m,n,k}$, where $h^k_{m,n}$ represents small-scale complex fading, and $\beta_{m,n,k}$ denotes large-scale fading, which includes both path loss and shadowing from the m-th F-AP to the n-th device.

A2: The system runs on a slotted-time basis, with optimization parameters being updated every TS. On one hand, we assume that small-scale fading remains constant in a given TS and varies from one TS to the next. On the other hand, we assume that large-scale fading remains the same over many TSs.

A3: The devices accurately perform perfect SIC to decode the common and private messages transmitted from their associated F-AP.

2.3.2 RSMA Strategy

Each F-AP broadcasts data to its associated user devices using a one-layer RSMA technique. Without a loss of generality, the rate expressions for RSMA-enabled data transmission from the m-th F-AP are as follows. First, the F-AP splits each user device's data into common and private parts. The common parts of all devices' data are combined into a common message, W_c, which is encoded to a common stream, s_c. Meanwhile, the private parts of the devices' data, $W_{p,n}$, are independently encoded into N individual private streams, $s_{p,n}$. The s_c and $s_{p,n}$ streams are then linearly precoded and simultaneously transmitted over the same RRB. The signal transmitted from the m-th F-AP is expressed as:

$$x_m = \sqrt{P_c}\, s_c + \sum_{n=1}^{N} \sqrt{P_{p,n}}\, s_{p,n}, \tag{2.1}$$

where P_c and $\{P_{p,n}\}$ denote the transmission powers assigned to the common and private messages, respectively, and $P = [P_c, P_{p,1}, .., P_{p,N}]$ denotes the power profile. At TS t, the received signal at the n-th device is given by:

$$y_n(t) = \sqrt{g^k_{m,n}(t)}\, x_m + n, \tag{2.2}$$

where $g^k_{m,n}(t)$ is the channel gain between the m-th F-AP and the n-th device over the k-th RRB at the t-th TS, and $n \sim \mathcal{N}(0, \sigma^2)$ is additive white Gaussian noise (AWGN) with variance σ^2. Without loss of generality, we assume all receivers have the same noise power spectral density (PSD). Each device first decodes the common message by treating interference from the private streams as noise and removing this interference using SIC. In a given small cell, the common message, s_c, must be decoded by each device. Therefore, the rate of the common stream for the set of devices served by the m-th F-AP over the k-th RRB is expressed as:

$$R_c = \min \left[\log_2 \left(1 + \frac{P_c\, g^k_{m,1}(t)}{\sum_{n=1}^{N} P_{p,n}\, g^k_{m,1}(t) + \sigma^2} \right), \cdots, \log_2 \left(1 + \frac{P_c\, g^k_{m,N}(t)}{\sum_{n=1}^{N} P_{p,n}\, g^k_{m,N}(t) + \sigma^2} \right) \right]. \tag{2.3}$$

After removing the common message's interference using SIC, each device decodes its private message by treating the interference coming from the other devices' private messages as noise. Therefore, the rate of the private stream for the n-th device served by the m-th F-AP over the k-th RRB is expressed as:

$$R_{n,p} = \log_2 \left(1 + \frac{P_{p,n}\, g^k_{m,n}(t)}{\sum_{\substack{n'=1 \\ n' \neq n}}^{N} P_{p,n'}\, g^k_{m,n}(t) + \sigma^2} \right). \tag{2.4}$$

The total rate for all devices served by the m-th F-AP is given by:

$$R_m = R_c + \sum_{n=1}^{N} R_{n,p}. \tag{2.5}$$

These rate expressions are valid when there are no HWIs at the transceivers. However, in practice, there are several distortions in the transmitted waveforms due to the presence of impairments in the RF chain of the transceivers caused by the non-linear power amplifier, the mixer, the filter, and low-resolution analog-to-digital/digital-to-analog conversion. To take the impact of HWIs into account, we also provide below the SINR expressions of the common and private messages in the presence of HWIs at the transceivers. In the presence of HWIs, the expression for the received signal at the n-th device in the TS t is instead $y_n(t) = \sqrt{g^k_{m,n}(t)}\ x_m + n + z$, where z represents the signal distortion caused by HWIs. In this paper, to evaluate the proposed algorithms' performance in the presence of HWIs, two well-known distortion models, namely, linear and non-linear distortion are considered. The SINR expressions for each distortion model are provided below.

1. **Linear Distortion**

   In this case, $z = \sqrt{g^k_{m,n}(t)}\ k\ x_n$, where k is a positive real-number (Matthaiou, Papadogiannis, Bjornson & Debbah, 2013) and (Zhang, Matthaiou, Coldrey & Björnson, 2015). Based on (2.1), if $S = \{s_c, s_{p,1}, ..., s_{p,N}\}$ follows a normal distribution $\mathcal{CN}(0, 1)$ and the elements in S are independent, the variance of z is expressed as:

$$\begin{aligned} &Var(z) = \mathbb{E}(z^2) - (\mathbb{E}(z))^2 = \mathbb{E}(z^2) \\ =&k^2 g^k_{m,n}(t) P_c\ Var(s_c) + \sum_{n=1}^{N} k^2 g^k_{m,n}(t) P_{p,n}\ Var(s_{p,n}) \\ =&k^2 g^k_{m,n}(t)(P_c + \sum_{n=1}^{N} P_{p,n})\ = k^2 g^k_{m,n}(t) P_t, \end{aligned} \tag{2.6}$$

   where $P_t = P_c + \sum_{n=1}^{N} P_{p,n}$ is the total power provided by the F-AP.

   The received SINRs of the common and private streams at the n-th device are expressed as:

$$\text{SINR}^{Linear}_{c,n} = \frac{P_c\ g^k_{m,n}(t)}{\sum_{n=1}^{N} P_{p,n}\ g^k_{m,n}(t) + K^2 g^k_{m,n}(t) P_t + \sigma^2} \tag{2.7}$$

and

$$\text{SINR}_{p,n}^{Linear} = \frac{P_{p,n}\, g_{m,n}^{k}(t)}{\sum_{\substack{n'=1 \\ n' \neq n}}^{N} P_{p,n'}\, g_{m,n}^{k}(t) + K^2 g_{m,n}^{k}(t) P_t + \sigma^2}, \tag{2.8}$$

respectively.

2. **Non-Linear Distortion**

This article considers the non-linear distortion introduced by the clipping of power amplifiers. We use the standard clipping distortion models to obtain the SINR expressions (Lee, 2021), (Lee, 2018). According to (Lee, 2021, eq. (16)), clipping distortion is a complex Gaussian random variable given by:

$$z \sim \mathcal{CN}\left(0, \chi(e^{-v^2} - \sqrt{\pi} v\ \text{erfc}(v))\right), \tag{2.9}$$

where χ represents oversampling and filtering for the clipping process and v represents the clipping ratio (CR). More specifically, v is a function of the total receive power at the received front-end (Lee, 2021, eq. (5)), and $\chi = 3.85$ when $v[dB]$ is in the interval $[0dB, 3dB]$ or $\chi = 2.85$ when $v > 3dB$. Using this model, the SINR expressions for the common and private streams are expressed as:

$$\text{SINR}_{c,n}^{Clipping} = \frac{P_c\, g_{m,n}^{k}(t)}{\sum_{n=1}^{N} P_{p,n}\, g_{m,n}^{k}(t) + \chi(e^{-v^2} - \sqrt{\pi} v\ \text{erfc}(v)) + \sigma^2} \tag{2.10}$$

and

$$\text{SINR}_{p,n}^{Clipping} = \frac{P_{p,n}\, g_{m,n}^{k}(t)}{\sum_{\substack{n'=1 \\ n' \neq n}}^{N} P_{p,n'}\, g_{m,n}^{k}(t) + \chi(e^{-v^2} - \sqrt{\pi} v\ \text{erfc}(v)) + \sigma^2}, \tag{2.11}$$

respectively.

The data rate in the presence of HWIs can be obtained by substituting (2.7), (2.8), (2.10), and (2.11) to (2.3) and (2.4), respectively.

Remark 1: Note that our proposed algorithms (eDQN- and DQN-based DDPA) learn the transmit power allocation from the received SINRs at the receivers. As a result, they are generic and also valid for other distortion models at the transmitters or receivers.

2.3.3 Problem Formulation

In this work, we aim to maximize the network's SE subject to transmit power constraints. The optimization problem can be formulated mathematically as follows:

$$\begin{aligned} &\max_{\boldsymbol{P}} \quad R_{total} \\ &\text{s.t.} \begin{cases} \text{(C1)} & P_{min} \leq P_c \leq P_{max} \\ \text{(C2)} & P_{min} \leq P_{p,n} \leq P_{max}, \end{cases} \end{aligned} \tag{2.12}$$

where R_{total} represents the overall throughput of the system, i.e., $R_{total} = \sum_{m=1}^{M} R_m$. The constraints (C1) and (C2) imply that the transmit powers of the common and private messages are bounded by P_{min} and P_{max}, which are system-defined parameters. P_{max} is the maximum PSD a transmitter can emit, and P_{min} represents the minimum transmit power.

Eq. (2.12) is generally non-convex and has been shown to be NP-hard (Luo & Zhang, 2008). Hence, it is computationally intractable to global-optimally solve (2.12). Moreover, due to its high computational complexity, is also challenging to solve (2.12) using traditional optimization methods, especially in a dynamically varying environment. In light of this challenge, we employ a DRL framework to efficiently solve (2.12). In the next section, we provide an overview of the DRL framework and then we propose two DRL-enabled algorithms to near-optimally solve (2.12).

Remark 2: The reasons behind selecting the DRL framework to solve (2.12) are as follows. First, due to the presence of both co-channel interference and HWI-induced signal distortion, (2.12) is a non-convex and NP-hard optimization problem. Hence, the standard iterative optimization methods fail to reach a globally optimal solution to solve (2.12). Second, the transmit power

allocation needs to be periodically updated due to the dynamic variation of the communication environment in F-RAN. More specifically, a transmit power allocation algorithm needs to be rapidly executed within the channel coherence time. However, the high computational complexity of the iterative optimization method makes it extremely challenging to adapt the transmit power with rapidly dynamic channel variation. Finally, standard optimization methods require accurate models of impairment. Unfortunately, in most practical cases, exact models of HWI-impaired signal distortions at the transceivers are not available. It is for these reasons that standard iterative optimization methods are not suitable to solve (2.12) for F-RANs in practice. We employ a DRL technique to solve (2.12) and address these challenges. On one hand, DRL is an effective methodology to near-optimally solve complex and highly-dimensional optimization problems (Naderializadeh *et al.*, 2021). On the other hand, DRL can effectively deal with the decision making problems associated with very large state and action spaces in unknown and dynamically changing environments. In particular, in our proposed DRL setting, the F-APs act as the distributed agents and learn the optimal policy for selecting suitable transmit power levels for their associated user devices based on the states they receive from the surrounding environment. Essentially, a DRL framework is suitable to solve (2.12) while taking the dynamic variation of co-channel interference, signal distortions, and communication channels into account.

2.4 Proposed Solution: DRL-based Power Allocation in F-RAN Systems

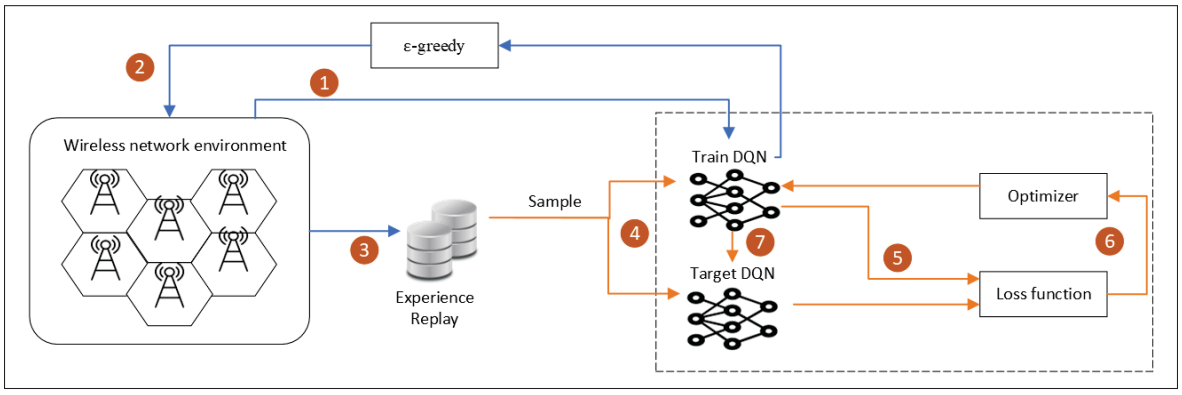


Figure 2.2 DQN-based DDPA

2.4.1 The Proposed DDPA

The transmit power control problem in the downlink F-RAN system is modeled as a MARL problem consisting of multiple RL agents and an environment that interacts with each other. As shown in Fig. 2.2, the F-APs are treated as the RL agents, and the RSMA-enabled F-RAN system is the environment. Each F-AP is an independent agent that determines it is own transmit power profile based on the state that it receives from the surrounding environment. Power is optimally allocated using the derived power profile, and the environment returns the reward function, which is the SE of the downlink F-RAN system.

In our work, we adopt the distributed training and execution scheme. Therefore, we have two procedures: an online decision making process and an offline training procedure. The online decision making process, indicated by a blue line in Fig. 2.2, involves the agent performing an action a_m^t based on the current state s_m^t and according to the ϵ-greedy process. The offline training procedure is indicated by an orange line in Fig. 2.2. Each agent's experience is stored in the experience replay memory, and the agent calculates the loss function using a mini-batch of experiences sampled from this memory. Then, the loss function is minimized using stochastic gradient descent, and the weights of the trained DQN are updated. The elements of the proposed DDPA are:

- **Agent:** The agents are the F-APs.
- **States:** A state in DRL includes all the information about the environment to ensure the RL agent makes an informed decision about which action to take. The first element of state is the set of the local CSI $(g_{m,1}^k(t), g_{m,2}^k(t), ..., g_{m,N}^k(t))$, which represent the most critical feature. The second element is the set of the SINR of the common stream $(SINR_{c,1}, SINR_{c,2}, ..., SINR_{c,N})$. Finally, the third element is the set of the SINR of the private stream $(SINR_{p,1}, SINR_{p,2}, ..., SINR_{p,N})$. In other words, the states of all agents in the system can be defined as:

$$S = \{s_1^t, s_2^t, \ldots, s_M^t\}, \tag{2.13}$$

where

$$s_m^t = \{(g_{m,1}^k(t), ..., g_{m,N}^k(t)), (SINR_{c,1}, ..., SINR_{c,N}), (SINR_{p,1}, ..., SINR_{p,N})\}. \quad (2.14)$$

- **Actions:** Similar to (Nasir & Guo, 2019; Ge, Liang, Joung & Sun, 2020), we use discrete power level taken between P_{min} and P_{max}. We assume that all agents use the same action space, i.e., $\mathcal{A}_i = \mathcal{A}_j, \forall i, j \in \mathcal{M}$. Therefore, the action space of the system is defined as:

$$\mathcal{A} = \{\{\mathcal{A}_1\}_n, \{\mathcal{A}_2\}_n, \ldots, \{\mathcal{A}_m\}_n, \ldots, \{\mathcal{A}_M\}_n\}, \quad (2.15)$$

where

$$\mathcal{A}_m = \{P_{min}, \frac{P_{max}}{|\mathcal{A}_m| - 1}, \frac{2P_{max}}{|\mathcal{A}_m| - 1}, \ldots, P_{max}\}, \quad (2.16)$$

and $|\mathcal{A}_m|$ represents the cardinality of the action space. For each small cell, we have $|\mathcal{A}_m|$ number of action for each user. Hence, for $\mathcal{M}$ small cells, we have $|\mathcal{A}| = \mathcal{M} \times |\mathcal{A}_m| \times (\mathcal{N}+1)$ number of actions.

- **Reward:** As described in Section 2.3, we aim to maximize the SE of the downlink F-RAN system. As a result, we design the reward function to be the average agent rate.

$$R_t = \sum_{m=1}^{M} R_m. \quad (2.17)$$

The reward function is immediately calculated after an agent takes action.

In Fig 2.2, we use numbers 1-7 to describe the whole process (both the offline training procedure and the online decision making process) of our proposed algorithm. These steps are briefly described below.

1. Agent m observes the state of the environment s_m^t.
2. Agent m performs an action a_m^t based on the current state and according to the ϵ-greedy process.

3. Agent m receives feedback from the environment and moves to the next state. The tuple $(s_m^t, a_m^t, r_m^{t+1}, s_m^{t+1})$ is saved in the experience replay memory.
4. A mini-batch of experiences is sampled from the replay memory. The trained DQN receives (s_m^t, a_m^t) as input and returns $q(s_m^t, a_m^t; \theta_m)$, while the target DQN receives (r_m^{t+1}, s_m^{t+1}) as input and returns $r_m^{t+1} + \gamma \max_{a'} q(s_m^{t+1}, a'; \theta'_m)$.
5. The loss function that defines the difference between the trained DQN and target DQN is executed.
6. The loss function (1.6) is minimized using the stochastic gradient descent optimizer, which results in new weights for the trained DQN.
7. The target DQN's weights are updated in accordance with the trained DQN's new weights once every time step T_{step}.

These steps are repeated at each time epoch. After a certain number of time epochs, a trained model is obtained to allocate power optimally in the real environment.

Our proposed DQN is a fully connected DNN that consists of four layers. The first layer contains the input state vector, which is equal to the number of state elements, i.e., for an F-AP supporting three users, the number of state elements is equal to $(3+3+3 = 9)$. The middle two hidden layers are used to learn the non-linearity of network optimization. Finally, the last layer represents the output layer, which contains the vector of the estimated Q-function. The agent (F-AP) selects from the output vector the action with the maximum transmit power.

2.4.1.1 Training Phase

In our proposed solution, each agent is trained independently based on the experiences of all agents. Each agent benefits from the experiences of all other agents to converge faster. The pseudocode of the proposed DQN-based DDPA scheme for F-RANs is shown in Algorithm 2.1.

Step 1: Define two Q-networks (trained DQN and target DQN), i.e., the number of layers, the activation function, and the hyperparameters. Initialize the two networks with random weights θ_m for the trained DQN and θ'_m for the target DQN. Initialize the experience replay memory

with zero. The training procedure comprises L episodes. The duration of each RL episode is denoted by T and it is defined as $T = N_R\, T_s$. Here, T_s is the duration of each TS and $N_R > 0$ is an appropriately selected integer by the system designer.

Step 2: In each TS, the agent observes the current state and chooses an action. In the first 200 episodes, the agent performs random actions; after that, it follows the ϵ-greedy learning strategy.

Step 3: Execute the selected action, and calculate the new state s_m^{t+1} and the corresponding reward. Store the transition $(s_m^t, a_m^t, r_m^{t+1}, s_m^{t+1})$ in the experience replay memory D. The memory is treated as a first-input-first-output (FIFO) buffer; the new experience replaces the old experience in the buffer.

Step 4: Randomly sample a mini-batch of experiences from the experience replay memory to minimize the occurrence of states of the system between consecutive time-steps, and avoid over-fitting of the DQN model. We use experience replay to enable the networks to learn from a set of uncorrelated tuples (Zhang, Wang & Xu, 2019b).

Step 5: Calculate the loss function, which represents the difference between the target DQN and the trained DQN as given in (1.6). The learning process aims to use the stochastic gradient descent optimizer to reduce the prediction error between the two networks. Consequently, each time step, the weights of the trained DQN are then updated using the backpropagation technique, which is expressed as follows:

$$\frac{\partial \mathcal{L}(\theta_m)}{\partial \theta_m} = \sum_{(s,a,r,s') \in D} (r' - q(s,a;\theta_m)) \bigtriangledown q(s,a;\theta_m). \tag{2.18}$$

Step 6: Once every time step T_{step}, update the weights of the target network. T_{step} is used to make sure the weights are changed slowly in order to improve the stability of the learning process.

Step 7: Repeat steps 2-6 to train the DQN for different episodes.

Step 8: The output of the proposed DDPA algorithm is a learned DQN model.

Algorithm 2.1 Pseudocode of the Proposed DQN-based DDPA Algorithm

Input: Create two DQNs (a trained DQN and a target DQN with weights θ_m and $\theta_m^{'}$, respectively). Then, create an empty experience replay memory D for the F-APs.
Initialize: Initialize the trained DQN with random weights, and set $\theta_m^{'} = \theta_m, \forall m \in \mathcal{M}$.
for $i = 1 : L$ **do**
  for $t = 1 : T$ **do**
    Agent m observes its state s_m^t.
    Agent m chooses an action a_m^t in accordance with the ϵ-greedy policy.
    Agent m performs the chosen action a_m^t, and gets an reward $r_m^{t+1} = R(s_m^t, a_m^t)$.
    Agent m observes a new state s_m^{t+1} in TS $t + 1$.
    Agent m saves its new experience $(s_m^t, a_m^t, r_m^{t+1}, s_m^{t+1})$ into the experience replay memory D.
    Agent m samples a random mini-batch of D_b experiences from D.
    Agent m uses the backpropagation method Eq. 2.18 to update the weights θ_m of the trained DQN.
    Agent m updates $\theta_m^{'}$ with θ_m once every time step T_{step}.
  end for
end for
Output: Learned DQN $q(s, a; \theta_m)$.

2.4.1.2 Testing Phase

In the testing phase, each F-AP uses its trained DQN agent to select the power profile. For an input state vector s, the optimal power profile represents the action item that maximizes the function $q(s, a; \theta_m), \forall m \in \mathcal{M}$. Thus, the power allocation solution is straightforwardly obtained without performing any iterative and time-consuming optimization procedure.

To show the near-optimality of the proposed DQN-based DDPA algorithm, we compare it with the exhaustive search algorithm (ESA), which is the most efficient power allocation algorithm used in the literature. To do this, we must examine all conceivable power combinations for all F-APs and return the optimum combination that yields the highest reward. For example, with two cells, two users, and four discrete power levels, we have $3^4 = 81$ power combinations at each F-AP. This method is feasible only when the number of devices is small. However, in practice,

the number of devices in an IoT network can be very large, which poses significant challenges for executing the ESA and motivating the adoption of our proposed method.

Furthermore, in the testing phase, the output of the proposed Algorithm 2.1 is also compared with traditional power control algorithms, such as WMMSE, FP, and random power (RP), in various network configurations and distortion circumstances (linear and non-linear distortions).

2.4.2 Ensemble DQN

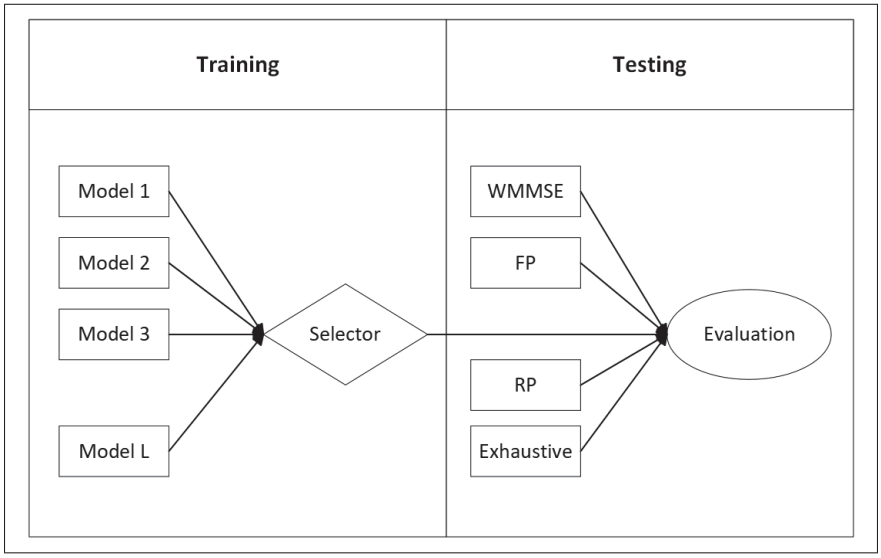


Figure 2.3 eDQN-based DDPA

2.4.2.1 Motivation

EL is a type of ML in which a group of weak learners are trained individually and then merged together to solve an ML problem. In this context and similar to (Liang, Shen, Yu & Wu, 2020), we propose the EL technique, which is shown in Fig. 2.3. The key idea is to choose a model from a group of trained learners that has the best performance for the training samples. The numerical results in Section 2.5 show that DQN-based DDPA performs significantly better than traditional optimization methods. However, the trained DQN may converge to a local optimal solution, especially for large-scale networks. Combining the DQN scheme with the EL method is an

effective way to prevent local optimality. Consequently, to further improve the performance of DDPA for large-scale networks, we propose eDQN by combining the DQN and EL frameworks.

2.4.2.2 Procedure

In eDQN, $\mathcal{L}$ DQN-based DDPA models are trained with different wireless network configurations. For each model, we vary the discount factor γ, which represents one of the critical features of RL algorithms that determine how much the RL agents care about rewards in the distant future relative to the immediate future. The discount factor plays an important role in the stability and convergence of DQN algorithms (François-Lavet, Fonteneau & Ernst, 2016). Various learned models are formed as a result of the variation of γ. This heterogeneity between models enhances the diversity of the $\mathcal{L}$ learners. For simplicity, we assume that the $\mathcal{L}$ DQN models have the same DL network structure and hyperparameters. This assumption simplifies the deployment of eDQN while also reducing the computing complexity of the system. After the training phase, the selector collects all models' power profiles and then selects the model with the highest sum rate $\{\max_p R_{Tot,l}, \forall l \in \mathcal{L}\}$. The pseudocode of the eDQN is shown in Algorithm 2.2.

Algorithm 2.2 Pseudocode of the Proposed eDQN-based DDPA Algorithm

Input: Define the list of the different values of γ.
for $l = 1 : \mathcal{L}$ **do**
  Algorithm 1
end for
Output: Trained DQN with the highest reward, $\{\max_p R_{Tot,l}, \forall l \in \mathcal{L}\}$.

2.4.3 Complexity of the DDPA Algorithm

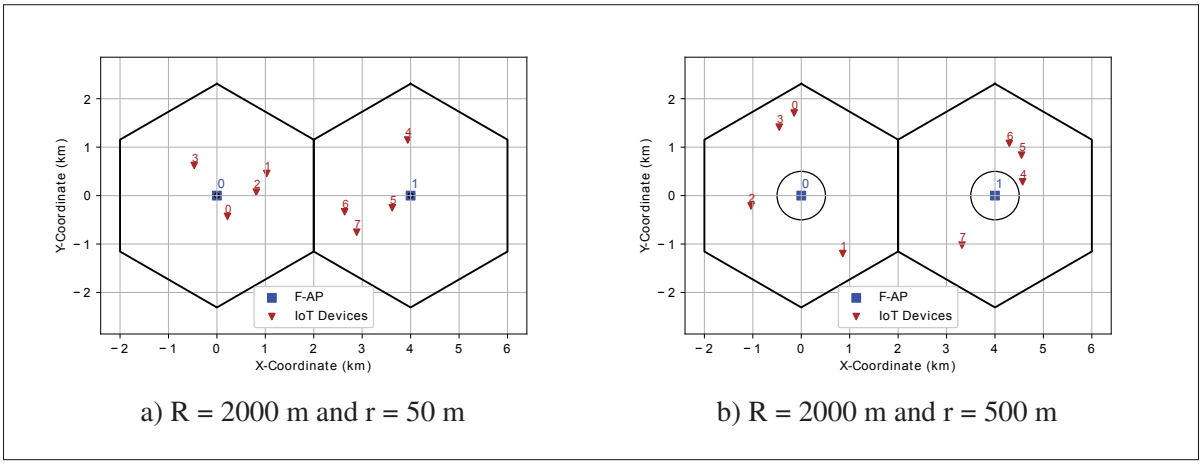
Since the number of episodes is set to L and the number of time steps is set to T, we may deduce from Algorithm 2.1 that the number of iterations is equal to LT. Furthermore, it has been established in the literature (Strehl, Li, Wiewiora, Langford & Littman, 2006) that the Q-learning algorithm's convergence complexity is proportional to the size of the action-state space. As a result, if the state space elements are denoted by $|\mathcal{S}|$ and the action space elements

are represented by $|\mathcal{A}|$, this complexity is given as $O(|\mathcal{S}||\mathcal{A}|)$. Therefore, the computational complexity of Algorithm 2.1 is $O(LT|\mathcal{S}||\mathcal{A}|)$.

2.5 Simulation Results

In this section, the effectiveness of our proposed solutions compared to state-of-the-art transmit power control methods is demonstrated via numerical simulations. More specifically, we compare the DQN-based DDPA scheme with an ESA to validate the near optimality of this algorithm compared to the optimal power control solution in the literature. We also numerically verify the performance of our proposed algorithms (eDQN- and DQN-based DDPA) when the number of devices increases and the distortion in the real environment is non-negligible.

2.5.1 Simulation Step



a) R = 2000 m and r = 50 m

b) R = 2000 m and r = 500 m

Figure 2.4 Network configuration, two F-APs, and four user devices in each small cell

Fig. 2.4 shows two network configurations at different TSs with two F-APs and eight user devices.[2] Section 2.5.7.2 discusses the scalability of our proposed algorithm for large-scale networks. In Fig. 2.4(a), the transmitters (F-APs) are located at the center of each cell, e.g., F-AP$_0$ is positioned at (0,0), and F-AP$_1$ is positioned at (4,0). The cell radius is set to R = 2 Km. We also create a small region of radius r = 0.05 Km with no active devices. The receivers (user

[2] It is noted that Fig. 2.4(b) is added to show the small region boundary in our network configuration.

devices) are placed randomly according to a uniform distribution between the boundary of the small region and that of the cell. In addition, we set P_{max} to 30 dBm and P_{min} to 5 dBm. It is noted that the values of R and P_{max} are selected in accordance with (Nguyen, 2017).

According to the LTE standard (3GPP, 2017), the path loss between the m-th F-AP and the n-th device depends on the distance $d_{m,n}$ and is expressed as $\beta_{m,n} = 120.9 + 37.6 \log_{10}(d_{m,n})$, where $d_{m,n}$ is the distance in kilometers. The log-normal shadowing standard deviation is set to 8 dB. The AWGN PSD σ^2 is set to -174 dBm (Jang & Yang, 2020). According to Jakes fading model (Liang, Kim, Jha, Sivanesan & Li, 2017), $h^k_{m,n}$ is expressed as a first-order complex Gauss-Markov process in which the time interval between adjacent instants is set to $T_s = 20\ ms$ (Meng *et al.*, 2020) and the correlation coefficient between two TSs is set to $\rho = 0.64$ (Ge *et al.*, 2020).

The hyperparameters adopted in our proposed DQN are as follows. Our DQN is composed of one input layer, two hidden layers, and one output layer. Section 2.4.1 describes the state space in depth. The input layer is 4x3. The first hidden layer has 64 neurons and the second has 32. For the output layer, we varied the cardinality of the action space to select the most suitable number of layers. It is worth noting from Fig. 2.5 that the average SE is changed with the cardinality of the action space. We emphasize that this number needs to be chosen carefully to obtain the power level that gives the higher SE (Meng *et al.*, 2020). Fig. 2.5 shows the average SE vs different power levels ($|\mathcal{A}_m| = 5$, $|\mathcal{A}_m| = 10$, $|\mathcal{A}_m| = 16$, $|\mathcal{A}_m| = 20$) when the number of devices per small cell is set to 4 and there is no distortion in the real environment. As the figure shows, the best output is achieved when the cardinality of the action space is set to $|\mathcal{A}_m| = 10$. However, further increasing the dimensions of the output layer does not necessarily improve the SE of the system. Hence, we chose 10 as the cardinality of our action space.

The activation function hyperbolic tangent (tanh) is adopted in each hidden layer. We also tested the rectifier linear unit (ReLU), however, tanh converges to an optimal power allocation faster than ReLU does. Moreover, to balance exploitation and exploration, we adopt the adaptive

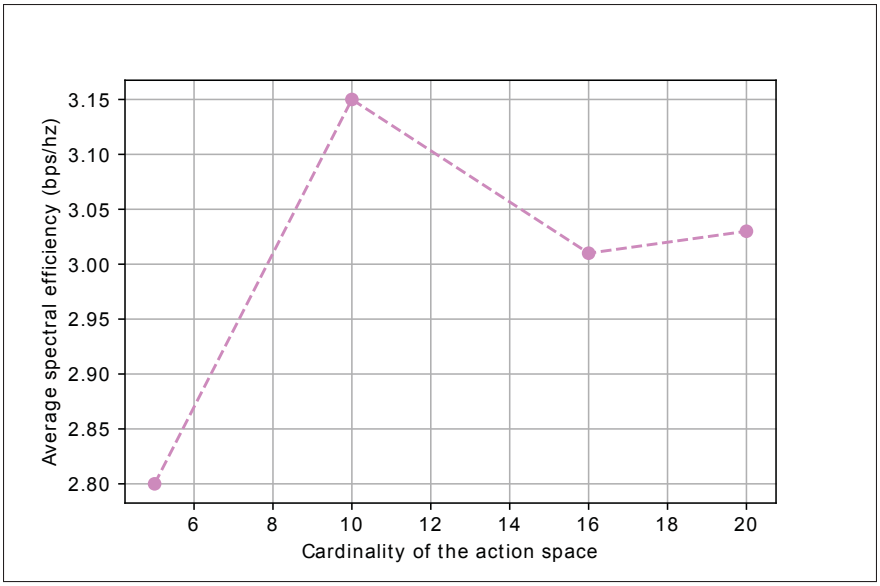


Figure 2.5 Average SE vs the cardinality of the action space ($|\mathcal{A}_m| = 5$, $|\mathcal{A}_m| = 10$, $|\mathcal{A}_m| = 16$, $|\mathcal{A}_m| = 20$)

Table 2.1 DQN's hyper-parameters

Parameter	**Value**
Initial learning rate ($\alpha(0)$)	$5e^{-3}$
Discount factor (γ)	0.8
Episode number (L)	5000
Replay memory buffer size (D)	5000
Mini-batch size (D_b)	32
Loss function	MSE
Optimizer	RMSprop
Target network update period T_{step}	100

ϵ-greedy policy algorithm, in which $\epsilon(t) = \max\{\epsilon_{min}, (1 - \lambda_\epsilon)\epsilon(t-1)\}$, where $\epsilon(0) = 0.7$, $\epsilon_{min} = 10^{-2}$, and $\lambda_\epsilon = 10^{-3}$.

Our DQN's training parameters are shown in Table 2.1. The size of each F-AP's experience memory is set to 5000 with a batch size of 32 so that samples are uniformly selected from the memory. $T_{step} = 100$, which means that once each 100 TSs, the weights of the target DQN θ'_m are updated to match the weights of the trained DQN θ_m. To update θ, we use an RMSprop

optimizer with an adaptive learning rate $\alpha(t)$, where $\alpha(t) = (1 - \lambda)\alpha(t - 1)$, with $\alpha(0) = 5e - 3$ and $\lambda = 10^{-3}$ as the starting learning rate. We found this optimal learning rate value provides better learning performance in our simulation results. The learning rate is generally the most critical hyperparameter that controls the change in weights of the proposed DQN during the training phase (Ahmed & Hossain, 2019).

Our simulation process involves two phases: a training phase and a testing phase. In the training phase, we train each agent similar to what is described in Algorithm 2.1. Training is performed for 5000 episodes to help agent m learn the action-value function $q(s_m, a_m; \theta)$ of each state-action pair. The duration of each RL episode is $T = 10 \times T_s = 200$ ms. At the end of this phase, we save the trained model.

In the testing phase, the performance of the trained model is evaluated for different network deployment scenarios. In this phase, the agent stops exploring the surrounding environment and exploits the trained model to perform power allocation tasks. We consider four benchmark algorithms to evaluate our proposed solution's performance. The first benchmark is the ESA. The second and the third algorithms are WMMSE and FP, both of which are the centralized transmit power control algorithms, defined in (Sun *et al.*, 2018) and (Shen & Yu, 2018), respectively. Finally, the fourth benchmark algorithm used is RP, in which the power profile of each F-AP at each TS is chosen randomly between 0 and P_{max} (Meng *et al.*, 2020) .

2.5.2 DQN-based DDPA Scheme Versus ESA

The ESA is an optimal power allocation algorithm since the F-AP chooses the power profile that provides the largest average achievable rate. In this section, the DQN-based DDPA algorithm is compared to the ESA with the number of devices equal to 2 and 4 and three distortions scenarios–no distortion, linear distortion, and clipping distortion. For the linear distortion scenario, we set $k = 0.1$ based on (Matthaiou *et al.*, 2013), and (Zhang *et al.*, 2015). For the non-linear distortion scenario, we set $v = 7$ dB and $\chi = 2.85$ in accordance with (Lee, 2018).

Table 2.2 Average SE comparison between DQN-based DDPA algorithm and ESA

Number of devices per small cell	Distortion	SE of DDPA (in bps/Hz)	SE of ESA (in bps/Hz)
N = 2	None	5.3	8.53
	Linear	4.16	5.1
	Clipping	2.6	3.14
N = 4	None	3.15	6.7
	Linear	2.9	4.6
	Clipping	2.1	2.8

We perform this comparison with a few devices since the ESA takes a long time to return the optimal power allocation solution. Table 2.2 shows that when the number of devices is equal to 2 (N = 2), we can draw the following conclusions. For the scenario with no distortion, the proposed DQN-based DDPA scheme can achieve up to 62.13% of the optimal SE of the system. For the other two distortion scenarios, it is clear that the achievable SE is much lower for both the DQN-based DDPA algorithm and the ESA. However, the DQN-based DDPA algorithm can achieve up to 81.56% of the optimal solution with linear distortion. Meanwhile, the DQN-based DDPA algorithm achieves a remarkable result for the clipping distortion scenario compared to the ESA–up to 82.8% of the optimal solution. Similarly, the DQN-based DDPA scheme's SE loss is reasonable when there are 4 devices per small cell. For instance, the DQN-based DDPA scheme achieves 47.01%, 63.04%, and 75% of the optimal SE for the no distortion, linear distortion, and clipping distortion scenarios, respectively. Note that the ESA suffers from high computational complexity because it checks all possible combinations of powers to find the transmit power profile that provides the highest sum rate. As a result, the proposed DQN-based DDPA algorithm is significantly less computationally complex than exhaustive search technique, especially for a large-scale network. We therefore conclude that the proposed DQN-based DDPA algorithm strikes a suitable balance between optimal SE and computational complexity.

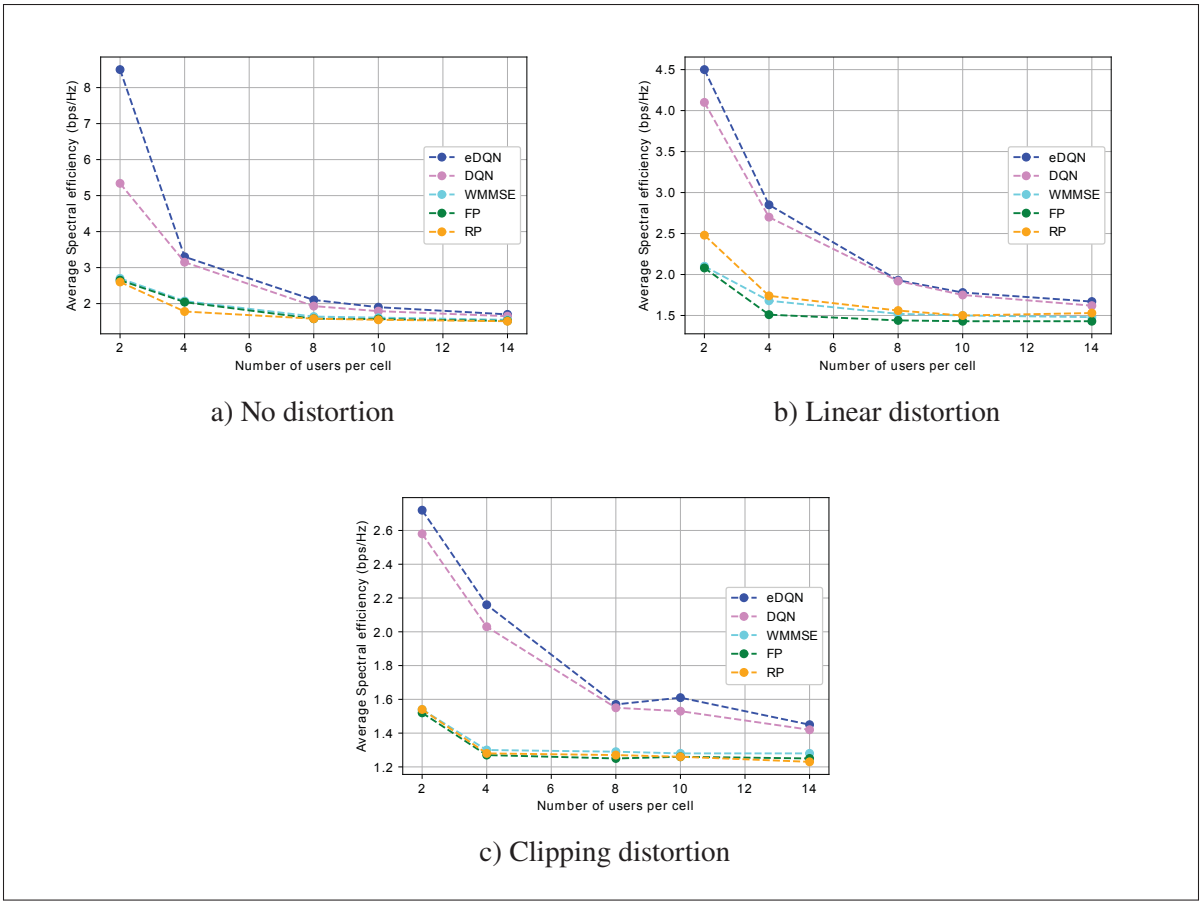


Figure 2.6 Proposed solutions versus traditional power allocation algorithms for the three distortion scenarios

2.5.3 eDQN- and DQN-based DDPA Algorithms Versus Traditional Power Allocation Algorithms

This section shows the impact of increasing the number of devices at the average sum rate of the proposed algorithms compared to state-of-the-art power control algorithms. The average rate is an important parameter to determine our algorithms' performance with regard to the power allocation problem.

Fig. 2.6 shows the average sum rate versus the number of devices per small cell for different power control algorithms. We evaluate the performance of the trained eDQN- and DQN-based DDPA algorithms against the benchmark algorithms by changing the number of devices per

small cell. The DQN-based DDPA algorithm is trained with a value of $\gamma = 0.8$, chosen because it provides the optimum combination of average sum rate for various devices per small cell. Besides, we use $\mathcal{L} = 4$ to build the eDQN-based DDPA algorithm, in which each model is trained with a value of γ in $\{0.3, 0.5, 0.7, 0.9\}$. As discussed in Section 2.4.2, the eDQN-based DDPA algorithm chooses the models that provide the best rewards. This evaluation is done for the three distortion scenarios mentioned earlier.

2.5.3.1 No Distortion

It is evident in Fig. 2.6(a) that the proposed eDQN- and DQN-based DDPA algorithms outperform existing power allocation algorithms in the testing settings, achieving the highest average sum rates. Furthermore, it is evident in Fig. 2.6(a) that increasing the number of devices per small cell gradually decreases the performance of both proposed algorithms and the benchmark schemes. This is because increasing the number of devices also increases the state space and the action space. Nevertheless, the proposed schemes outperform the benchmark schemes in terms of average sum rate for both small and large numbers of devices. For instance, Fig. 2.6(a) shows that the eDQN-based DDPA algorithm achieves a 3.16 bps/Hz, 5.8 bps/Hz, 5.85 bps/Hz, and 5.9 bps/Hz higher average sum rates than the DQN-based DDPA, WMMSE, FP, and RP algorithms when the number of devices is set to N = 2. Similarly, when the number of devices is increased to 4, the eDQN-based DDPA algorithm achieves a 0.15 bps/Hz and 1.23 bps/Hz greater average sum rates than the DQN-based DDPA and WMMSE algorithms, respectively.

In summary, the disparity between the two proposed solutions and the benchmark methods is substantially wider when the number of devices is minimal. However, when the number of devices increases, the disparity narrows.

2.5.3.2 Linear Distortion

Fig. 2.6(b) shows the average sum rate versus the number of devices per small cell when there is non-negligible linear distortion in the real environment. As expected, the distortion reduces the achievable SE. It is observed that both the eDQN- and DQN-based DDPA algorithms outperform all the benchmark schemes for both small and large numbers of devices in the system. For instance, when the number of devices is set to 2, the eDQN-based DDPA algorithm achieves a 0.4 bps/Hz, 2.4 bps/Hz, 2.42 bps/Hz, and 2.02 bps/Hz greater average sum rate than the DQN-based DDPA, WMMSE, FP, and RP algorithms, respectively.

2.5.3.3 Clipping Distortion

Fig. 2.6(c) depicts the average sum rate of the system versus the number of devices per small cell in the presence of clipping distortion. Fig. 2.6(c) shows that both the eDQN- and DQN-based DDPA algorithms are more efficient than the state-of-art power allocation algorithms for various numbers of user devices in the system with non-negligible clipping distortion. When the number of devices is set to 2, the eDQN-based DDPA algorithm achieves a 0.14 bps/Hz, 1.18 bps/Hz, 1.2 bps/Hz, and 1.18 bps/Hz greater average sum rate than the DQN-based DDPA, WMMSE, FP, and RP algorithms, respectively.

2.5.4 Comparisons of Power Control Schemes over TSs

This section provides the results of an example testing episode in which the number of devices per small cell is set to N = 4. In Fig. 2.7, we illustrate the average sum rate of different power allocation schemes with respect to the number of TSs. We can see that the eDQN-based DDPA algorithm achieves the highest SE performance. The reason is that it uses both DRL and EL. DRL-based frameworks are able to perform the best action (e.g., select the most suitable transmit power profile) that yield the best outcomes (e.g., average sum rate) according to the state at the current TS. On the other hand, the EL selects the DQN model with the best performance for a set of trained models. It should be noted that all the results are averaged by taking a window of

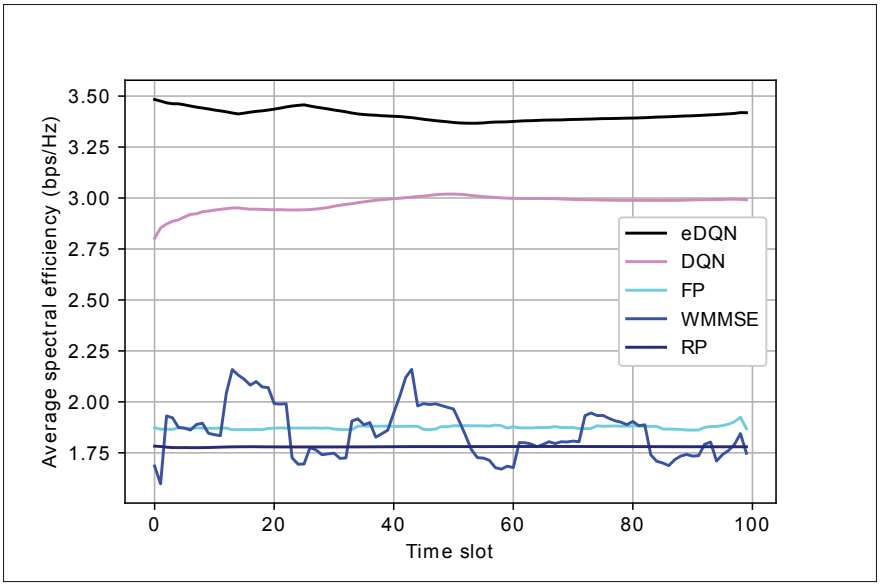


Figure 2.7 Comparisons of all five power allocation schemes over 100 TSs

100 TSs. The different power control algorithms' performance is not stable, which is explained by the fact that the small-scale fading component changes at each TS. Fig. 2.7 depicts that the eDQN-based DDPA algorithm achieves around 15% improvement in the average sum rate compared to the DQN-based DDPA scheme for a system with two F-APs and four devices. Fig. 2.7 interestingly shows that both of our proposed solutions have improved the stability compared to optimization-based approaches (i.e., WMMSE).

2.5.5 Performance of the Proposed eDQN-based DDPA Algorithm Versus Number of Trained DQN Models

The proposed eDQN-based DDPA algorithm aims to return the model that provides the highest sum rate in the training phase. To assess the quality of the proposed eDQN-based DDPA scheme, we plot in Fig. 2.8 the average sum rate versus the number of trained models when the number of devices is set to four and there is no distortion in the real environment. The average sum rate was taken from the training phase. We can see that when the number of models increases, the average sum rate of the system also increases. The eDQN-based DDPA algorithm performs poorly when

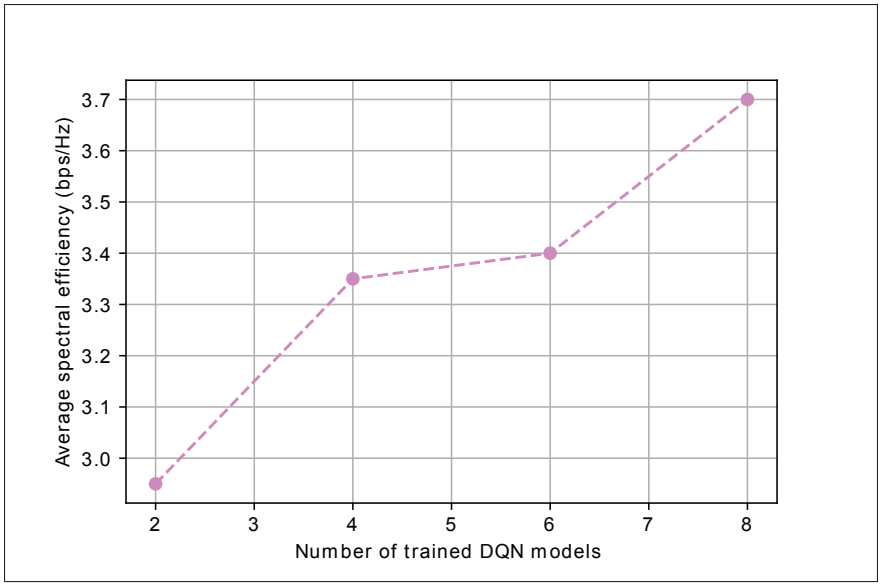
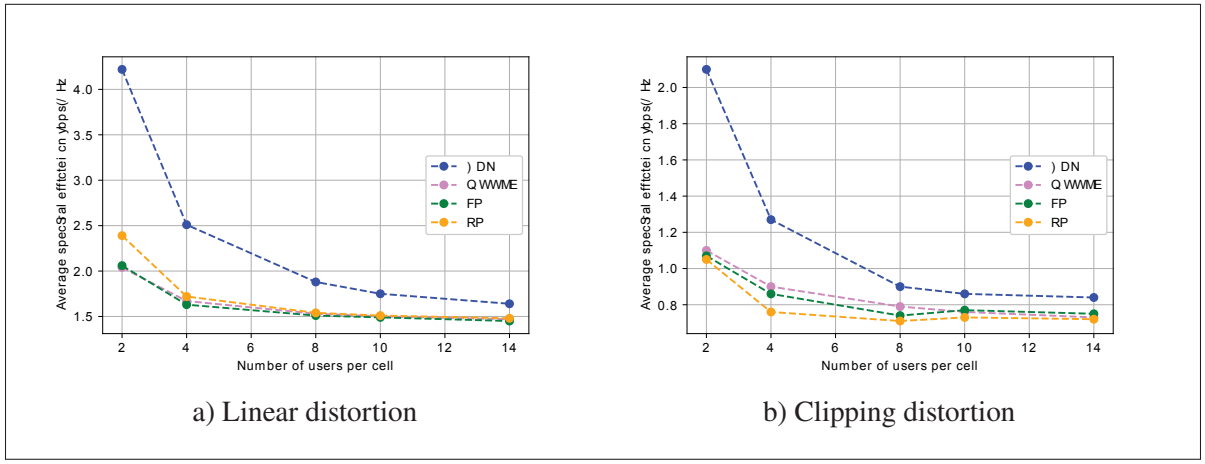


Figure 2.8 Average SE vs number of trained DQN models

$\mathcal{L} = 2$, moderately when $\mathcal{L} = 4$ and $\mathcal{L} = 6$, and the best when $\mathcal{L} = 8$. Fig. 2.8 depicts that the proposed DQN-based DDPA algorithm achieves a 25.42% improvement in average sum rate when $\mathcal{L} = 8$ than when $\mathcal{L} = 2$. In summary, the eDQN-based DDPA algorithm is an efficient power management algorithm when the latency is not the system's key performance indicator.

2.5.6 Performance Comparison for Non-uniform Distortions



a) Linear distortion b) Clipping distortion

Figure 2.9 DQN-based DDPA versus traditional power allocation algorithms for non-uniform distortions scenarios

HWIs in physical transceivers are known to have deleterious effects on communications systems (Matthaiou *et al.*, 2013). In the real-word environment, transceivers suffer from impairments that cause a mismatch between the symbols that are generated and those that are emitted. These HWIs are randomly generated and have a negative effect on the received signal at the user devices. Practically speaking, transceivers can have a variety of different HWIs. To be more precise, in our work, we define a range of values from which the distortion parameter is selected randomly for each transceiver. For linear distortion, we consider $\mathrm{k} \in \{0.05, 0.1, 0.15, 0.2\}$. The transceiver selects a random value of k from the range for each TS, and the selected value changes from one TS to the next. These values are chosen in accordance with (Matthaiou *et al.*, 2013), (Zhang *et al.*, 2015). Fig. 2.9(a) shows the average sum rate versus the number of users per small cell in the presence of non-uniform linear distortion. As expected, the proposed DQN-based DDPA algorithm outperforms the benchmark schemes with various numbers of user devices. For example, when the number of devices is set to 4, the DQN-based DDPA algorithm achieves a 0.84 bps/Hz, 0.88 bps/Hz, and 0.79 bps/Hz greater average sum rate than the WMMSE, FP, and RP algorithms, respectively.

Similarly, for clipping distortion, we create a list of CR values. Each transceiver's CR value is randomly selected from $v \in \{5, 6, 7, 8\}$ and can vary from one simulation instance to another. Such values are consistent with the existing literature (Lee, 2021). Fig. 2.9(b) depicts the average sum rate of the system versus the number of users per small cell when there is non-negligible non-uniform clipping distortion. It is observed that the DQN-based DDPA algorithm outperforms the state-of-the-art schemes with different numbers of devices. For instance, when the number of users is set to 4, the DQN-based DDPA algorithm achieves a 0.37 bps/Hz, 0.41 bps/Hz, and 0.51 bps/Hz higher average sum rate than the WMMSE, FP, and RP algorithms, respectively.

2.5.7 Extensive Simulations

2.5.7.1 Robustness to Different Network Configurations

In Table 2.3, we vary the value of both the radius of the small cell, R, and the radius of the small region, r, to validate the proposed algorithm's performance against the findings in the literature. The first three scenarios were taken from (Sun *et al.*, 2018), (Nasir & Guo, 2019), and (Meng *et al.*, 2020). The last scenario is the network configuration that we adopted in our work (Section 2.5.1). It is evident from Table 2.3 that our proposed solution is capable of finding an optimal transmit power for different radius configurations. More importantly, our proposed solution still outperforms the benchmark schemes in terms of SE and also performs well as discussed in the previous sections. Consequently, we emphasize that our proposed solution is consistent with the setups given in the literature and our proposed framework is valid for other radius values as well.

Table 2.3 Average SE for variant number of R and r.
M=2, N= 4, and without distortion

		Average SE (bps/Hz)			
R (m)	r (m)	DQN-Based DDPA	WMMSE	FP	RP
100	50	4.25	2.18	2.16	1.79
500	10	3.82	2.09	2.07	1.78
1000	10	3.22	2.08	2.06	1.77
2000	50	3.15	2.08	2.04	1.78

2.5.7.2 Scalability for Large-Scale Networks

To investigate the scalability of our proposed solution, we simulate the proposed DQN-based DDPA algorithm and benchmark schemes for large-scale networks by increasing both the number of F-APs and the number of users per small cell. Table 2.4 shows the average SE of the proposed solution and benchmark schemes for the different numbers of F-APs and users considered. As expected, the proposed DQN-based DDPA algorithm still outperforms the benchmark schemes

in terms of average SE. For example, when the number of F-APs is set to 25 and the number of users per cell is set to 50, the DQN-based DDPA algorithm achieves a 0.02 bps/Hz, 0.03 bps/Hz, and 0.05 bps/Hz higher average SE than the WWMSE, FP, and RP algorithms, respectively. However, the proposed algorithm's average SE drops as the number of users and F-APs become greater. This is because increasing the number of users and F-APs increases the action and state spaces of the proposed algorithm, which then requires more exploration to evade sub-optimal solutions. For instance, the DQN-based DDPA algorithm achieves a 1.5 bps/Hz average SE when the number of F-APs is set to 25 and the number of devices per cell is set to 50, and a 1.48 bps/Hz average SE when we double those two quantities.

It is also worth noting that the benchmark schemes considered (e.g., WMMSE and FP) are centralized, and as a result, their iterative optimization frameworks are significantly more computationally complex and require much more signaling overhead for large-scale networks. In contrast, the DQN-based DDPA algorithm follows a distributed training and execution approach, with the training performed offline. Hence, the proposed DQN-based DDPA algorithm is much less complex to implement than the existing centralized transmit power allocation schemes. Therefore, in the context of scalability, the proposed DQN-based DDPA algorithm has clear merit over the existing transmit power allocation methods considered.

Table 2.4 Average SE for large-scale networks.
R = 2000 m, r = 50 m

		Average SE (bps/Hz)			
F-APs	Users	DQN-Based DDPA	WMMSE	FP	RP
25	50	1.5	1.48	1.47	1.45
50	100	1.48	1.46	1.46	1.44
100	150	1.46	1.44	1.45	1.42

2.5.8 Discussion

From the simulation results, it is evident that DRL is an effective approach to maximize system SE that performs better than traditional optimization approaches, especially in the presence of both interference and HWIs. In particular, both the eDQN- and DQN-based DDPA algorithms

are incredibly efficient when the system is impaired by both co-channel interference and HWIs distortion. We emphasize that the eDQN-based DDPA algorithm must train $\mathcal{L}$ models for each agent, which increases the learning time. Hence, the proposed eDQN-based algorithm is particularly suitable for transmit power allocation for non-time-critical F-RAN systems. In contrast, each F-AP in the DQN-based DDPA algorithm autonomously learns the network dynamics online from locally gathered information. Hence, the DQN-based scheme is suitable for time-critical F-RAN systems due to its low training and execution complexity.

We also emphasize that our proposed algorithm does not require any prior knowledge of hardware errors at different transmitters. More importantly, the HWI-resilient optimal power allocation can be performed by observing the values of SINR and CSI. Meanwhile, conventional solutions require exact mathematical models of HWI, and they do not learn anything from SINR. Therefore, our proposed algorithm also has advantages over the conventional optimization methods for non-uniform HWIs.

2.6 Conclusion

We investigated the transmit power allocation problem to maximize the network SE of downlink RSMA-enabled F-RANs in the presence of realistic impairments, namely, co-channel interference and HWI-induced signal distortion. Due to the existence of both interference and HWI-induced distortions, it is challenging to obtain optimal power control using traditional optimization approaches. To overcome this challenge, we proposed a DRL-based power control solution. We first developed a distributed DQN-based DDPA algorithm that takes the CSI and the SINR of the private and common streams into account to perform optimal power allocation in the real environment. Then, we proposed an eDQN-based DDPA algorithm by combining the DQN and EL frameworks. The simulation results demonstrated that the eDQN-based DDPA algorithm outperforms the DQN-based DDPA algorithm at maximizing the network's SE. However, the DQN scheme requires less training time and is less complex than the eDQN scheme. The simulation results confirm that our proposed DQN-based DDPA achieves a near-optimal sum rate compared to ESA with significantly less computational complexity. The simulation results

also demonstrate that the proposed algorithm is more scalable and resilient to both interference and HWIs than state-of-the-art transmit power allocation schemes considered.

CHAPTER 3

RSMA-ENABLED INTERFERENCE MANAGEMENT FOR INDUSTRIAL INTERNET OF THINGS NETWORKS WITH FINITE BLOCKLENGTH CODING AND HARDWARE IMPAIRMENTS

Nahed Belhadj Mohamed[1] , Md. Zoheb Hassan[2], Georges Kaddoum[1]

[1] Electrical Engineering Department, École de technologie supérieure (ETS), 1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3
[2] Electrical and Computer Engineering Department, Université Laval, 2325 Rue de l'Université, Québec, QC G1V 0A6

Paper published in *IEEE Transactions on Machine Learning in Communications and Networking*, vol. 2, pp. 1319-1340, September 2024.

3.1 Abstract

The increasing proliferation of IIoT devices requires the development of efficient radio resource allocation techniques to optimize spectrum utilization. In densely populated IIoT networks, the interference that results from simultaneously scheduling multiple IIoT devices over the same RRBs severely degrades a network's achievable capacity. This paper investigates an interference management problem for IIoT networks that considers both FBL-coded transmission and signal distortions induced by HWIs arising from practical, low-complexity radio-frequency front ends. We use the RSMA scheme to effectively schedule multiple IIoT devices in a cluster over the same RRB(s). To enhance the system's achievable capacity, a joint clustering and transmit power allocation problem is formulated. To tackle the optimization problem's inherent computational intractability due to its non-convex structure, a two-step DCPM framework is proposed. First, the DCPM framework obtains a set of clustered devices for each access point by employing a greedy clustering algorithm while maximizing the clustered devices' SINR. Then, the DCPM framework employs a MADRL framework to optimize transmit power allocation among the clustered devices. The proposed DRL algorithm learns a suitable transmit power allocation policy that does not require precise information about instantaneous signal distortions. Our simulation results demonstrate that our proposed DCPM framework adapts seamlessly to varying

channel conditions and outperforms several benchmark schemes with and without HWI-induced signal distortions.

3.2 Introduction

The IIoT has emerged as a transformative force that is propelling industries into a new era of automation, data-driven decision making, and operational efficiency. IIoT networks incorporating devices, robots, and healthcare systems that require low-latency (around 1 ms) communication are anticipated to grow in scale in future-generation wireless networks (Mahmood *et al.*, 2022). Significant challenges must be overcome to meet such stringent QoS requirements, particularly over time-varying fading channels. FBL codes are used to reduce the latency of over-the-air transmission in applications, such as industrial automation where control packets of around 100 bits are transmitted to provide instructions to robots and autonomous vehicles (Ranjha *et al.*, 2022). This is in stark contrast to the conventional practice of transmitting arbitrary long codes for negligible error. Moreover, next-generation IIoT networks for 6G-based Industry 5.0 are expected to support much higher data rates, traffic volumes, and connection density levels (devices/km^2) than IIoT networks designed for Industry 4.0 (Chi, Wu, Huang, Tsang & Radwan, 2023). In such large-scale IIoT networks, the fact that spectrum resources are limited means that RRBs must be shared among multiple IIoT devices, which leads to co-channel interference. Additionally, IIoT devices often use low-complexity RF front ends that are power-efficient and cost-effective but usually result in HWI-induced signal distortions. Co-channel interference and HWI-induced signal distortions make it difficult to achieve the QoS levels that IIoT networks require over time-varying fading channels. Thus, effective radio resource allocation that takes into consideration FBL-coded transmission, co-channel interference, and HWI-induced signal distortions is crucial for improving spectrum utilization in IIoT networks.

Device clustering and transmit power allocation are critical for managing interference and improving spectrum utilization in wireless networks (Hassan, Hossain, Cheng & Leung, 2021c). More specifically, clustering IIoT devices can improve the utilization of the limited RRBs that are available, and optimizing transmit power allocation in MA systems can enhance system capacity

over the inferring channels. Accordingly, this work aims to develop a device clustering and transmit power allocation optimization framework to improve resource utilization and manage interference in IIoT networks.

3.2.1 Motivation

Despite significant advancements in resource optimization having been made, the current literature fails to address some limitations, particularly in terms of clustering IIoT devices and allocating transmit power among device clusters[1] while considering FBL, co-channel interference, and HWI-induced distortions. Most radio resource allocation studies overlook the detrimental impact HWIs have on network resource optimization. Note that, HWIs caused by non-linearity in low-cost and low-power IIoT devices can significantly reduce the average sum rate achieved. Practical radio equipment, such as power amplifiers and filters, often have inherent limitations that cannot be ignored (Mohammadi & Marojevic, 2021). Additionally, it is impractical to assume Shannon channel capacity in the context of IIoT networks since the transmitted packets' blocklengths are short due to the stringent delay requirements. Consequently, FBL information theory must be employed in place of the classical Shannon rate formula to accurately estimate the rate performance of finite packet transmission. Notably, the optimization of transmit power and device clustering in a downlink FBL-coded IIoT network is a computationally intractable problem when HWI-induced distortions are present. Off-the-shelf optimization tools usually fail to provide scalable solutions for these types of optimization problems. Furthermore, these approaches require accurate HWI models for devices, which are not always available in practice. These challenges motivate our work to develop an optimized device clustering and power allocation framework in order to improve the system capacity of FBL-coded IIoT networks with non-negligible co-channel interference and HWI-induced distortions.

[1] In this work, an IIoT device cluster implies the set of IIoT devices that are scheduled over the same RRB. We consider a dense IIoT network and assume that IIoT devices are clustered to ensure efficient spectrum use.

3.2.2 Contributions and Paper Organization

This work presents a comprehensive resource optimization framework to enhance the system capacity of RSMA-enabled IIoT networks in the presence of FBL-coded data transmission and HWI-induced signal distortions. This work's specific contributions are summarized below.

1. This work investigates a downlink data transmission scheme for an FBL-coded multi-cell IIoT network. Each AP in the envisioned network utilizes dedicated orthogonal RRBs for data transmission to overcome inter-cell interference. Each AP's devices are grouped into non-overlapping clusters such they are concurrently scheduled over the same RRB. Using the RSMA strategy enables the IIoT devices in each cluster to receive data over the same RRB, which makes the proposed framework particularly suitable for resource-constrained IIoT networks. Note that the network's system capacity is impaired by both intra-cell co-channel interference and HWI-induced signal distortions arising from the practical low-complexity RF front ends. A sum rate maximization problem is formulated to optimize the clusters of IIoT devices and select a suitable transmit power allocation for the interfering IIoT links in order to mitigate the HWI-induced signal distortions. To the best of the authors' knowledge, this is the first work to develop a device clustering and transmit power allocation solution for IIoT networks that explicitly considers the detrimental effects of FBL, co-channel interference, and HWI-induced signal distortions.
2. The joint device clustering and transmit power allocation problem is proved to be NP-hard and computationally intractable. A DCPM framework is proposed to efficiently solve the joint problem. The framework decomposes the joint problem into multiple device clustering and transmit power allocation sub-problems (one for each AP) and solves them sequentially utilizing only local CSI.
3. We propose a greedy clustering algorithm to solve the device clustering sub-problem for a given transmit power allocation. In the algorithm, each AP's devices are grouped into multiple clusters to enhance their SINRs in an RSMA setup. We also devise a MADRL-empowered power allocation algorithm to solve the power allocation sub-problem. The proposed algorithm efficiently distributes the AP's transmit power among the IIoT devices

in each cluster. Our proposed power allocation algorithm is innovative because it learns a suitable transmit power allocation policy without requiring precise and instantaneous knowledge of HWI. This learned policy facilitates the cooperative allocation of the most suitable transmit power across all APs to support dynamic adaptation to changing channel conditions and network variables.

4. Extensive simulations are conducted to verify the influence of several system parameters, including HWI, FBL, block error probability, total available power at the AP, and number of user devices, on the proposed DCPM framework's system capacity. The provided simulation results confirm the framework is able to adapt and effectively learn an appropriate power allocation strategy for a range of IIoT networking scenarios. The simulation results also validate that the DCPM framework consistently outperforms state-of-the-art transmit power allocation algorithms and interference management schemes, which attests that our proposed DCPM is able to enhance the IIoT network's resilience to co-channel interference.

The rest of the paper is organized as follows. Section 3.3 provides an overview of the system and the problem formulation. The sub-problem solutions are detailed in Sections 3.4 and 3.5. Sections 3.6 and 3.7 present the overall algorithm and the simulation results, respectively. Finally, Section 3.8 contains the conclusion. A list of symbols and notations used in this paper is given in Table 3.1.

3.3 System Model and Problem Formulation

3.3.1 System Overview

We consider a multi-device downlink IIoT network of M cells each equipped with one AP and K devices randomly distributed within their coverage areas (Fig. 3.1). For simplicity, let $\mathcal{M} = \{1, 2, \cdots, M\}$ be the set of all APs, $\mathcal{K} = \{1, 2, \cdots, K\}$ be the set of IIoT devices in each cell, and $\mathcal{N} = \{1, 2, \cdots, N\}$ be the set of RRBs in each cell. We ignore the inter-AP interference by assigning the neighboring APs orthogonal RRBs and reusing the same RRBs at the far APs. This consideration can be explained by assuming that a total of $S > N$ RRBs are available in the

Table 3.1 Table of variables and descriptions

Variable	Description	Variable	Description
M	Number of APs	θ_c	SIC error coefficient
K	Number of devices per AP	n_b	Blocklength
N	Number of clusters	ϵ	Block error probability
K_D	Number of devices per cluster	$R_{Cn}^{(m)}$	Achievable sum-rate of the n-th device-cluster
$h_{k,n}^{(m)}$	Small-scale block Rayleigh fading	$R_{\text{total}}^{(m)}$	Total achievable sum-rate of the m-th AP
$\beta_{k,n}^{(m)}$	Large-scale fading	B	Total system bandwidth
$g_{k,n}^{(m)}$	Channel gain	$\mathcal{U}_{MU}$	Set of unassigned IIoT devices
ρ	Correlation coefficient	$\mathcal{U}_{MR}$	Set of RRBs that can accommodate at least one IIoT device
f_d	Doppler Frequency	D_n	Number of IIoT devices assigned to the n-th RRB
T_s	Time interval	R	Cell radius
$Pr_{\text{Los}}, Pr_{\text{NLos}}$	LoS and NLoS probabilities	r_s	Small region radius
$PL_{\text{Los}}, PL_{\text{NLos}}$	LoS and NLoS path losses	P_t, $P_{total,n}$	Total power, total power provided by the AP to the n-th cluster
$\eta_{\text{Los}}, \eta_{\text{NLos}}$	LoS and NLoS additional attenuation factors	$P_{\max}$	Maximum power
$d_{k,AP}$	3D distance between the AP and the k-th IIoT device	$P_{\min}$	Minimum power
f_c	Carrier frequency	s_t, a_t	State and action at t
H	Height of the AP	r_{t+1}	Reward at $t+1$
h_k	Height of the k-th IIoT device	R_t	The discounted accumulated reward
d_{clutter}	Typical clutter size	$Q(\dots)$	Q-function
h_c	Effective clutter height	$s_{n,t}^{(m)}$	State space
r	Clutter density	$\mathcal{A}_u^{(n)}$	Action space
s_c	Common stream	α	Learning rate
$s_{p,k}$	Private stream of the k-th IIoT device	γ	Discount factor
$P_{c,n}^{(m)}, P_{p,k,n}^{(m)}$	Powers assigned for the common and private messages	Z	Episode number
x_n	The signal transmitted by the m-th AP to the n-th device cluster	D	Replay memory buffer size
$C_n^{(m)}$	The set of devices in the n-th device cluster	D_b	Mini-batch size
y_k	The received signal at the k-th IIoT device	$\pi(\cdots)$	Policy of the agent
n_a	AWGN	θ, θ^-	Weights of the trained and target network
σ^2	The AWGN power spectral density	$\mathcal{L}oss$	Loss function
σ_t, σ_r	Level of HWI at the transmitter and receiver	T_{step}	Time step
$\gamma_{c,k}^{(n)}, \gamma_{p,k}^{(n)}$	SINRs for the common and private streams for the k-th device	$\mathcal{Z}^{(m)}$	The set of learning agents associated with the m-th AP
$R_{c,k}^{(n)}, R_{p,k}^{(n)}$	Achievable rates for the common and private streams for the k-th device	T	Duration of each RL episode

system, and the RRBs are divided into a total of L non-overlapping RRB groups (RRBGs) (each containing N RRBs), which are denoted by $\mathcal{G}_1 = \{1, 2, \cdots, N\}$, $\mathcal{G}_2 = \{N+1, N+2, \cdots, 2N\}$, $\mathcal{G}_3 = \{2N+1, 2N+2, \cdots, 3N\}$, $\cdots$, $\mathcal{G}_L = \{S-N+1, S-N+2, \cdots, S\}$. Each AP is assigned an RRBG in such a way that its L nearest APs are assigned orthogonal RRBGs and its $(L+1)$-th nearest AP reuses its RRBG. Also, the AP's down-tilt angles can be optimized as very little power (interference) radiates from the neighboring cell. As a result, inter-cell interference from distant APs can be ignored due to path loss and shadowing without any performance degradation. In this work, we assume that both RRBG assignments and the down-tilt angles

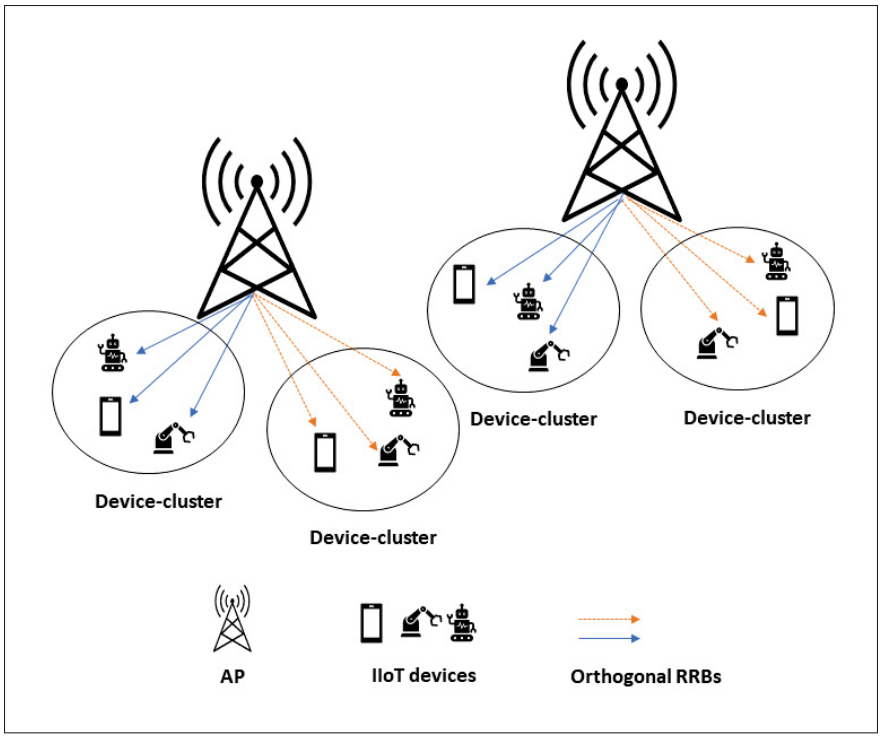


Figure 3.1 An RSMA-enabled downlink IIoT network with two APs, and multiple IIoT devices in each cell

have been pre-optimized to keep inter-AP interference at a minimum[2]. However, in the IIoT network considered, $K >> N$ (corresponding to a dense network) leads to there being fewer resources available than are needed to accommodate the vast number of devices. We address this challenge by concurrently scheduling multiple IIoT devices over the same RRB. We mitigate the resultant interference and improve system capacity by incorporating a one layer RSMA strategy in each cluster so that multiple devices can be scheduled over the same RRB simultaneously. A two-step procedure is then followed to optimize the RSMA framework's ability to manage interference. In the first step, we divide the K IIoT devices that are in each cell into a total of N clusters, and each cluster is assigned to one orthogonal RRB. In the second step, we execute the DRL algorithm to optimize power allocation and maximize each cluster's capacity.

[2] The mitigation of both inter-cell and intra-cell interference in an IIoT network is left for future work.

For analytical tractability, the following assumptions are made. **A1:** Each AP is aware of the CSI of the devices in its cell. However, APs do not have any information about the presence of instantaneous HWI-induced signal distortions at the devices. **A2:** The clusters are non-overlapping to ensure that inter-cluster interference is negligible. **A3:** Each device can be associated with a maximum of one AP and one cluster. **A4:** The time horizon is decomposed into multiple non-overlapping TSs. We consider all communication links to exhibit quasi-static channel fading, i.e., the CSI of the links remains constant in each TS and can change independently from one TS to the next.

3.3.2 Channel Model

We model the downlink channel gain from the m-th AP to the k-th IIoT device in the n-th cluster the same way as it is modeled in (Nasir & Guo, 2019), where each channel is subjected to large-scale fading $\beta_{k,n}^{(m)}$ and small-scale block Rayleigh fading $h_{k,n}^{(m)}$. In the t-th TS, the channel gain is expressed as:

$$g_{k,n}^{(m)}(t) = |h_{k,n}^{(m)}(t)|^2 \beta_{k,n}^{(m)}. \tag{3.1}$$

According to the Jakes fading model (Liang *et al.*, 2017), $h_{k,n}^{(m)}(t)$ can be expressed as a first-order complex Gauss-Markov process:

$$h_{k,n}^{(m)}(t) = \rho h_{k,n}^{(m)}(t-1) + \sigma^{'}, \tag{3.2}$$

where $\rho = J_0(2\pi f_d T_s)$ is the correlation coefficient between two TSs, with $J_0(.)$ being the zeroth-order Bessel function, f_d being the Doppler frequency, and T_s being the duration of each TS, and $\sigma^{'}$ is a random variable with a distribution $\sigma^{'} \sim \mathcal{CN}(0, 1-\rho^2)$. The large-scale fading includes both path loss and shadowing. We consider the path loss model proposed by 3GPP (3GPP, 2020) for a scenario involving an indoor factory with space clutter and a high BS height (InF-SH) (Maldonado *et al.*, 2021). The average path loss (in dB) between the AP and the k-th

IIoT device is given by:

$$PL = Pr_{LoS}PL_{LoS} + Pr_{NLoS}PL_{NLoS}, \tag{3.3}$$

where Pr_{LoS} and Pr_{NLoS} are the probability of having a line-of-sight (LoS) and a non-line-of-sight (NLoS) link, respectively, between the AP and the k-th IIoT device. PL_{LoS} and PL_{NLoS} represent the path loss (in dB) between the AP and the k-th IIoT device for the LoS and NLoS links, respectively. They are expressed as (3GPP, 2020, Table 7.4.1-1):

$$PL_{LoS} = 31.84 + 21.5\ log_{10}(d_{k,AP}) + 19\ log_{10}(f_c) + \eta_{LoS} \tag{3.4}$$

and

$$\begin{aligned} PL_{NLoS} = &\max(32.4 + 23\ log_{10}(d_{k,AP}) + 20\ log_{10}(f_c) + \\ &\eta_{NLoS}, PL_{LoS}), \end{aligned} \tag{3.5}$$

where $d_{k,AP} = \sqrt{(dx_k - X)^2 + (dy_k - Y)^2 + (h_k - H)^2}$ represents the 3D distance between the AP and the k-th IIoT device, with (dx_k, dy_k) being the position of the k-th IIoT device, (X, Y) being the position of the AP, H denoting the height of the AP, and h_k denoting the height of the k-th IIoT device, f_c denotes the carrier frequency and η_{LoS} and η_{NLoS} represent additional attenuation factors due to the LoS and NLoS connections, respectively.

The probability values can be obtained from (3GPP, 2020, Table 7.4.2-1):

$$Pr_{LoS} = e^{(\frac{-d}{k_{subsce}})} \tag{3.6}$$

and

$$Pr_{NLoS} = 1 - Pr_{LoS}, \tag{3.7}$$

where

$$k_{subsce} = -\frac{d_{clutter}}{ln(1-r)}\frac{H-h_k}{h_c-h_k}, \tag{3.8}$$

with $d_{clutter}$ being the typical clutter size, h_c representing the effective clutter height, and r representing the clutter density.

3.3.3 RSMA Strategy

We adopt the well-known one-layer RSMA[3] strategy that splits the received data of the devices in a cluster into common and private parts. All the receivers' common parts are combined into a common message, which is encoded in a common stream s_c, while the private parts are encoded in a set of private streams $\{s_{p,k}\}$, one per device (Mishra *et al.*, 2022). The s_c and $\{s_{p,k}\}$ streams are then linearly precoded and transmitted together over the same RRB. The signal transmitted by the m-th AP to the n-th device cluster is expressed as:

$$x_n = \sqrt{P_{c,n}^{(m)}}\, s_c + \sum_{n \in C_n^{(m)}} \sqrt{P_{p,k,n}^{(m)}}\, s_{p,k}, \tag{3.9}$$

where $C_n^{(m)}$ is the set of devices in the n-th device cluster and $|C_n^{(m)}|$ is the cardinality of the $C_n^{(m)}$ set, $P_{c,n}^{(m)}$ and $\{P_{p,k,n}^{(m)}\}$ denote the transmission power assigned to the common and private messages, respectively, and $P_n = [P_{c,n}^{(m)}, P_{p,1,n}^{(m)}, .., P_{p,|C_n^{(m)}|,n}^{(m)}]$ denotes the power profile. The signal received at the k-th IIoT device in the t-th TS is given by:

$$y_k(t) = \sqrt{g_{k,n}^{(m)}(t)}(\, x_n + z) + n_a, \tag{3.10}$$

where z represents the signal distortion caused by HWI and n_a is AWGN with variance σ^2. In this article, we consider the linear distortion introduced by the imperfect power amplifier.

[3] It is noteworthy that our DCPM framework is generic and can be applied to IIoT networks utilizing the NOMA scheme. While NOMA-based DCPM has the potential to achieve higher system capacity compared to RSMA-based DCPM, its implementation is challenging due to the need for multiple SIC operations at user devices.

According to (Chen, Yang, Zhang & Alouini, 2021), $z \sim \mathcal{CN}\left(0, (\sigma_t^2 + \sigma_r^2)P_{total,n}\right)$, where $P_{total,n}$ is the total power provided by the AP to the n-th cluster, and σ_t and σ_r denote the level of HWI at the transmitter and receiver, respectively.

In the RSMA strategy considered, each receiver first decodes the common messages by treating the interference from all the private streams as noise. Each receiver then removes the interference from the decoded common stream by applying SIC and decodes its private message while treating the other devices' private messages as noise. Thus, the SINRs of the common and private streams for the k-th IIoT device, $\forall k \in C_n^{(m)}$, are expressed as:

$$\gamma_{c,k}^{(n)} = \frac{P_{c,n}^{(m)}\, g_{k,n}^{(m)}(t)}{\sum_{k=1}^{|C_n^{(m)}|} P_{p,k,n}^{(m)}\, g_{k,n}^{(m)}(t) + I_{HWI} + \sigma^2} \tag{3.11}$$

and

$$\gamma_{p,k}^{(n)} = \frac{P_{p,k,n}^{(m)}\, g_{k,n}^{(m)}(t)}{\sum_{\substack{k'=1 \\ k' \neq k}}^{|C_n^{(m)}|} P_{p,k',n}^{(m)}\, g_{k,n}^{(m)}(t) + I_{SIC} + I_{HWI} + \sigma^2}, \tag{3.12}$$

respectively, where $I_{HWI} = g_{k,n}^{(m)}(t)(\sigma_t^2 + \sigma_r^2)P_{total,n}$ represents the interference caused by HWI-induced signal distortions and $I_{SIC} = \theta_c g_{k,m,n}(t) P_{c,n}^{(m)}$ is the interference caused by applying SIC to decode the common message, with $\theta_c \in [0, 1]$ representing the SIC error coefficient. $\theta_c = 0$ and $\theta_c = 1$ represent the scenarios with ideal SIC and no SIC, respectively.

The common stream's achievable rate for the k-th IIoT device when considering FBL is expressed as (He *et al.*, 2021):

$$\mathrm{R}_{c,k}^{(n)} = \log_2(1 + \gamma_{c,k}^{(n)}) - \frac{\mathrm{Q}^{-1}(\epsilon)}{\sqrt{n_b}} \sqrt{\mathrm{V}(\gamma_{c,k}^{(n)})}, \tag{3.13}$$

where $\mathrm{V}(\gamma_{c,k}^{(n)})$ is defined as:

$$\mathrm{V}(\gamma_{c,k}) = 1 - \frac{1}{(1 + \gamma_{c,k}^{(n)})^2}, \tag{3.14}$$

and n_b and ϵ represent the blocklength and block error probability, respectively. Moreover, $Q^{-1}(.)$ is the inverse of the Gaussian Q-function $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt$. It should be noted that the common message needs to be decoded by each device, and, therefore, the common stream's achievable rate is equal to the lowest of all the receivers' common rates:

$$\mathrm{R}_c^{(n)} = \min \left[\mathrm{R}_{c,1}^{(n)}, \cdots, \mathrm{R}_{c,|C_n^{(m)}|}^{(n)} \right]. \tag{3.15}$$

Once the interference has been removed from the common message using SIC, each device decodes its private message by considering the interference from the other devices in the device cluster as noise. The private stream's achievable rate for the k-th IIoT device, $\forall k \in C_n^{(m)}$, is obtained as:

$$\mathrm{R}_{k,p}^{(n)} = \log_2(1 + \gamma_{p,k}^{(n)}) - \frac{\mathrm{Q}^{-1}(\epsilon)}{\sqrt{n_b}} \sqrt{\mathrm{V}(\gamma_{p,k}^{(n)})}. \tag{3.16}$$

We consider that each AP has a total of N RRBs, and consequently, each AP has a maximum of N IIoT device clusters. We denote the m-th AP's device clusters by the sets $C_1^{(m)}$, $C_2^{(m)}$, $\cdots$, $C_N^{(m)}$. The achievable sum rate of the n-th device cluster in the m-th AP is obtained as

$$\mathrm{R}_{C_n}^{(m)} = \frac{B}{MN} \left(\mathrm{R}_c^{(n)} + \sum_{k \in C_n^{(m)}} \mathrm{R}_{k,p}^{(n)} \right), \tag{3.17}$$

where B is the total system bandwidth, which is divided equally among all the APs and their device clusters. The total achievable sum rate of the m-th AP is obtained as:

$$\mathrm{R}_{total}^{(m)} = \sum_{n=1}^{N} \mathrm{R}_{C_n}^{(m)}. \tag{3.18}$$

3.3.4 Problem Formulation

The joint clustering and transmit power allocation optimization problem is formulated as (3.19) at the top of the next page. Constraints (C1) and (C2) imply that the transmit power of the common

$$\text{P0:} \quad \max_{\boldsymbol{P},\{C_n^{(m)}\}} \sum_{m=1}^{M} R_{total}^{(m)}$$
$$\text{s.t.} \begin{cases} \text{C1: } P_{min} \le P_{c,n}^{(m)} \le P_{max}, \forall n \in \mathcal{N}, m \in \mathcal{M} \\ \text{C2: } P_{min} \le P_{p,k,n}^{(m)} \le P_{max}, \forall k \in C_n^{(m)}, n \in \mathcal{N}, m \in \mathcal{M} \\ \text{C3: } \sum_{n=1}^{N} \sum_{k \in C_n^{(m)}} (P_{p,k,n}^{(m)} + P_{c,n}^{(m)}) \le P_t, \forall k \in C_n^{(m)}, n \in \mathcal{N}, m \in \mathcal{M} \\ \text{C4: } |C_n^{(m)}| \le K_D, \forall n \in \mathcal{N}, m \in \mathcal{M} \\ \text{C5: } C_n^{(m)} \cap C_t^{(m)} = \varnothing, \forall n, t \in \mathcal{N}, n \ne t, m \in \mathcal{M} \end{cases} \tag{3.19}$$

and private messages must be bounded by P_{min} and P_{max}, which represent the minimum and maximum transmit power limits of an IIoT device, respectively. Constraint (C3) implies that the total power provided by an AP must be less than the total power an AP can achieve. Constraint (C4) implies that each device cluster can support a maximum of K_D IIoT devices. Constraint (C5) ensures that the device clusters belonging to the same AP do not overlap.

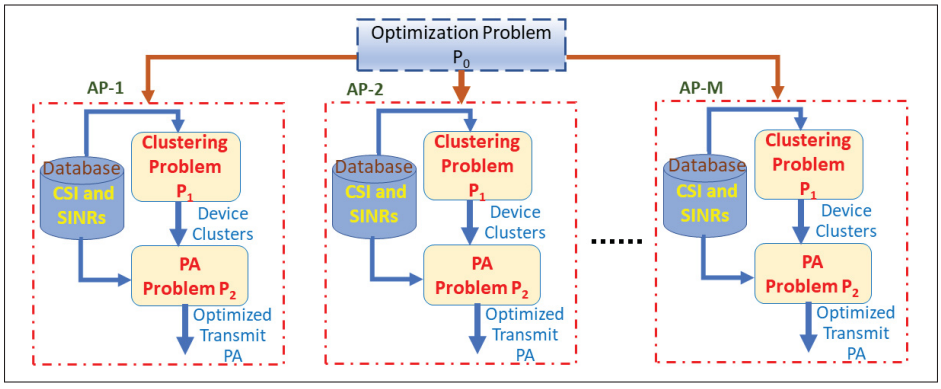


Figure 3.2 Proposed DCPM framework

Proposition 1: P0 is an NP-hard optimization problem.

Proof: The proof is provided in Appendix I.

Since P0 is an NP-hard and mixed-integer non-linear programming (MINLP) problem, it is computationally intractable to obtain its global optimal solution using standard optimization techniques. Therefore, we propose to mitigate this computational intractability with a sequential optimization framework. The framework involves fixing one block of optimization variables so that P0 can be decomposed into two sub-problems, namely, a device clustering one and a power allocation one. If we assume the power allocation is fixed, i.e., $P_{c,n}^{(m)} = P_0$ and $P_{p,k,n}^{(m)} = P_0$, the device clustering sub-problem can be formulated as:

$$\begin{aligned} \text{P1:} \quad & \max_{\{C_n^{(m)}\}} \sum_{m=1}^{M} R_{total}^{(m)} \\ & \text{s.t. C4, C5.} \end{aligned} \tag{3.20}$$

Meanwhile, if we consider that each AP has a given set of device clusters, the power allocation sub-problem can be formulated as:

$$\begin{aligned} \text{P2:} \quad & \max_{\mathbf{P}} \sum_{m=1}^{M} R_{total}^{(m)} \\ & \text{s.t. C1, C2, C3.} \end{aligned} \tag{3.21}$$

We propose to use a distributed optimization framework entitled the DCPM framework, which is illustrated in Fig. 3.2, to sequentially solve P1 and P2. Each AP consists of a database that stores the instantaneous local CSI and past SINRs of its associated devices. Each AP obtains its set of device clusters by solving P1 using the local CSI extracted from its database. Then, each AP obtains its device clusters' transmit power allocation by solving P2. We built this optimization framework by first developing a greedy clustering algorithm to solve P1 (see Section 3.4). We then developed a MADRL approach to solve P2 (see Section 3.5). The overall DCPM algorithm leverages the greedy clustering and DRL approaches to solve P0 and is presented in Section 3.6.

Remark 1: The motivation for using the DRL approach to solve P2 is as follows. P2 is a non-convex and computationally intractable optimization problem that becomes maximizing a sum-of-function-ratios, which is NP-complete. Solving P2 using a conventional optimization

technique that relies on satisfying the Karush-Khun-Tucker (KKT) conditions is practically challenging. Conventional optimization techniques require precise knowledge of devices' instantaneous SINRs to derive the KKT conditions. However, it is practically unfeasible to know the exact value of HWI-induced signal distortions at the devices, as these distortions vary randomly from device to device. We specifically consider a practical setting in which the devices' exact non-linearity models are unknown to the network controller. The SIC error coefficients and the interference caused by imperfect SIC at the receivers are also unknown to the network controller. Therefore, even though the instantaneous CSI is assumed to be available, the devices' *exact instantaneous* SINR values are not available to the network controller in practice. Such a fact renders conventional optimization techniques ineffective at solving P2. In contrast, the DRL approach learns a suitable policy for allocating transmit power based on the devices' historical SINRs and current CSI[4]. The DRL approach neither requires precise knowledge of the devices' instantaneous SINRs nor the exact value of HWI-induced signal distortions to do this. Moreover, once the DRL agent is fully trained, it can also quickly solve P2 and thus converges in much less time than the conventional optimization method considered, which requires several iterations to converge. This efficiency is particularly advantageous in dynamic environments where quick decision making is crucial. Therefore, the DRL approach is a robust and scalable means of solving P2.

Remark 2: The optimality of the overall solution can be improved by alternately optimizing P1 and P2 instead of solving them only once sequentially. However, this iterative optimization approach increases the computational complexity, signaling overhead, and solution time. In this approach, device clusters need to be updated at each iteration based on the transmit powers obtained in the previous iteration. Afterward, the SINRs (with HWI-induced distortions) must be estimated for the newly formed device clusters. Finally, the power allocation needs to be updated based on the new SINR values. It is practically non-trivial to perform these three steps

[4] We assume that the IIoT devices provide accurate received SINR feedback at the end of each TS. Hence, the AP knows the SINRs of the common and private streams of the associated IIoT devices from the previous TS. AP needs to compute power allocation decisions for the current TS using that information.

iteratively several times within the channel coherence time. A sequential optimization approach is therefore adopted to solve P0 in a computationally efficient manner.

3.4 Solution to P1: Device Clustering Algorithm

Algorithm Development: Since there is no interference among the APs, P1 can be decomposed into a total of M independent optimization problems, one for each AP. The device clustering sub-problem for the m-th AP, $\forall m \in \mathcal{M}$, is formulated as:

$$\text{P1.1:} \quad \max_{\{C_n^{(m)}\}} \sum_{n=1}^{N} R_{C_n}^{(m)} \quad \text{s.t. C4, C5.} \tag{3.22}$$

The computational complexity involved in solving P1.1 optimally can be obtained from the Stirling number of the second kind, which is defined as the number of ways K different objects (i.e., IIoT devices) can be partitioned into N non-overlapping device clusters. The Stirling number of the second kind is expressed as:

$$\left\{ {K \atop N} \right\} \triangleq \frac{1}{N!} \sum_{j=0}^{N} (-1)^{N-j} \binom{N}{j} j^K. \tag{3.23}$$

For $K \gg N$, the Stirling number of the second kind is approximated as $\mathcal{O}(N^{K-N})$ (Ben Ghorbel *et al.*, 2018). Clearly, the computational complexity involved in solving P1.1 optimally increases exponentially with the number of IIoT devices, and, as a result, arriving at a theoretically optimal solution to P1.1 is computationally intractable for practical IIoT networks. However, it is possible to develop a sub-optimal yet efficient algorithm to solve P1.1 by exploiting the problem's characteristics. To this end, we assume that certain clusters of $K' \geq 1$ devices are known[5]. Doing so reduces P1.1 to an optimization problem that involves assigning the remaining $(K - K')$ devices to N RRBs in such a way as to maximize the total sum rate. The optimal

[5] One can obtain initial set of device clusters by assigning K' devices to their most suitable RRBs such that each device is associated with only one RRB and each RRB is associated with only one device.

device–RRB assignment needs to maximize (a) the device's SINR and (b) the minimum channel gain of the devices in the cluster. An efficient device–RRB assignment can be obtained by selecting the remaining $(K - K')$ devices in a suitable order and sequentially assigning them to their most appropriate device cluster[6].

In light of the above discussion, P1.1 can be alternatively solved by finding the clusters of IIoT devices that maximize the clustered devices' total SINRs, which are the sum of the SINRs of their common and private messages. Without loss of generality, we assume that the k-th IIoT device is part of the n-th cluster. The total SINR of the k-th IIoT device is obtained as $\overline{\gamma_k^{(n)}} = \gamma_{c,k}^{(n)} + \gamma_{p,k}^{(n)}$, where

$$\gamma_{c,k}^{(n)} = \begin{cases} \dfrac{P_o}{|C_n^{(m)}| P_o + \dfrac{\sigma^2}{\min\{\min\limits_{k' \in C_n^{(m)}} g_{k',n}^{(m)}, g_{k,n}^{(m)}\}}} \text{ if } |C_n^{(m)}| \geq 1 \\ \dfrac{P_o}{\dfrac{\sigma^2}{g_{k,n}^{(m)}}} \text{ if } |C_n^{(m)}| = 0 \end{cases} \tag{3.24}$$

and

$$\gamma_{p,k}^{(n)} = \begin{cases} \dfrac{P_o}{\left(|C_n^{(m)}| - 1 + \theta_c\right) P_o + \dfrac{\sigma^2}{g_{k,n}^{(m)}}} \text{ if } |C_n^{(m)}| \geq 1 \\ \dfrac{P_o}{\dfrac{\sigma^2}{g_{k,n}^{(m)}}} \text{ if } |C_n^{(m)}| = 0, \end{cases} \tag{3.25}$$

where P_o is the fixed transmit power of both the device's common and all its private messages[7]. Let us introduce the variable $z_{k,n}^{(m)}$, which equals 1 if $k \in C_n^{(m)}$ and 0 otherwise. The modified

[6] This idea is inspired by the greedy approach used to solve the generalized assignment problem (GAP) and exploits the similarity that exists between P1.1 and a GAP (Romeijn & Morales, 2000).

[7] Recall that instantaneous information about HWI-induced signal distortions and SIC error coefficients is not available at the devices. Consequently, interference caused by HWI-induced signal distortions and imperfect SIC is ignored in (3.24) and (3.25), i.e., during the device clustering stage.

device clustering sub-problem can be formulated as:

$$\text{P1.2:} \quad \max_{\{z_{k,n}^{(m)} \in \{0,1\}} \sum_{n=1}^{N} \sum_{k=1}^{K} z_{k,n}^{(m)} \left(\gamma_{c,k}^{(n)} + \gamma_{p,k}^{(n)} \right)$$
$$\text{s.t.} \begin{cases} \sum_{k=1}^{K} z_{k,n}^{(m)} \leq K_D, \forall n \in \mathcal{N} \\ \sum_{n=1}^{N} z_{k,n}^{(m)} = 1, \forall k \in \mathcal{K}. \end{cases} \tag{3.26}$$

We develop a greedy clustering algorithm, which is summarized in Algorithm 3.1, to solve P1.2. Algorithm 3.1 iteratively selects a device and assigns it to an RRB (i.e., device cluster). It maintains two sets – (i) $\mathbf{UM}_U$, the set of unassigned IIoT devices; and (ii) $\mathbf{UM}_R$, the set of RRBs that can accommodate at least one IIoT device. Furthermore, it keeps track of the number of IIoT devices that are assigned to the n-th RRB, which is denoted by D_n. When a device is associated with an RRB, it simultaneously obtains a profit by increasing its own SINR gain and incurs a cost by either reducing the minimum channel gain of its cluster (and thereby, reducing the common message rate) or reducing the SINRs of other devices' private messages. Hence, attention must be paid to the order in which devices are selected and assigned to their most suitable clusters, as it can enhance the net profit. To select the most suitable device cluster for a given IIoT device at each iteration of Algorithm 3.1, we consider the following score

$$\Delta_k = \overline{\gamma_k^{(n_k^*)}} - \max_{\substack{n \in \mathbf{UM}_R \\ n \neq n_k^*}} \overline{\gamma_k^{(n)}}, \tag{3.27}$$

where n_k^* denotes the index of the k-th IIoT device's most suitable cluster, and the first and second terms of (3.27) provide the k-th IIoT device's highest and second-highest SINRs, respectively, obtained from the available device clusters in $\mathbf{UM}_R$. This score captures the net profit that would be obtained from assigning the k-th IIoT device to its most suitable device cluster. In other words, it provides a measure of the importance of including the IIoT device in its most suitable cluster.

The key steps in Algorithm 3.1 are summarized as follows. Step 1 initializes the set $\mathbf{UM}_U$, the variable D_n and the device cluster sets $\{C_n\}$, $\forall n \in \mathcal{N}$. Steps 3-10 iteratively compute the score defined in (3.27) for each unassigned device in $\mathbf{UM}_U$ when it is paired with each available device cluster in $\mathbf{UM}_R$. Step 11 selects the device that achieves the highest score. Step 12 assigns the selected device to its most suitable device cluster in $\mathbf{UM}_R$, and updates $\mathbf{UM}_U$ and $\{D_n\}$. The aforementioned steps continue until either $\mathbf{UM}_U$ or $\mathbf{UM}_R$ is empty.

Computational Complexity: We first determine the computational complexity of executing a single loop iteration of Algorithm 3.1, i.e., Steps 2-12. The worst-case computational complexity of executing Steps 4-10 is $\mathcal{O}\left(|\mathbf{UM}_U|N\right)$, where $|\mathbf{UM}_U|$ denotes the cardinality of $\mathbf{UM}_U$. The computational complexity of executing Step 11 is $\mathcal{O}\left(|\mathbf{UM}_U|\right)$, and that of the remaining steps is $\mathcal{O}(1)$. The total computational complexity of a single loop iteration of Algorithm 3.1 is therefore $\mathcal{O}\left(|\mathbf{UM}_U|N + |\mathbf{UM}_U|\right)$. Note that since Algorithm 3.1 sequentially clusters a device and removes it from $\mathbf{UM}_U$, the variable $|\mathbf{UM}_U|$ is reduced by 1 at each iteration. In the end, the overall computational complexity of Algorithm 3.1 is obtained as $\mathcal{O}\left(\sum_{l=1}^{K}(Nl + l)\right) \approx \mathcal{O}\left(K^2N\right)$.

Algorithm 3.1 Proposed IIoT Device Clustering Algorithm for the m-th AP

Input: Number of IIoT devices K, number of RRBs N, channel gains $\{g_{k,n}^{(m)}\}$, fixed PAs $P_{c,n}^{(m)} = P_0$ and $P_{p,k,n}^{(m)} = P_0, \forall k \in C_n^{(m)}, n \in \mathcal{N}, m \in \mathcal{M}$.

Initialize: $\mathbf{UM}_U = \{1, 2, \cdots, K\}$, $D_n = 0, \forall n \in \{1, 2, \cdots, N\}$; IIoT device clusters $C_n = \varnothing$, $\forall n \in \{1, 2, \cdots, N\}$.

while $\mathbf{UM}_U \neq \varnothing$ *and* $\mathbf{UM}_R \neq \varnothing$ **do**

  Find $\mathbf{UM}_R = \{n \in \{1, 2, \cdots, N\} \mid D_n < K_D\}$.

  for $k = 1 : |\mathbf{UM}_U|$ **do**

    for $n = 1 : |\mathbf{UM}_R|$ **do**

      Compute $\overline{\gamma_k^{(n)}} = \gamma_{c,k}^{(n)} + \gamma_{p,k}^{(n)}$ using Eq. (3.24) and Eq. (3.25).

    end for

    Find $n_k^* = \arg\max_{n \in \mathbf{UM}_R} \overline{\gamma_k^{(n)}}$.

    Compute $\Delta_k = \overline{\gamma_k^{(n_k^*)}} - \max_{\substack{n \in \mathbf{UM}_R \\ n \neq n_k^*}} \overline{\gamma_k^{(n)}}$.

  end for

  Find $k^* = \arg\max_{k \in \mathbf{UM}_U} \Delta_k$.

  Assign $C_{n_k^*} \leftarrow C_{n_k^*} \cup \{k^*\}$; update $D_{n_k^*} = D_{n_k^*} + 1$ and $\mathbf{UM}_U \leftarrow \mathbf{UM}_U \setminus \{k^*\}$.

end while

Output: IIoT device clusters $C_1, C_2, \cdots, C_N$.

3.5 Solution to P2: DRL-Empowered Power Allocation Algorithm

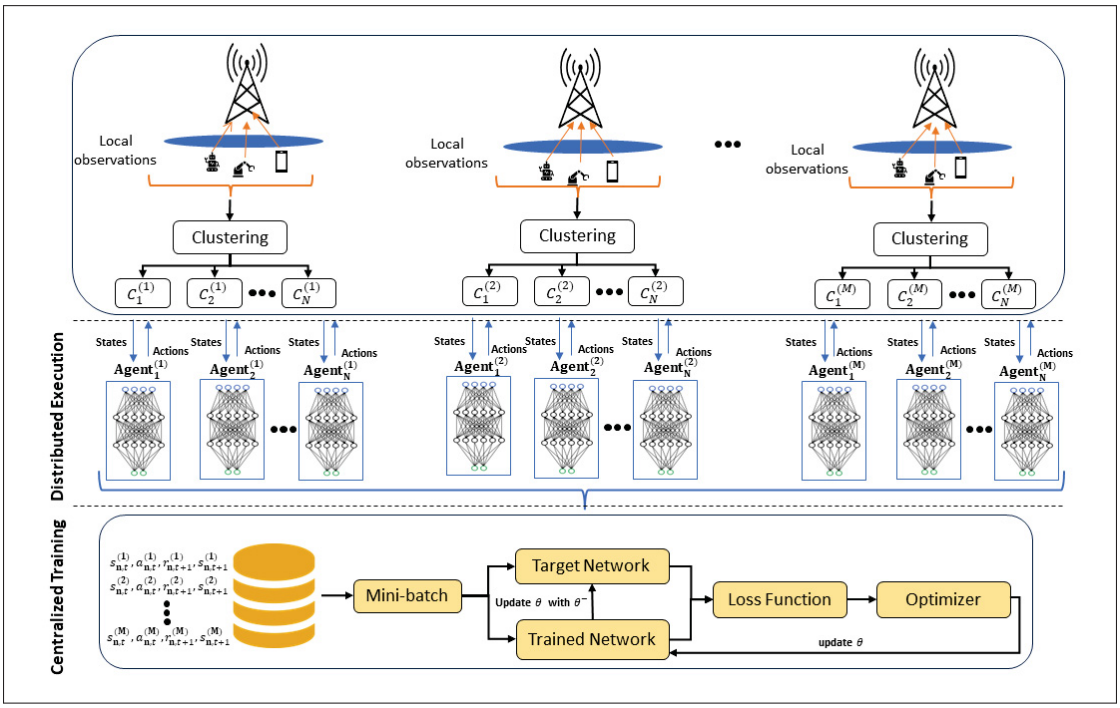


Figure 3.3 Illustration of the proposed MADRL algorithm with centralized training and distributed execution

In this work, we develop a DQN-based MADRL algorithm to optimize power allocation at each AP. The main constituents of our proposed DRL scheme are defined as follows:

1. **Agents:** Each device cluster is a learning agent. The set of learning agents associated with the m-th AP is denoted by $\mathcal{Z}^{(m)} = \{1, 2, \cdots, n\}$, and thus $\mathcal{Z} = \cup_{m=1}^{M} \mathcal{Z}^{(m)}$ represents an overall set of learning agents.
2. **States:** The state space refers to the set of all possible states that can apply to the environment during the learning process. In our context, the state space of the n-th agent in the m-th AP comprises three key components:
   (i) The set of local CSI at the t-th TS $(g_{1,n}^{(m)}, ..., g_{|C_n^{(m)}|,n}^{(m)})$,
   (ii) The set of SINRs of the common streams at the $(t-1)$-th TS $(\gamma_{c,1}^{(n)}, \gamma_{c,2}^{(n)}, ..., \gamma_{c,|C_n^{(m)}|}^{(n)})$,
   (iii) The set of SINRs of the private streams at the $(t-1)$-th TS $(\gamma_{p,1}^{(n)}, \gamma_{p,2}^{(n)}, ..., \gamma_{p,|C_n^{(m)}|}^{(n)})$.

Therefore, the state space of the n-th agent in the m-th AP, $\forall n \in \mathcal{Z}^{(m)}$, at the t-th TS is formally defined as:

$$s_{n,t}^{(m)} = \{(g_{1,n}^{(m)}, ..., g_{|C_n^{(m)}|,n}^{(m)}), (\gamma_{c,1}^{(n)}, \gamma_{c,2}^{(n)}, ..., \gamma_{c,|C_n^{(m)}|}^{(n)}), (\gamma_{p,1}^{(n)}, \gamma_{p,2}^{(n)}, ..., \gamma_{p,|C_n^{(m)}|}^{(n)})\}. \quad (3.28)$$

At every TS, the state space provides information about the current state of the environment, including the instantaneous channel gains. Additionally, feedback is provided on the outcome of previous actions through the SINRs of the previous TS(s). This information helps the agents learn a suitable policy while also keeping track of the history of the effectiveness of their chosen actions over the fading channels.

3. **Actions:** Each action taken by each agent represents a set of discrete power levels between P_{min} and P_{max} for the agent's associated IIoT devices. The action space of the u-th user device in the n-th device cluster is defined as:

$$\mathcal{A}_u^{(n)} = \{P_{min}, \frac{P_{max}}{NP-1}, \frac{2P_{max}}{NP-1}, \ldots, P_{max}\}, \quad (3.29)$$

where NP is the total number of discrete power levels. Note that for a device cluster of K_D IIoT devices, a total of (K_D+1) discrete power levels (i.e., one level for the common message and K_D levels for the private messages) are selected. Thus, the action space of the n-th agent in the m-th AP, $\forall n \in \mathcal{Z}^{(m)}$ and $\forall m \in \mathcal{M}$, is defined as $\mathcal{A}_n = \mathcal{A}_1^{(n)} \times \mathcal{A}_2^{(n)} \times \cdots \mathcal{A}_{K_D}^{(n)} \times \mathcal{A}_{K_D+1}^{(n)}$, where $\mathcal{A}_i^{(n)}, i = 1 \cdots K_D$, denotes the set of discrete power levels for the i-th private message and $\mathcal{A}_{K_D+1}^{(n)}$ denotes the discrete power level for the common message.

4. **Reward:** The agents (i.e., device clusters) select transmit power levels for the IIoT devices to maximize the sum rate. We consider a collaborative learning framework in which all the agents jointly maximize the AP's total sum capacity. Accordingly, the reward function of the n-th agent associated with the m-th AP, $\forall m \in \mathcal{M}$, is defined as $\hat{R}_n^{(m)} = \mathrm{R}_{total}^{(m)}, \forall n \in \mathcal{Z}^{(m)}$, where $\mathrm{R}_{total}^{(m)}$ is defined in (3.18).

Fig. 3.3 shows the architecture of our proposed MADRL framework. The environment comprises M APs that each host K IIoT devices randomly distributed within their coverage area. The position of each IIoT device changes from one TS to the next. Therefore, in each TS, the proposed solution begins by clustering the devices associated with each AP using Algorithm 3.1 to obtain, for example, the set of clusters $\{C_1^{(m)}, C_2^{(m)}, \cdots, C_N^{(m)}\}$ for the m-th AP, $\forall m \in \mathcal{M}$. These device clusters are our MADRL framework's learning agents. In each TS, these RL agents observe the states of the clustered IIoT devices and then allocate the AP's transmit power levels to the clustered devices. The RL agents can be considered independent DQN agents and trained by employing the DQN framework discussed in Section 1.4.2. However, extending the single-agent DRL (SADRL) framework to a multi-agent scenario in such a straightforward way entails the following two challenges. First, due to the total transmit power constraint C3, an agent's state transition depends on the policies adopted by the other agents belonging to the same AP. Since the agents continuously update their policies during training, the learning environment becomes non-stationary and violates the Markov properties of state transitions, which results in instability in DQN training. Second, a total of MN DQNs need to be trained simultaneously since each agent's policy is provided by an independent DQN, and this significantly limits the scalability of the learning framework. We overcome these challenges by employing a centralized training and distributed execution (CTDE) approach with parameter sharing (Nasir & Guo, 2019). In this approach, each agent maintains an identical DQN. The DQN is centrally trained by a centralized controller based on the experiences collected by all the agents and shared with all the agents. Unlike in (Seid, Boateng, Mareri, Sun & Jiang, 2021), the CTDE approach significantly reduces the computational complexity involved and the amount of memory required, as only a single DQN needs to be trained. Furthermore, since all the agents use the same DQN simultaneously, the action space (i.e., DQN size) does not increase with the number of agents. Consequently, this approach is scalable and suitable for large-scale networks. Moreover, the CTDE approach trains the DQN model based on the collective experience of all the agents. This not only ensures better stability (and avoids biased training), but also enables collaborative learning so the agents can select suitable actions. As a result, the system reward (i.e., the network sum rate) improves.

The CTDE approach is empirically shown to converge for MARL with homogeneous agents (Gupta, Egorov & Kochenderfer, 2017).

Algorithm 3.2 provides the overall steps for training our proposed MADRL framework using the CTDE approach. We consider episodic learning with Z RL episodes and T steps per RL episode, each of which represents a TS. Thus, the channel gains and user positions vary at each step. At each TS of an episode, Algorithm 3.2 randomly generates each device's location and channel gain, and clusters the devices using the proposed Algorithm 3.1. The inclusion of device clustering in DRL model training enables the DQN model to learn the inherent relationship that exists between power control and device clustering for decision making.

Algorithm 3.2 is summarized as follows. Algorithm 3.2 initializes the replay memory, the power allocation agent (i.e., the DQN to be trained), and the target DQN. At each step in each RL episode, the channel gains and device locations vary (Line 7). A set of device clusters (i.e., agents) is formed for each AP using Algorithm 3.1 (Line 9). Each agent collects state information from the environment (Line 11). The clustered devices' transmit power levels are selected using the ϵ-greedy approach (Lines 12-21). Note that although each agent uses the same DQN, the actions they select are different since their observed states are different. Line 22 normalizes the transmit power of all the devices associated with an AP so that constraint C3 is satisfied. The updated transmit power values are used to determine the agent's rewards and next states (Line 23). Each agent's experience, such as $(s_{n,t}^{(m)}, a_{n,t}^{(m)}, r_{n,t+1}^{(m)}, s_{n,t+1}^{(m)})$, is stored in the replay memory D (Line 24). A mini-batch of experiences is randomly sampled from the replay memory to train the DQN agent (Line 26). This makes it possible to train the DQN agent based on previous experiences instead of the most recent ones, which are potentially correlated and may introduce bias in the training. The DQN's parameters are updated by applying the backpropagation technique and (1.7) (Line 27). Finally, the target DQN's weights are updated with the new weights of the trained DQN after every T_{step} number of steps (Line 28). Lines 5-28 are iteratively repeated for all the steps and RL episodes. Algorithm 3.2's convergence is confirmed via simulations (see Fig. 3.8).

Remark 3: In practice, the number of devices associated with an AP and the number of devices per cluster can vary. We fix the dimensions of the trained DQN's input and output layers to make it robust in all scenarios. Since the maximum number of devices per cluster is K_D consistent with the state space description, the dimension of the DQN's input layer is set to $3K_D$. The dimension of the DQN's output layer is set to NP (i.e., the total number of discrete power levels). If an AP has more IIoT devices than the total number device clusters are allowed, Algorithm 3.1 selects the best K_D devices for each cluster at each TS. Meanwhile, a device cluster's state space is padded with zeros when the cluster has fewer than K_D devices (Naderializadeh *et al.*, 2021).

3.6 Overall DCPM Algorithm

3.6.1 Description of DCPM Algorithm

All the steps of the proposed DCPM framework are set out in Algorithm 3.3, which is summarized as follows. Each AP is considered to have a copy of the trained DQN-enabled power allocation agent. At each TS, the APs first collect the instantaneous channel gains of their associated IIoT devices (Line 4). Next, they form N device clusters by executing Algorithm 3.1. Then, they determine the transmit power levels of the common and private messages in their associated device clusters by executing Lines 6-15 using the trained DQN. Finally, they transmit data to the devices in each cluster using the optimized power allocation. Note that Lines 4-15 can be executed simultaneously and independently at each AP without needing to exchange any information. Consequently, Algorithm 3.3 can be implemented in a distributed manner at each AP. Distributed implementation enables rapid algorithm execution, which makes Algorithm 3.3 particularly advantageous for delay-sensitive IIoT applications.

3.6.2 Computational Complexity of DCPM Algorithm

We first determine the computational complexity of executing Lines 4-15. Line 5's computational complexity is $\mathcal{O}(K^2N)$. Note that each device cluster can accommodate a maximum of K_D devices. Hence, the worst-case computational complexity of executing Lines 6-11 is $\mathcal{O}(NK_DNP)$,

Algorithm 3.2 Algorithm for Training DQN-Enabled Power Allocation Agent

Input: Maximum number of episodes Z and maximum number of steps per episode T.
Initialize: the replay memory D to zero.
Create a DQN-enabled power allocation agent, $q(:,:;\theta)$ with random weights θ.
Create a target DQN $q(:,:;\theta^{-})$ with $\theta^{-} = \theta$.
for $episode \leftarrow 1 : Z$ **do**
  Initializes channel fading gains for all the devices and APs.
  for $t \leftarrow 1 : T$ **do**
    Vary the devices' locations within APs' coverage zone and channel fading gains using (3.1).
    for $m \leftarrow 1 : M$ **do**
      Update N IIoT device clusters $\left\{C_1^{(m)}, C_2^{(m)}, \cdots, C_N^{(m)}\right\}$ using Algorithm 3.1.
      for $n \leftarrow 1 : |\mathcal{Z}^{(m)}|$ **do**
        Observe the state of the environment, $s_{n,t}^{(m)}$. Initialize $a_{n,t}^{(m)} \leftarrow \varnothing$.
        for $u \leftarrow 1 : |C_n^{(m)}| + 1$ **do**
          Generate a random number z.
          if $z \geq \epsilon$ **then**
            Determine $i = \arg\max_{a \in \mathcal{A}_u^{(n)}} q(s_{n,t}^{(m)}, a, \theta)$.
          else
            Select $i \in \{1, 2, \cdots, |\mathcal{A}_u^{(n)}|\}$ randomly.
          end if
          Assign $a_{n,t}^{(m)} \leftarrow a_{n,t}^{(m)} \cup \mathcal{A}_u^{(n)}[i]$.
        end for
      end for
      Normalize the common and private messages' transmit power over all the clusters so that $\frac{|a_{n,t}^{(m)}|_1}{P_t} = 1$ is satisfied.
      Transmit using the normalized transmit power and measure the instantaneous SINRs at the devices. Determine the immediate reward $r_{n,t+1}^{(m)} = \mathrm{R}_{total}^{(m)}$ and the next state $s_{n,t+1}^{(m)}$, $\forall n \in \mathcal{Z}^{(m)}$.
      Save the experience $\left(s_{n,t}^{(m)}, a_{n,t}^{(m)}, r_{n,t+1}^{(m)}, s_{n,t+1}^{(m)}\right)$, $\forall n \in \mathcal{Z}^{(m)}$, in replay memory D.
    end for
    Sample a random mini-batch of experience from D.
    Update the weight of power allocation agent, θ, by applying (1.7) and the backpropagation method.
    After every T_{step} time step, update $\theta^{-} = \theta$.
  end for
end for
Output: Trained power allocation agent $q^{*}(s, a; \theta)$.

where NP is the number of available discrete power levels. The computational complexity of the remaining lines is $O(1)$. Thus, the computational complexity of making clustering and power allocation decisions for a single AP is $O(N(K^2 + K_D NP))$. Since there are a total of M APs in the system, the overall computational complexity of executing Algorithm 3.3 in a single TS is $O(MN(K^2 + K_D NP))$.

Algorithm 3.3 Distributed Clustering and Power Management Algorithm

Input: Number of APs M; number of IIoT devices K and RRBs N per AP; maximum number of devices per cluster K_D; trained DQN for power allocation $q^*(s, a; \theta)$; total number of TSs T_s.

Initialize: TS index $t = 1$.
while $t \leq T_s$ **do**
  for $m \leftarrow 1 : M$ **do**
    Collect the instantaneous channel gains $\{g_{k,n}^{(m)}\}$, for the associated IIoT devices.
    Determine N device clusters $\left\{\mathcal{C}_1^{(m)}, \mathcal{C}_2^{(m)}, \cdots, \mathcal{C}_N^{(m)}\right\}$ using Algorithm 3.1.
    for $n \leftarrow 1 : N$ **do**
      Acquire the state information $s_{n,t}^{(m)}$ described in Section 3.5.
      for $u \leftarrow 1 : |\mathcal{C}_n^{(m)}| + 1$ **do**
        Determine the optimal power allocation action $a_u^* = \arg\max_{a \in \mathcal{A}_u^{(n)}} q^*(s_{n,t}^{(m)}, a; \theta)$.
      end for
    end for
    for $n \leftarrow 1 : N$ **do**
      Determine the transmit power of the common message as

$$P_{c,n}^{(m)} = \frac{a^*_{|\mathcal{C}_n^{(m)}|+1}}{\sum_{n=1}^{N} \sum_{u=1}^{|\mathcal{C}_n^{(m)}|+1} a_u^*} \times P_t$$

      Determine the transmit power of the k-th device's private message as

$$P_{p,k,n}^{(m)} = \frac{a_k^*}{\sum_{n=1}^{N} \sum_{u=1}^{|\mathcal{C}_n^{(m)}|+1} a_u^*} \times P_t, \forall k \in \mathcal{C}_n^{(m)}$$

    end for
  end for
  Perform downlink data transmission at all cells using the optimized device clusters and PA; increase the TS index $t = t + 1$.
end while
Output: Device clusters and transmit power allocation solution to P0 at each TS.

Remark 4: For the optimal performance of Algorithm 3.3, the environment used to train the DQN agent in Algorithm 3.2 and the environment observed during online inference need to be similar. Nevertheless, in practice, an IIoT network usually exhibits several dynamic factors that make it difficult to keep the environment identical throughout model training and online inference. This challenge can be addressed by periodically updating the trained DQN as follows. Train the initial DQN model using Algorithm 3.2 and a realistic network simulator or DT[8]. Then,

[8] Note that Algorithm 3.2 requires several rounds of exploration (i.e., trial and error) to collect sufficient samples and optimize the DQN network. However, conducting exploration in a live network is expensive. Algorithm 3.2 can be executed in a virtual domain while considering the realistic network simulator as the environment to reduce costs.

deploy the trained model in a live network (i.e., in Algorithm 3.3), and periodically collect the experiences of the agents (i.e., device clusters) from the APs and send them to a replay buffer. Once a sufficient number of sample experiences have been collected, the weights of the DQN model can be updated by executing Lines 26-28 of Algorithm 3.2. Periodically updating the DQN in this way will enable Algorithm 3.3 to maintain its adaptability to dynamic IIoT networks. It is worth noting that the following two types of communication overhead are required at each TS for DQN model training: (1) APs send their experiences (i.e., observed states, actions, and rewards) to the central server, and (2) the updated DQN model is sent back to the APs by the central server. While the amount of information exchanged increases with the number of APs, TSs, and RL episodes, we assume that the APs are connected to the server via high-speed wired (e.g., Ethernet or optical fiber) links. As a result, the overhead can be afforded. Additionally, it is important to note that training is a one-time data-intensive process. Once the model has been trained, deploying it does not require significant communication overhead, as each AP can make distributed decisions independently using the trained model.

3.7 Simulation Results

3.7.1 Benchmark Schemes

This section presents our numerical simulation results and evaluates the performance of the proposed scheme for downlink RSMA-enabled IIoT networks that considers both FBL and HWI-induced distortions. In the subsequent numerical results, unless specified, our DCPM algorithm considers perfect SIC at the IIoT devices. For performance comparison, we consider the following benchmark schemes.

3.7.1.1 Clustering Algorithms

To demonstrate the superiority of our proposed clustering algorithm, we compare its performance to that of the following clustering algorithms.

- **Random clustering:** This scheme allocated devices to clusters in a purely random fashion, with no specific pattern or criteria governing allocation.
- **Clustering using channel gain:** In this case, device clusters are obtained by assigning the IIoT devices to their most suitable RRBs from the perspective of channel gain. This clustering scheme follows the same steps that are mentioned in Algorithm 3.1 except for when it comes to calculating the score Δ_k for the k-th device, $\forall k$. In this scheme, the score is calculated as the difference between the highest and second-highest channel gains of the k-th device obtained from the RRBs, which implies the importance of assigning the k-th device to its most suitable RRB. The pseudocode of the channel gain-based device clustering algorithm is summarized in Algorithm 3.4.

Once the IIoT devices have been clustered, the transmit power allocation is selected using the trained DQN from Algorithm 3.2.

3.7.1.2 Interference Management Schemes

We compare the performance of the proposed DCPM algorithm (i.e. Algorithm 3.3) to that of the following interference management schemes.

- **Treat interference as noise (TIN):** In this scheme, no common message is transmitted only the private messages are. Each device decodes its intended message by treating the interference from other devices in the same cluster as pure noise.
- **RSMA with imperfect SIC (ImpSIC):** This scheme considers that the receivers' SIC process is imperfect. Consequently, residual interference from the common message is encountered when decoding private messages. We take this interference into account by considering $\theta_c > 0$ in (3.12).

Since both the TIN and RSMA with ImpSIC schemes employ the DCPM framework, they are denoted by DCPM-TIN and DCPM-ImpSIC, respectively, in the ensuing discussion.

3.7.1.3 PA Algorithms

We compare the performance of the proposed DCPM algorithm to that of the following benchmark power allocation algorithms. Note that the benchmark power allocation algorithms are applied only after Algorithm 3.1 has been executed to obtain each AP's device clusters.

- **WMMSE:** The conventional WMMSE algorithm, which is given in (Sun *et al.*, 2018), is used to optimize the transmit power allocation of the clustered devices.
- **RP:** Each IIoT device's transmit power level at each TS is chosen randomly between 0 and P_{max}.
- **Maximum power (MaxP):** The IIoT devices in each cluster are given the maximum power level P_{max} at each TS.

Algorithm 3.4 IIoT Device Clustering Algorithm for the m-th AP Based on the Channel Gain

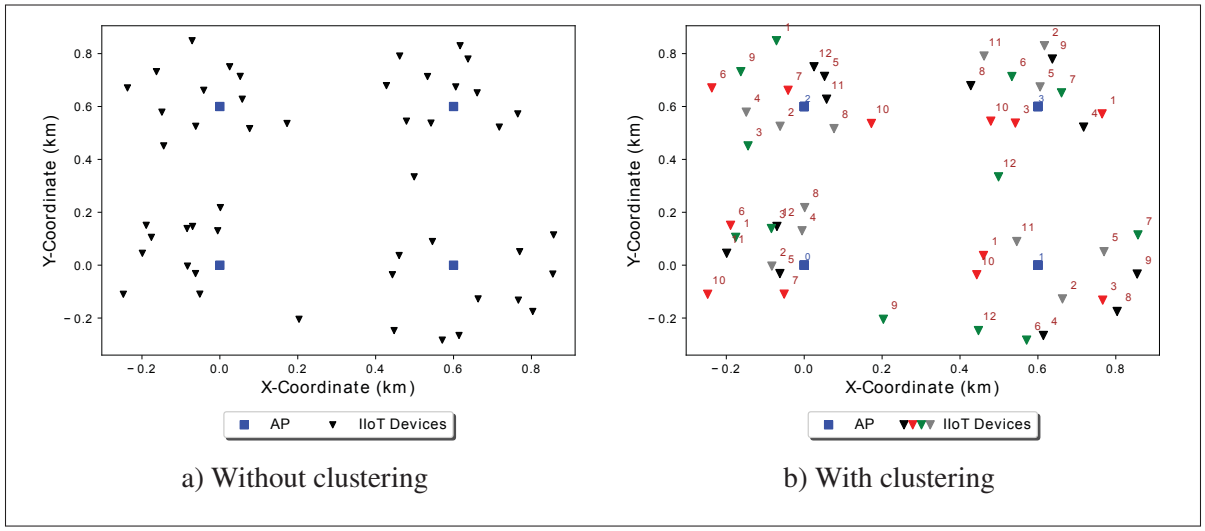
Input: Number of IIoT devices K, number of RRBs N, channel gains $\{g_{k,n}^{(m)}\}$.
Initialize: $\mathbf{UM}_U = \{1, 2, \cdots, K\}$, $D_n = 0$, $\forall n \in \{1, 2, \cdots, N\}$; IIoT device clusters $C_n = \varnothing$, $\forall n \in \{1, 2, \cdots, N\}$.
while $\mathbf{UM}_U \neq \varnothing$ *and* $\mathbf{UM}_R \neq \varnothing$ **do**
  Find $\mathbf{UM}_R = \{n \in \{1, 2, \cdots, N\} | D_n < K_D\}$.
  for $k = 1 : |\mathbf{UM}_U|$ **do**
    Calculate $n_k^* = \arg\max_{n \in \mathbf{UM}_R} g_{k,n}^{(m)}$.
    Calculate $\Delta_k = g_{k,n_k^*}^{(m)} - \max_{\substack{n \in \mathbf{UM}_R \\ n \neq n_k^*}} g_{k,n}^{(m)}$.
  end for
  Find $k^* = \arg\max_{k \in \mathbf{UM}_U} \Delta_k$.
  Assign $C_{n_k^*} \rightarrow C_{n_k^*} \cup \{k^*\}$; update $D_{n_k^*} = D_{n_k^*} + 1$ and $\mathbf{UM}_U \rightarrow \mathbf{UM}_U \setminus \{k^*\}$.
end while
Output: IIoT device clusters $C_1, C_2, \cdots, C_N$.

3.7.2 Simulation Settings

Our simulation model comprises four IIoT cells that each contain one AP and K IIoT devices placed randomly around the AP. To ensure that the position of the devices and the AP do not overlap, a small device-free region of radius r_s is maintained around each AP. Fig. 3.4(a) shows an example of a network configuration for 4 APs and 48 IIoT devices without clustering. When the IIoT devices are clustered using Algorithm 3.1 in such a way that the SINR is maximized,

the network configuration becomes as depicted in Fig 3.4(b). Note that the devices shown in the same color in the figure are part of the same cluster. In our specific setup, we consider that each AP can have $N = 4$ clusters, and each cluster can support a maximum of $K_D = 3$ devices. The fixed power level used to execute Algorithm 3.1 is obtained as $P_0 = \frac{P_t}{(K_D+1)N}$.

Our network simulation settings are shown in Table 3.2. The path loss parameter values provided in Table 3.2 were selected in accordance with (3GPP, 2020, Table 7.2-4). We use a linear distortion model to simulate HWIs and consider $\sqrt{\sigma_t^2 + \sigma_r^2} = 0.1$ in accordance with (Matthaiou *et al.*, 2013).



a) Without clustering b) With clustering

Figure 3.4 Network configuration for 4 APs and 48 IIoT devices

In our proposed solution, each agent trains the DQN with one input layer, two fully connected hidden layers, and one output layer. The input layer N_1 contains the state elements described in Section 3.5. The number of neurons in the two hidden layers (N_2, N_3) is (64, 32). Finally, we set the output layer $N_4 = NP = 20$ outputs. The hyperparameters adopted to train the DQN are listed in Table 3.3. The weight-update time step, T_{step}, is set to 100. This indicates that every 100 time steps, the weights of the trained DQN, which are denoted by θ, are aligned with the weights of the target DQN, which are represented by θ^-. When updating θ, we employ an RMSprop optimizer with an adaptive learning rate $\alpha(t)$, where $\alpha(t) = (1-\lambda)\alpha(t-1)$. The initial learning rate was set to $\alpha(0) = 5e-3$, and λ , to 10^{-3}. This initial learning rate was determined through

Table 3.2 IIoT network's parameters

Parameter	Value
Network Parameters	
Number of AP M	4
Total number of devices	48
Number of clusters N	4
Number of devices per cluster K_D	3
Cell radius R	300 m
Small region radius r_s	50 m
Total power P_t	38 dBm
Maximum power P_{max}	30 dBm
Minimum power P_{min}	5 dBm
AWGN power σ^2	-174 dBm
Blocklength n_b	256
Bit error rate ϵ	10^{-5}
Total bandwidth B	100 MHz
Doppler Frequency f_d	15 Hz
Time slot duration T_s	20 ms
Path Loss Parameters	
Height of AP H	3 m
Height of IIoT devices h_k	1 m
Effective clutter height h_c	2 m
Carrier frequency f_c	1 GHz
Typical clutter size $d_{clutter}$	10 m
Clutter density r	0.4

Table 3.3 Hyper-parameters of DQN-based power allocation

Parameter	Value
Initial learning rate $\alpha(0)$	$5e^{-4}$
Discount factor γ	0.8
Number of episodes Z	1000
Replay memory buffer size D	5000
Mini-batch size D_b	32
Loss function	MSE
Optimizer	RMSprop
Activation function	tanh
Number of power levels	20
Time step T_{step}	100

experimentation and was found to improve learning performance in our simulation results. Furthermore, we utilize the adaptive ϵ-greedy algorithm to balance exploration and exploitation during the learning process. This algorithm adjusts the exploration rate ϵ over time to ensure the agent gradually transitions from exploration to exploitation as it learns more about the environment. The formula used to update ϵ is $\epsilon(t) = \max\{\epsilon_{\min}, (1 - \lambda_\epsilon)\epsilon(t-1)\}$, where $\epsilon_{\min}$ represents the minimum exploration probability and λ_ϵ is the decay rate. In our implementation, we initialize $\epsilon(0)$ at 0.7 and set $\epsilon_{\min}$ to 10^{-2} and λ_ϵ to 10^{-3}.

We performed training using Algorithm 3.2 for 1000 episodes to help the agent learn the optimal power allocation policy for different channel conditions. To train the DQN of the DCPM framework, experiences (i.e., datasets) were collected by iteratively interacting the DQN with a simulated IIoT environment having the parameters indicated in Table 3.2. The experiences, which take the form of (state, action, reward, and next state) tuples, are stored in a reply buffer of size (5,000) and randomly sampled in a mini-batch to iteratively train the DQN model. Note that the experiences generated include a wide range of states and actions that reflect various network conditions, devices' HWI configurations, and dynamic channel gains. This ensures that the DQN agent is exposed to a wide variety of scenarios during training, which enhances its ability to generalize and perform well in real-world situations. Once the DQN was trained, it was deployed in each AP and Algorithm 3.3 was executed. The sum rate of the ensuing numerical results is from the online execution of Algorithm 3.3 in different channel conditions. For simplicity, we consider a similar networking environment for both model training and online inference. We implemented our program in Python 3 and ran it on a 64-bit Windows 10 machine with an Intel Core i7-6700 CPU with a 3.40 GHz processor and 8 GB of RAM.

3.7.3 RSMA Versus OFDMA: Number of Clusters Versus Jain Fairness Index

The Jain fairness index quantifies the fairness of resource allocation among multiple devices in a network or system. From (Sediq, Gohary, Schoenen & Yanikomeroglu, 2013), we obtain Jain's

fairness index as:

$$\mathcal{J}(x) = \frac{(\sum_{k=1}^{K_D} x_k)^2}{K_D \sum_{k=1}^{K_D} x_k^2}, \tag{3.30}$$

where x_k is the achievable rate of the k-th device and K_D is the total number of devices allowed per RRB (i.e., device cluster). The Jain fairness index measures how fairly the bandwidth of an RRB is allocated among its associated devices, on a scale from 0 to 1 (Sediq *et al.*, 2013), with 1 representing perfect fairness and indicating that all users receive an equitable share of the available resources. Conversely, a Jain index value that approaches 0 indicates substantial unfairness, meaning that some users receive disproportionately more resources, while others are given comparatively less resources. Essentially, the Jain fairness index quantitatively measures how resources are distributed and is a valuable tool for evaluating fairness in different network and resource-sharing scenarios. Note that each AP has 4 RRBs in the setup considered. Hence, at any given time, OFDMA can support one device per RRB and a maximum of 4 devices per AP. In contrast, RSMA can support 12 devices per AP with intra-cluster interference. As a result, the sum rate is not a rational metric for comparing the two schemes. We therefore use the Jain fairness index to assess and compare the RSMA and OFDMA schemes' ability to fairly allocate resources.

Table 3.4 Jain fairness index for RSMA and OFDMA technologies

Number of clusters	Number of devices per cluster (K_D)	Fairness index for RSMA	Fairness index for OFDMA
2	6	0.361	1/6
4	3	0.437	1/3
6	2	0.603	1/2
12	1	1	1

In Table 3.4, we evaluate the number of clusters (i.e., RRBs) per AP and the fairness index for the OFDMA and RSMA schemes. It is evident from the table that increasing the number of clusters per AP decreases the fairness index. When it comes to the OFDMA scheme, each RRB (or cluster) can support only one user at a time. Hence, its fairness index is always $1/K_D$.

Meanwhile, the RSMA scheme simultaneously supports all the clustered devices with certain data rates by effectively mitigating intra-cluster interference. Accordingly, for a given number of RRBs, RSMA outperforms OFDMA when it comes to the fairness index, as Table 3.4 shows. For example, in the third scenario, where $N = 6$ and $K_D = 2$, RSMA and OFDMA achieve Jain fairness index values of 0.603 and 0.5, respectively.

In the last scenario, we consider 12 RRBs and thus 12 clusters per AP, with each cluster supporting only one device. In this context, the RSMA and OFDMA schemes align, which means they both obtain same fairness index. We conclude that the RSMA scheme allocates resources more fairly than the OFDMA scheme in scenarios where there are fewer RRBs than devices.

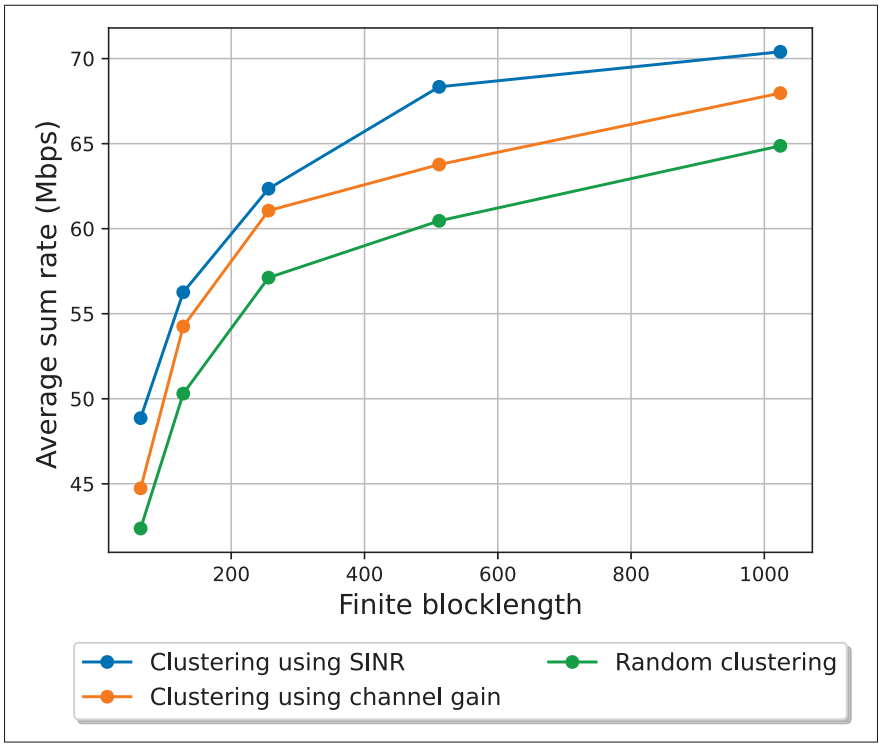


Figure 3.5 Average sum rate for different blocklengths and clustering algorithms

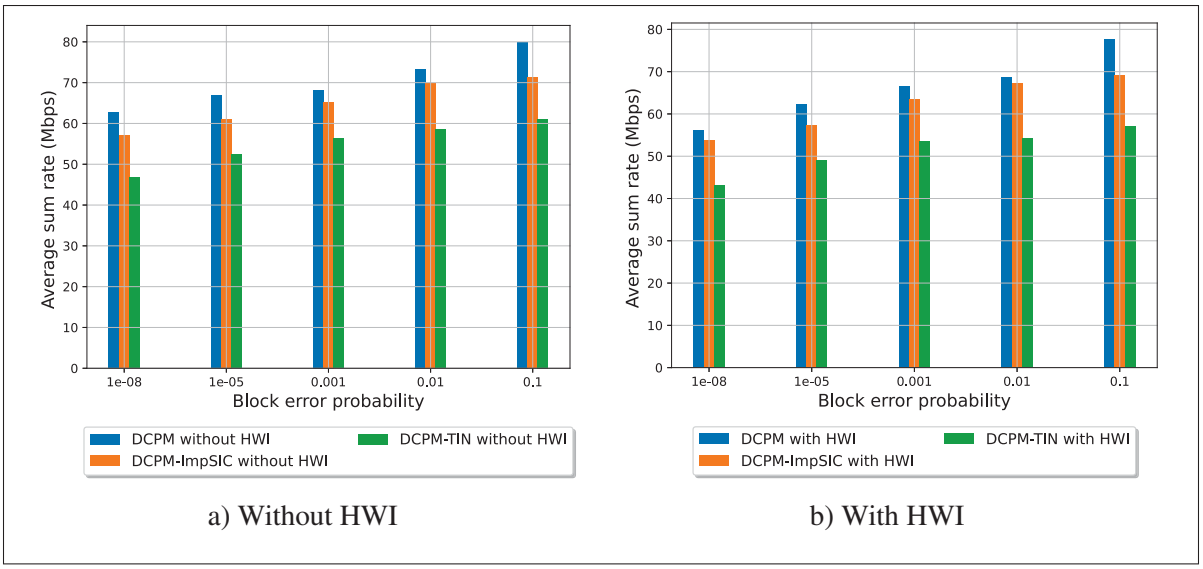


Figure 3.6 Average sum rate for different block error probability and interference management scheme

3.7.4 Average Sum Rate for Different Blocklengths and Clustering Algorithms

Fig. 3.5 plots our proposed solution' average sum rate for various blocklengths and clustering algorithms. We consider that $\epsilon = 10^{-5}$ and $n_b \in \{64, 128, 256, 512, 1024\}$. It is evident from the figure that the average sum rate increases as the blocklength value increases. For instance, the proposed DCPM framework[9] achieves an average sum rate of 48.86 Mbps and 70.4 Mbps when $n_b = 64$ and $n_b = 1024$, respectively. This observation is logically intuitive, as larger blocklengths bring the average data rate closer to the near-Shannon channel capacity. It is worth noting that our proposed DCPM framework achieves a non-negligible average rate in the short blocklength regime, which makes it a compelling scheme for URLLC scenarios.

Fig. 3.5 also highlights that our proposed clustering algorithm outperforms the benchmark clustering algorithms for both short and long blocklengths. For example, when the blocklength is $n_b = 256$, the proposed DCPM framework achieves an average sum rate that is 1.29 Mbps higher than the average sum rate of the DQN-based power allocation with clustering using channel gain algorithm and 5.23 Mbps higher than the average rate of the DQN-based power allocation with random clustering algorithm. These outcomes are anticipated, given that our proposed

[9] In Fig. 3.5, the legend entry "*Clustering using SINR*" implies with the DCPM framework.

clustering algorithm (i.e., Algorithm 3.1) is designed to cluster IIoT devices with the objective of improving the SINR and, consequently, enhancing system capacity. In contrast, Algorithm 3.4 assigns IIoT devices to the most suitable RRBs without considering interference. Hence, it does not necessarily maximize the SINR(s) and thus reduces the system capacity. Both Algorithms 3.1 and 3.4 have the same computational complexity. Meanwhile, the random clustering scheme can result in severe intra-cluster interference and consequently achieve the lowest sum rate. We therefore conclude that our proposed SINR-based device clustering algorithm has clear merit over both the channel gain-based and random device clustering schemes.

3.7.5 Average Sum Rate for Different Block Error Probability and Interference Management Schemes

Figs. 3.6(a)-(b) plot the average sum rate with respect to the block error probability considering $n_b = 256$ and $\epsilon \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-5}, 10^{-8}\}$. The average sum rate increases as ϵ increases. This is intuitively expected since $\mathrm{Q}^{-1}(\epsilon) \to 0$ as $\epsilon \to 1$ in (3.16), and consequently, the device's average data rate approaches Shannon capacity as the tolerable block error probability increases.

In Fig. 3.6(a), we also compare the average sum rate of the proposed DCPM framework with that of different interference management schemes without considering any HWI-induced distortions. Fig. 3.6(a) clearly depicts that the proposed DCPM framework outperforms both the DCPM-TIN and DCPM-ImpSIC schemes. This observation can be explained by the following arguments. First, for the imperfect SIC scheme with $\theta_c = 10^{-4}$, the SINRs of the private streams decrease in RSMA (3.12) due to uncancelled interference from the common message. This leads to a reduction in the average sum rate of the DCPM-ImpSIC scheme. Meanwhile, no interference cancellation is considered in the TIN scheme for decoding the devices' messages, and consequently, the scheme usually exhibits less capacity than RSMA. Hence, the proposed DCPM framework also outperforms the DCPM-TIN scheme. For instance, Fig. 3.6(a) depicts that at $\epsilon = 10^{-3}$, the DCPM framework achieves an average sum rate that is 2.93 Mbps higher than that of the DCPM-ImpSIC scheme and 11.9 Mbps higher than that of the DCPM-TIN scheme.

In Fig. 3.6(b), we compare the average sum rate of the proposed DCPM framework with that of different interference management schemes while taking into account the impact of HWI-induced distortions. It is worth noting that HWI-induced distortions have a detrimental effect on the performance of all schemes. For example, when HWI-induced distortions are considered, the average sum rate of the DCPM, DCPM-ImpSIC, and DCPM-TIN schemes decreases approximately 2 to 6 Mbps, 2 to 4 Mbps, and 3 to 4 Mbps, respectively. Despite the reduction in capacity, our proposed DCPM framework also consistently outperforms both the DCPM-TIN and DCPM-ImpSIC schemes in the presence of HWI-induced distortions. Overall, our proposed DCPM framework is better able to manage interference than both the DCPM-TIN and DCPM-ImpSIC schemes are.

3.7.6 Average Sum Rate for Different Total Power Values Per AP and Power Management Schemes

Fig. 3.7 compares the average sum rate of the proposed DCPM scheme with that of three benchmark schemes, namely WMMSE, RP, and MaxP, for an IIoT network with $n_b = 256$ and $\epsilon = 10^{-5}$. It is evident from Fig. 3.7 that the average sum rates of all the power allocation schemes gradually increase as the total power available at each AP increases. For instance, when the total power $P_t = 20$ dBm, the DCPM scheme achieves an average sum rate of 57.9 Mbps. However, when the total power is increased to $P_t = 42$ dBm, the DCPM algorithm's average sum rate increases to 63.44 Mbps.

Fig. 3.7 shows that our proposed DCPM algorithm achieves a higher average sum rate than the state-of-the-art transmit power allocation schemes for both small and large transmit power limits. For instance, when $P_t = 30$ dBm, the DCPM scheme's average sum rate exceeds that of the WWMSE, RP, and MaxP schemes by 5.46 Mbps, 25.61 Mbps, and 52.22 Mbps, respectively. We emphasize that the WMMSE scheme (i) is more time-consuming/computationally complex due to iterative optimization, especially in the context of large-scale networks, (ii) is primarily optimized for interference channels in the infinite blocklength regime and (iii) does not take into consideration any HWI-induced signal distortions. Due to these limitations, it not only is

sub-optimal for interference channels in FBL-coded IIoT networks but also suffers from high computational and implementation complexity. Meanwhile, the RP scheme does not guarantee any performance gain over fading channels, and the MaxP scheme generates severe intra-cell interference in the network. In contrast to these power allocation schemes, the proposed DCPM algorithm can learn and adapt to dynamic wireless channel conditions. When the devices' wireless channel conditions vary over time, the optimal power allocation strategy also changes. The DQN-based approach is able to effectively solve this problem by learning to adapt the power allocation strategy based on past experiences. Furthermore, it can choose suitable power levels to jointly mitigate interference and HWI-induced signal distortions. In essence, the proposed DCPM approach's resilience to intra-cell interference and HWI-induced signal distortions gives it clear merit over the existing transmit power allocation methods.

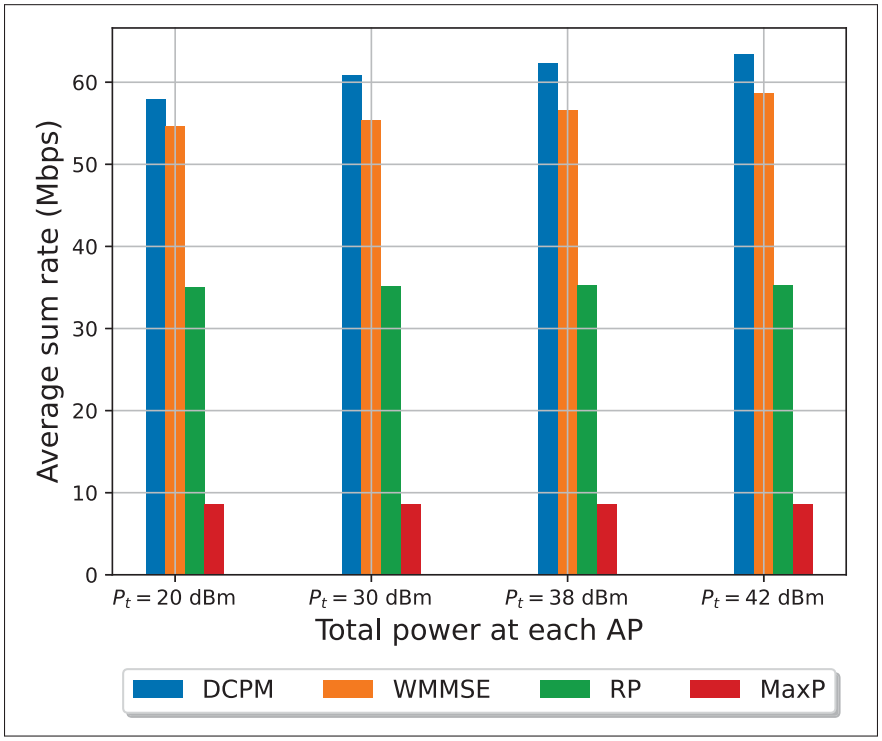


Figure 3.7 Average sum rate for different total power values per AP and power management schemes

3.7.7 Convergence and Scalability of the DCPM Algorithm

3.7.7.1 Convergence of the DCPM

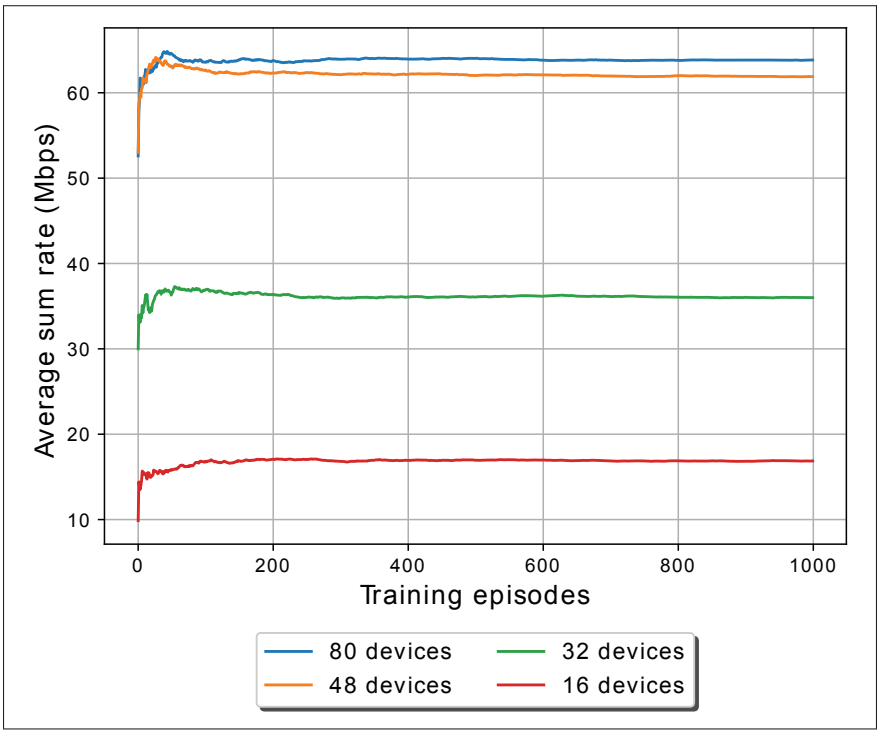


Figure 3.8 Average sum rate for different numbers of IIoT devices and numbers of training episodes

Fig. 3.8 plots the DCPM algorithm's average sum rate with respect to the number of training episodes considered in the power allocation scheme for different numbers of IIoT devices in the network. Fig. 3.8 shows that the DCPM algorithm converges to a stable system capacity in approximately 200 episodes for the device quantities considered. As a result, its convergence is guaranteed. Fig. 3.8 also shows that the average sum rate increases as the number of devices increases. This observation is expected since each AP's average rate (which is given by (3.18)) is equal to the sum of the rates of all the devices in its associated clusters. However, intra-cluster interference also increases as the number of devices per cluster increases. We emphasize that the proposed DCPM algorithm efficiently conducts device clustering and power allocation, and thereby mitigates intra-cluster interference and achieves multi-user capacity gain. Note that

we consider $N = 4$ and $k_D = 3$ for each AP, and consequently, each AP can accommodate a maximum of 12 devices. As a result, the system can support a total of 48 IoT devices at any given time. The DCPM algorithm selects the top 12 devices per AP that have the highest SINRs when there are 20 IIoT devices per AP. Hence, the DCPM scheme achieves a nearly identical system capacity in the scenarios with 48 and 80 devices. For instance, its average sum rate is 62.07 Mbps and 63.93 Mbps for the scenarios with 48 and 80 IIoT devices, respectively.

3.7.7.2 Run Time Duration of the Trained DCPM Algorithm

Table 3.5 Required time versus different numbers of IIoT devices

	4 devices	8 devices	12 devices	20 devices
Time required (ms)	16	35	62	150

In our proposed DCPM framework, the DQN model is saved after training for 1000 episodes, and the subsequent evaluations are conducted by applying the trained model in Algorithm 3.3 across various network configurations. Note that this algorithm runs in parallel in a distributed manner at each AP.

Table 3.5 shows the average amount of time required to obtain the optimal decision variables (i.e., device clustering and transmit power allocation) for a single AP and a single TS. We observe that the DCPM framework's execution time increases with the number of devices. This observation is logically intuitive, as the computational complexity of Algorithm 3.3 increases (quadratically) with the number of devices. For instance, the DCPM framework's execution time for 4 and 20 IIoT devices per AP is 16 ms and 150 ms, respectively. Note that this duration includes the time it takes to obtain all the required information (e.g., instantaneous CSI and device positions), compute the SINRs, cluster all the IIoT devices, and apply power allocation strategy. We note that all the evaluations presented in this section were conducted on a personal machine. The run time duration of the trained DCPM algorithm could be further reduced by using hardware with powerful computational capabilities, such as graphic processor units and cloud computing.

3.7.7.3 Scalability of the DCPM Algorithm

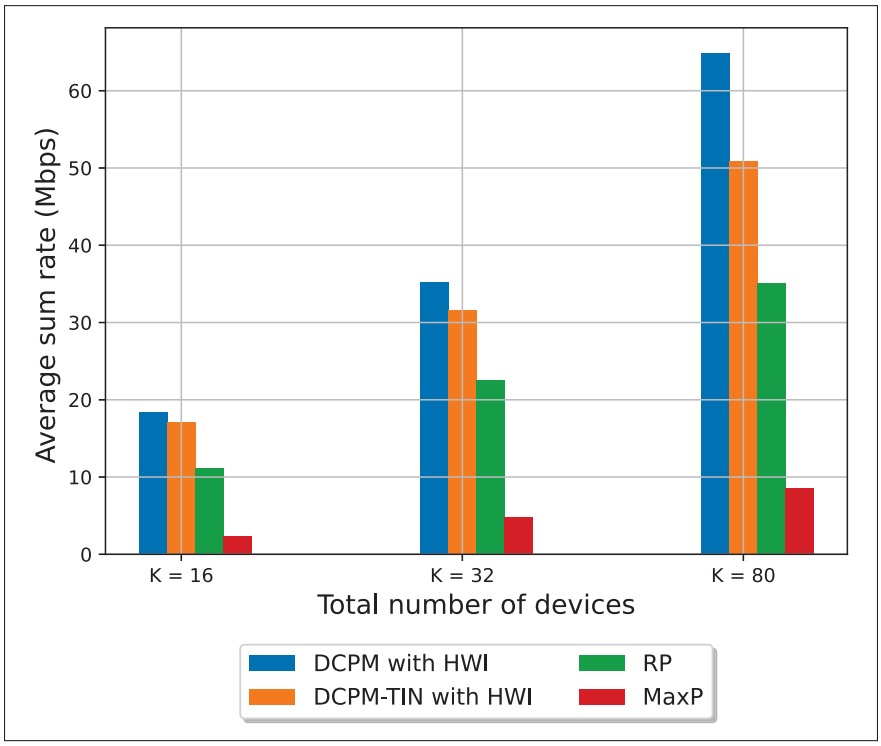


Figure 3.9 Average sum rate of DCPM and other interference management schemes for different numbers of IIoT devices

To assess the scalability of our proposed solution, we conducted simulations using the DCPM algorithm and benchmark schemes with different numbers of IIoT devices. Fig. 3.9 plots the average sum rate of different benchmark schemes with respect to the number of IoT devices while taking into account the impact of HWI-induced distortions. It is evident that our proposed algorithm outperforms the benchmark schemes in terms of average sum rate for all the device counts considered. For example, when the number of devices is set to 80, the proposed algorithm achieves an average sum rate that exceeds that of the DCPM-TIN with HWI, RP, and MaxP schemes by 13.99 Mbps, 29.8 Mbps, and 56.4 Mbps, respectively. Thus, the proposed DCPM algorithm is scalable and has a clear advantage over the benchmark schemes when it comes to mitigating interference in large-scale IIoT networks.

3.7.7.4 Impact of SIC on the DCPM Framework's Performance

Fig. 3.10 plots the DCPM algorithm's average sum rate with imperfect SIC and HWI-induced signal distortions with respect to the total number of IIoT devices considered. It is expected

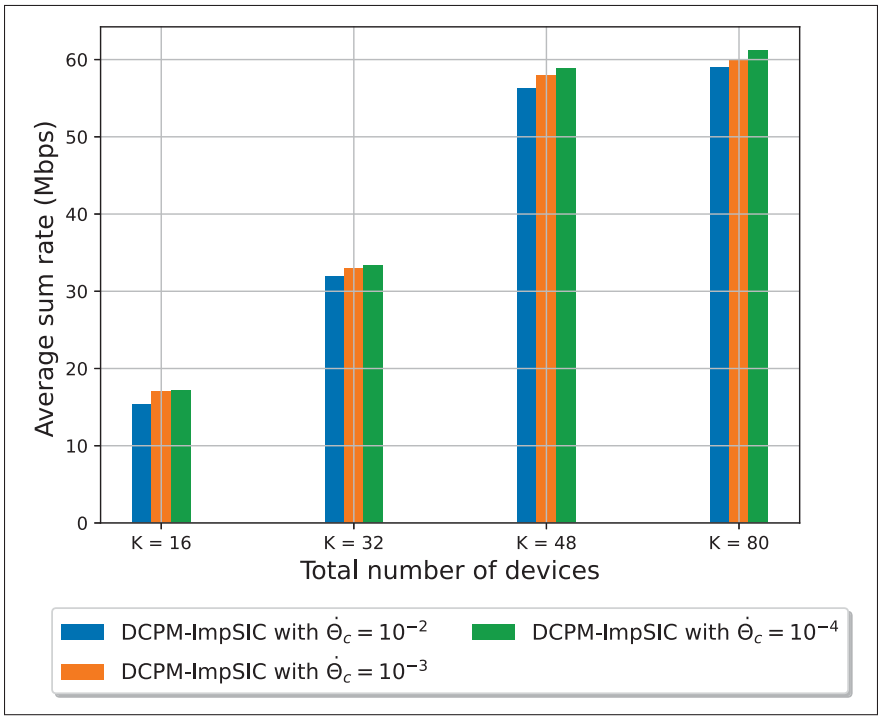


Figure 3.10 Average sum rate for different numbers of IIoT devices and different SIC capabilities

that the DCPM algorithm's average sum rate would decrease as the SIC error coefficient θ_c increases. This is because as the SIC error coefficient θ_c increases, the residual interference from the common message also increases during private message decoding at the devices. However, we observe that the imperfect SIC does not significantly deteriorate the DCPM algorithm's performance. For instance, if 80 IIoT devices are considered, the DCPM-ImpSIC with HWI scheme's average sum rate is 58.98 Mbps, 59.82 Mbps, and 61.18 Mbps when θ_c is set to 10^{-2}, 10^{-3}, and 10^{-4}, respectively. Furthermore, even when the DCPM algorithm has imperfect SIC, its average sum rate increases as the number of IIoT devices increases. Based on these observations, we can conclude that our proposed DCPM algorithm is robust to different SIC capabilities, which makes it a practical solution for RSMA-based IIoT networks.

3.7.7.5 Impact of the Number of Training Episodes on the DCPM Framework's Performance

In this section, we investigate how varying the number of training episodes affects the DCPM framework's performance. We consider $Z = \{20, 50, 200, 1000, 5000\}$ training episodes. After each specified number of episodes, the DQN model is saved and then tested in our environment. Fig. 3.11 illustrates the DCPM algorithm's average sum rate corresponding to the different numbers of training episodes. The results clearly show that increasing the number of episodes improves the proposed scheme's performance. For instance, when $Z = 20$ and $Z = 100$, the DCPM framework achieves an average sum rate of 56.2 Mbps and 59.62 Mbps, respectively. However, beyond 1000 episodes, the performance gains plateau and the DCPM algorithm's performance changes very little. Therefore, based on these observations, we limit our training episodes to $Z = 1000$ to balance training efficiency and performance improvement. This decision ensures that we achieve efficient performance without unnecessary computational overhead from additional episodes that provide diminishing returns.

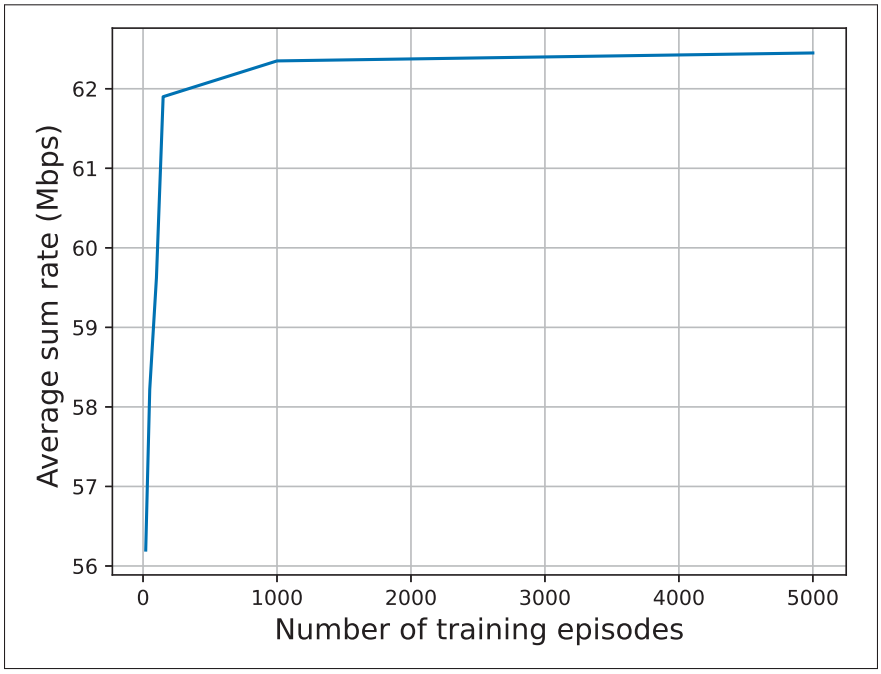


Figure 3.11 Average sum rate for different numbers of training episodes

3.7.8 Comparison of DCPM using the Sequential Technique and DCPM using Alternating Optimization

In this section, we compare the performance of the proposed sequential technique and the alternating optimization (AO) method to evaluate their performance gap. AO is renowned for iteratively refining solutions by optimizing different sets of variables having the potential to improve performance. In the AO approach, the device clusters are updated at each iteration based on the transmit powers obtained in the previous iteration. Afterward, the SINRs (with HWI-induced distortions) are estimated for the newly formed device clusters. Finally, the power allocation is updated based on the new SINR values. These three steps are iteratively repeated 1000 times for each channel fading state. After each iteration, the solutions (i.e., device clusters and power allocation) and sum rates obtained are saved. Once all the iterations have been completed, the best sum rate and its corresponding cluster and power allocation are selected. As can be seen in Table 3.6, the DCPM using the AO scheme achieves a higher average sum rate than our proposed scheme for the different values of P_{max} considered. For instance, when $P_{max} = 20$ dBm, the DCPM using the sequential technique scheme achieves an average sum rate of 45.51 Mbps, whereas the DCPM using AO scheme achieves an average sum rate of 83.48 Mbps. However, AO requires significantly more time to achieve its final solution. More specifically, the AO approach took approximately 1000 times longer than the sequential method did to obtain its average sum rates reported in Table 3.6. Note that sequential optimization takes around 62 ms when there are 12 devices per AP. For the same number of devices, the AO approach takes around 60.54 s. Clearly, this latter approach is not practically feasible for FBL-coded IIoT networks, which require fast computation for latency-sensitive control operations. Therefore, despite its improved sum rate, the AO approach lacks scalability, which makes it impractical for large-scale IIoT networks. Furthermore, it is clear from our simulation results that the gap between the approaches' performance increases as P_{max} increases. Thus, our solution is particularly suitable for scenarios with low P_{max} values.

Table 3.6 Average sum rate for different values of P_{max}

	P_{max} = 10 dBm	P_{max} = 20 dBm	P_{max} = 30 dBm
DCPM using the sequential technique	35.80 Mbps	45.51 Mbps	62.34 Mbps
DCPM using AO	42.35 Mbps	83.48 Mbps	138.36 Mbps

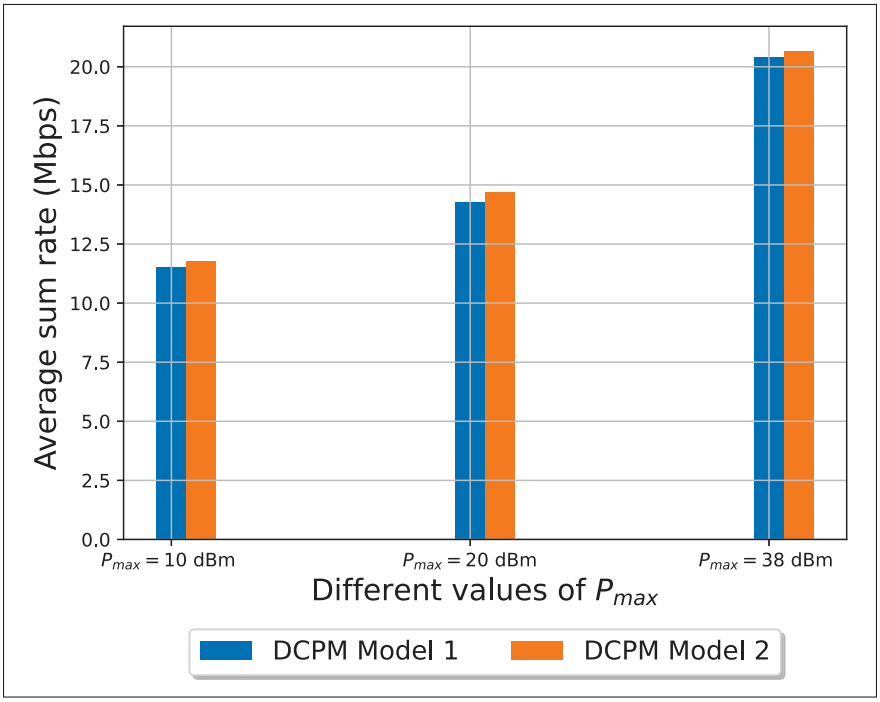


Figure 3.12 Average sum rate for different values of P_{max}

3.7.9 Performance Evaluation of the Proposed Algorithm in Unfamiliar HWI Conditions

In this section, we evaluate the DCPM algorithm's performance in the presence of another HWI model outlined in the literature (Lee, 2021). To this end, we consider the following two scenarios.

- DCPM Model 1: This model was trained using the HWI model considered in this study.
- DCPM Model 2: This model was trained using the non-linear distortion model outlined in (Lee, 2021).

After the training phase, both models were saved and subsequently tested in an environment where only the non-linear distortion model outlined in (Lee, 2021) was present. The aim of this analysis was to demonstrate that even when the DCPM model has been trained in a different environment (with alternate HWI models), it does not perform significantly worse than models that have been trained specifically for the testing environment in question. For instance, when P_{max} = 10 dBm, DCPM Model 1 and DCPM Model 2 achieve an average sum rate of 11.53 Mbps and 11.78 Mbps, respectively. Fig. 3.12 underscores that the proposed scheme is more adaptable than various HWI models considered in the literature. Based on our findings, we conclude that our proposed algorithm is generic in nature since it learns to achieve efficient resource allocation while interacting with the system. Therefore, it can effectively work with unfamiliar HWI models.

3.8 Conclusion

In this paper, we proposed a resource optimization framework to manage co-channel interference and maximize system capacity in a downlink RSMA-enabled FBL-coded IIoT network with dynamic channel variation and HWI-induced signal distortions. The proposed resource optimization problem was proven to be NP-hard, and a DCPM algorithm was proposed to address its computational intractability. The proposed DCPM algorithm first divides the IIoT devices associated with each AP into multiple non-overlapping RSMA clusters to enhance the clustered devices' SINRs. It then employs a trained DQN (obtained via MARL) in each cluster to allocate the AP's transmit power to different devices in the presence of realistic impairments (e.g., co-channel interference, HWI-induced distortions, and FBL). The proposed DCPM scheme does not require iterative optimization or instantaneous information about signal distortions for transmit power allocation. Extensive simulations confirm the DCPM algorithm can effectively maximize the capacity of the FBL-coded IIoT network. The simulation results confirm that the DCPM algorithm: (i) is able to adapt to changes in channel condition, blocklength, block error probability, and IIoT device count; (ii) is more resilient to co-channel interference and HWI-induced signal distortions than existing power allocation and interference management

schemes; and (iii) converges and can be scaled up for large-scale IIoT networks without a significant increase in computation time.

CHAPTER 4

ENERGY-EFFICIENT CLUSTERING AND POWER ALLOCATION IN RSMA-ENABLED IOT NETWORKS WITH FINITE BLOCKLENGTH CODING AND HARDWARE IMPAIRMENTS

Nahed Belhadj Mohamed[1] , Md. Zoheb Hassan[2], Georges Kaddoum[1]

[1] Electrical Engineering Department, École de technologie supérieure (ETS), 1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3
[2] Electrical and Computer Engineering Department, Université Laval, 2325 Rue de l'Université, Québec, QC G1V 0A6

Paper published in *2024 IEEE 10th World Forum on Internet of Things (WF-IoT)*, Ottawa, ON, Canada, pp. 438-443, 2024.

4.1 Abstract

As the number of IoT devices grows rapidly, ensuring EE in IoT networks is becoming a crucial concern. In dense IoT networks, scheduling multiple IoT devices over the same RRBs simultaneously causes interference that can severely degrade a network's EE. This paper investigates a resource optimization problem for IoT networks that considers both FBL-coded transmission and signal distortions caused by practical low-complexity RF front ends that lead to HWIs. We exploit RSMA to schedule multiple IoT devices over the same RRB. We propose to jointly optimize device clustering and transmit power allocation in order to enhance the network's EE while managing interference. A two-step framework is devised to solve the optimization problem in a distributed and computationally efficient manner. First, IoT devices are clustered into non-overlapping groups and the RSMA strategy is implemented in each cluster. Second, a DQN-based MARL framework is employed to near-optimally allocate transmit power among the devices in each cluster. Our simulation results shown that our proposed framework enhances the EE of a downlink RSMA IoT network notably more than state-of-the-art transmit power allocation schemes do in the presence of FBL coding and HWI-induced distortions.

4.2 Introduction

In recent years, the number of IoT devices has increased exponentially in various domains, such as healthcare, which requires low-latency communication (Mahmood *et al.*, 2022). Meeting the healthcare sector's QoS requirements, especially strict latency constraints over time-varying fading channels, presents significant challenges. FBL codes have emerged as a solution to address this issue by reducing the latency in over-the-air transmission applications. This contrasts with conventional practices that involve arbitrary long codes being transmitted to minimize error. Furthermore, the surge in IoT devices has led to RRBs being shared by numerous devices. While resource sharing makes it possible for a large number of IoT devices to be accommodated in a given bandwidth, it also introduces co-channel interference, which poses additional challenges. Furthermore, IoT devices frequently employ low-complexity RF front ends, which can lead to signal distortions due to HWIs (Chu *et al.*, 2022). Resource optimization framework that considers FBL-coded transmission, co-channel interference, and HWI-induced signal distortions is therefore necessary to improve spectrum utilization in IoT networks.

Furthermore, since IoT devices typically have limited battery capacity, EE is a crucial metric for IoT systems. EE maximization in IoT networks has been investigated in the state-of-the-art literature. In (Cao *et al.*, 2019), the authors propose a DRL-based channel and power allocation framework to improve EE in UAV-enabled IoT systems. Similarly, in (Wang, Zhang, Shen, Xu & Zheng, 2020b), the authors propose a DRL-enabled framework for joint channel assignment and power allocation to maximize the EE of uplink NOMA systems. The stochastic computation offloading problem was also investigated using DRL to maximize the EE of dynamic mobile edge computing-aided IoT systems (Ansere *et al.*, 2023). However, the state-of-the-art literature lacks practical solutions to enhance EE while simultaneously addressing co-channel interference, FBL, and HWI-induced distortions in large-scale IoT networks. In light of this fact, we propose in this work a joint clustering and power allocation framework that is able to enhance system-level, manage interference, and improve RRB utilization in dense IoT networks. The specific contributions of this work are summarized as follows.

• A RSMA-enabled and FBL-coded downlink multi-cell IoT network is envisioned. In this system, each AP utilizes dedicated orthogonal RRBs for data transmission to eliminate inter-cell interference. Furthermore, the devices associated with each AP are organized into non-overlapping clusters by channel gain. Data is transmitted to all the devices in a given cluster over the same RRB using RSMA; thus, the achievable capacity of the network is limited only by the intra-cell co-channel interference. An EE maximization problem is formulated to optimize the clusters of IoT devices and select a suitable transmit power allocation for different devices in each cluster while considering interfering channels, FBL, and HWI-induced distortions.

• The proposed optimization problem is non-convex and computationally intractable. This is addressed by decomposing the proposed optimization problem into a device clustering sub-problem and a power allocation sub-problem. The device clustering sub-problem is solved using a many-to-one matching algorithm. A DQN–based MADRL algorithm is proposed to optimize power allocation in each device cluster. The proposed algorithm is able to learn, without accurate or instantaneous knowledge of the HWIs at play, the optimal policy to guide all APs to cooperatively allocate the optimal transmit power in response to variations in channel characteristics and network conditions. A CTDE method is developed by considering each cluster as a distributed agent with partial observability of the system states. A distributed algorithm is devised to maximize the system's EE by leveraging the sub-problems' solutions.

• The proposed scheme's performance is evaluated via extensive system-level simulations. The simulation results confirm the advantages of clustering IoT devices and show that the proposed scheme can learn a suitable power allocation strategy for various networking scenarios and achieves notably higher EE than do the state-of-the-art transmit power allocation schemes considered.

4.3 System Model and Problem Formulation

4.3.1 System Overview

We consider a downlink multi-cell IoT network involving M cells, each of which has 1 AP positioned at its center and K devices distributed randomly within it. The set of APs is represented as $\mathcal{M} = \{1, .., M\}$, the set of devices, as $\mathcal{K} = \{1, .., K\}$, and the set of RRBs, as $\mathcal{N} = \{1, .., N\}$. The IoT network can accommodate a substantial number of IoT devices, which results in considerable interference and means ample resources are needed to cater to the devices. In response to these challenges, the devices in each cell are grouped into N clusters, which are denoted by $C_1^{(m)}, C_2^{(m)}, \cdots, C_N^{(m)}$, to match the number of available RRBs. All the devices in a cluster receive data from the cluster's AP over the same RRB. Hence, the network's capacity depends on mitigating inter-cell, inter-cluster and intra-cluster interference. We emphasize that mitigating inter-cell interference requires coordinating the APs. Thus, for simplicity, we consider that each AP is allocated a dedicated set of orthogonal RRBs to combat the inter-cell interference. Similarly, to combat the inter-cluster interference, we assume that each cluster is assigned to one orthogonal RRB.[1] Meanwhile, to combat the intra-cell interference, we apply a one-layer RSMA strategy to the devices that are served by the same RRB. We emphasize that this setup can be implemented with frequency planning so that neighboring cells/clusters are assigned orthogonal channels. Inter-cell/inter-cluster interference from distant APs/clusters can then be ignored due to path loss and shadowing without any performance degradation. While the aforementioned setup simplifies our analysis, it also leads to a distributed resource allocation algorithm. The one-layer RSMA technique divides the data received by each device in a given cluster into a common part and a private part. While the private parts are encoded in a set of private streams $\{s_{p,k}\}$, all the common parts are concatenated into a common message that is encoded in a common stream s_c. The s_c and $\{s_{p,k}\}$ streams then undergo linear precoding and are transmitted together over the same RRB. The signal transmitted by the m-th AP to the n-th

[1] The mitigation of both inter-cell and inter-cluster interference in an IoT network is left for future work.

device cluster is expressed as:

$$x_n = \sqrt{P_{c,n}^{(m)}}\, s_c + \sum_{n \in C_n^{(m)}} \sqrt{P_{p,k,n}^{(m)}}\, s_{p,k}, \tag{4.1}$$

where $P_{c,n}^{(m)}$ and $\{P_{p,k,n}^{(m)}\}$ denote the transmission power allocated to the common and private messages, respectively. At TS t, the signal received at the k-th device is given by:

$$y_k(t) = \sqrt{g_{k,n}^{(m)}(t)}\, x_n + \eta_x + n_x, \tag{4.2}$$

where $g_{k,n}^{(m)}(t) = |h_{k,n}^{(m)}(t)|^2 \beta_{k,n}^{(m)}$ represents the channel gain from the m-th AP to the k-th IoT device in the n-th cluster, with $h_{k,n}^{(m)}(t)$ representing small-scale complex fading (Nasir & Guo, 2019) and $\beta_{k,n}^{(m)}$ denoting large-scale fading. The variable n_x is AWGN with variance σ^2, and η_x is distortion noise from HWIs. Theoretical measurements have shown that $\eta_x \sim CN(0, d^2 g_{m,n}^k(t) P_{total,n})$ (Mohamed, Hassan & Kaddoum, 2023), where $P_{total,n}$ is the total power provided by the AP to the n-th cluster and d is a positive real-number. In the RSMA strategy, each receiver decodes the common messages by treating the interference from private streams as noise and removes this interference from the common data using SIC. At the k-th device, the received SINRs of the common and private streams are expressed as:

$$\gamma_{c,k}^{(n)} = \frac{P_{c,n}^{(m)}\, g_{k,n}^{(m)}(t)}{\sum_{k=1}^{|C_n^{(m)}|} P_{p,k,n}^{(m)}\, g_{k,n}^{(m)}(t) + d^2 g_{k,n}^{(m)}(t) P_{total,n} + \sigma^2} \tag{4.3}$$

and

$$\gamma_{p,k}^{(n)} = \frac{P_{p,k,n}^{(m)}\, g_{k,n}^{(m)}(t)}{\sum_{\substack{k'=1 \\ k' \neq k}}^{|C_n^{(m)}|} P_{p,k',n}^{(m)}\, g_{k,n}^{(m)}(t) + d^2 g_{k,n}^{(m)}(t) P_{total,n} + \sigma^2}, \tag{4.4}$$

respectively.

When FBL is taken into account, the common stream rate for the k-th device is expressed as (He *et al.*, 2021):

$$\mathrm{R}_{c,k}^{(n)} = \log_2(1+\gamma_{c,k}^{(n)}) - \frac{\mathrm{Q}^{-1}(\epsilon)}{\sqrt{n_b}}\sqrt{\mathrm{V}(\gamma_{c,k}^{(n)})}, \tag{4.5}$$

where $\mathrm{V}(\gamma_{c,n})$ is defined as:

$$\mathrm{V}(\gamma_{c,k}^{(n)}) = 1 - \frac{1}{(1+\gamma_{c,k}^{(n)})^2}, \tag{4.6}$$

and n_b and ϵ represent the blocklength (in bits) and the block error probability, respectively. The function $Q^{-1}(.)$ represents the inverse of the Gaussian Q-function, which is expressed as $Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty e^{-\frac{t^2}{2}}dt$. It is important to note that since each receiver must decode the common message, the rate of the common stream, $\mathrm{R}_c^{(n)}$, is determined by the lowest common rate of all the receivers. In other words, $\mathrm{R}_c^{(n)} = \min[\ \mathrm{R}_{c,1}^{(n)}, \cdots, \mathrm{R}_{c,|C_n^{(m)}|}^{(n)}]$.

Each device decodes its private message by treating the interference from the other devices in the cell as noise after the common message's interference has been eliminated using SIC. The rate of the private stream, $\mathrm{R}_{k,p}^{(n)}$, is obtained by substituting $\gamma_{p,k}^{(n)}$ into (4.5). The sum rate of the n-th device cluster in the m-th AP is determined as follows:

$$\mathrm{R}_{C_n}^{(m)} = \frac{B}{MN}\left(\mathrm{R}_c^{(n)} + \sum_{k\in C_n^{(m)}} \mathrm{R}_{k,p}^{(n)}\right). \tag{4.7}$$

The EE of the n-th device cluster in the m-th AP is defined as:

$$\mathrm{EE}_n^{(m)} = \frac{\mathrm{R}_{C_n}^{(m)}}{P_{total,c} + p_c}, \tag{4.8}$$

where p_c is the amount of power consumed by the device itself. Finally, the m-th AP's overall EE is obtained as $\mathrm{EE}^{(m)} = \sum_{n=1}^{N} \mathrm{EE}_n^{(m)}$.

4.3.2 Problem Formulation

The joint clustering and transmit power allocation optimization problem is formulated as (4.9). The objective function aims to maximize the overall EE of the system subject to the following constraints.

$$\text{P0:} \quad \max_{\boldsymbol{P},\{C_n^{(m)}\}} \sum_{m=1}^{M} \text{EE}^{(m)}$$

$$\text{s.t.} \begin{cases} \text{C1: } P_{min} \leq P_{c,n}^{(m)} \leq P_{max}, \forall n \in \mathcal{N}, m \in \mathcal{M} \\ \text{C2: } P_{min} \leq P_{p,k,n}^{(m)} \leq P_{max}, \forall k \in C_n^{(m)}, n \in \mathcal{N}, m \in \mathcal{M} \\ \text{C3: } \sum_{n=1}^{N} (\sum_{k \in C_n^{(m)}} P_{p,k,n}^{(m)} + P_{c,n}^{(m)}) \leq P_t, \forall k \in C_n^{(m)}, n \in \mathcal{N} \\ \text{C4: } |C_n^{(m)}| \leq K_D, \forall n \in \mathcal{N}, m \in \mathcal{M} \\ \text{C5: } C_n^{(m)} \cap C_t^{(m)} = \varnothing, \forall n, t \in \mathcal{N}, n \neq t, m \in \mathcal{M} \end{cases} \tag{4.9}$$

Constraints (C1) and (C2) ensure that the transmit power of the common and private messages for each device is bounded by P_{min} and P_{max}. Constraint (C3) limits the total power provided by each AP to ensure it is less than or equal to the AP's maximum capacity. Constraint (C4) restricts the number of devices in each cluster to be no more than K_D. Lastly, constraint (C5) ensures that the device clusters with the same AP do not overlap. The problem in (4.9) is a non-convex optimization problem and is provably NP-hard. Hence, it cannot be solved by applying conventional optimization techniques. To solve it we decompose it into two sub-problems, namely, a device clustering one (P1) and a power allocation one (P2), as follows.

$$\text{P1:} \quad \max_{\{C_n^{(m)}\}} \sum_{m=1}^{M} \text{EE}^{(m)} \quad \text{s.t. C4, C5.} \tag{4.10}$$

$$\text{P2:} \quad \max_{\mathbf{P}} \sum_{m=1}^{M} \text{EE}^{(m)} \quad \text{s.t. C1, C2, C3.} \tag{4.11}$$

The solutions to sub-problems P1 and P2 are provided in Sections 4.4.1 and 4.4.2, respectively. Finally, a distributed algorithm to solve P0 is proposed in Section 4.4.3.

4.4 Proposed Solutions

4.4.1 Solution to P1: Device Clustering

To cluster the IoT devices, we consider Algorithm 3.4. The computational complexity of this algorithm is dominated by Step 5, which has a worst-case complexity of $\mathcal{O}\left(|\mathbf{UM}_R|\right)$ at each iteration. Since $|\mathbf{UM}_R|$ is reduced by 1 at each iteration, the overall complexity of Algorithm 3.4 is $\mathcal{O}\left(\sum_{n=1}^{K} n\right) \approx \mathcal{O}\left(K^2\right)$.

4.4.2 Solution to P2: Transmit PA

The DQN algorithm is applied to each cluster individually to determine the power level of all the devices in a cluster. The key components of the proposed scheme are:

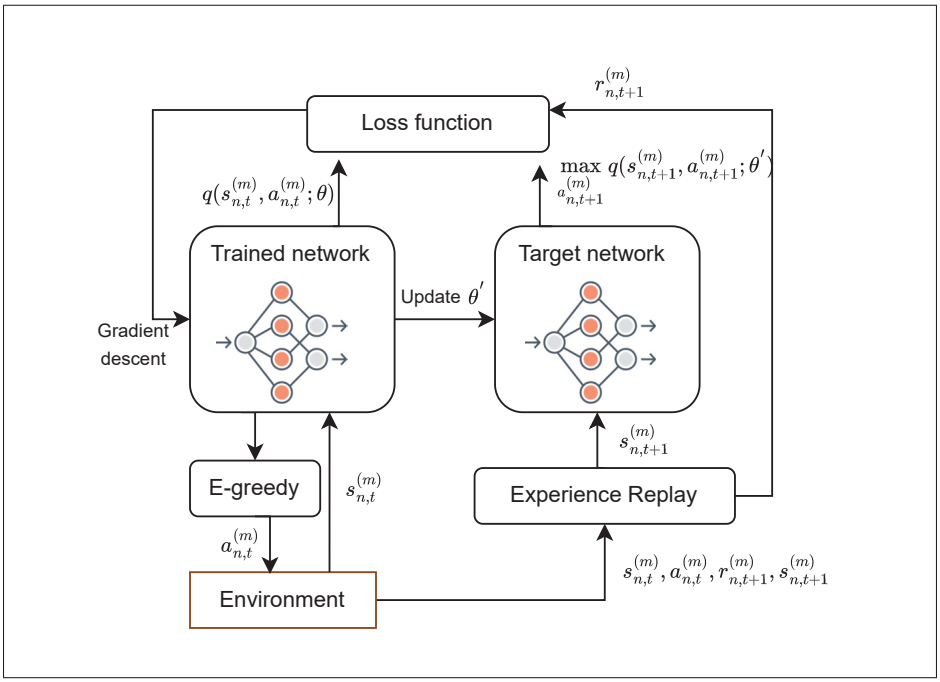


Figure 4.1 Framework of deep Q-network in our system

1. **Agents:** Each device cluster represents a learning agent.
2. **States:** Our state space contains (i) the local CSI at TS t, (ii) the SINRs of the common streams, and (iii) the SINRs of the private streams. Therefore, the state space is defined as:

$$s_{n,t}^{(m)} = \left\{ (g_{1,n}^{(m)}, ..., g_{|C_n^{(m)}|,n}^{(m)}), (\gamma_{c,1}^{(n)}, \gamma_{c,2}^{(n)}, ..., \gamma_{c,|C_n^{(m)}|}^{(n)}), (\gamma_{p,1}^{(n)}, \gamma_{p,2}^{(n)}, ..., \gamma_{p,|C_n^{(m)}|}^{(n)}) \right\}. \quad (4.12)$$

3. **Actions:** Our action space contains discrete power levels between P_{min} and P_{max}.
4. **Rewards:** Our reward function is the system's EE.

Fig. 4.1 shows the architecture of our proposed solution. In each TS, the RL agent observes the states of the clustered devices $s_{n,t}^{(m)}$ and then allocates the AP's transmit power level $a_{n,t}^{(m)}$ to the clustered devices. Our proposed solution incorporates a centralized trainer that compiles all the RL agents' knowledge and stores it in the experience replay memory so that it can be trained on the memory and then develop global policy to ensure stability. We consider a parameter sharing approach in which a single agent is trained and its parameters are shared with all the other agents rather than training agents in parallel. Although each agent has the same set of parameters, they have different actions since they select their actions independently and their observed states are different. This implementation accelerates the learning process and reduces the amount of computational resources required for training. As shown in Fig. 4.1, each agent's experience at TS t is stored in the replay memory D. First, the single agent samples a mini-batch of experience from D to calculate the loss function in accordance with (1.6). Then, it uses the optimizer to minimize the loss function and update the weights of the trained network using the backpropagation method. Finally, the target DQN's weights are updated with the new weights of the trained DQN every T_{step}.

4.4.3 Overall Algorithm to Solve P0

Algorithm 4.1 shows the pseudocode of the overall algorithm to solve P0 in a distributed manner. At each TS, the APs compile the instantaneous channel gains of their respective IoT devices. Then, each AP forms N device clusters by employing Algorithm 3.1. Following cluster formation,

Algorithm 4.1 Overall Proposed Algorithm

Input: M; K and N ; K_D; trained DQN for power allocation $q^*(s, a; \theta)$; total number of TSs T_s.
Initialize: TS index $t = 1$.
while $t \leq T_s$ **do**
  for $m \leftarrow 1 : M$ **do**
    Collect the channel gains for the associated IoT devices.
    Determine N device clusters using Algorithm 3.4.
    for $n \leftarrow 1 : N$ **do**
      Acquire the state information $s_{n,t}^{(m)}$ described in Section 4.4.2.
      Determine the optimal power allocation action using the trained DQN for all the common and private messages in n-th device cluster.
    end for
    Scale the transmit power of the common and private messages of all the device clusters such that constraint C3 is respected.
  end for
  Perform downlink data transmission at all cells using the optimized device clusters and PA; increase the TS index $t = t + 1$.
end while
Output: Device clusters and transmit power allocation solution to P0 at each TS.

the APs utilize the trained DQN agent to determine how much transmit power to allocate to the common and private messages in their respective clusters. After power allocation, the APs perform downlink data transmission to their associated devices.

4.5 Performance Evaluation

Our simulation setup involves four cells, each of which has 1 AP at the center and 12 devices distributed randomly within it. The default simulation settings are presented in Table 4.1. We consider that each AP can accommodate four clusters, each of which is capable of supporting three devices using the RSMA strategy. We defined the path loss model in accordance with 3GPP's urban micro (UMi) model (3GPP, 2020). We selected the parameters in Table 4.1 in accordance with (Nasir & Guo, 2019).

In our proposed solution, each agent trains the DQN with one input layer, three fully connected hidden layers, and one output layer. The input layer N_1 contains the state elements. The hidden layers N_2, N_3, and N_4 contain 256, 128, and 64 neurons, respectively. Finally, we set $N_5 = |\mathcal{A}| = 20$ as the number of outputs. The hyperparameters adopted to train the DQN are shown in Table 4.1.

Table 4.1 Simulation settings

Parameter	Value
Network Parameters	
Number of AP M	4
Total number of devices per AP	12
Number of clusters N	4
Number of devices per cluster K_D	3
Cell radius R	300 m
Small region radius r_s	50 m
Total power P_t	38 dBm
Maximum power P_{max}	30 dBm
Minimum power P_{min}	5 dBm
Power consumed by the device p_c	20 dBm
AWGN power σ^2	-174 dBm
Blocklength n_b	256
Block error probability ϵ	10^{-5}
Total bandwidth B	100 MHz
Doppler Frequency f_d	15 Hz
Time interval T_s	20 ms
Level of impairment d	0.1
Path Loss Parameters	
Height of AP H	3 m
Height of IIoT devices h_k	1 m
Effective clutter height h_c	2 m
Carrier frequency f_c	1 GHz
Typical clutter size $d_{clutter}$	10 m
Clutter density r	0.4
DQN's Hyper-Parameters	
Initial learning rate $\alpha(0)$	$1e^{-3}$
Discount factor γ	0.9
Episode number Z	2000
Replay memory buffer size D	5000
Mini-batch size D_b	32
Optimizer	RMSprop

4.5.1 Importance of Including the SINRs in the State Space

This section shows the importance of including the SINRs of the common and private streams in our state space. We illustrate this by exploring two distinct scenarios. In the first scenario, we

utilize the state space as defined in (4.12), which incorporates the SINRs and the CSI. Conversely, the second scenario includes only the CSI in the state space.

Fig. 4.2 compares the EE of both scenarios at different levels of impairments. Our proposed scheme outperforms the DQN-based power allocation approach with only CSI in the state space. For example, when $d = 0$, the proposed solution makes the system 2.24 Mbps/W more EE than does the DQN-based power allocation approach with only CSI in the state space. Additionally, increasing the impairment level adversely affects EE in both scenarios, as it is depicted in Fig. 4.2. However, these issues can be managed by incorporating the SINRs in the state space. For instance, when $d = 0.05$, $d = 0.1$, and $d = 0.2$, the proposed scheme achieves an EE that is 2.8 Mbps/W, 3.75 Mbps/W, and 3.2 Mbps/W higher than that achieved by the DQN-based power allocation approach with only CSI in the state space, respectively. We emphasize that including the SINRs in the state space helps the DQN agent capture the impact of co-channel interference and HWI-induced distortions, enabling it to learn an effective policy to mitigate them.

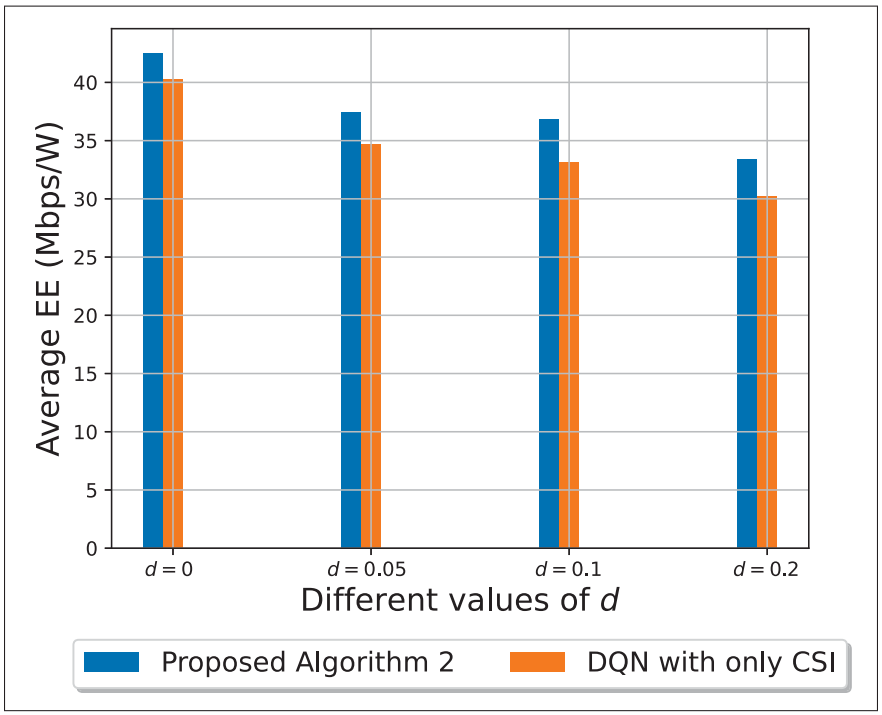


Figure 4.2 EE versus different levels of impairments

4.5.2 Advantage of Clustering the IoT Devices

Here, we explore the benefits of dividing the IoT devices associated with each AP into non-overlapping device clusters. To this end, we compare our proposed approach with a scenario in which all the IoT devices in a given cell are served over the same RRB without clustering. It is worth noting that clustering the IoT devices has a clear advantage in terms of the average EE of the entire system. This observation is to be expected, as serving more IoT devices over the same RRB leads to increased intra-cell interference. For example, when $\epsilon = 10^{-3}$, the proposed framework achieves an average EE that is 35.18 Mbps/W higher than that achieved by the scenario without clustering. Furthermore, it is clear from Fig. 4.3 that the average EE increases as ϵ increases. For instance, the proposed framework achieves an average EE of 40.62 Mbps/W when $\epsilon = 10^{-1}$ and 30.551 Mbps/W when $\epsilon = 10^{-8}$.

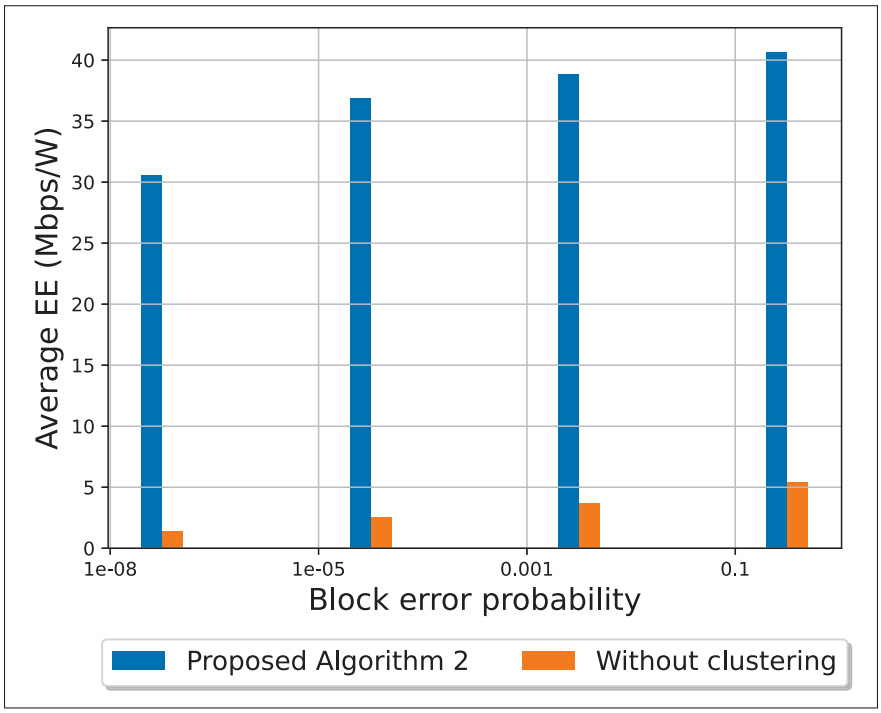


Figure 4.3 EE for different block error probability

4.5.3 EE Versus Training Episodes for Different p_c Values

Fig. 4.4 illustrates the average EE achieved with different numbers of training episodes and p_c values. Intuitively, the EE decreases as the p_c value increases. For instance, the proposed scheme achieves an average EE of 40.94 Mbps/W when p_c = 10 dBm, 34.519 Mbps/W when p_c = 20 dBm, and 21.26 Mbps/W when p_c = 30 dBm. Furthermore, Fig. 4.4 shows that the proposed framework converges to a stable system EE in approximately 100 episodes for all the p_c values considered. Its convergence is therefore guaranteed.

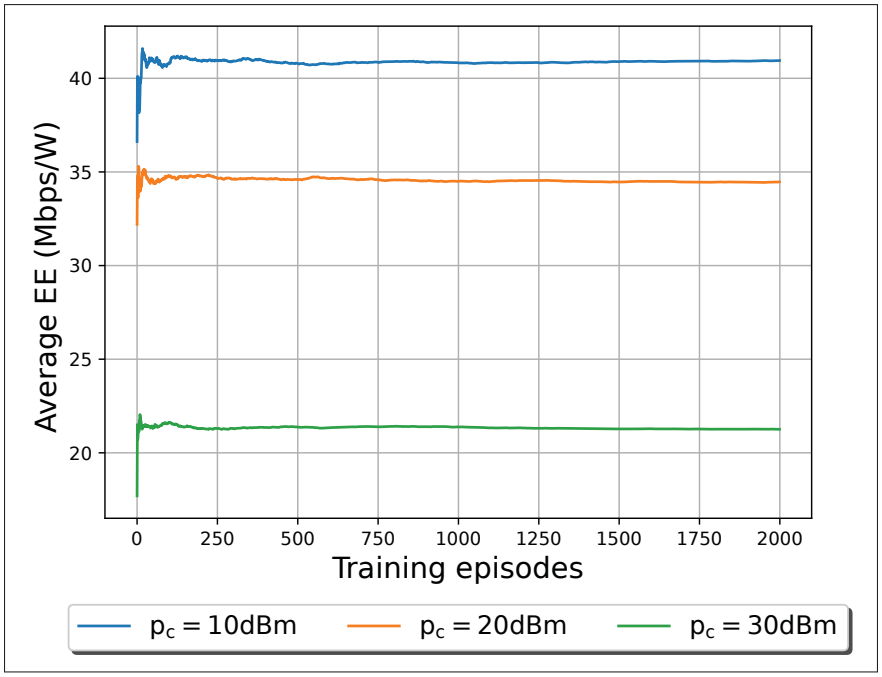


Figure 4.4 EE versus training episodes for different p_c values

4.5.4 EE Comparison with the Benchmark Schemes Considered

Fig. 4.5 compares the EE of the proposed scheme and three benchmark power allocation schemes for IoT networks, namely, WMMSE (Nasir & Guo, 2019), RP, and MaxP (Mohamed *et al.*, 2023). Algorithm 3.1 is applied to cluster the IoT devices for all four schemes considered. In the RP scheme, the transmit power of each device is randomly selected between 0 and P_{max}, whereas in

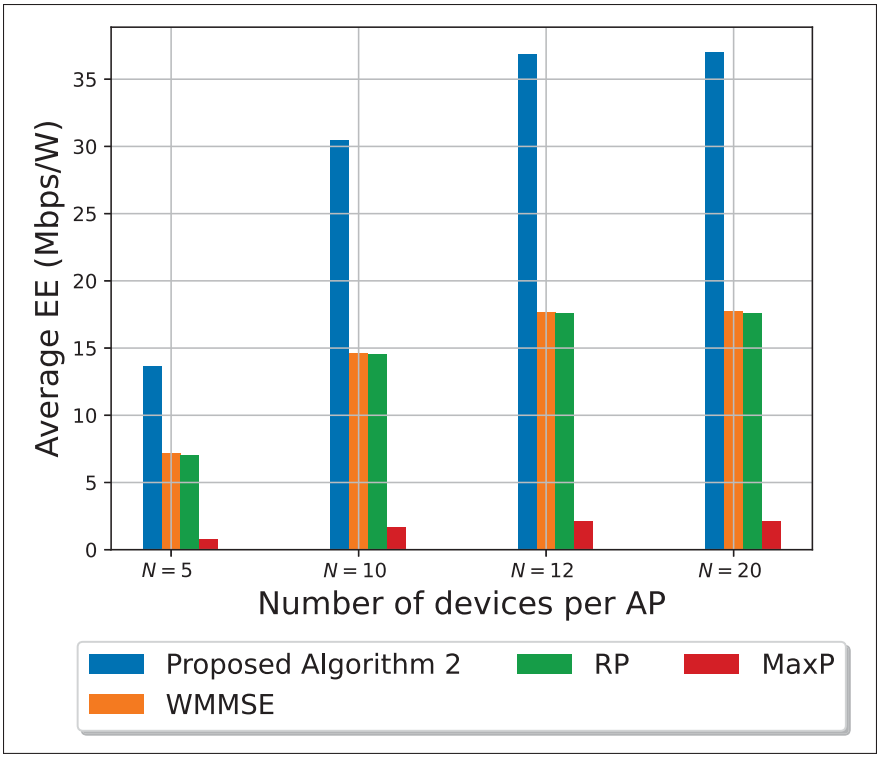


Figure 4.5 EE of different power allocation approaches versus the number of devices

the MaxP scheme, each device is allocated the maximum transmit power P_{max}. It is evident from Fig. 4.5 that our proposed scheme achieves higher EE than the benchmark schemes considered regardless of the number of devices. This is due to its resilience to intra-cell interference and HWI-induced distortions. For instance, when $N = 10$, the proposed scheme achieves an EE that exceeds that achieved by the WMMSE, RP, and MaxP schemes by 15.86 Mbps/W, 15.96 Mbps/W, and 28.85 Mbps/W, respectively. We emphasize that the WMMSE approach can achieve a near-optimal solution only for interference channels with infinite blocklength. HWI-induced signal distortion also needs to be accurately measured in the WMMSE approach. In FBL-coded IoT networks, the WMMSE approach is not only sub-optimal for interference channels overall, but it is also quite computationally complex. As for the other benchmark schemes, the MaxP strategy significantly increases intra-cell interference in the network, and the RP scheme offers no guarantee of performance across fading channels. Furthermore, it can be observed in Fig. 4.5 that increasing the number of devices increases the EE. For example, with 5 and 20 IoT devices per AP, the proposed framework achieves an average EE of 13.63 Mbps/W

and 37.02 Mbps/W, respectively. Overall, our proposed algorithm is able to maximize the EE of large-scale IoT networks in a computationally efficient manner.

4.6 Conclusion

In this paper, we proposed a distributed device clustering and power allocation framework to maximize the EE of a downlink RSMA-enabled and FBL-coded IoT network with dynamic channel variation and HWI-induced distortions. The proposed framework first groups the IoT devices into non-overlapping clusters based on their channel gain and then applies a multi-agent DQN to perform power allocation in each cluster. The DQN-based power allocation scheme does not require iterative optimization or prior knowledge of signal distortions, which makes it particularly efficient in IoT systems. Our simulation results demonstrate that our proposed scheme is more resilient to co-channel interference and HWI-induced distortions than state-of-the-art transmit power allocation schemes, especially in large-scale IoT networks.

CHAPTER 5

DIGITAL TWIN-DRIVEN CONTINUAL DEEP REINFORCEMENT LEARNING FOR COEXISTENCE OF MULTIPLE RADIO ACCESS TECHNOLOGY IOT LINKS WITH NONLINEAR RECEIVERS

Nahed Belhadj Mohamed[1] , Georges Kaddoum[1], Md. Zoheb Hassan[2]

[1] Electrical Engineering Department, École de technologie supérieure (ETS), 1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3
[2] Electrical and Computer Engineering Department, Université Laval, 2325 Rue de l'Université, Québec, QC G1V 0A6

Paper Accepted for publication to *IEEE Transactions on Machine Learning in Communications and Networking*, 24 February 2026.

5.1 Abstract

This paper investigates the coexistence of downlink IoT links enabled by multiple RATs, including LTE and 5G NR. The coexistence of multiple RAT IoT links is significantly challenged by ACI and HWI that arise from practical low-complexity radio-frequency front ends. To mitigate these challenges, we propose a radio resource optimization scheme that dynamically adjusts link adaptation parameters (transmit power, modulation, and coding rate) to maximize overall throughput while explicitly accounting for ACI and HWI. However, the proposed optimization is an NP-hard MINLP problem that requires global CSI and centralized optimization, making it impractical for large-scale, dynamic multi-RAT IoT networks. To enable distributed optimization under ACI and HWI, we reformulate the problem as a Markov game and develop a MADRL framework that derives equilibrium link adaptation policies from local observations. Direct DRL training in real networks, however, incurs high communication overhead and can create adverse effects due to the random explorations. To overcome these limitations, we introduce a context-aware DTN that provides a safe and efficient virtual environment for training. In particular, we propose a novel DTN-empowered MADRL scheme that employs a replay memory-based continual model updating strategy, enabling policies to be learned from DT-generated experiences and periodically refined with real network data. This approach alleviates the need for frequent physical network interactions and significantly reduces communication overhead.

Extensive simulations demonstrate that the proposed framework is scalable, computationally efficient, and robust in dynamic IoT environments, while outperforming 3GPP-standardized link adaptation in the presence of non-negligible ACI and HWI.

5.2 Introduction

The IoT has witnessed rapid growth in recent years, with an ever-expanding number of interconnected devices and objects. With the growth in the number of IoT devices, the demand for higher data rates, bandwidth, capacity, and throughput has increased exponentially (Shafique *et al.*, 2020). This rapid expansion underscores the critical need for advanced multi-RAT systems—such as LTE and 5G NR—to provide seamless, efficient, and reliable connectivity across diverse IoT applications (Andrews *et al.*, 2014).

In large-scale IoT networks, where multiple RATs operate in close proximity, strong ACI may occur when links operate on adjacent channels. This interference is caused by the non-linear response of the RF front end, which produces distortion and intermodulation, degrading signal quality and receiver sensitivity. Notably, when IoT links employ spectrum-agile devices with wideband pre-selection filters to operate across multiple RATs and frequency bands, they may receive multiple unwanted signals from adjacent channels (Mohammadi, AlQwider, Rahman & Marojevic, 2022). These interfering signals can originate from the same or different RATs and, due to the receiver's inherent non-linearity, the resulting interference can be significant—particularly in dense IoT networks, where it is non-trivial to maintain a large distance between adjacent channel transmitters and victim receivers. Besides the ACI, IoT devices frequently use low-complexity RF front ends that induce signal distortions due to HWIs. These HWIs originate from the impact of the phase noise, quantization error, amplifier non-linearity, and so on (Chu *et al.*, 2022).

Both ACI and HWI-induced distortions can considerably reduce the achievable throughput. In such scenarios, conventional model-based optimization methods experience difficulties for several reasons. First, these impairments make the optimal resource allocation in large-scale IoT networks

computationally intractable (i.e., NP-hard) (Mohamed, Hassan & Kaddoum, 2024). Second, conventional iterative optimization methods (e.g., successive convex approximation) usually require significant computation overhead, long converge time, and high energy consumption, particularly in scenarios with large numbers of IoT devices. Third, these methods frequently rely on analytical models to capture ACI- and HWI-induced distortions. However, existing models may not accurately reflect impairments in dynamic IoT systems. For example, classical Rapp (Jayati & Sipan, 2020) and Saleh (Shammasi & Safavi, 2012) models are derived under idealized assumptions, including memoryless behavior, device-agnosticity, and static operating conditions (Pedro & Maas, 2005). In contrast, HWI and ACI in practical IoT networks are vendor- and manufacturer-specific, and may vary with device conditions such as aging, operating temperature, and hardware features. Consequently, in real-world IoT networks, optimization approaches that rely on predefined analytical impairment models while overlooking these aspects may experience severe performance degradation.

In this context, DRL emerges as a promising paradigm to tackle complex and non-convex optimization in IoT networks by learning efficient resource allocation policies from environmental observations. However, pre-trained DRL models are susceptible to catastrophic forgetting when adapting to new scenarios (Davaslioglu, Kompella, Erpek & Sagduyu, 2024). To overcome this concern, a viable solution is continual DRL that trains the DRL model using both past and new experiences (stored in the experience replay memory), ensuring knowledge retention and adaptability. However, training such algorithms directly in real-world environments not only requires significant communication overhead, but also poses significant risks due to the potential for erroneous decisions during exploration (Guo, Tang & Kato, 2023b). To mitigate these challenges, a transformative solution that has recently emerged is the concept of a DT. A DTN provides a virtual replica of the physical network, enabling realistic simulations of multi-RAT IoT scenarios. DTNs can serve as controllable and trustworthy virtual environments to train DRL algorithms while eliminating the high costs of frequent interactions with the physical environment and the risk of disrupting network operations during exploration. This

approach enhances practicality and performance of continual learning-based resource allocation in dynamic IoT networks.

5.2.1 Contributions and Paper Organization

Emerging downlink IoT applications, such as real-time video surveillance, industrial automation, and augmented reality, require high downlink data rates and stringent reliability. Meeting these requirements with legacy systems like NB-IoT or a single RAT is challenging, especially for dense IoT environments. Accordingly, it becomes imperative to integrate multi-RATs, as doing so enables flexible spectrum usage, higher aggregate throughput, and robust connectivity across heterogeneous IoT scenarios.

In this paper, we investigate the coexistence of multi-RAT-enabled IoT systems, a key enabler of next-generation Internet-of-Everything (IoE) networks that must support diverse QoS demands and an exponentially growing number of IoT links (Sarkar *et al.*, 2025). Recent studies demonstrated the importance ensuring coexistence between multiple RATs in the emerging cellular bands (International Telecommunication Union (ITU), 2024), (Baldesi, Restuccia & Melodia, 2022). Nevertheless, managing the large-scale coexistence of multi-RAT IoT links is confronted by the following challenges: (**C1**) insufficient physical distance between links operating on adjacent channels, particularly when unlicensed services are involved; (**C2**) the deleterious impact of ACI and HWI resulting from non-linear hardware commonly used in cost-constrained IoT devices; and (**C3**) the absence of centralized mechanisms and global network information required to optimize the transmission parameters of distributed multi-RAT IoT links at scale.

To address these challenges, in this work, we propose a DTN-enhanced and MADRL-empowered decentralized radio resource optimization framework to facilitate the harmonious coexistence of multi-RAT IoT links. In the proposed framework, each IoT link determines its suitable link adaptation parameters—namely, transmit power, modulation, and coding rate—based on its local observation. To the best of our knowledge, our study is the first to consider both ACI and

HWI-induced distortions in the radio resource optimization of multi-RAT IoT systems. The specific contributions of this work are summarized below.

• **Design of DTN of multi-RAT IoT networks:** A DTN for a multi-RAT IoT system is designed to enhance resource allocation in dynamic and heterogeneous environments. Its key features include: (1) network topology modeling; (2) awareness of time-varying fading channels in multi-RAT IoT networks; and (3) incorporation of non-linear device characteristics to model ACI- and HWI-induced distortions.

• **Radio resource management (RRM) problem formulation:** An optimization problem is formulated to maximize the long-term sum throughput in downlink multi-RAT IoT networks by optimally selecting link adaptation parameters (power level, modulation order, and coding rate) for the coexisting IoT links. Solving this RRM problem is challenged by: (1) computational intractability due to its NP-hardness and MINLP structure, and (2) the requirement of global CSI[1], which is impractical to acquire in real time, particularly in large-scale networks (Nasir & Guo, 2019).

• **Distributed resource allocation policy:** To address these challenges and enable a distributed solution, the RRM problem is reformulated as a Markov game, and a MADRL approach is employed to learn the equilibrium policy. In this framework, each IoT link uses a trained DRL model to select appropriate link adaptation parameters based solely on its local state [2], without requiring global network information. This design makes the solution well-suited for large-scale multi-RAT IoT networks with partial observability.

• **DTN-assisted continual DRL framework:** Since training a DRL model directly in live networks is costly and impractical due to large exploration overhead, we exploit DTN as a reliable virtual environment for training and online updating of the DRL model. We propose a

[1] Global CSI comprises two components: (i) the individual CSI of each IoT link and (ii) the CSI of adjacent interfering links.

[2] The proposed DRL-based link-adaptation framework is generic and can be applied to any wireless system where standard-form of channel quality feedback (such as CSI and SINR) is available.

DTN-empowered, replay memory–based continual learning framework that first learns an initial policy from DT-generated experiences and then periodically refines the policy using experiences collected from the physical network. This approach offers the following key advantages: (i) fast convergence; (ii) reduced communication overhead; and (iii) continual adaptation to evolving network conditions.

• **Extensive performance evaluation:** Extensive simulations are conducted to evaluate the performance of the proposed solution under ACI and HWI-induced distortions. Simulation results demonstrate (a) the robustness and scalability of the proposed DTN-empowered continual-DRL framework in dynamic networks with ACI and HWI uncertainties, and (b) that it achieves up to 114.16% higher sum throughput than the 3GPP-standardized channel quality indicator (CQI)-based link adaptation scheme[3] in dense IoT networks under high HWI levels.

The remainder of this paper is organized as follows. Section 5.3 presents an overview of the system model and the proposed framework. Section 5.4 provides the description of DTN. Section 5.5 formulates the RRM problem. Section 5.6 details the proposed framework, while Section 5.7 discusses the simulation results. Finally, Section 5.8 concludes.

5.3 System Model

5.3.1 System Overview

As shown in Fig. 5.1, we consider a downlink IoT network comprising multiple RAT APs and uniformly distributed IoT devices. Let $\mathcal{K} = \{1, 2, \cdots, K\}$ be the set of all RAT APs, $\mathcal{N} = \{1, 2, \cdots, N\}$ be the set of IoT devices, and $\mathcal{R} = \{1, 2, \cdots, R\}$ be the set of RRBs, where each RRB provides the minimum granularity of bandwidth allocated to an IoT link. In the proposed system, IoT devices are clustered with their nearest RAT APs based on proximity. Each RAT AP simultaneously transmits data to its associated IoT devices using the OFDMA

[3] In this work, 3GPP-specified SINR–MCS mapping is used solely as a representative example of a standardized link-adaptation scheme. Our proposed learning framework itself is not restricted to 3GPP systems and remains applicable to any deployment where suitable CQIs can be obtained.

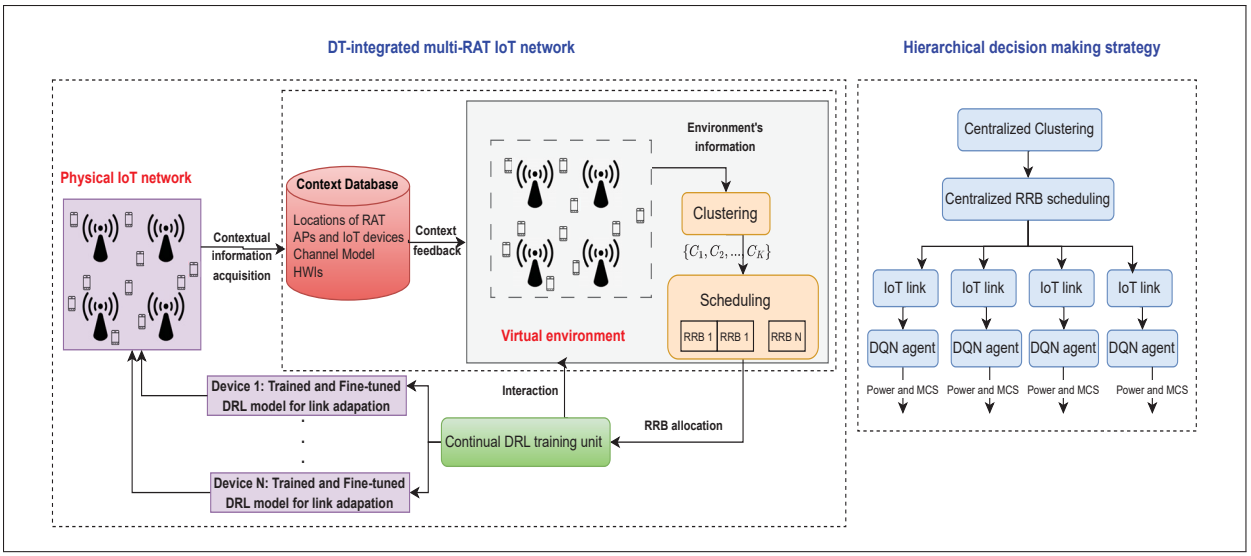


Figure 5.1 Proposed DTN-empowered resource allocation framework for multiple RAT APs IoT networks

technique. Meanwhile, a centralized network controller assigns an orthogonal RRB to each device[4] by PF RRB scheduling algorithm, thus ensuring efficient resource distribution and maintaining fairness across the network. Owing to these considerations, co-channel interference among devices—both within a cluster and across clusters—is considered to be negligible.

Importantly, the devices associated with different RAT APs can be scheduled to adjacent RRBs. However, due to the low-cost receivers that are typically employed in IoT devices, the system is susceptible to HWI and ACI. Specifically, non-ideal filters cause undesired signals to leak into adjacent frequency bands, resulting in ACI for devices allocated to neighboring RRBs. While ACI and HWI explicitly depend on the selected transmit power level, because of the fundamental power–modulation–coding trade-off, they also implicitly depend on the chosen MCS schemes. On one hand, increasing transmit power and signal-to-noise ratio enables the use of higher-order MCS at the cost of amplifying HWI-induced distortion and ACI toward neighboring IoT links. On the other hand, decreasing transmit power minimizes the impact of non-linear impairments, but reduces throughput, as only lower-order MCS can be supported to maintain transmission reliability. In essence, there exists an inherent relationship between link adaptation parameters

[4] Without the loss of generality, throughout this paper, we use the terms device" and link" interchangeably.

and the impairments caused by ACI and HWI. In the present study, we use this relationship to optimize link adaptation and enable harmonious coexistence among multi-RAT IoT links.

5.3.2 Proposed Framework

This section provides an overview of the proposed RRM framework, which comprises the following three key components: (a) a context database; (b) a digital twin network; and (c) a continual DRL training unit. Fig. 5.1 illustrates these components and the hierarchical resource allocation decision making strategy.

5.3.2.1 Context Database

The context database gathers data from the real-world environment and stores this data to support the creation of a virtual representation of the physical network. To maintain the relevance and accuracy of the virtual environment, the contextual data are categorized into the following two types:

• Static contextual data: Environmental parameters such as the radio propagation model, RAT AP information (e.g., number, location, height, maximum/minimum transmit power), frequency band allocation, and a theoretical HWI model.

• Dynamic contextual data: Network parameters such as RRB allocation, IoT device locations, and individual channel measurements of IoT links[5].

The static contextual data remain fixed over extended periods of time and are collected only once during the DT model construction phase. In contrast, dynamic contextual data is collected periodically through various key performance measurement reports, widely available in wireless standards. Importantly, network parameters are periodically collected at RAT APs over standard

[5] We assume that each IoT node estimates its own CSI and reports it to the RAT AP over an uplink control feedback channel, consistently with off-the-shelf wireless standards.

control feedback channels and provided as dynamic contexts to the DTN once in every continual learning period (or DRL refinement period) spanning over several TSs.

5.3.2.2 Digital Twin Network

The DTN serves as a virtual emulation environment designed to train DRL algorithms in a controlled and reliable setting. The DTN developed in this study maintains a virtual network topology with a digital abstraction of multi-RAT APs, IoT devices, and radio resources. It also incorporates a baseline representation of time-varying fading channels and approximate models of HWI and ACI (given in Section 5.4.2.2). These initial models, while simplified, provide a starting point for generating realistic datasets and enabling the training of DRL-based RRM policies. Importantly, the DTN leverages continual learning to refine the DRL agent's policy with observations periodically collected from the real deployment, rather than relying solely on theoretical models. Such an adaptive learning mechanism ensures that the trained policies remain resilient even when the actual device nonlinearities and interference patterns deviate from the initial assumptions. The DTN integrates virtual radio resource control functionalities such as clustering, scheduling, and data transmission with adaptive modulation (AM) and coding selection. In creating the DTN, we relied on the following assumptions. **A1:** Each device is aware of its local CSI. **A2:** We consider a time-slotted fading channel where the small-scale gain remains constant in each TS and changes from one TS to the next. Besides, we assume that large-scale fading remains the same over T TSs. **A3:** Each device is associated with only one RAT AP. **A4:** For the direct collection of channel-specific datasets at the DTN from the physical network, as well as for transferring the trained DRL model from the DTN to the physical network, following the DT literature (Elloumi *et al.*, 2025a), (Li *et al.*, 2025), and (Elloumi *et al.*, 2025b), we assume the existence of trustworthy and high-bandwidth physical-to-digital (P2D) and digital-to-physical (D2P) feedback links between the multi-RAT APs in the physical network and the DT. These P2D/D2P links are also used to disseminate updated DQN parameters during continual refinement, while no inter-link communication or synchronization is required during online operation. Taken together, these assumptions provide a foundation for accurate modeling

and simulating the physical IoT network within the DTN. Further detail on the design of this DTN is provided in Section 5.4.

5.3.2.3 Continual DRL Training Unit

This unit is responsible for delivering a trained and fine-tuned DRL agent for deployment in the coexisting multi-RAT IoT links. Each link operates as an independent agent and makes resource allocation decisions based on local observations. Given the dynamic nature of IoT networks with uncertain channels, ACI, and HWI, the DRL agent is continuously trained through interactions with both the DTN and the physical environment. The continual learning strategy leverages replay memory–based periodic fine-tuning that (a) enhances model stability by storing and reusing past experiences and (b) updates the model with new data to adapt to evolving network dynamics. The continual DRL algorithm is discussed in detail in Section 5.6.

To support adaptive decision making, we divide the mission duration into several episodes, each consisting of T_s TSs. Device clustering around multi-RAT APs is performed once at the beginning of each episode based on device proximity to the APs, whereas PF-based RRB scheduling and DRL-based link adaptation are executed at every TS. Fig. 5.2 illustrates a representative timing diagram showing this periodic coordination across clustering, scheduling, and link adaptation.

5.4 Digital Twin Network

This section provides a comprehensive overview of designing the DTN for multi-RAT IoT systems, including the overall network environment and the modeling parameters.

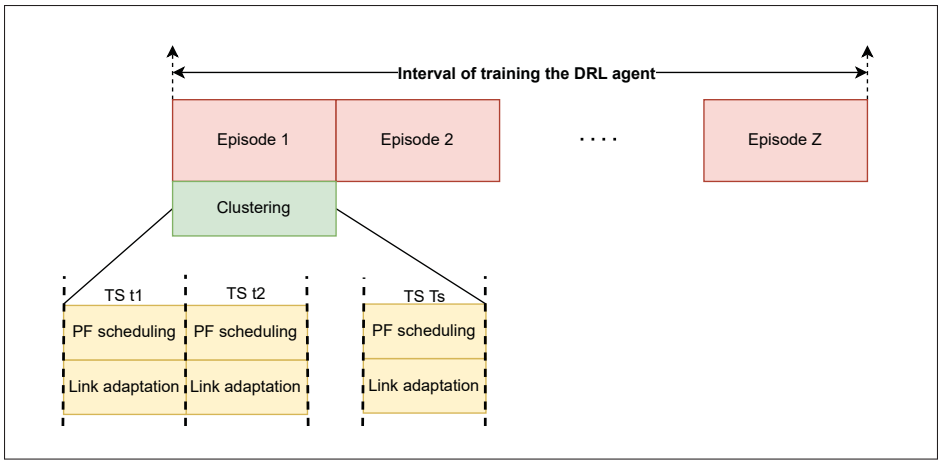


Figure 5.2 Timing diagram

5.4.1 Environment of the Digital Twin

5.4.1.1 RAT APs

Each RAT AP has a coverage radius of 300 m, with a distance of 600 m between neighboring RAT APs. Following 3GPP UMi specifications, the RAT APs' heights are set to 10 m. Each RAT AP operates in the 6.5 GHz (Hetting, 2023) frequency band. In addition, each RAT AP contains nonlinear components contributing to HWI.

5.4.1.2 IoT Devices

IoT devices, which are uniformly distributed within the coverage area, experience varying path losses based on their proximity to RAT APs and the network's dynamic conditions. Following 3GPP UMi specifications, their heights are set to 1.5 m. Similarly to the RAT APs, IoT devices also contain nonlinear components that result in HWI.

5.4.1.3 Time Varying Channel

A fully synchronized, time-slotted system with a slot duration of T_s is considered. The channel gain accounts for both small-scale and large-scale fading. Small-scale fading remains constant within each TS and varies between TSs, modeled using the Jakes fading model. The TS interval,

T_s, is set to 20 ms, representing the channel coherence time to capture fading effects. In its turn, large-scale fading incorporates path loss and shadowing to reflect the overall propagation environment, following the 3GPP UMi path loss model.

5.4.1.4 Device Clustering

Devices are clustered to the nearest RAT AP based on their proximity. The number of devices associated with each RAT AP can vary across RAT APs and from deployment to deployment. Let $\{C_1, C_2, \cdots, C_K\}$ represent the set of clusters formed.

5.4.1.5 RRB-Device Scheduling

Following (Naderializadeh *et al.*, 2021), the PF algorithm is used to schedule the device to the appropriate available RRB. The PF scheduling algorithm is discussed in Section 5.4.3.3.

5.4.2 Digital Twin Modeling Parameters

5.4.2.1 Channel Model

We denote the location of the n-th devices in TS t by $(x_n(t), y_n(t), H_n)$, $\forall n \in \mathcal{N}$, and the location of the RAT AP by (x_k, y_k, H_k). In TS t, the 2D and 3D distances between the RAT AP and the n-th device are expressed, respectively, as shown in Eqs. (5.1)-(5.2).

$$d_{2D} = \sqrt{(x_n(t) - x_k)^2 + (y_n(t) - y_k)^2}, \tag{5.1}$$

and

$$d_{3D} = \sqrt{(x_n(t) - x_k)^2 + (y_n(t) - y_k)^2 + (H_n - H_k)^2}. \tag{5.2}$$

The downlink channel gain from the k-th RAT AP to its associated device is modeled following (Nasir & Guo, 2019), where each channel is subjected to large-scale fading $\beta_{n,k}$ and small-scale

block Rayleigh fading $h_{n,k}$. The corresponding channel gain is $g^t_{n,k} = |h_{n,k}(t)|^2\beta_{n,k}$. According to the Jakes fading model (Liang *et al.*, 2017), $h_{n,k}$ can be expressed as a first-order complex Gauss-Markov process (see Eq. (5.3)).

$$h_{n,k}(t) = \rho h_{n,k}(t-1) + \sigma, \tag{5.3}$$

where ρ is the correlation coefficient between two TSs, σ is a random variable with a distribution $\sigma \sim \mathcal{CN}(0, 1-\rho^2)$, and $h(0)$ is a random variable with a normal distribution $h(0) \sim \mathcal{CN}(0,1)$. The large-scale fading component $\beta_{n,k} = 10^{\frac{-(PL+\sigma_s)}{10}}$ depends on both path loss PL and shadowing σ_s. In the present study, we consider the path loss model proposed by 3GPP (3GPP, 2020), specifically focusing on the UMi scenario. The average path loss (PL) (in dB) between the RAT AP and the n-th device is given by Eq. (5.4).

$$PL = Pr_{LoS}PL_{LoS} + Pr_{NLoS}PL_{NLoS}, \tag{5.4}$$

where Pr_{LoS} and Pr_{NLoS} are the probabilities of having a LoS and NLoS between the RAT AP and the n-th device, respectively. PL_{LoS} and PL_{NLoS} represent the path loss between the RAT AP and the n-th device for the LoS and NLoS links (in dB), respectively. These values were computed based on (3GPP, 2020, Table 7.4.1-1). The PL_{LoS} is expressed as shown in Eq. (5.5).

$$PL_{LoS} = \begin{cases} PL_1 & 10\text{m} \leq d_{2D} \leq d_{BP} \\ PL_2 & d_{BP} \leq d_{2D} \leq 5\text{km} \end{cases} \tag{5.5}$$

where PL_1 and PL_2 are defined as follows (see Eqs. (5.6)-(5.7)).

$$PL_1 = 32.4 + 21\ log_{10}(d_{3D}) + 20\ log_{10}(f_c), \tag{5.6}$$

and

$$PL_2 = 32.4 + 40\ log_{10}(d_{3D}) + 20\ log_{10}(f_c) + \eta_{LoS}. \tag{5.7}$$

The PL_{NLoS} is given by Eq. (5.8).

$$PL_{NLoS} = \max(22.4 + 35.3\ log_{10}(d_{3D}) + 21.3\ log_{10}(f_c) + \eta_{NLoS}, PL_{LoS}), \tag{5.8}$$

where f_c denotes the carrier frequency, η_{LOS} and η_{NLoS} represent additional attenuation factors due to the LoS and NLoS connections, respectively. Parameter d_{BP} represents the breakpoint distance, expressed as shown in Eq. (5.9).

$$d_{BP} = \frac{4}{c}(H_k - 1)(H_n - 1)f_c, \tag{5.9}$$

where c is the speed of light. From (3GPP, 2020, Table 7.4.2-1), we obtain $Pr_{LoS} = \frac{18}{d} + e^{(\frac{-d_{2D}}{36})}(1 - \frac{18}{d_{2D}})$ and $Pr_{NLoS} = 1 - Pr_{LoS}$. Of note, ensuring analysis is also valid for other channel fading and path loss models.

5.4.2.2 ACI and HWI Model in DT

We consider the effects of both ACI and HWI-induced distortions on the system performance.

• **ACI:** After scheduling each device to its appropriate RRB, we consider the ACI as previously proposed in (ACI, 2009). Specifically, the ACI from the first and second adjacent channels dominate as compared to the remaining adjacent channels. For instance, a device allocated to RRB_r will experience ACI from devices assigned to RRB_{r-2}, RRB_{r-1}, RRB_{r+1}, and RRB_{r+2}. The ACI power is expressed as shown in Eq. (5.10).

$$P_{ACI}(t) = \sum_{l=1}^{L} \frac{g_l^t P_l^t}{A_l}, \tag{5.10}$$

where L is the total number of adjacent channels, P_l^t is the power received from the l-th adjacent channel, g_l^t denotes the channel gain for the l-th adjacent channel at time t, and A_l is the ACI ratio that quantifies the interference attenuation for the l-th adjacent channel (ACI, 2009, Table 10).

• **Linear distortion:** In the present study, we analyze the impact of linear distortion arising from imperfections in the power amplifier. Following (Chen *et al.*, 2021) and (Matthaiou *et al.*, 2013), the distortion noise, denoted as z, is modeled as $z \sim \mathcal{CN}\left(0, (\sigma_t^2 + \sigma_r^2)P_{total,k}\right)$, where $P_{total,k}$ denotes the total transmit power of the k-th RAT AP to its associated IoT devices (Mohamed *et al.*, 2024). Here, $\sigma_t^2 \geq 0$ and $\sigma_r^2 \geq 0$ denote the degree of HWI at the transmitter and receiver, respectively. Specifically, they represent error vector magnitude of RF transceivers, which quantifies the ratio of the average distortion magnitude to the average reference signal magnitude (Baghel *et al.*, 2025).

5.4.2.3 ACI and HWI Model in Physical Network

Due to factors such as hardware nonlinearity, temperature drift, aging, and manufacturing tolerances, the HWI and ACI observed in practice diverge from those modeled in the DTN. Since these effects cannot be precisely captured by a static theoretical model, a mismatch between the impairments in the DTN and those in the physical environment is both inevitable and unknown a priori. Without loss of generality, we model these deviations using linear functions and introduce a scalar parameter ϕ to represent them. Specifically, ϕ quantifies the relative differences between the digital and physical representations of ACI power and HWI-induced distortion with respect to the theoretically modeled values. The ACI power and HWI-induced distortion noise in the physical environment are defined as shown in Eqs. (5.11)-(5.12)

$$P_{ACI}^{\text{PHY}}(t) = \sum_{l=1}^{L} \frac{g_l^t P_l^t}{A_k}(1+\phi), \tag{5.11}$$

and

$$z^{\text{PHY}} \sim \mathcal{CN}\left(0, (\sigma_t^2 + \sigma_r^2)P_{total,k}(1+\phi)\right), \tag{5.12}$$

respectively. Evidently, small and large values of this scalar parameter represent scenarios where the ACI and HWI in the physical network are more and less severe, respectively. Importantly, our continual DRL framework is agnostic to the choice of deviation model, as the link-adaptation DRL agent is periodically refined using experiences collected from the physical network. Consequently, our framework is also applicable to other impairment deviation models.

5.4.3 Virtual RRM Functionality

5.4.3.1 Data Transmission Model and Signal-to-Interference-Plus-Noise Ratio

In what follows, we derive the SINR while taking into account the deleterious impact of both ACI and HWI-induced distortions. Without the loss of generality, we consider that the n-th IoT device is associated with the k-th RAT AP. The received baseband signal from the k-th RAT AP to the n-th IoT device is expressed by Eq. (5.13).

$$\mathrm{r_n(t)} = \sqrt{g^t_{n,k}}\left(\sqrt{P^t_{n,k}}x_n + z\right) + \sum_{l=1}^{L}\sqrt{\frac{g^t_l P^t_l}{A_l}}x_l + n_x, \tag{5.13}$$

where x_n is baseband modulated symbol with $\mathbb{E}[|x_n|^2] = 1$, $P^t_{n,k}$ denotes the transmit power at t TS, z denotes the signal distortion caused by HWIs (Javed, Amin, Shihada & Alouini, 2019; Soleymani, Lameiro, Santamaria & Schreier, 2019), x_l refers to baseband symbol of the l-th interfering link with $\mathbb{E}[|x_l|^2] = 1$, and $n_x \sim \mathcal{N}(0, \sigma^2)$ is the AWGN with variance σ^2. Using both the ACI and HWI-induced distortions models, the downlink SINR expression for the n-th device is expressed as shown in Eq. (5.14).

$$\mathrm{sinr^t_{n,k}} = \frac{P^t_{n,k}\, g^t_{n,k}}{g^t_{n,k}(\sigma_t^2 + \sigma_r^2)P_{total,k} + P_{ACI}(t) + \sigma^2}. \tag{5.14}$$

Of note, when the SINR for IoT links is computed in the real system, rather than in the DT (i.e., based on measurements from the physical environment), the expression in Eq. (5.14) incorporates real-world parameter such as $P^{\mathrm{PHY}}_{\mathrm{ACI}}(t)$ and z^{PHY}.

5.4.3.2 Modulation and Coding Rate

In this study, we consider AM with error-correcting codes, where $\mathcal{M}$ modulation schemes, either Gray-coded phase-shift keying (PSK) or quadrature amplitude modulation (QAM), and C coding schemes are available. For the n-th IoT device, the selected MCS at TS t are denoted as $m^t_{n,k}$ and $c^t_{n,k}$, respectively. Let $r_{m,c}$ (in bits per symbol) represent the transmission efficiency, assuming that N_p denotes the number of symbols per packet or frame. The achievable data rate (measured in bits per frame) for the n-th IoT device is then given by Eq. (5.15) (Mashhadi, Ghiasi, Farahmand & Razavizadeh, 2021).

$$\mathrm{R}^t_{n,k} = r_{m,c}\left(1 - \rho_{m,c}(\mathrm{sinr}^t_{n,k})\right) N_p, \tag{5.15}$$

where $\rho_{m,c}$ denotes the packet error rate (PER) and is defined as shown in Eq. (5.16) (Mashhadi *et al.*, 2021).

$$\rho_{m,c}(\mathrm{sinr}^t_{n,k}) = 1 - \left(1 - f_{m,c}(\mathrm{sinr}^t_{n,k})\right)^{N_p}. \tag{5.16}$$

In the above expression, $f_{m,c}(\cdot)$ represents the symbol error rate (SER), which is given by Eq. (5.17).

$$f_{m,c}(\mathrm{sinr}^t_{n,k}) = 2\left(1 - \frac{1}{\sqrt{m^t_{n,k}}}\right) Q\left(\sqrt{\frac{3\, G^t_{n,k} \log_2(m^t_{n,k})\ \mathrm{sinr}^t_{n,k}}{m^t_{n,k} - 1}}\right), \tag{5.17}$$

where $Q(.)$ is the tail distribution function of the standard Gaussian distribution. Coding gain G is defined as $G^t_{n,k} = c^t_{n,k} d_{free,H}$, where $d_{free,H}$ is the Hamming free distance.

5.4.3.3 PF Scheduling Algorithm

We consider the PF scheduling algorithm to schedule RRBs among the coexisting devices while striking a suitable balance between fairness and throughput efficiency (Naderializadeh *et al.*,

2021). The PF ratio for device n at TS t is defined as[6] shown in Eq. (5.18).

$$PF_n = w_n(t)\ log_2(1 + \frac{P_c\ g^t_{n,k}}{\sigma^2}), \tag{5.18}$$

where $w_n(t) = \frac{1}{\hat{\mathrm{R}}^t_{n,k}}$ is the weight assigned to device n at each TS, and P_c is a fixed value of transmit power for all IoT links. Without loss of generality, we consider $P_c = P_{min}$ for all IoT links. The long-term average rate of device n at t TS denoted as $\hat{\mathrm{R}}^t_{n,k}$ is updated using an exponential moving average to smooth variations over time (see Eq. (5.19)).

$$\hat{\mathrm{R}}^t_{n,k} = (1 - \alpha_r)\hat{\mathrm{R}}^{t-1}_{n,k} + \alpha_r\, \mathrm{R}^{t-1}_{n,k}, \tag{5.19}$$

where $\alpha_r \in [0, 1]$ determines the smoothing factor or the window size for the exponential moving average. $\mathrm{R}^{t-1}_{n,k}$ is the achieved rate of device n at the previous time step, which depends on the modulation m and coding c schemes selected at $t - 1$. At each TS, the PF ratios of all devices are sorted, and available RRBs are allocated to devices accordingly. All steps of the RRB scheduling algorithm are set out in Algorithm 5.1.

Algorithm 5.1 PF Scheduling Algorithm

Input: Total scheduling duration T, number of devices N, R available RRBs.
Initialization: Initialize with random scheduling.
for $t = 1 : T$ **do**
    Compute the PF ratio for all devices.
    Sort devices in the descending order based on their PF ratio.
    Select the top R devices from the sorted list.
    Assign a single orthogonal RRB to each selected device.
end for
Output: RRB scheduling for devices at each TS.

Remark 1: In the present study, assuming that the multi-RAT APs always have data to transmit to IoT devices, we adopt PF scheduling and a full-buffer traffic model. This setup emulates persistent downlink demand and enables performance evaluation under worst-case load conditions. While typical IoT traffic is often bursty or periodic, emerging IoE applications are increasingly

[6] PF scheduling not only prioritizes the devices with the best channel gain, but also with its long-term behavior.

characterized by continuous or semi-continuous data flows (Bauwens, Ruckebusch, Giannoulis, Moerman & Poorter, 2020b). However, to accommodate delay-sensitive or freshness-critical traffic, the proposed DRL-based link adaptation framework is scheduler-agnostic and can be readily extended to alternative scheduling policies.

5.5 Radio Resource Management

Problem formulation: In the present study, we aim to maximize the long-term sum-throughput of the network by optimizing the IoT links' transmit power, modulation orders, and coding rates at each TS. The RRM problem is formulated as shown in Eq. (5.20).

$$\text{P0:} \quad \max_{\boldsymbol{P}^t_{n,k}, \boldsymbol{m}^t_{n,k}, \boldsymbol{c}^t_{n,k}} \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \sum_{n=1}^{N} \quad R^t_{n,k}$$
$$\text{s.t.} \begin{cases} \text{C1: } P_{min} \leq P^t_{n,k} \leq P_{max}, \forall n \in \mathcal{N}, \forall t, \forall k \\ \text{C2: } m^t_{n,k} \in \mathcal{M}, \forall n, \forall t, \forall k \\ \text{C3: } c^t_{n,k} \in C, \forall n, \forall t, \forall k \end{cases} \tag{5.20}$$

Constraint (C1) implies that the transmit power between the k-th AP and the n-th IoT device is bounded by P_{min} and P_{max} that represent the minimum and maximum transmit power limits of a RAT AP, respectively. Constraints (C2) and (C3) imply that the optimal MCS are selected from $\mathcal{M}$ and C, respectively. P0 is a MINLP problem, as it involves both continuous and discrete optimization variables, and its throughput function is non-convex.

Lemma 1. *P0 is an NP-hard optimization problem.*

Proof. The proof is provided in Appendix II. □

Motivation for using DRL to solve P0: Problem P0 is NP-hard, and obtaining its optimal solution requires an exhaustive search over the joint space of transmit power, modulation, and coding, which is infeasible in large-scale systems. Beyond this combinatorial complexity,

conventional optimization techniques face several fundamental limitations. First, the objective function of P0 is implicit due to the lack of precise knowledge of HWI and ACI parameters, which vary across IoT devices because of dynamic and nonlinear hardware behaviors. This renders classical gradient-based methods inapplicable (Andrews, Humphreys & Ji, 2024). Second, optimal centralized optimization requires global CSI, including both direct- and adjacent-link channel gains, the acquisition of which incurs prohibitive signaling overhead in dense deployments (Nasir & Guo, 2019). Third, even assuming perfect CSI and exact HWI/ACI models, solving P0 via a centralized MINLP solver entails excessive computational complexity, making per-device link adaptation at each TS impractical.

Given these challenges, we cast P0 as a learning problem rather than a traditional optimization problem, with the objective of learning a policy that maps system states to transmission parameters under uncertainty and dynamics. Although supervised DL could, in principle, approximate this mapping, it requires labeled optimal solutions over a vast state space, whose generation is computationally intractable due to the NP-hardness of P0 and the continuous network dynamics. To address this, we reformulate P0 as a decentralized Markov game and adopt an MADRL framework. In this model-free setting, agents learn their policies directly through interaction with the environment. The DRL-based approach enables: (i) implicit handling of interference and hardware uncertainties; (ii) real-time, low-complexity decision making across a large number of agents; and (iii) scalable and distributed link adaptation without requiring full CSI.

Why DTN? Training DRL models directly in a live network is impractical due to safety risks, high operational costs, excessive overhead, and suboptimal exploration. DTNs provide a high-fidelity virtual environment for training and continuously updating DRL models by accurately replicating key network dynamics. Nevertheless, DTNs inevitably deviate from real-world conditions (e.g., due to channel variations, HWI, and ACI), leading to a *sim-to-real gap*. To mitigate this, we adopt a replay-memory-based continual learning strategy (Tong *et al.*, 2025). Specifically, the DRL model trained in the DT is deployed at RAT APs, where a subset of real-world experiences is periodically collected and stored in a replay memory. These samples

are then used to fine-tune the DTN and continuously adapt the DRL model using both simulated and real experiences.

5.6 Digital Twin-Enabled Continual MADRL Framework

We first reformulate P0 as a multi-player stochastic game (also known as a Markov game). This approach is motivated by the need for a distributed solution to P0 while taking interdependence of devices' decisions within the system. To manage the computational complexity, we discretize the decision space. Specifically, we represent transmit power levels using a discrete set[7], as defined in Eq. (5.25). Due to the presence of ACI, each IoT link's achievable downlink rate is affected not only by the transmit power of its serving RAT AP, but also by the power levels allocated to other coexisting IoT links, particularly those operating on adjacent channels. Hence, a game-theoretic approach is well-suited to solve P0 in a decentralized manner. Furthermore, the IoT network exhibits stochastic dynamics driven by both the environment and the actions of individual links, which naturally calls for a Markov game formulation. In this setting, each IoT link is modeled as a player that iteratively explores actions (i.e., transmit power and MCS) according to a well-defined strategy and periodically refines it, enabling autonomous learning of optimal link adaptation strategy from local state information (as in Eq. (5.21)) without requiring knowledge of other IoT links' strategies.

5.6.1 Proposed Multi-Player Stochastic Game

Game formulation: P0 is reformulated as a Markov game, formally defined as tuple $\mathcal{G} = (\mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{R}, \pi)$ with the following components:

• $\mathcal{N}$ is the set of all agents. The stochastic game is conducted in a virtual space and a virtual agent is created for each IoT link.

[7] Although theoretical optimization models assume continuous power allocation, hardware constraints and implementation ease require that power allocation be discrete and quantized. Consequently, devices can only choose from predefined power levels (e.g., level 1, level 2, level 3, etc.).

• $\mathcal{S}$ is the state space of the game that comprises representative features extracted from each agent's interaction with the environment. To accurately characterize the environment, the state space includes the following key elements:

(i) The local CSI between the k-th RAT AP and the n-th IoT device at TS t, $g_{n,k}^{t}$.

(ii) Historical data including SINR of the IoT devices at TS $t-1$, denoted as $\text{sinr}_{n,k}^{t-1}$, and average transmission rate at TS $t-1$, represented as $R_{n,k}^{t-1}$.

(iii) Actions taken by the agent in the previous TS, including $P_{n,k}^{t-1}$, $m_{n,k}^{t-1}$, and $c_{n,k}^{t-1}$.

Therefore, for each agent, the state space is formally defined as shown is Eq. (5.21).

$$\mathcal{S}_n = \{g_{n,k}^{t}, \text{sinr}_{n,k}^{t-1}, R_{n,k}^{t-1}, P_{n,k}^{t-1}, m_{n,k}^{t-1}, c_{n,k}^{t-1}\}. \tag{5.21}$$

The overall state space for all agents is expressed as shown in Eq. (5.22).

$$\mathcal{S} = \bigcup_{n=1}^{N} \mathcal{S}_n. \tag{5.22}$$

Incorporating historical data (e.g., past SINR, throughput, and actions) into the state space enables the agent to (i) track the effectiveness of past actions and (ii) capture inherent temporal correlations across consecutive actions arising from correlated channel realizations. This approach enhances decision making by leveraging past experiences to improve future performance.

• $\mathcal{A}$ is the overall action space of the game denoted as $\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \times \cdots \times \mathcal{A}_N$. In consistent with P0 formulation, the goal of each player in the Markov game is to find the optimal combination of the transmit power, modulation, and coding rate at the beginning of each game round (i.e., each TS). To describe each of these schemes, we use a set of discrete values. Specifically, for the transmit power, we consider discrete power levels ranging from P_{min} to P_{max}. The modulation and coding action space encompasses all available MCS levels. Consequently, the action space of each agent is designed as shown in Eq. (5.23).

$$\mathcal{A}_n = \{m_{n,k}^{t} \in \mathcal{M}, P_{n,k}^{t} \in \mathcal{P}, c_{n,k}^{t} \in C\}, \tag{5.23}$$

where

$$\mathcal{M} = \{M_1, M_2, \ldots, M_{|\mathcal{M}|}\}, \tag{5.24}$$

$$\mathcal{P} = \{P_{min}, \frac{P_{max}}{|\mathcal{P}|-1}, \frac{2P_{max}}{|\mathcal{P}|-1}, \ldots, P_{max}\}, \tag{5.25}$$

$$C = \{C_1, C_2, \ldots, C_{|C|}\}, \tag{5.26}$$

$\mathcal{M}$, $\mathcal{P}$, and C represent the modulation, power, and coding rate action spaces, respectively. Furthermore, we use $|.|$ to denote the cardinality of each action space. For implementation in the proposed DRL-based framework, each joint action $(m^t_{n,k}, P^t_{n,k}, c^t_{n,k}) \in \mathcal{M} \times \mathcal{P} \times C$ is mapped to a single discrete action index corresponding to one unique combination of transmit power, modulation, and coding rate. Consequently, the total number of discrete actions for each agent is $|\mathcal{A}_n| = |\mathcal{M}| \times |\mathcal{P}| \times |C|$.

• $\mathcal{R}$ is the agents' common reward given by the sum throughput, i.e., $\mathcal{R} = \sum_{n=1}^{N} R^t_{n,k}$.

• $\Pi = \{\pi_1, \pi_2, \ldots, \pi_N\}$ is the overall strategy space of the game. In particular, $\pi_n : \mathcal{S} \times \mathcal{A} \rightarrow [0.1]$ provides the resource allocation policy of the n-th agent, $\forall n \in \mathcal{N}$. The resource allocation policy is essentially a probability mass function (PMF) over possible actions in a given state. More specifically, $\pi_n(s_{n,t}, a_{n,t}) = [\pi_n(s_{n,t}, a_{n,t})]_{a_{n,t} \in \mathcal{A}_n}$, where $\pi_n(s_{n,t}, a_{n,t}) \in [0, 1]$ denotes the n-th agent's probability of selecting action $a_{n,t}$ from action space $\mathcal{A}_n$ in state $s_{n,t}$, $\forall n \in \mathcal{N}$ and $\forall s_{n,t} \in \mathcal{S}_n$. In this case, $\sum_{a_{n,t} \in \mathcal{A}_n} \pi_n(s_{n,t}, a_{n,t}) = 1$ holds. To describe the solution to this Markov game, we first introduce the notion of state value function as follows (Han, Niyato, Saad & Başar, 2019).

Definition 1. For an infinite-horizon Markov game, state value function $V_n(s_{n,t}, \pi_n, \pi_{-n})$ represents the n-th agent's expected utility over the state transitions starting from state $s_{n,t}$ when

the agents employ the resource allocation policy $\{\pi_n, \pi_{-n}\}$ and is defined as shown in Eq. (5.27).

$$V_n(s_{n,t}, \pi_n, \pi_{-n}) = \mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i R_{i|s_0=s_{n,t},\pi_n,\pi_{-n}}\right] = \bar{\mathrm{R}} + \gamma \sum_{s_{n,t+1}\in\mathbf{S}_n} \mathcal{P}_{s_{n,t},s_{n,t+1}}(\pi_n, \pi_{-n}) V_n(s_{n,t+1}, \pi_n, \pi_{-n}), \tag{5.27}$$

where $\bar{\mathrm{R}} = \mathbb{E}_\pi[R_0(s_{n,t}, \pi_n, \pi_{-n})]$ indicates the reward to be expected in state $s_{n,t}$ according to resource allocation policy $\{\pi_n, \pi_{-n}\}$, $\mathcal{P}_{s_{n,t},s_{n,t+1}}(\pi_n, \pi_{-n})$ presents the probability of transitioning from state $s_{n,t}$ to state $s_{n,t+1}$ with resource allocation policy $\{\pi_n, \pi_{-n}\}$, $\gamma \in (0, 1)$ represents the discount factor that captures how important recent experiences are in the state value function, and, finally, π_{-n} denotes the resource allocation policies of all agents except the n-th agent.

Solution to game $\mathcal{G}$: In a Markov game, each agent's optimal policy corresponds to its best response to the policies of the other agents, and it is obtained by maximizing the agent's state-value function, expressed as follows.

$$\pi_n^* = \arg\max_{\pi\in\Pi} V_n(s_{n,t}, \pi_n, \pi_{-n}), \forall s_{n,t} \in \mathbf{S}_n, n \in \mathcal{N}. \tag{5.28}$$

Definition 2. The optimal solution to the Markov game is expressed in terms of the Nash equilibrium (NE) resource allocation policies $\Pi^* = \{\pi_1^*, \pi_2^*, \ldots, \pi_N^*\}$, where each agent's policy is its best response (i.e., solution to (5.28)) with respect to the other agents' policy. Thus, in NE, the condition in Eq. (5.29) is satisfied for the n-th agent, $\forall n \in \mathcal{N}$.

$$V_n(s_{n,t}, \pi_n^*, \pi_{-n}^*) \geq V_n(s_{n,t}, \pi_n, \pi_{-n}^*), \forall s_{n,t} \in \mathcal{S}_n, \forall \pi_n, \tag{5.29}$$

where π_{-n}^* denotes the optimal resource allocation policies of all agents except the n-th agent.

Since the state value function does not have any closed-form expression and it depends on the neighbor agents' actions, it is non-trivial to derive optimal policies by (directly) solving (5.28). MARL enables agents to iteratively learn a set of equilibrium policies by iteratively interacting with the environment without any specific information about the agents' state

transition probabilities (Han *et al.*, 2019). Accordingly, we develop a MARL framework to obtain the agents' NE resource allocation policies.

Learning-based optimal solution to game $\mathcal{G}$: To solve (5.28) via learning, we use the multi-agent Q-learning framework to determine the optimal state value function from the optimal Q-function. The Q-function denoted by $Q_m(s_{n,t}, a_{n,t}, \pi)$ represents the n-th agent's expected utility over the state transitions starting from state $s_{n,t}$ and action $a_{n,t}$ using resource allocation policy π, $\forall n \in \mathcal{N}$, and it is expressed by

$$\begin{aligned} Q_n(s_{n,t}, a_{n,t}, \pi) &= \mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i R_{i|s_0=s_{n,t}, a_0=a_{n,t}, \pi}\right] \\ =&\bar{\mathrm{R}} + \gamma \sum_{s_{n,t+1} \in \mathbf{S}_n} \mathcal{P}_{s_{n,t}, s_{n,t+1}}(\pi_n, \pi_{-n}) \sum_{a_{n,t+1} \in \mathbf{A}_n} \pi(s_{n,t+1}, a_{n,t+1}) Q_n(s_{n,t+1}, a_{n,t+1}, \pi), \end{aligned} \tag{5.30}$$

If we compare Eq. (5.27) and Eq. (5.30), with $\forall n \in \mathcal{N}$ and $s_n^t \in \mathbf{S}_n$, we obtain Eq. (5.31).

$$V_n(s_n, \pi_n, \pi_{-n}) = \sum_{a_n \in \mathbf{A}_n} \pi(s_n, a_n) Q_n(s_n, a_n, \pi). \tag{5.31}$$

The optimal state value function (Cui, Liu & Nallanathan, 2020, Eq. (30)) can be expressed as shown in Eq. (5.32).

$$\begin{aligned} V_n(s_{n,t}, \pi_n^*, \pi_{-n}^*) &= \max_{\pi_n} V_n(s_n, \pi_n, \pi_{-n}^*) = \max_{\pi_n} \sum_{a_{n,t} \in \mathbf{A}_n} \pi(s_{n,t}, a_{n,t}) Q_n(s_{n,t}, a_{n,t}, \pi) \\ =& \sum_{a_{n,t} \in \mathbf{A}_n} \pi(s_{n,t}, a_{n,t}) Q_n^*(s_{n,t}, a_{n,t}) \leq \max_{a_{n,t} \in \mathbf{A}_n} Q_n^*(s_{n,t}, a_{n,t}) \end{aligned} \tag{5.32}$$

where $Q_n^*(s_{n,t}, a_{n,t}, \pi) = \max_{\pi_n} Q_n(s_{n,t}, a_{n,t}, \pi)$ is the optimal Q-function. The maximum value of the state value function is obtained from the optimal Q-function of the most profitable action. Accordingly, the optimal solution to (5.28) for the n-th agent in in state $s_{n,t}$, $\forall n \in \mathcal{N}$ and

$s_{n,t} \in \mathcal{S}_n$, is obtained as shown in Eq. (5.33).

$$\pi(s_n^t, a_n^t) = \begin{cases} 1, \text{ if } a_{n,t} = a_{n,t}^* \\ 0, \text{ if } a_{n,t} \neq a_{n,t}^* \end{cases} \tag{5.33}$$

where $a_{n,t}^* = \arg\max_{a_n \in \mathbf{A}_n} Q_n^*(s_n, a_n)$. Therefore, the task of determining the agents' optimal resource allocation policies boils down to learning optimal Q-functions. However, since the state space is continuous (see Eq. (5.21)), conventional Q-learning suffers from the curse of dimensionality; therefore, we employ a multi-agent DQN approach. In what follows, we provide an overview of the DQN and the proposed DT-enabled continual MADRL learning framework.

5.6.2 DTN-Enabled Continual DRL Framework

5.6.2.1 Description of the Proposed Framework

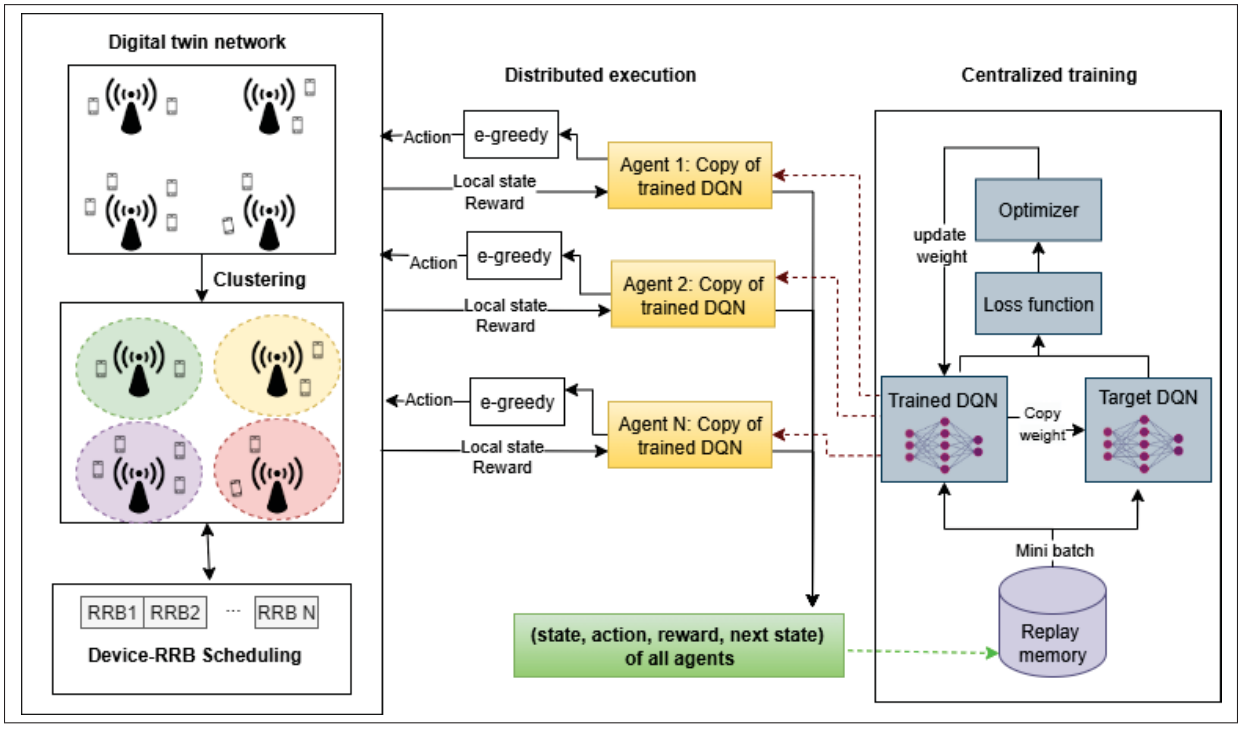


Figure 5.3 Overview of the proposed DT-enabled continual DRL framework

In this study, we propose a continual DQN algorithm where the IoT link acts as the RL agent, while the DTN serves as the environment. Fig. 5.3 provides an overview of the architecture of the proposed solution. The DTN environment comprises K RAT APs and N uniformly distributed IoT devices. The framework begins by clustering IoT devices to the nearest RAT AP, forming clusters $\{C_1, C_2, \cdots, C_K\}$. After clustering, the devices are scheduled to RRBs based on the PF approach (Algorithm 5.1). In the proposed framework, each IoT link is modeled as an independent RL agent. At each step of an RL episode, an agent receives its state from the DTN environment and subsequently performs the corresponding action. Importantly, each IoT link operates independently, resulting in a total of N agents. To address the scalability issue caused by simultaneous training a total of N agents, we adopt a CTDE approach (Nasir & Guo, 2019). In this approach, all links share an identical DQN that is centrally trained by a centralized controller using the experiences collected from all agents. This design ensures scalability and efficiency while maintaining decentralized decision making during execution. Furthermore, training the DQN based on collective experiences of all agents enhances stability and promotes collaborative learning. During execution, each agent operates independently, relying on the shared DQN to select actions based on its own state. While the agents use the same DQN, their actions differ, as each of them encounters unique states, leading to varied action selections. Importantly, during training, the DTN has access to complete system information, allowing it to accurately compute global rewards and state transitions for each IoT link and to store the corresponding experience tuples in the replay memory. This memory is then used to train the DRL model. All policy updates are performed centrally, and the refined DQN model is redistributed and applied synchronously across all IoT links over reliable control channels, thus ensuring that all agents operate with a consistent policy version at any given time.

Algorithm 5.2 provides the overall steps for training our proposed framework using the CTDE approach. The input of our algorithm is the maximum number of training episodes Z and the maximum number of steps per episode T. The algorithm begins by constructing the DTN environment, followed by initializing the replay memory, the trained DQN, and the target DQN. It then proceeds as follows: within each training episode, the IoT devices are clustered to

the nearest RAT AP using a proximity-based clustering mechanism (Line 5). For each step within an episode, the PF scheduling algorithm is executed to determine the RRB allocation for each device (Line 7). For each IoT link, the state of the environment is observed, and the DQN agent selects an action based on the ϵ-greedy policy (Lines 8-15). This policy allows the agent to balance exploration and exploitation by selecting the action with the highest estimated Q-value with probability $1 - \epsilon$ or choosing a random action with probability ϵ. On performing the selected action, the algorithm interacts with either the physical environment or the DT environment. Of note, the algorithm interacts with the physical environment every Z_k RL training episodes, where Z_k is judiciously chosen to strike a balance between overhead and accuracy. During these interactions, the physical twins of the agents (i,e., IoT links) deploy the link adaptation parameters selected by the most recently trained DRL model in the real-world channel and observe the resulting rewards. The IoT links then forward such channel-specific dataset (comprising states, actions, and rewards) to the context database of the DTN via RAT APs[8]. These datasets are subsequently stored in the replay memory for fine-tuning the DRL model within the DT domain (Lines 16-21). For the sake of simplicity, we assume a uniform TS structure and a fixed interaction frequency across all devices during the training phase.

After collecting experiences (from either DT or physical environment), a mini-batch is randomly sampled from the replay memory (Line 23). The trained DQN updates its weights using the backpropagation method by minimizing the temporal difference loss Eq. (1.6) (Line 24). The weights of the target DQN are periodically updated to match the trained DQN, thus ensuring stability in learning (Line 25). Finally, the algorithm outputs the trained DQN policy, where actions are chosen based on the maximum Q-value for the observed state. This policy is subsequently deployed for resource allocation in the multi-RAT IoT system. Importantly, while our study focuses on the downlink, the proposed architecture and learning model can also be readily extended to handle uplink scenarios.

[8] Of note, the channel-specific data collection from the physical network to DTN is required only during the DRL training phase. Once the DRL model is fully trained, inference and decision making are carried out in a fully distributed manner by the individual IoT links, without requiring any real-time CSI exchanges between the IoT links and DTN.

Remark 2: The proposed framework follows a continual learning paradigm, where a DT-trained DRL agent is incrementally fine-tuned using a sequential stream of experiences generated by multiple IoT links operating under time-varying environmental conditions. To preserve knowledge acquired from prior interactions, an experience replay mechanism is employed, whereby the agent is trained using a combination of newly collected experiences D_t (i.e., experiences collected at TS t) and a subset of previously stored experiences $(D_0, D_1, \ldots, D_{t-1})$ drawn from a shared replay buffer. Since replay memory aggregates experiences across multiple IoT links and diverse environment regimes, the resulting policy learns representations that generalize both spatially (across IoT links) and temporally (across evolving channel conditions). As a result, the proposed framework fundamentally differs from conventional periodic retraining approaches that retrain models from scratch for each new dataset or experience batch, and thus lack knowledge retention capability and require large overhead.

5.6.2.2 Signaling Overhead

In this section, we compare the signaling overhead of training our algorithm solely in a physical network versus our proposed approach—predominantly training in the DT domain while periodically collecting physical network data.

• **Training solely in the physical environment:** Here, we consider that Algorithm 5.2 trains our proposed framework exclusively in the physical environment. At each episode, Algorithm 5.2 computes the distance between N existing devices and K RAT APs, clustering each device to its nearest RAT AP. Without the loss of generality, we consider T_D as the amount of information required from each device to accomplish a task. Accordingly, each new clustering step involves a total of NKT_D information exchanges between RAT APs and devices. Next, the algorithm calculates the instantaneous CSI for each device with its associated RAT AP, resulting in an additional NT_D information exchanges. Subsequently, devices are scheduled to available RRBs, and the PF ratio in Eq. (5.18) is computed, leading to NT_D information exchanges between the centralized scheduler and the devices. For each device, the algorithm observes the current state, performs an action, receives a reward, and observes the next state, resulting in a total of $4NT_D$

Algorithm 5.2 Algorithm for Training the DQN-Based Resource Allocation

Input: **DT Construction Phase:** Create a virtual DT environment to emulate the real system behavior, by (a) placing multi-RAT APs and IoT devices, (b) creating propagation channel and non-linear impairments (ACI and HWI), (c) enabling RRM functionalities (device clustering, RRB scheduling, and link adaptation), and (d) integrating the capability to compute each IoT link's SINR and throughput based on the RRM decisions, as described in Section 5.4.
Input: Maximum number of episodes Z, maximum number of steps per episode T, and frequency of interaction with physical environment Z_k.
Initialize replay memory D to zero (**Start of Training Phase**).
Create trained DQN with random weights θ.
Create target DQN with $\theta^{'} = \theta$.
for $i = 1 : Z$ **do**
  Cluster IoT devices to the nearest RAT AP.
  for $t = 1 : T$ **do**
    Perform the PF scheduling algorithm.
    for $n = 1 : N$ **do**
      Observe state of the environment $s_{n,t}$.
      Generate random number $\eta \in [0, 1]$.
      if $\eta > \epsilon$ **then**
        Select $a_{n,t} = \arg\max_{a_{n,t} \in \mathcal{A}_n} q(s_{n,t}, a_{n,t}; \theta)$, where q is estimated by the trained network.
      else
        Randomly select action $a_{n,t}$.
      end if
      if $i \mod Z_k = 0$ **then**
        Observe real reward $r_{n,t+1}$ and real new state $s_{n,t+1}$.
      else
        Observe DT reward $r_{n,t+1}$ and DT new state $s_{n,t+1}$.
      end if
      Save new experience $(s_{n,t}, a_{n,t}, r_{n,t+1}, s_{n,t+1})$ into experience replay memory D.
    end for
    Sample random mini-batch D_b experiences from D.
    Use the backpropagation method to update the weights of the trained DQN.
    Update weights of the target network $\theta^{'}$ by weights of the trained network θ every T_{step}.
  end for
end for
Output: Trained resource allocation agent $q^*(s, a; \theta)$.

information exchanges across all devices. Experience tuple $(s_{n,t}, a_{n,t}, r_{n,t+1}\ s_{n,t+1})$ is then stored in the replay memory, adding another N information exchange. Updating the DQN model's weights requires exchanging updated parameters, which contributes D_b exchanges. Similarly, updating the target network weights requires additional D_b exchanges. Therefore, at each TS of Algorithm 5.2, the total information exchange amounts to $(N(6T_D + KT_D + 1) + 2D_b)$. Over the entire execution of the algorithm, the total signaling overhead is given by the following: $ZT(N(6T_D + KT_D + 1) + 2D_b)$.

• **Our proposed RL training:** As outlined in Algorithm 5.2, in our proposed scheme, the DRL model is trained in the DT environment and periodically streamlines using the experience data collected from the physical network. Since the training in DT is purely software-based, it does not introduce any communication overhead T_D for device-specific information acquisition. However, every Z_k episodes, the algorithm transfers the actions selected by the most recent DRL model to the RAT APs and collects real-world experiences to ensure alignment with actual network conditions. Therefore, the total signaling overhead can be expressed as shown in Eq. (5.34).

$$\frac{Z}{Z_K}T(N(6T_D + KT_D + 1) + 2D_b) + (Z - \frac{Z}{Z_K})T(NK + 7N + 2D_b). \tag{5.34}$$

In Eq. (5.34), the first term represents the over-the-air information exchanges resulting from interactions with the physical environment, whereas the second term corresponds to the total information exchanges within the DT environment. The proposed approach explicitly reduces over-the-air signaling overhead during DRL training.

5.6.3 Overall Algorithm

5.6.3.1 Description of the Overall Algorithm

After training the DQN-based resource allocation model using Algorithm 5.2, the trained model is saved and deployed in the physical network (i.e., IoT devices). Algorithm 5.3 outlines the steps involved in executing the resource allocation process during testing. The algorithm begins by capturing the positions of IoT devices (Line 2) and clustering them to their nearest RAT APs (Line 3). For each TS, each RAT AP collects the instantaneous channel gains for its associated devices (Line 5). Next, PF scheduling is applied using Algorithm 5.1 to assign available RRBs to devices based on fairness (Line 6). Finally, each IoT link observes the current state of the environment (Line 8) and performs the optimal resource allocation action (Line 9). This process is repeated until the total number of TSs T_s is reached. The final output of the algorithm includes device clusters, device-to-RRB scheduling, and an optimal resource allocation solution for

problem P0 at each TS. Algorithm 5.3 adopts a semi-centralized architecture that combines centralized device clustering and RRB scheduling with distributed link adaptation. Specifically, the centralized controller performs PF-based RRB scheduling across heterogeneous multi-RAT IoT links, thus achieving a balanced tradeoff between throughput and fairness. Meanwhile, each IoT link independently performs link adaptation using its local state and the trained DQN model to select the appropriate transmit power level and MCS index. By decoupling global RRB coordination from local link adaptation, Algorithm 5.3 remains scalable and well-suited for large-scale IoT networks.

Algorithm 5.3 DRL-Enabled ACI and HWI Aware Distributed Link Adaptation Algorithm

Input: Number of RAT APs K; number of IoT devices N and RRBs R; trained DRL model $q^*(s, a; \theta)$; total number of TSs T_s.
1 **Initialize:** TS index $t = 1$.
2 Determine the positions of IoT devices.
3 Cluster IoT devices to the nearest RAT AP.
4 **while** $t <= T_s$ **do**
5 Collect instantaneous channel gains $\{g^t_{n,k}\}$, for the existing IoT devices.
6 Perform the PF scheduling using Algorithm 5.1.
7 **for** $n = 1 : N$ **do**
8 Observe state of the environment $s_{n,t}$.
9 Determine the optimal set of transmission parameters (power level and MCS): $a^*_{n,t} = \arg\max_{a_{n,t} \in \mathbf{A}_n} q^*(s_{n,t}, a_{n,t}, \theta)$.
10 Inform the associated RAT-AP of the selected transmission parameters over a reliable control channel for downlink data transmission.
11 **end for**
12 $t \leftarrow t + 1$
13 **end while**
Output: Device clusters, devices-RRBs scheduling, and resource allocation solution to P0 at each TS.

5.6.3.2 Computational Complexity

Computational complexity of Algorithm 5.3 is determined by analyzing its key steps. Capturing the positions of IoT devices (Line 2) requires $O(N)$ operations, while clustering each device to the nearest RAT AP (Line 3) involves checking all K APs per device, leading to a worst-case complexity of $O(NK)$. Next, collecting the instantaneous channel gains for all devices (Line 5) requires $O(NK)$ computations, while performing PF scheduling (Line 6) typically incurs a complexity of $O(N \log N)$. Resource allocation decisions for all devices (Lines 8-9) require

selecting an action from the available action space, resulting in a complexity of $O(N|\mathcal{A}_n|)$. Summing these steps, the overall complexity of Algorithm 5.3 is then $O(2NK+N\log N+N|\mathcal{A}_n|)$.

5.6.3.3 Control Overhead of the Proposed Algorithm

The centralized device clustering step requires collecting IoT device positions, incurring N uplink control signaling exchanges. Note that, to limit signaling overhead, such position information is collected infrequently—namely, once in several TSs. The centralized RRB scheduling step requires per-link CSI, i.e., the channel gain between each device and its serving AP, at every TS. This CSI is estimated at the associated RAT AP using standard-compliant procedures and then is forwarded to the centralized controller via fronthaul/backhaul links, resulting in N uplink feedback exchanges per TS. The distributed link adaptation step at each IoT link requires only local CSI for the current TS and the received SINR from the previous TS, also requiring N uplink feedback exchanges per TS. Overall, the feedback required by Algorithm 5.3 is standard-compliant, as it does not require global or cross-cell CSI acquisition (e.g., CSI from non-serving RAT APs operating on adjacent channels) and any inter-agent coordination signaling during online operation.

5.7 Simulation Results

5.7.1 Benchmark Schemes

• **Random algorithm (RA)**: Power, modulation, and coding rate are randomly selected from sets $\mathcal{P}$, $\mathcal{M}$, and C.

• **SADRL:** This approach involves a centralized agent that observes the global state of the environment and performs joint action selection for all devices.

• **ESA**: In this scheme, the power level, modulation order, and coding rate are jointly selected via an exhaustive search over the entire feasible set of configurations. The combination that

maximizes the sum-throughput is then selected. Perfect knowledge of ACI and HWI is considered in order to achieve an upper bound of sum-throughput.

• **ESA and FP**: This benchmark serves as an upper bound for performance evaluation. Specifically, we perform an exhaustive search over all possible combinations of modulation schemes and coding rates selected from the discrete sets $\mathcal{M}$ and $\mathcal{C}$. For each (m, c) pair, we then optimize the transmission power using FP, since power is treated as a continuous variable. Within the nested loop structure, the FP algorithm computes the optimal power level p for the given (m, c), and the resulting sum throughput is calculated and stored. After evaluating all combinations, the configuration that yields the highest overall sum throughput is selected. Similarly to the ESA method, perfect knowledge of ACI and HWI is assumed.

• **Maximum power 3GPP-CQI based MCS selection (MaxP-3GPP-MCS):** In this benchmark scheme, all RAT APs transmit at a fixed maximum power level ($P_{\max}$). The MCS for each IoT link is selected based on the SINR, following the standard 3GPP CQI-based AMC method. Specifically, for each IoT link, the SINR is first calculated by applying the maximum transmit power $P_{\max}$ to (5.14). Then, the corresponding modulation order and coding rate are selected by comparing the computed SINR with predefined SINR thresholds from the standard CQI-MCS lookup table (as defined in (Chang, 2021, Table 2.1)). This ensures that the selected MCS is supported under the current channel conditions.

5.7.2 Simulation Settings

Our simulation model comprises 4 RAT APs and 20 uniformly distributed IoT devices (Fig. 5.4). The RAT AP's coverage is set to $R = 0.3$ km (Nasir & Guo, 2019). In addition, to ensure that the position of the device will never be confused with that of the RAT AP, we define a small region of radius $r = 0.03$ km with no active devices. For the large-scale fading, the path loss parameters are selected following (Nasir & Guo, 2019) and are presented in Table 5.1. To simulate the HWIs, we consider that $\sqrt{\sigma_t^2 + \sigma_r^2} = \sigma_{rt} = 0.05$ as previously indicated in (Matthaiou *et al.*, 2013). Furthermore, we consider $f_c = 6.5$ GHz in accordance with (Hetting, 2023).

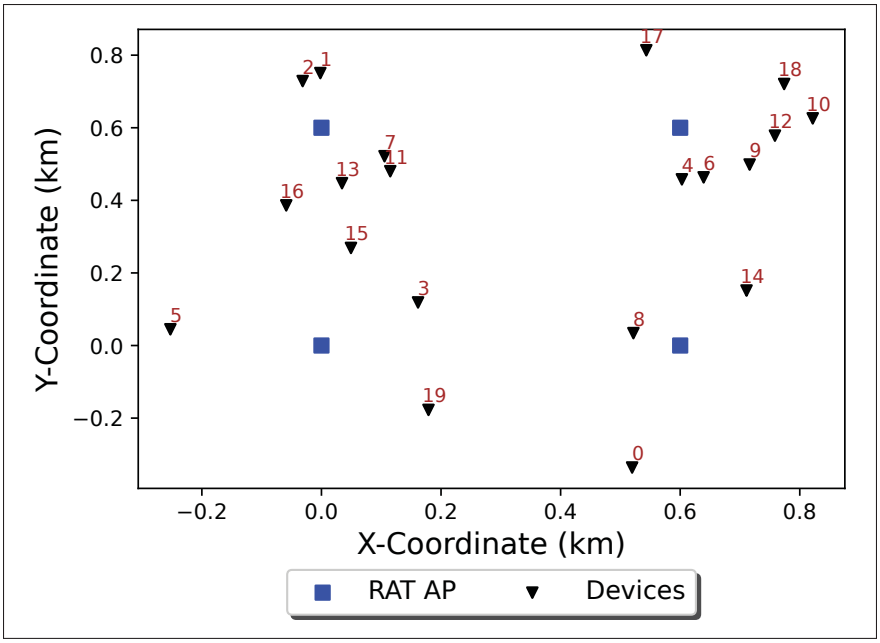


Figure 5.4 Network configuration for 4 RAT APs and 20 devices

The hyper-parameters used for our proposed algorithm's architecture are shown in Table 5.2. The DQN model comprises one input layer, two hidden layers, and one output layer. The hidden layers consist of $N_1 = 64$, $N_2 = 128$ neurons, respectively. The input size is equal to the number of elements in the state vector, which is 6. For the output layer, we consider that $|\mathcal{M}_m| = 4$, $|\mathcal{P}| = 4$, and $|C| = 3$; accordingly, the DQN has a total of 48 outputs. In the present study, we consider an error-control system where the devices support multiple modulation schemes, such as BPSK, QPSK, 16QAM, and 64QAM, as illustrated in Table 5.3. Table 5.3 provides the SER expressions for the modulation schemes considered. For each scheme, the coding rate is selected from the set $\{\frac{1}{2}, \frac{3}{4}, \frac{5}{6}\}$.

For our DQN algorithm, we employ the tanh function as the activation function. Moreover, we use the adaptive ϵ-greedy algorithm to perform actions, where $\epsilon(t) = \max\{\epsilon_{min}, (1-\lambda_\epsilon)\epsilon(t-1)\}$. Here, $\epsilon(0)$ is set to 0.7, ϵ_{min} to 10^{-2}, and λ_ϵ to 10^{-3}. To update the parameter vector θ, we use the RMSprop optimizer with adaptive learning rate $\alpha(t) = (1-\lambda)\alpha(t-1)$, where $\alpha(0) = 5e^{-3}$ is the initial learning rate and $\lambda = 1e^{-3}$ is the learning rate decay. The implementation was

carried out in Python 3 and executed on a 64-bit Windows 10 platform equipped with an Intel Core i7-6700 CPU (3.40 GHz) and 8 GB of RAM.

Table 5.1 Default simulation parameters

Parameter	**Value(s)**
Network's parameters	
Maximum power (P_{max})	30 dBm
Minimum power (P_{min})	5 dBm
RAT AP coverage (R)	300 m
Small region radius (r)	30 m
AWGN PSD	-174 dBm/Hz
Time slot duration (T_s)	20 ms
Number of transmitted symbol (N_p)	1000 packet/frame
Smoothing factor (α_r)	0.1
Scalar parameter (ϕ)	0.4
Level of HWI (σ_{rt})	0.05
Path loss parameters	
Height of RAT AP (H_k)	10 m
Height of IIoT devices (H_n)	1.5 m
Speed of light (c)	3 10^8 m/s
Carrier frequency (f_c)	6.5 GHz

Table 5.2 DQN's hyper-parameters

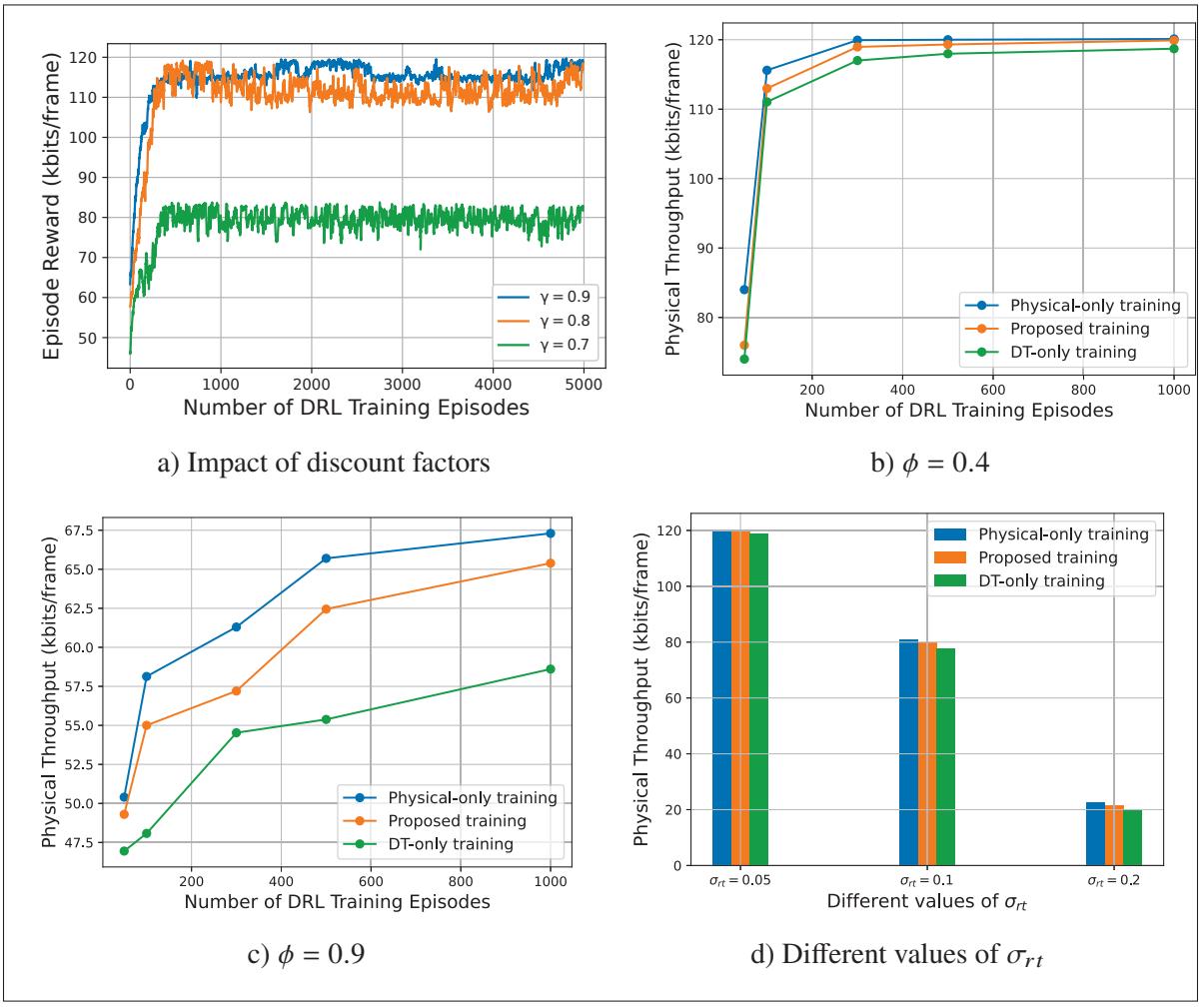
Parameter	**Value**
Episode number (Z)	2000
Learning rate (α)	$5e^{-3}$
Replay memory buffer size (D)	5000
Mini-batch size (D_b)	64
Time step (T_{step})	500

5.7.3 Impact of Discount Factor

Fig. 5.5a evaluates the episodic reward of the DQN agent with respect to the discount factor that determines the relative importance of future rewards compared to the current reward. In other words, the discount factor governs the agent's decision making process by influencing how much weight is assigned to the rewards expected in the future. Therefore, selecting an

Table 5.3 Considered modulation schemes and corresponding SERs

MCS	SER
BPSK	$f_{1,c}(\text{sinr}^t_{n,k}) = Q(\sqrt{2\, G^t_{n,k}\, \text{sinr}^t_{n,k}})$
QPSK	$f_{2,c}(\text{sinr}^t_{n,k}) = 2\,(1-\frac{1}{\sqrt{4}})\, Q(\sqrt{\frac{3\, G^t_{n,k} \log_2(4)\, \text{sinr}^t_{n,k}}{4-1}})$
16QAM	$f_{3,c}(\text{sinr}^t_{n,k}) = 2\,(1-\frac{1}{\sqrt{16}})\, Q(\sqrt{\frac{3\, G^t_{n,k} \log_2(16)\, \text{sinr}^t_{n,k}}{16-1}})$
64QAM	$f_{4,c}(\text{sinr}^t_{n,k}) = 2\,(1-\frac{1}{\sqrt{64}})\, Q(\sqrt{\frac{3\, G^t_{n,k} \log_2(64)\, \text{sinr}^t_{n,k}}{64-1}})$



a) Impact of discount factors

b) $\phi = 0.4$

c) $\phi = 0.9$

d) Different values of σ_{rt}

Figure 5.5 Comparison among different RL training approaches

appropriate value for the discount factor is essential to achieving optimal performance. In this evaluation, we vary the discount factor of the DQN-based resource allocation while keeping

other parameters fixed. Fig. 5.5a shows the DQN's episodic reward (evaluated in the physical twin environment) vs. number of training episodes for different values of the discount factor, such as $\gamma = 0.9$, $\gamma = 0.8$, and $\gamma = 0.7$. As shown in Fig. 5.5a, the highest reward is achieved when $\gamma = 0.9$. Therefore, we choose $\gamma = 0.9$ as the discount factor in the ensuing performance evaluations.

5.7.4 Advantage of the Proposed RL Training Approach

5.7.4.1 Comparison Among Different RL Training Approaches

Here, we train our algorithm for a varying number of DRL training episodes under the following three distinct scenarios:

• Physical-only training (Scenario 1): The DRL model is trained solely in the physical environment.

• Proposed training (Scenario 2): The model is primarily trained in the DT environment, with interactions and experience collection from the physical environment occurring every 100 episodes.

• DT-only training (Scenario 3): The model is trained exclusively in the DT environment.

To compare performance across these scenarios, we test the trained models' performance in terms of the physical throughput[9]. Fig. 5.5b and 5.5c show the physical throughput for each scenario when $\phi = 0.4$ and $\phi = 0.9$, respectively. The results indicate that our proposed training approach achieves throughput levels comparable to those obtained through physical-only training. For instance, as shown in Fig. 5.5b, after 300 episodes of DRL training, scenarios 1, 2, and 3 achieve throughputs of 119.94 kbits/frame, 118.96 kbits/frame, and 117 kbits/frame, respectively. Moreover, increasing the deviation of ACI/HWI (ϕ) from the theoretical model

[9] In our simulations, physical throughput refers to the throughput obtained by applying the selected transmit power, modulation, and coding schemes to the physical IoT network.

widens the performance gap between scenarios 1 and 3, since the DT environment in scenario 3 is unaware of such uncertainties, rendering the trained DRL model sub-optimal in the physical environment. For example, as shown in Fig. 5.5c, after 500 episodes of DRL training, scenarios 1, 2, and 3 achieve throughputs of 65.7 kbits/frame, 62.45 kbits/frame, and 55.38 kbits/frame, respectively. These findings confirm that our DT-empowered continual-learning DRL approach (Algorithm 5.2) achieves high accuracy and adaptability under ACI and HWI uncertainties, while significantly reducing signaling overhead.

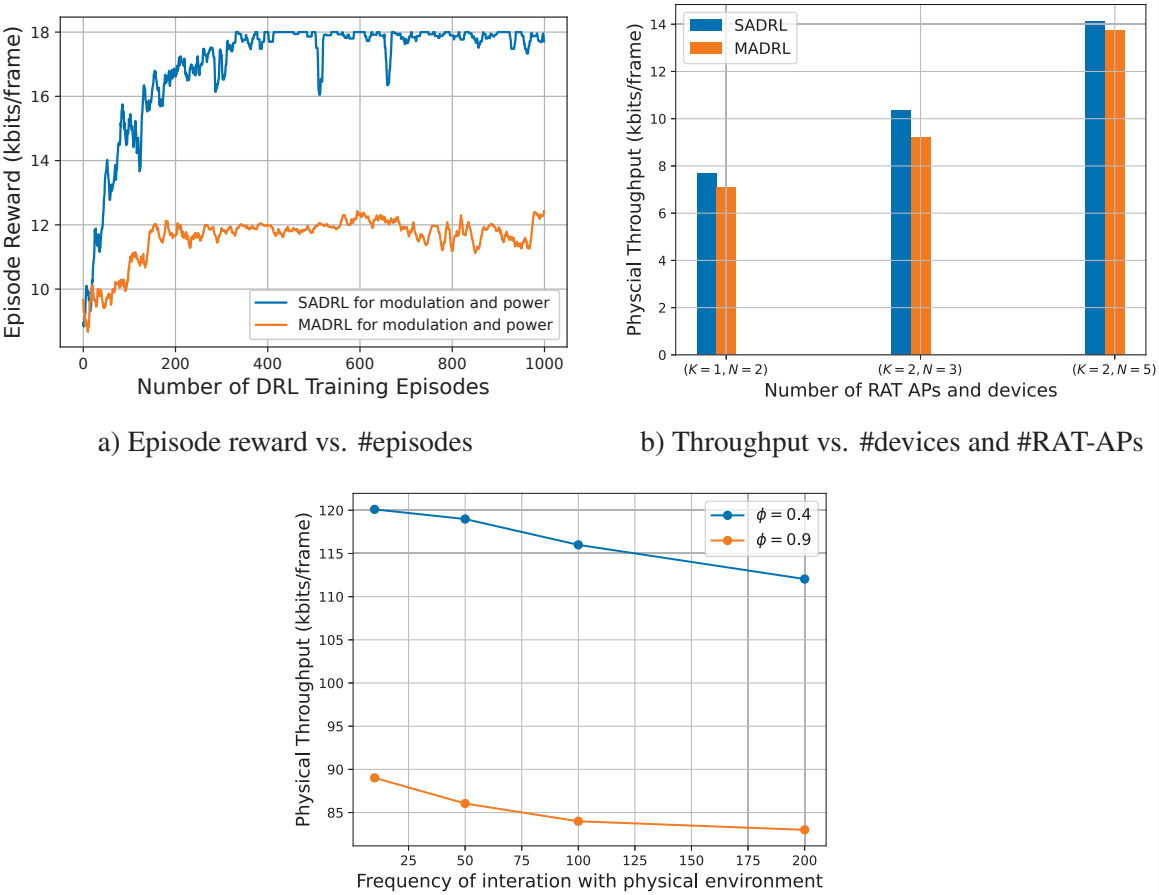
5.7.4.2 Comparison of RL Training Approaches under Varying Impairment Levels

Fig. 5.5d compares the performance of our proposed and other training approaches under different levels of HWIs. The results shown in Fig. 5.5d demonstrate that our approach achieves comparable performance across different impairment levels as when compared to those afforded by training exclusively in the physical environment. For instance, when $\sigma_{rt} = 0.1$, the proposed framework achieves a throughput of 80.98 kbits/frames, 79.86 kbits/frames, and 77.58 kbits/frames for scenario 1, scenario 2, and scenario 3, respectively. Fig. 5.5d also intuitively shows that with an increase in the impairment levels, the achievable throughput decreases.

5.7.5 Advantage of MADRL Framework Compared to SADRL Framework

5.7.5.1 Convergence of SADRL and MADRL for Different Number of Episodes

In this section, we evaluate the performance of a MADRL framework against that of a SADRL framework. To this end, the coding rate for all devices is fixed at $\frac{1}{2}$. The SADRL framework is designed such that a centralized DRL agent receives state information from all devices and performs joint action selection. Given the high computational complexity of SADRL, we consider a scenario with 2 RAT APs and 3 IoT devices. Fig. 5.6a plots the throughput obtained in the DT network, where the interaction with the physical environment occurs every 100 episodes. As shown in Fig. 5.6a, the SADRL framework converges to a higher throughput than the proposed MADRL framework in this specific scenario. The throughput gain of SADRL



a) Episode reward vs. #episodes

b) Throughput vs. #devices and #RAT-APs

c) Throughput vs. #DT-PHY interactions

Figure 5.6 Comparison between SADRL and MADRL scheme and impact of the number of DT-PHY interactions

is intuitive, as SADRL considers global state information and jointly determines the actions of all devices in the network. However, the action space for the SADRL framework increases exponentially with the number of devices. For instance, with 2 RAT APs and 3 devices, the action space size is $(|\mathcal{M}_m| \times |\mathcal{P}|)^3 = 4096$. With the growth in the number of devices, this exponential increase in action space size imposes severe challenges, including excessive computational complexity and memory requirements to explore and optimize the extensive action space, making the SADRL impractical for large-scale multi-RAT IoT networks.

5.7.5.2 Performance Comparison of MADRL vs. SADRL under Varying Numbers of RAT APs and IoT Devices

Fig. 5.6b illustrates the physical throughput achieved by SADRL and MADRL under varying numbers of RAT APs and IoT devices. As shown, SADRL consistently achieves higher throughput than MADRL across different configurations. For instance, when $K = 2$ and $N = 5$, SADRL attains a throughput of 14.13 kbits/frame, compared to 13.73 kbits/frame for MADRL. This performance advantage can be attributed to the centralized nature of SADRL, which jointly determines the link-adaptation parameters for all IoT links. However, as the number of devices increases, the joint action space in SADRL grows exponentially, making training and decision making more challenging. For example, with just 5 devices and 4 power levels and 4 modulation levels, the joint action space size becomes $(4 \times 4)^5 = 1,048,576$. Furthermore, Fig. 5.6b shows that the performance gap between SADRL and MADRL narrows as the networks scales. In contrast to SADRL, the MADRL framework adopts a decentralized learning paradigm in which decision making is distributed among agents. Each agent (i.e., IoT link) in MADRL framework operates within a relatively small local action space based solely on local observations, without requiring information exchange or signaling with neighboring agents. This design enhances scalability, accelerates convergence, and reduces computational overhead with a reasonable loss of optimality, all of which make MADRL well-suited for large-network settings.

5.7.6 Impact of Interaction Frequency with Physical Environment

In this section, we train our algorithm with $Z_K \in \{10, 50, 100, 200\}$, meaning the algorithm interacts with the physical environment every 10, 50, 100, or 200 DRL training episodes. Fig. 5.6c shows the throughput in the physical environment vs. the interaction frequencies for different values of ϕ. As can be seen in Fig. 5.6c, decreasing the interaction frequency with the physical environment leads to a reduction in throughput. This outcome is expected, as the DRL agent has a limited understanding of the physical environment, resulting in less accurate knowledge of its characteristics. For instance, considering $\phi = 0.4$, the proposed framework achieves a throughput of 120.11 kbits/frame and 112.04 kbits/frame when $Z_k = 10$ and $Z_k = 200$,

respectively. However, increasing interaction frequency also incurs higher communication overhead. Therefore, it is crucial to select an optimal interaction frequency. Based on this trade-off, we set $Z_K = 100$ in our experiments, i.e., the DTN periodically collects feedback from the physical network after every 100 episodes of DRL training.

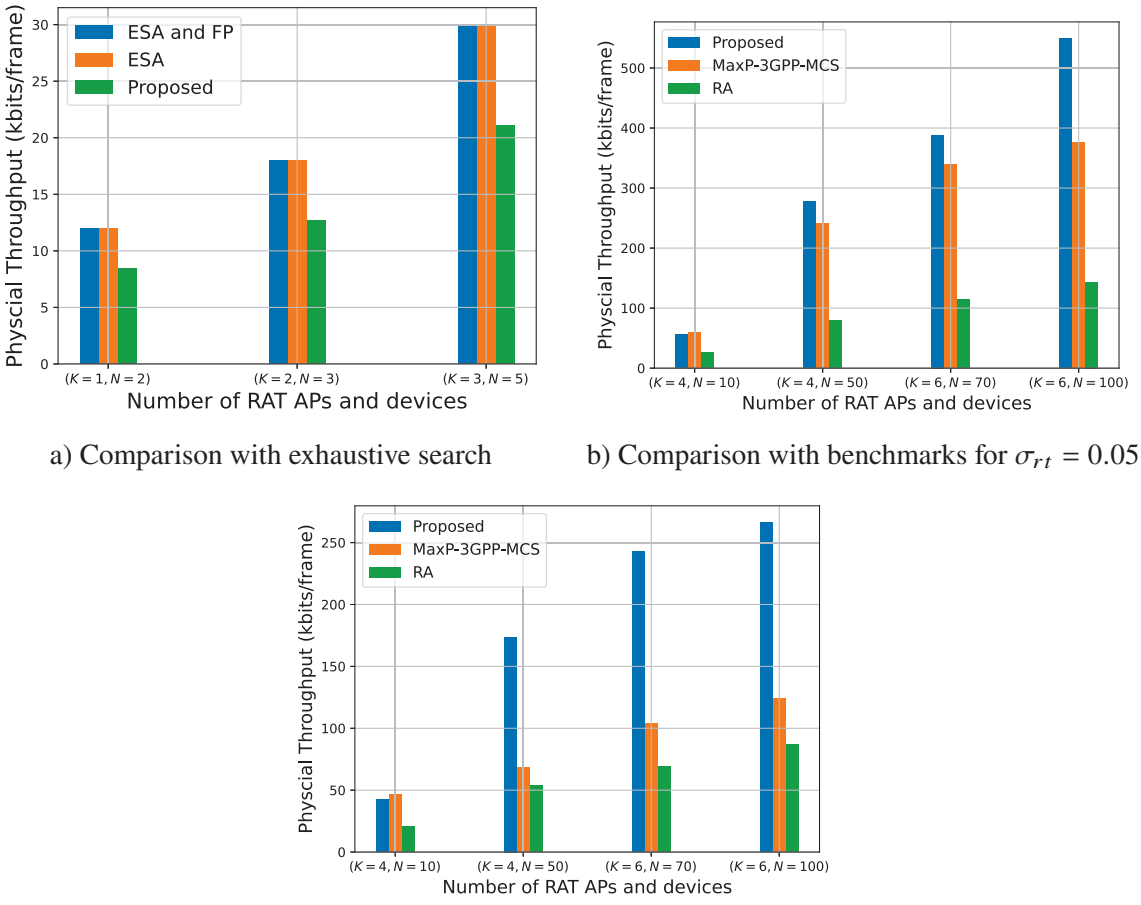
In addition, Fig. 5.6c shows that increasing ϕ decreases the sum throughput. For instance, if $Z_k = 50$ is considered, the proposed framework achieves a throughput of 118.98 kbits/frames and 86.04 kbits/frames when $\phi = 0.4$ and $\phi = 0.9$, respectively. This outcome is expected, as increasing ϕ amplifies uncertainty in modeling ACI and HWI in the DTN, requiring more frequent interactions with the physical network to refine the DQN model.

5.7.7 Performance Comparison Against Benchmark Schemes

5.7.7.1 Comparison with the ESA and ESA-FP Benchmarks

In Fig. 5.7a, we compare performance of the proposed resource allocation algorithm against the exhaustive-search-based optimal schemes. Due to the significant complexity of performing exhaustive search, this evaluation is conducted in a small-scale scenario with a limited number of IoT devices and RAT APs. Both benchmark schemes achieve the same optimal throughput for the considered small-scale scenarios ($(K = 1, N = 2)$, $(K = 2, N = 3)$, and $(K = 3, N = 5)$), while the proposed DRL-based method attains approximately 70.4% of this optimum.

It is noteworthy that both ESA approaches require large computational complexity even for the small network settings. For example, with only three devices, the exhaustive search iterates over $4^3 = 64$ modulation combinations and $3^3 = 27$ coding-rate combinations, yielding $64 \times 27 = 1728$ unique (m, c) configurations. For each of these, FP must be applied to optimize transmit power, further increasing complexity. In contrast, ESA over (m, p, c) requires $1728 \times 64 = 110{,}592$ evaluations, making the complexity orders of magnitude higher. Consequently, for large-scale IoT networks, both ESA-FP and ESA methods become infeasible due to prohibitively increasing



a) Comparison with exhaustive search

b) Comparison with benchmarks for $\sigma_{rt} = 0.05$

c) Comparison with benchmarks for $\sigma_{rt} = 0.2$

Figure 5.7 Throughput comparison between the proposed scheme, exhaustive search, and benchmark link adaptation schemes

computational cost. This observation underscores that our proposed DRL solution achieves a suitable trade-off between performance and computational complexity.

5.7.7.2 Comparison with MaxP-3GPP-MCS and RA Benchmarks

In Fig. 5.7b and 5.7c, we compare the proposed framework with state-of-the-art algorithms under varying numbers of RAT APs and IoT devices. The benchmarks considered are the RA allocation scheme and the MaxP-3GPP-MCS. As the number of RAT APs and IoT devices increases, distinct performance trends emerge. For small-scale settings with few IoT links, the proposed scheme achieves slightly lower throughput than MaxP-3GPP-MCS. This is because

with only a few RAT IoT links, transmissions can be scheduled over non-adjacent RRBs, making ACI negligible. In the absence of ACI, the MaxP-3GPP-MCS strategy can achieve near-optimal sum throughput performance. However, as the number of devices and RAT-APs increases, the number of adjacent interfering links also grows. In this context, the proposed scheme outperforms the benchmark schemes by intelligently optimizing the link adaptation parameters to mitigate the deleterious impact of ACI. For instance, in Fig. 5.7b, when $N = 70$ and $K = 6$, the proposed framework's sum throughput is 49.01 kbits/frame and 274.21 kbits/frame higher than that of the MaxP-3GPP-MCS and RA schemes, respectively. Likewise, the proposed framework achieves a 46% higher sum throughput compared to the MaxP-3GPP-MCS scheme, evaluated for $K = 6$ RAT APs and $N = 100$ IoT devices.

The performance advantage of our framework becomes more pronounced as the HWI level increases (see Fig. 5.7c). Notably, higher impairment levels increase the denominator of the SINR expression. In such scenarios, judicious selection of transmit power, modulation, and coding schemes becomes imperative to enhance the sum throughput. Thanks to its DRL-based link adaptation strategy, which intelligently adjusts parameters by leveraging both current and past states of IoT links, our proposed scheme achieves notable performance gain over the benchmarks for large HWI levels. For instance, when $N = 70$ and $K = 6$, the proposed framework's sum throughput is 138.46 kbits/frame and 173.52 kbits/frame higher than that of the MaxP-3GPP-MCS and RA schemes, respectively. Moreover, under high HWI levels, the proposed framework achieves 114.16% higher sum throughput than the MaxP-3GPP-MCS scheme, in a system with $K = 6$ RAT APs and $N = 100$ IoT devices. Overall, the results confirm robustness of our learning-based framework in dense IoT deployments with practical impairments caused by ACI and HWI.

5.7.8 Run Time Duration of the Proposed Algorithm

The DQN model is saved after training for 2000 episodes, and subsequent testing is conducted using the trained model in a testing scenario. Table 5.4 presents the average decision making time (i.e., time required to compute the power level and select the MCS) for a single device and

Table 5.4 Required time vs. different numbers of IoT devices

	$N = 1$	$N = 20$	$N = 50$	$N = 100$
Time required (ms)	10	12.5	13.9	15.1

a single TS. This duration includes the time required for data collection (e.g., device locations, ACI and channel gain estimation, SINR computation, and PF calculation), device clustering and RRB scheduling, and the final link adaptation decisions. Consistently with our expectations, the execution time increases with the number of devices, since the clustering and scheduling processes must accommodate all existing devices. For instance, the proposed framework achieves an execution time of 12.5 ms for $N = 20$ IoT devices and 15.1 ms for $N = 100$ IoT devices. These observations indicate that the proposed scheme enables efficient resource allocation with minimal execution delay, making it well-suited for timely and effective decision making in multi-RAT IoT networks.

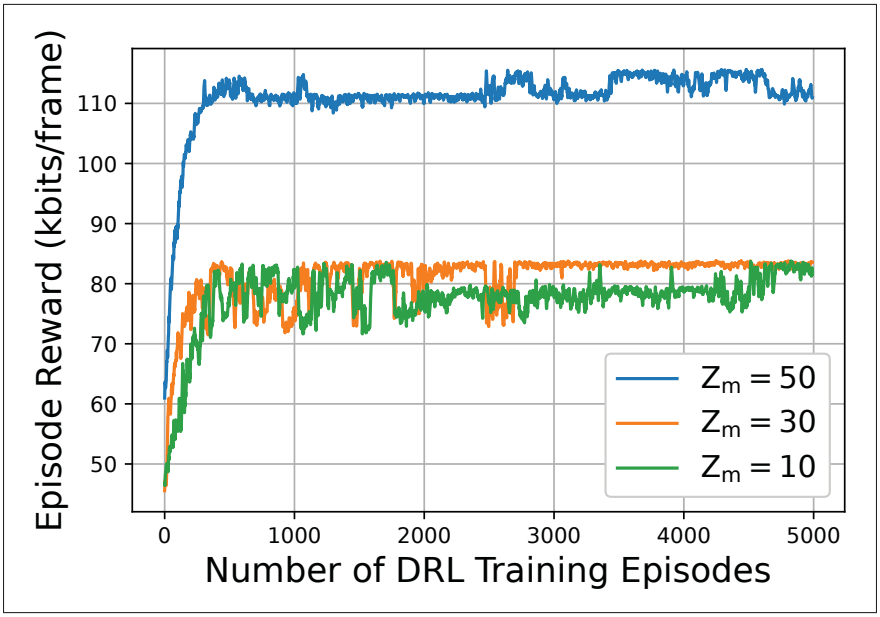


Figure 5.8 Physical throughput vs. DRL training episodes in non-stationary environment

5.7.9 Performance Evaluation of the Proposed Algorithm under Non-Stationary Environment

To evaluate the continual learning capability of the proposed DRL agent, we simulate a non-stationary environment where key parameters, including shadowing (σ_s) and the non-linearity uncertainty parameter ϕ, are kept constant for a block of $Z_m \in \{10, 30, 50\}$ consecutive episodes and then randomly re-sampled for each IoT link at the beginning of the next block. The shadowing component σ_s is independently sampled for each IoT link from a normal distribution with zero mean and standard deviation of 4, 6, or 8 dB, while the non-linearity uncertainty parameter ϕ is randomly selected from the set $\{0, 0.4, 0.6, 0.9\}$. This piecewise-stationary design induces controlled distribution shifts, thus creating a globally non-stationary environment over the entire training horizon. The agent interacts with this environment and updates its policy incrementally using experiences drawn from a shared replay memory. Here, smaller values of Z_m correspond to more frequent environment changes, creating a rapidly varying scenario, while larger values of Z_m represent slower changes, allowing the agent more time to adapt before the next distribution shift. Fig. 5.8 shows that the proposed framework converges under all considered scenarios, while intuitively generating the highest and lowest reward for slow and fast varying non-stationary environments. Consequently, the proposed framework allows the DRL agent to continuously adapt to changing channel conditions while retaining knowledge from previously encountered states.

5.7.10 Discussion on Performance–Complexity Tradeoff

Table 5.5 A qualitative comparison of performance and complexity trade-offs across different schemes

Criteria	ESA-FP	MaxP-3GPP-MCS	RA	Proposed
Complexity	Very High	Low	Low	Moderate
Execution Time	Very long	Low	Low	Low
Scalability	Poor	Excellent	Poor	Excellent
Adaptability	Low	High	Moderate	High
Optimality	Optimal	Suboptimal	Suboptimal	Near-optimal

This subsection analyzes the tradeoff between performance and complexity to identify the most suitable scheme for large-scale IoT deployments. Five key criteria, such as **complexity**, **execution time**, **scalability**, **adaptability**, and **optimality** are considered. Table 5.5 provides a qualitative comparison across the considered schemes. The ESA-FP scheme achieves the highest optimality but comes with very high complexity and execution time, making it impractical beyond small-scale settings. The MaxP-3GPP-MCS and RA baselines offer low complexity but at the cost of reduced optimality, thus serving as feasible but performance-limited solutions. By contrast, the proposed continual MADRL-based method achieves a balanced trade-off across the five criteria. While its computational complexity is slightly higher than that of simple heuristics, it provides a near-optimal solution with reasonably low execution time, remains scalable as the network size increases, and adapts to dynamic IoT conditions with reduced overhead. These features make the proposed scheme instrumental for addressing multi-RAT IoT coexistence in the presence of inevitable ACI and HWIs.

5.8 Conclusion

In this paper, we proposed a DT-enhanced continual DRL framework for radio resource allocation in multiple RAT IoT networks. A radio resource optimization problem was devised aimed at maximizing network sum throughput by jointly optimizing transmit power allocation and MCS selection while considering the challenges posed by ACI and HWI arising from low-cost RF front-end components. By treating each IoT link as an independent learning agent, we devised an MARDL approach to solve this optimization problem in a distributed and computationally efficient manner. Unlike conventional methods, our innovative MADRL training framework uses a DTN to enhance learning efficiency and mitigate real-world deployment risks. Specifically, our framework enables DRL model training within a DTN, with periodic updates based on data collected from the physical network. Such continual learning approach improves the model's adaptability and accuracy in dynamic environments. Our simulation results confirm the following key findings: (i) The proposed MADRL achieves performance close to SADRL, with throughput

differences within 2.8–10.9%. (ii) The continual learning strategy provides up to 14.4% higher throughput compared to DT-only training. (iii) The proposed scheme achieves near-optimal performance—within 70.4% of the ESA-FP benchmark—while maintaining scalability for large-scale IoT deployments. (iv) Compared to the MaxP-3GPP-MCS baseline, the proposed framework achieves up to 46% and 114.16% higher sum throughput in dense IoT networks under two different HWI levels.

CONCLUSION AND RECOMMENDATIONS

6.1 Conclusion and Lessons Learned

With the growing number of IoT and IIoT devices and the increasing complexity of networks, new challenges emerge in managing interference, HWIs, and resource scarcity under dynamic wireless conditions. In this context, the present thesis investigated intelligent and adaptive resource allocation strategies aimed at enhancing efficiency, scalability, and reliability of IoT networks. Our specific focus was on developing distributed DRL-based frameworks that would effectively allocate resources, mitigate interference, and meet stringent QoS demands. In addition, we addressed the impact of FBL constraints and HWIs, proposing solutions to ensure robust and energy-efficient communication in large-scale IoT systems.

In Chapter 2, we analyzed downlink power allocation in F-RANs, taking into account the detrimental effects of HWIs and co-channel interference. A DRL-based distributed framework was developed, where each F-AP operated as an autonomous agent. To further improve the learning process and decision quality, an EL mechanism was integrated, thus enabling the selection of the most effective policy during training. Simulation results demonstrated that the proposed DDPA scheme achieves higher SE than state-of-the-art methods.

In Chapters 3 and 4, we extended this work to IIoT networks, where we investigated joint device clustering and power allocation under FBL constraints. Recognizing stringent QoS demands and sensitivity of IIoT applications to delay and reliability, we proposed a two-step framework combining greedy clustering with a MADRL algorithm. By modeling each cluster as an independent agent, the proposed solution was found to enhance SINR, adaptability, and scalability, thereby addressing critical requirements of industrial automation systems. Simulation results confirmed effectiveness of our proposed scheme and its suitability for large-scale, delay-sensitive IIoT deployments.

Finally, in Chapter 5, we tackled the resource allocation problem in multi-RAT IoT networks, where ACI and HWIs add further complexity. To this end, we developed a MADRL framework where each device operated as an independent learning agent. This design was then found to allow devices to adapt their transmission parameters—such as modulation, power, and coding rate—based solely on local observations. Furthermore, we integrated a DTN to train agents safely in a virtual environment, thereby improving learning stability and reducing real-world risks. This framework underscores the potential of fully decentralized intelligence for future 6G-enabled IoT ecosystems.

Based on the results, the lessons learned in the present thesis can be summarized as follows:

- Effectively addressing both HWIs and interference, including co-channel and ACI, is crucial for robust and scalable resource allocation in large-scale IoT and IIoT networks.
- Distributed DRL proved to be a powerful approach for dynamic and adaptive resource management under time-varying wireless channel conditions and partial observability.
- Integrating EL with DRL agents enhances decision accuracy and accelerates convergence in F-RANs, particularly when optimizing SE in the presence of HWIs.
- Greedy clustering combined with MADRL is an effective strategy for joint device clustering and power allocation in IIoT systems under FBL constraints, enabling reliable low-latency communication.
- Modeling each device as an autonomous agent in multi-RAT environments improves scalability and flexibility, thus enabling the learning of customized resource allocation policies (e.g., modulation, power, and coding rate) based solely on local device observations.
- DTNs provide a safe and effective virtual environment for training distributed learning algorithms, reducing real-world risks, and enhancing policy generalization in practical IoT deployments.

In conclusion, the results of this thesis demonstrate that intelligent, DRL-based distributed frameworks offer promising solutions to overcome limitations of conventional resource allocation strategies in heterogeneous and interference-prone IoT ecosystems. These findings lay down the foundation for future research into scalable, autonomous, and QoS-aware communication systems in next-generation IoT and 6G networks.

6.2 Future Research

In this section, we outline future research directions identified based on the literature review and the findings of this Ph.D. thesis.

6.2.1 Robust Resource Allocation in IoT Networks under Channel Uncertainty Using DRL

Because of practical limitations such as feedback delays, quantization errors, hardware imperfections, and unpredictable environmental dynamics, perfect CSI is rarely available in real-world IoT deployments (Cao, Zhang, Jin & Roy, 2023; Kumar, Jayakody & Upadhyay, 2024). These uncertainties can significantly degrade the performance of resource allocation schemes, especially those that rely on the assumption of accurate and instantaneous CSI. This challenge becomes even more pronounced in fast-fading or highly dynamic environments, which are common in large-scale or mobile IoT scenarios.

Yet, a promising direction for future research is the development of robust DRL-based resource allocation frameworks capable of effectively operating under channel uncertainty. One potential approach involves using partially observable Markov decision processes (POMDPs) to model the decision making process under incomplete or noisy observations. By integrating a POMDP framework into the DRL architecture, learning agents can derive optimal policies that are resilient to inaccuracies in channel measurements and adaptable to dynamic environments.

6.2.2 Federated Deep Reinforcement Learning for Scalable and Privacy-Preserving Resource Allocation in IoT Networks

As IoT networks continue to scale, distributed and privacy-preserving learning methods become increasingly critical. Federated DRL (FDRL) has emerged as a promising solution to these challenges (Guo, Tang & Kato, 2023a). In an FDRL, multiple IoT devices or network entities (e.g., APs or edge nodes) collaboratively train DRL models without sharing raw data. Each agent conducts local training and transmits only model updates—such as gradients or weights—to a central aggregator, which then constructs a global model through aggregation. This decentralized learning paradigm provides the following advantages: it enhances data privacy, reduces communication overhead, and improves scalability in large and heterogeneous IoT deployments (Muhammad Fahad Noman, Dimyati, Ariffin Noordin, Hanafi & Abdrabou, 2024). Future research can focus on developing robust and communication-efficient FDRL frameworks specifically designed for dynamic network environments and devices with varying computational and energy capabilities. Doing so will help to address challenges such as model heterogeneity, asynchronous updates, and robustness to non-IID (non-independent and identically distributed) data distributions across agents.

6.2.3 Mobility-Aware Resource Allocation in Dynamic IoT Environments

In real-world IoT deployments, mobility of both devices and APs introduces significant challenges to maintaining reliable and efficient communication. Mobile IoT devices—such as autonomous vehicles, drones, and wearable electronics—frequently encounter rapidly varying network conditions due to handovers, fluctuating interference patterns, and changes in channel quality (Biswas & Wang, 2023). Similarly, mobile APs, including UAV-mounted BSs and moving edge servers, add further dynamic complexity. However, most available resource allocation schemes assume quasi-static network topologies, which limits their applicability in highly mobile environments.

In this context, a promising direction for future research is the development of mobility-aware resource allocation frameworks that would incorporate predictive models to anticipate device trajectories and AP movement patterns. By integrating mobility prediction into the decision making process of DRL agents, these frameworks will enable proactive and adaptive resource management, thereby enhancing QoS, minimizing handover disruptions, and reducing communication latency in dynamic IoT scenarios.

6.2.4 Towards Distributed and Context-Aware Resource Management for 6G Networks

The upcoming 6G wireless network requires a paradigm shift in network operation and control. Unlike its predecessors, 6G will not be a single network, but instead a collection of interconnected networks, including both terrestrial and non-terrestrial components, presenting a range of complex challenges. The centralized resource management approach used in current 5G systems frequently results in increased latency, which is unacceptable for many 6G applications, such as brain-computer interaction (Akbar, Hussain, Ikram, Sheng & Mukhopadhyay, 2025) and human sense-based technologies (Alwis *et al.*, 2021). These applications require ultra-low latency in the range of 10–100 ms. Consequently, adopting distributed resource management becomes imperative for 6G systems. Recent advancements in ML, such as DRL and federated learning (FL), offer promising pathways for transitioning from centralized to distributed allocation frameworks, ensuring more efficient and low-latency resource management. However, training such solutions is associated with significant challenges. Developing ML algorithms for 6G resource management requires extensive data, and training in real-world scenarios is often infeasible due to safety concerns, high exploration costs, and the challenges associated with data collection. Traditional network simulators model networks in simplified ways, thus overlooking dynamic, site-specific contexts. Consequently, ML models trained with such simulators lack the optimality needed for complex and dynamic wireless networking environments. In these

settings, DTNs provide a promising solution to these challenges by enabling real-time situational awareness and emulating physical wireless networks within a virtual environment. DTNs also produce realistic datasets for training AI and ML models and support the safe training of RL algorithms without disrupting the actual wireless network's operations. Importantly, the 3GPP has recognized DTNs as crucial tools for generating trustworthy data and training ML models for next-generation wireless networks (3GPP Technical Specification Group Services and System Aspects, 2024). Therefore, to conclude, DTNs are pivotal tools in developing context-aware resource management systems for next-generation wireless networks, addressing critical challenges and enabling the seamless integration of AI-driven solutions into 6G.

6.2.5 Explainable Reinforcement Learning for Resource Allocation in IoT Networks

As DRL-based resource allocation frameworks grow increasingly complex and data-driven, one significant limitation persists, namely, the lack of interpretability, which is critical for ensuring trust, accountability, and reliability—particularly in safety- and latency-sensitive IoT applications (Tang, Mao, Kato & Gui, 2021). The "black-box" nature of DNNs complicates understanding the rationale behind specific decisions or actions taken by the learning agent. Explainable reinforcement learning (XRL) has emerged as a promising research direction aimed at addressing this limitation by offering human-interpretable insights into the behavior of DRL agents (Sun *et al.*, 2025). Therefore, future research can focus on integrating XRL techniques into DRL-based resource allocation frameworks to enhance transparency and operational trust. For instance, XRL can provide post-hoc explanations of decisions, such as why certain power levels, subcarrier assignments, or modulation schemes were selected under specific network conditions. In addition, incorporating explainability into resource management can enable human-in-the-loop optimization, where expert feedback is used to validate, guide, or refine the policy learned by the agent. This collaborative approach not only improves transparency,

but also contributes to safer, more adaptable, and trustworthy decision making in dynamic and complex IoT environments.

APPENDIX I

APPENDIX OF CHAPTER 3

The completion of the proof for Proposition 1 involves reducing the optimal coalition structure (OCS) problem to $P0$. To this end, we first describe the OCS problem as follows. Let A denote a set of K distinct agents, which are denoted as $\mathrm{A} = \{a_1, \cdots, a_K\}$. A coalition structure (CS) is defined as the partitioning of agents into disjoint coalitions, with each agent belonging to exactly one coalition. Given a CS denoted as CS and value $v_s \geq 0$ assigned to each coalition, where $s \in CS$, the total value of CS is expressed as $V(CS) = \sum_{S \in CS} v_{S_n}$. The objective of the OCS problem is to find the optimal CS^*, where $CS^* = \arg\max_{CS \in U} V(CS)$, with U representing the universal set of all possible CSs. In other words, the OCS problem aims to find the CS with the highest value.

To reduce the OCS problem to an instance of $P0$, we make the initial assumption that the transmit power allocation of all the APs in $P0$ is known. In this scenario, $P0$ becomes a device-to-cluster association problem. Each device is considered an agent in set A. Thus, the set of devices associated with a cluster forms a coalition, and these coalitions are disjoint since a device can be associated with only one cluster. Each AP in the IIoT network has a total of N clusters, and a CS consists of at most N disjoint coalitions, which can be expressed as $CS = \{S_1, S_2, \cdots, S_N\}$, where S_n represents the set of devices associated with the n-th cluster. The value of coalition S_n is defined as $v_{S_n} = R_{C_n}^{(m)}$, with $R_{C_n}^{(m)}$ given by (3.17). Consequently, the total value of CS is determined as $V(CS) = \sum_{S_n \in CS} v_{S_n} = \sum_{n=1}^{N} \mathrm{R}_{C_n}^{(m)}$.

With this reduction, an optimal solution to $P0$ determines the optimal set of device clusters (or optimal CS) that has the maximum system capacity (or largest CS value). Clearly, an optimal solution to $P0$ corresponds to an optimal solution to the OCS problem. Stated differently, the ability to optimally solve $P0$ in polynomial time implies it is possible to solve the OCS problem optimally in polynomial time. As the OCS problem is known to be NP-hard (Sandholm, Larson, Andersson, Shehory & Tohmé, 1999, Proposition 2), it follows that $P0$ is also NP-hard.

APPENDIX II

APPENDIX OF CHAPTER 5

By definition, an optimization problem is NP-hard if any instance of a known NP-complete problem can be reduced to it in polynomial time. Of note, the MCS for the coexisting IoT links can be typically determined using a traditional lookup table approach (e.g., (3GPP, 2024, Table 7.2.3-1)). This method, which maps a reported CQI to a suitable MCS, operates in polynomial time and does not account for the detrimental effects of ACI and HWI-induced distortions. Assuming that the MCS selection for all IoT devices is obtained via this lookup table, the original problem P0 can be reduced to a transmit power allocation problem, denoted by $\hat{\text{P0}}$ (see Eq. (A II-1)):

$$\begin{aligned} \hat{\text{P0}} \quad & \max_{\boldsymbol{P}^t_{\boldsymbol{n,k}}} \sum_{n=1}^{N} \quad R^t_{n,k} \\ & \text{s.t. C1: } P_{min} \leq P^t_{n,k} \leq P_{max}, \forall n \in \mathcal{N} \end{aligned} \tag{A II-1}$$

Here, $\hat{\text{P0}}$: aims to maximize the network's total capacity via power allocation only. Since the reduction from P0 to $\hat{\text{P0}}$ involves only the polynomial-time lookup table for MCS determination, this reduction is itself polynomial.

It is known that optimizing a sum of functions of ratios (as in $\hat{\text{P0}}$) is a classic NP-complete problem (Shen & Yu, 2018). Accordingly, P0 can be reduced in polynomial time to an NP-complete problem. Hence, P0 must be an NP-hard optimization problem. □

BIBLIOGRAPHY

(2009). *Derivation of a Block Edge Mask (BEM) for Terminal Stations in the 2.6 GHz Frequency Band (2500-2690 MHz)*. Dublin.

3GPP. (2017). *Universal Mobile Telecommunications System (UMTS); Radio Frequency (RF) system scenarios (3GPP TR 25.942 version 14.0.0 Release 14)*. ETSI.

3GPP. (2020). *5G; Study on channel model for frequencies from 0.5 to 100 GHz (3GPP TR 38.901 version 16.1.0 Release 16)*. ETSI.

3GPP. (2024). *Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Layer Procedures (Release 18)*.

3GPP Technical Specification Group Services and System Aspects. (2024). *Management and orchestration; Study on management aspects of Network Digital Twin (Release 19)*.

Abanto-Leon, L. F., Krishnamoorthy, A., Garcia-Saavedra, A., Sim, G. H., Schober, R. & Hollick, M. (2024). Radio Resource Management Design for RSMA: Optimization of Beamforming, User Admission, and Discrete/Continuous Rates With Imperfect SIC. *IEEE Transactions on Mobile Computing*, 23(12), 11498-11518. doi: 10.1109/TMC.2024.3396389.

Abdelnasser, A., Hossain, E. & Kim, D. I. (2014). Clustering and Resource Allocation for Dense Femtocells in a Two-Tier Cellular OFDMA Network. *IEEE Transactions on Wireless Communications*, 13(3), 1628-1641. doi: 10.1109/TW.2014.011614.131163.

Ahmadzadeh, S., Parr, G. & Zhao, W. (2021). A Review on Communication Aspects of Demand Response Management for Future 5G IoT- Based Smart Grids. *IEEE Access*, 9, 77555-77571. doi: 10.1109/ACCESS.2021.3082430.

Ahmed, K. I. & Hossain, E. (2019). A Deep Q-Learning Method for Downlink Power Allocation in Multi-Cell Networks. Retrieved from: https://arxiv.org/abs/1904.13032.

Akbar, M. S., Hussain, Z., Ikram, M., Sheng, Q. Z. & Mukhopadhyay, S. C. (2025). On challenges of sixth-generation (6G) wireless networks: A comprehensive survey of requirements, applications, and security issues. *Journal of Network and Computer Applications*, 233, 104040. doi: https://doi.org/10.1016/j.jnca.2024.104040.

Alwarafy, A., Abdallah, M., Ciftler, B. S., Al-Fuqaha, A. & Hamdi, M. (2022). Deep Reinforcement Learning for Radio Resource Allocation and Management in Next Generation Heterogeneous Wireless Networks: A Survey. *IEEE Open Journal of the Communications Society*, 3, 322–365. doi: 10.1109/ojcoms.2022.3153226.

Alwis, C. D., Kalla, A., Pham, Q.-V., Kumar, P., Dev, K., Hwang, W.-J. & Liyanage, M. (2021). Survey on 6G Frontiers: Trends, Applications, Requirements, Technologies and Future Research. *IEEE Open Journal of the Communications Society*, 2, 836-886. doi: 10.1109/OJCOMS.2021.3071496.

Andrews, J. G., Buzzi, S., Choi, W., Hanly, S. V., Lozano, A., Soong, A. C. K. & Zhang, J. C. (2014). What Will 5G Be? *IEEE Journal on Selected Areas in Communications*, 32(6), 1065-1082. doi: 10.1109/JSAC.2014.2328098.

Andrews, J. G., Humphreys, T. E. & Ji, T. (2024). 6G Takes Shape. *IEEE BITS the Information Theory Magazine*, 4(1), 2-24. doi: 10.1109/MBITS.2024.3504521.

Ansere, J. A., Gyamfi, E., Li, Y., Shin, H., Dobre, O. A., Hoang, T. & Duong, T. Q. (2023). Optimal Computation Resource Allocation in Energy-Efficient Edge IoT Systems With Deep Reinforcement Learning. *IEEE Transactions on Green Communications and Networking*, 7(4), 2130-2142. doi: 10.1109/TGCN.2023.3286914.

Baghel, A., Singh, D., Parihar, A. S., Bhatia, V., Rajatheva, N. & Latva-aho, M. (2025). Robustness of NOMA in HetNets: Impact of Hardware Impairments and Imperfect CSI. *IEEE Wireless Communications Letters*, 1-1. doi: 10.1109/LWC.2025.3605209.

Baldesi, L., Restuccia, F. & Melodia, T. (2022). ChARM: NextG Spectrum Sharing Through Data-Driven Real-Time O-RAN Dynamic Control. *IEEE INFOCOM 2022 - IEEE Conference on Computer Communications*, pp. 240-249. doi: 10.1109/INFOCOM48880.2022.9796985.

Bany Salameh, H. A., Almajali, S., Ayyash, M. & Elgala, H. (2018). Spectrum Assignment in Cognitive Radio Networks for Internet-of-Things Delay-Sensitive Applications Under Jamming Attacks. *IEEE Internet of Things Journal*, 5(3), 1904-1913. doi: 10.1109/JIOT.2018.2817339.

Bauwens, J., Ruckebusch, P., Giannoulis, S., Moerman, I. & Poorter, E. D. (2020a). Over-the-Air Software Updates in the Internet of Things: An Overview of Key Principles. *IEEE Communications Magazine*, 58(2), 35-41. doi: 10.1109/MCOM.001.1900125.

Bauwens, J., Ruckebusch, P., Giannoulis, S., Moerman, I. & Poorter, E. D. (2020b). Over-the-Air Software Updates in the Internet of Things: An Overview of Key Principles. *IEEE Communications Magazine*, 58(2), 35-41. doi: 10.1109/MCOM.001.1900125.

Ben Ghorbel, M., Berscheid, B., Bedeer, E., Hossain, M. J., Howlett, C. & Cheng, J. (2018). Principal Component-Based Approach for Profile Optimization Algorithms in DOCSIS 3.1. *IEEE Transactions on Network and Service Management*, 15(3), 934-945. doi: 10.1109/TNSM.2018.2828704.

Benhiba, B. E., Madi, A. A. & Addaim, A. (2018). Comparative Study of The Various new Cellular IoT Technologies. *2018 International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, pp. 1-4. doi: 10.1109/ICECOCS.2018.8610508.

Bennis, M. & Niyato, D. (2010). A Q-learning based approach to interference avoidance in self-organized femtocell networks. *2010 IEEE Globecom Workshops*, pp. 706-710. doi: 10.1109/GLOCOMW.2010.5700414.

Biswas, A. & Wang, H.-C. (2023). Autonomous Vehicles Enabled by the Integration of IoT, Edge Intelligence, 5G, and Blockchain. *Sensors*, 23(4). doi: 10.3390/s23041963.

Boshkovska, E., Ng, D. W. K., Dai, L. & Schober, R. (2018). Power-Efficient and Secure WPCNs With Hardware Impairments and Non-Linear EH Circuit. *IEEE Transactions on Communications*, 66(6), 2642-2657. doi: 10.1109/TCOMM.2017.2783628.

Bruno, R., Masaracchia, A. & Passarella, A. (2014). Robust Adaptive Modulation and Coding (AMC) Selection in LTE Systems Using Reinforcement Learning. *2014 IEEE 80th Vehicular Technology Conference (VTC2014-Fall)*, pp. 1-6. doi: 10.1109/VTCFall.2014.6966162.

Busoniu, L., Babuska, R. & De Schutter, B. (2008). A Comprehensive Survey of Multiagent Reinforcement Learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(2), 156-172. doi: 10.1109/TSMCC.2007.913919.

Cao, J., Zhu, X., Jiang, Y., Liu, Y. & Zheng, F.-C. (2020). Short Frame Structure Optimization for Industrial IoT with Heterogeneous Traffic and Shared Pilot. *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, pp. 1-6. doi: 10.1109/GLOBECOM42002.2020.9348000.

Cao, L., Zhang, L., Jin, S. & Roy, S. (2023). Efficient MIMO PHY Abstraction With Imperfect CSI for Fast Simulations. *IEEE Wireless Communications Letters*, 12(3), 530-534. doi: 10.1109/LWC.2022.3233542.

Cao, Y., Zhang, L. & Liang, Y.-C. (2019). Deep Reinforcement Learning for Channel and Power Allocation in UAV-enabled IoT Systems. *2019 IEEE Global Communications Conference (GLOBECOM)*, pp. 1-6. doi: 10.1109/GLOBECOM38437.2019.9014055.

Chang, H.-H. (2021). *Deep Reinforcement Learning for Next Generation Wireless Networks with Echo State Networks*. (Ph.D. dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA). Retrieved from: Available online if applicable.

Chen, Y., Yang, Z., Zhang, J. & Alouini, M.-S. (2021). Further Results on Detection and Channel Estimation for Hardware Impaired Signals. *IEEE Transactions on Communications*, 69(11), 7167-7179. doi: 10.1109/TCOMM.2021.3100430.

Cheng, P., Chen, Y., Ding, M., Chen, Z., Liu, S. & Chen, Y.-P. P. (2023). Deep Reinforcement Learning for Online Resource Allocation in IoT Networks: Technology, Development, and Future Challenges. *IEEE Communications Magazine*, 61(6), 111-117. doi: 10.1109/MCOM.001.2200526.

Chi, H. R., Wu, C. K., Huang, N.-F., Tsang, K.-F. & Radwan, A. (2023). A Survey of Network Automation for Industrial Internet-of-Things Toward Industry 5.0. *IEEE Transactions on Industrial Informatics*, 19(2), 2065-2077. doi: 10.1109/TII.2022.3215231.

Chiang, M., Hande, P., Lan, T. & Tan, C. W. (2008). Power Control in Wireless Cellular Networks. *Foundations and Trends® in Networking*, 2(4), 381-533. doi: 10.1561/1300000009.

Chu, Z., Zhong, J., Xiao, P., Mi, D., Hao, W., Tafazolli, R. & Feresidis, A. P. (2022). RIS Assisted Wireless Powered IoT Networks With Phase Shift Error and Transceiver Hardware Impairment. *IEEE Transactions on Communications*, 70(7), 4910-4924. doi: 10.1109/TCOMM.2022.3175833.

Cisco. (2016, June). Internet of things. Retrieved from: http://www.audentia-gestion.fr/cisco/pdf/at-a-glance-c45-731471.pdf.

Clerckx, B., Mao, Y., Jorswieck, E. A., Yuan, J., Love, D. J., Erkip, E. & Niyato, D. (2023). A Primer on Rate-Splitting Multiple Access: Tutorial, Myths, and Frequently Asked Questions. *IEEE Journal on Selected Areas in Communications*, 41(5), 1265-1308. doi: 10.1109/JSAC.2023.3242718.

Cui, J., Liu, Y. & Nallanathan, A. (2020). Multi-Agent Reinforcement Learning-Based Resource Allocation for UAV Networks. *IEEE Transactions on Wireless Communications*, 19(2), 729-743. doi: 10.1109/TWC.2019.2935201.

Cui, Y., Lv, T., Ni, W. & Jamalipour, A. (2023). Digital Twin-Aided Learning for Managing Reconfigurable Intelligent Surface-Assisted, Uplink, User-Centric Cell-Free Systems. *IEEE Journal on Selected Areas in Communications*, 41(10), 3175-3190. doi: 10.1109/JSAC.2023.3310050.

Dahrouj, H. & Yu, W. (2011). Multicell Interference Mitigation with Joint Beamforming and Common Message Decoding. *IEEE Transactions on Communications*, 59(8), 2264-2273. doi: 10.1109/TCOMM.2011.060911.100554.

Dai, L., Wang, B., Yuan, Y., Han, S., Chih-lin, I. & Wang, Z. (2015). Non-orthogonal multiple access for 5G: solutions, challenges, opportunities, and future research trends. *IEEE Communications Magazine*, 53(9), 74-81. doi: 10.1109/MCOM.2015.7263349.

Dai, L., Wang, B., Ding, Z., Wang, Z., Chen, S. & Hanzo, L. (2018). A Survey of Non-Orthogonal Multiple Access for 5G. *IEEE Communications Surveys and Tutorials*, 20(3), 2294-2323. doi: 10.1109/COMST.2018.2835558.

Davaslioglu, K., Kompella, S., Erpek, T. & Sagduyu, Y. E. (2024). Continual Deep Reinforcement Learning to Prevent Catastrophic Forgetting in Jamming Mitigation. *MILCOM 2024 - 2024 IEEE Military Communications Conference (MILCOM)*, pp. 740-745. doi: 10.1109/MILCOM61039.2024.10773861.

Deng, L., Hinton, G. & Kingsbury, B. (2013). New types of deep neural network learning for speech recognition and related applications: an overview. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 8599-8603. doi: 10.1109/ICASSP.2013.6639344.

Dinh, T. H. L., Kaneko, M., Fukuda, E. H. & Boukhatem, L. (2021). Energy Efficient Resource Allocation Optimization in Fog Radio Access Networks With Outdated Channel Knowledge. *IEEE Transactions on Green Communications and Networking*, 5(1), 146-159. doi: 10.1109/TGCN.2020.3034638.

Elloumi, M., Hassan, M. Z. & Kaddoum, G. (2025a). Spectrum Sharing in Internet-of-Vehicles Networks: Digital Twin-Empowered Proactive Interference Management Approach. *IEEE Transactions on Network and Service Management*, 1-1. doi: 10.1109/TNSM.2025.3541977.

Elloumi, M., Kaddoum, G., Hassan, M. Z. & Selim, B. (2025b). Digital Twin-Empowered Interference Management for Multi-hop Internet-of-Vehicles Networks Over Millimeter Wave Bands. *IEEE Internet of Things Journal*, 1-1. doi: 10.1109/JIOT.2025.3540750.

Fan, Z., Gu, X., Nie, S. & Chen, M. (2017). D2D power control based on supervised and unsupervised learning. *2017 3rd IEEE International Conference on Computer and Communications (ICCC)*, pp. 558-563. doi: 10.1109/CompComm.2017.8322607.

Fitriasari, H. I., Lestari, A. I., Luhurkinanti, D. L. & Sari, R. F. (2021). Performance Evaluation of Downlink Multi-user OFDMA Scheduling in 5G New Radio (NR). *2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pp. 219-223. doi: 10.1109/ECTI-CON51831.2021.9454694.

François-Lavet, V., Fonteneau, R. & Ernst, D. (2016). How to Discount Deep Reinforcement Learning: Towards New Dynamic Strategies. Retrieved from: https://arxiv.org/abs/1512.02011.

Ge, J., Liang, Y.-C., Joung, J. & Sun, S. (2020). Deep Reinforcement Learning for Distributed Dynamic MISO Downlink-Beamforming Coordination. *IEEE Transactions on Communications*, 68(10), 6070-6085. doi: 10.1109/TCOMM.2020.3004524.

Ghanem, W. R., Jamali, V., Sun, Y. & Schober, R. (2019). Resource Allocation for Multi-User Downlink URLLC-OFDMA Systems. *2019 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1-6. doi: 10.1109/ICCW.2019.8756746.

Guo, Q., Tang, F. & Kato, N. (2023a). Federated Reinforcement Learning-Based Resource Allocation for D2D-Aided Digital Twin Edge Networks in 6G Industrial IoT. *IEEE Transactions on Industrial Informatics*, 19(5), 7228-7236. doi: 10.1109/TII.2022.3227655.

Guo, Q., Tang, F. & Kato, N. (2023b). Resource Allocation for Aerial Assisted Digital Twin Edge Mobile Network. *IEEE Journal on Selected Areas in Communications*, 41(10), 3070-3079. doi: 10.1109/JSAC.2023.3310065.

Gupta, J. K., Egorov, M. & Kochenderfer, M. (2017). Cooperative Multi-agent Control Using Deep Reinforcement Learning. *Autonomous Agents and Multiagent Systems*, pp. 66–83.

Guruacharya, S., Niyato, D., Hossain, E. & Kim, D. I. (2010). Hierarchical Competition in Femtocell-Based Cellular Networks. *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, pp. 1-5. doi: 10.1109/GLOCOM.2010.5683278.

Haider, M., Ahmed, I., Hassan, Z., O'Shea, T. J., Liu, L. & Rawat, D. B. (2025). Digital Twin Enabled Site Specific Channel Precoding: Over the Air CIR Inference. Retrieved from: https://arxiv.org/abs/2501.16504.

Han, Z., Niyato, D., Saad, W. & Başar, T. (2019). *Game theory for next generation wireless and communication networks: Modeling, analysis, and design*. Cambridge University Press.

Hartung, S. (2024). *State of IoT 2024: Number of connected IoT devices growing 13% to 18.8 billion globally* [Version]. Retrieved from: https://iot-analytics.com/wp/wp-content/uploads/2024/09/INSIGHTS-RELEASE-Number-of-connected-IoT-devices-vf.pdf.

Hassan, M. Z., Hossain, M. J., Cheng, J. & Leung, V. C. M. (2021a). Joint Throughput-Power Optimization of Fog-RAN Using Rate-Splitting Multiple Access and Reinforcement-Learning Based User Clustering. *IEEE Transactions on Vehicular Technology*, 70(8), 8019-8036. doi: 10.1109/TVT.2021.3090083.

Hassan, M. Z., Hossain, M. J., Cheng, J. & Leung, V. C. M. (2021b). Energy-Spectrum Efficient Content Distribution in Fog-RAN Using Rate-Splitting, Common Message Decoding, and 3D-Resource Matching. *IEEE Transactions on Wireless Communications*, 20(8), 4929-4946. doi: 10.1109/TWC.2021.3063283.

Hassan, M. Z., Hossain, M. J., Cheng, J. & Leung, V. C. M. (2021c). Device-Clustering and Rate-Splitting Enabled Device-to-Device Cooperation Framework in Fog Radio Access Network. *IEEE Transactions on Green Communications and Networking*, 5(3), 1482-1501. doi: 10.1109/TGCN.2021.3079369.

Hassan, M. Z., Kaddoum, G. & Akhrif, O. (2022). Interference Management in Cellular-Connected Internet of Drones Networks With Drone-Pairing and Uplink Rate-Splitting Multiple Access. *IEEE Internet of Things Journal*, 9(17), 16060-16079. doi: 10.1109/JIOT.2022.3152382.

Hatoum, A., Langar, R., Aitsaadi, N., Boutaba, R. & Pujolle, G. (2014). Cluster-Based Resource Management in OFDMA Femtocell Networks With QoS Guarantees. *IEEE Transactions on Vehicular Technology*, 63(5), 2378-2391. doi: 10.1109/TVT.2013.2290125.

He, S., An, Z., Zhu, J., Zhang, J., Huang, Y. & Zhang, Y. (2021). Beamforming Design for Multiuser uRLLC With Finite Blocklength Transmission. *IEEE Transactions on Wireless Communications*, 20(12), 8096-8109. doi: 10.1109/TWC.2021.3090197.

Hetting, C. (2023). Wi-Fi Industry Scores Important 6 GHz Victory at WRC-23. Retrieved from: https://wifinowglobal.com/news-blog/our-take-wi-fi-industry-scores-important-6-ghz-victory-at-wrc-23/.

Huang, Y., Zhang, C., Wang, J., Jing, Y., Yang, L. & You, X. (2018). Signal Processing for MIMO-NOMA: Present and Future Challenges. *IEEE Wireless Communications*, 25(2), 32-38. doi: 10.1109/MWC.2018.1700108.

Hussain, F., Hassan, S. A., Hussain, R. & Hossain, E. (2020). Machine Learning for Resource Management in Cellular and IoT Networks: Potentials, Current Solutions, and Open Challenges. *IEEE Communications Surveys and Tutorials*, 22(2), 1251-1275. doi: 10.1109/COMST.2020.2964534.

International Telecommunication Union (ITU). [Available: https://www.itu.int/wrc-23/]. (2024). Report on Activities and Proposals for the World Radiocommunication Conference 2024 (WRC-24).

Jamshidiha, S., Pourahmadi, V., Mohammadi, A. & Bennis, M. (2020). Link-Level Throughput Maximization Using Deep Reinforcement Learning. *IEEE Networking Letters*, 2(3), 101-105. doi: 10.1109/LNET.2020.3000334.

Jang, J. & Yang, H. J. (2020). Deep Reinforcement Learning-Based Resource Allocation and Power Control in Small Cells With Limited Information Exchange. *IEEE Transactions on Vehicular Technology*, 69(11), 13768-13783. doi: 10.1109/TVT.2020.3027013.

Javed, S., Amin, O., Shihada, B. & Alouini, M.-S. (2019). Improper Gaussian Signaling for Hardware Impaired Multihop Full-Duplex Relaying Systems. *IEEE Transactions on Communications*, 67(3), 1858-1871.

Jayati, A. E. & Sipan, M. (2020). Impact of Nonlinear Distortion with the Rapp Model on the GFDM System. *2020 Third International Conference on Vocational Education and Electrical Engineering (ICVEE)*, pp. 1-5. doi: 10.1109/ICVEE50212.2020.9243295.

Joris, L., Dupont, F., Laurent, P., Bellier, P., Stoukatch, S. & Redouté, J.-M. (2019). An Autonomous Sigfox Wireless Sensor Node for Environmental Monitoring. *IEEE Sensors Letters*, 3(7), 01-04. doi: 10.1109/LSENS.2019.2924058.

Kakkavas, G., Nyarko, K. N., Lahoud, C., Kühnert, D., Küffner, P., Gabriel, M., Ehsanfar, S., Diamanti, M., Karyotis, V., Mößner, K. & Papavassiliou, S. (2022). Teleoperated Support for Remote Driving over 5G Mobile Communications. *2022 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*, pp. 280-285. doi: 10.1109/MeditCom55741.2022.9928745.

Kapoor, S. (2018). Multi-Agent Reinforcement Learning: A Report on Challenges and Approaches. *CoRR*, abs/1807.09427. Retrieved from: http://arxiv.org/abs/1807.09427.

Katwe, M., Singh, K., Clerckx, B. & Li, C.-P. (2022). Rate-Splitting Multiple Access and Dynamic User Clustering for Sum-Rate Maximization in Multiple RISs-Aided Uplink mmWave System. *IEEE Transactions on Communications*, 70(11), 7365-7383. doi: 10.1109/TCOMM.2022.3211975.

Kaur, A. & Kumar, K. (2020). Energy-Efficient Resource Allocation in Cognitive Radio Networks Under Cooperative Multi-Agent Model-Free Reinforcement Learning Schemes. *IEEE Transactions on Network and Service Management*, 17(3), 1337-1348. doi: 10.1109/TNSM.2020.3000274.

Khalifeh, A., Aldahdouh, K. A., Darabkh, K. A. & Al-Sit, W. (2019). A Survey of 5G Emerging Wireless Technologies Featuring LoRaWAN, Sigfox, NB-IoT and LTE-M. *2019 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*, pp. 561-566. doi: 10.1109/WiSPNET45539.2019.9032817.

Kim, M. & Ko, I.-Y. (2015). An Efficient Resource Allocation Approach Based on a Genetic Algorithm for Composite Services in IoT Environments. *2015 IEEE International Conference on Web Services*, pp. 543-550. doi: 10.1109/ICWS.2015.78.

Kumar, A., Jayakody, D. N. K. & Upadhyay, R. K. (2024). Secure and Reliable IoT Communications in Underlay CRN With Imperfect CSI. *IEEE Internet of Things Journal*, 11(11), 20531-20546. doi: 10.1109/JIOT.2024.3372185.

Lee, B. M. (2018). Improved Energy Efficiency of Massive MIMO-OFDM in Battery-Limited IoT Networks. *IEEE Access*, 6, 38147-38160. doi: 10.1109/ACCESS.2018.2851591.

Lee, B. M. (2021). Energy-Efficient Operation of Massive MIMO in Industrial Internet-of-Things Networks. *IEEE Internet of Things Journal*, 8(9), 7252-7269. doi: 10.1109/JIOT.2020.3039236.

Li, L., Chai, K., Li, J. & Li, X. (2020). Resource Allocation for Multicarrier Rate-Splitting Multiple Access System. *IEEE Access*, 8, 174222-174232. doi: 10.1109/ACCESS.2020.3025635.

Li, L., Xu, Y., Zhang, Z., Yin, J., Chen, W. & Han, Z. (2019). A Prediction-Based Charging Policy and Interference Mitigation Approach in the Wireless Powered Internet of Things. *IEEE Journal on Selected Areas in Communications*, 37(2), 439-451. doi: 10.1109/JSAC.2018.2872429.

Li, T., Long, Q., Chai, H., Zhang, S., Jiang, F., Liu, H., Huang, W., Jin, D. & Li, Y. (2025). Generative AI Empowered Network Digital Twins: Architecture, Technologies, and Applications. *ACM Comput. Surv.*, 57(6). doi: 10.1145/3711682.

Li, X., Zheng, Y., Zeng, M., Liu, Y. & Dobre, O. A. (2023). Enhancing Secrecy Performance for STAR-RIS NOMA Networks. *IEEE Transactions on Vehicular Technology*, 72(2), 2684-2688. doi: 10.1109/TVT.2022.3213334.

Liang, F., Shen, C., Yu, W. & Wu, F. (2020). Towards Optimal Power Control via Ensembling Deep Neural Networks. *IEEE Transactions on Communications*, 68(3), 1760-1776. doi: 10.1109/TCOMM.2019.2957482.

Liang, L., Kim, J., Jha, S. C., Sivanesan, K. & Li, G. Y. (2017). Spectrum and Power Allocation for Vehicular Communications With Delayed CSI Feedback. *IEEE Wireless Communications Letters*, 6(4), 458-461. doi: 10.1109/LWC.2017.2702747.

Liu, Z., Garg, N. & Ratnarajah, T. (2024a). Dynamic Energy Efficient Resource Allocation for Massive MIMO Networks Using Randomized Ensembled Double Q-learning Algorithm. *IEEE Transactions on Cognitive Communications and Networking*, 1-1. doi: 10.1109/TCCN.2024.3524640.

Liu, Z., Zhang, J., Shi, E., Liu, Z., Niyato, D., Ai, B. & Shen, X. (2024b). Graph Neural Network Meets Multi-Agent Reinforcement Learning: Fundamentals, Applications, and Future Directions. *IEEE Wireless Communications*, 31(6), 39-47. doi: 10.1109/MWC.015.2300595.

Luo, Z.-Q. & Zhang, S. (2008). Dynamic Spectrum Management: Complexity and Duality. *IEEE Journal of Selected Topics in Signal Processing*, 2(1), 57-73. doi: 10.1109/JSTSP.2007.914876.

Luong, N. C., Hoang, D. T., Gong, S., Niyato, D., Wang, P., Liang, Y.-C. & Kim, D. I. (2019). Applications of Deep Reinforcement Learning in Communications and Networking: A Survey. *IEEE Communications Surveys and Tutorials*, 21(4), 3133-3174. doi: 10.1109/COMST.2019.2916583.

Mahmood, A., Beltramelli, L., Fakhrul Abedin, S., Zeb, S., Mowla, N. I., Hassan, S. A., Sisinni, E. & Gidlund, M. (2022). Industrial IoT in 5G-and-Beyond Networks: Vision, Architecture, and Design Trends. *IEEE Transactions on Industrial Informatics*, 18(6), 4122-4137. doi: 10.1109/TII.2021.3115697.

Makki, B., Chitti, K., Behravan, A. & Alouini, M.-S. (2020). A Survey of NOMA: Current Status and Open Research Challenges. *IEEE Open Journal of the Communications Society*, 1, 179-189. doi: 10.1109/OJCOMS.2020.2969899.

Maldonado, R., Karstensen, A., Pocovi, G., Esswie, A. A., Rosa, C., Alanen, O., Kasslin, M. & Kolding, T. (2021). Comparing Wi-Fi 6 and 5G Downlink Performance for Industrial IoT. *IEEE Access*, 9, 86928-86937. doi: 10.1109/ACCESS.2021.3085896.

Mao, H., Alizadeh, M., Menache, I. & Kandula, S. (2016). Resource Management with Deep Reinforcement Learning. *Proceedings of the 15th ACM Workshop on Hot Topics in Networks*, (HotNets '16), 50–56. doi: 10.1145/3005745.3005750.

Mao, Y., Clerckx, B. & Li, V. O. (2019). Rate-Splitting for Multi-User Multi-Antenna Wireless Information and Power Transfer. *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1-5. doi: 10.1109/SPAWC.2019.8815494.

Mashhadi, S., Ghiasi, N., Farahmand, S. & Razavizadeh, S. M. (2021). Deep Reinforcement Learning Based Adaptive Modulation With Outdated CSI. *IEEE Communications Letters*, 25(10), 3291-3295. doi: 10.1109/LCOMM.2021.3098419.

Masood, Z., Park, H., Jang, H. S., Yoo, S., Jung, S. P. & Choi, Y. (2021). Optimal Power Allocation for Maximizing Energy Efficiency in DAS-Based IoT Network. *IEEE Systems Journal*, 15(2), 2342-2348. doi: 10.1109/JSYST.2020.3013693.

Matthaiou, M., Papadogiannis, A., Bjornson, E. & Debbah, M. (2013). Two-Way Relaying Under the Presence of Relay Transceiver Hardware Impairments. *IEEE Communications Letters*, 17(6), 1136-1139. doi: 10.1109/LCOMM.2013.042313.130191.

Mekki, K., Bajic, E., Chaxel, F. & Meyer, F. (2018). Overview of Cellular LPWAN Technologies for IoT Deployment: Sigfox, LoRaWAN, and NB-IoT. *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pp. 197-202. doi: 10.1109/PERCOMW.2018.8480255.

Meng, F., Chen, P., Wu, L. & Cheng, J. (2020). Power Allocation in Multi-User Cellular Networks: Deep Reinforcement Learning Approaches. *IEEE Transactions on Wireless Communications*, 19(10), 6255-6267. doi: 10.1109/TWC.2020.3001736.

Mishra, A., Mao, Y., Dizdar, O. & Clerckx, B. (2022). Rate-Splitting Multiple Access for 6G—Part I: Principles, Applications and Future Works. *IEEE Communications Letters*, 26(10), 2232-2236. doi: 10.1109/LCOMM.2022.3192012.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S. & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature 518*, 529–533. doi: https://doi.org/10.1038/nature14236.

Moges, T. H., Lakew, D. S., Nguyen, N. P., Dao, N.-N. & Cho, S. (2023). Cellular Internet of Things: Use cases, technologies, and future work. *Internet of Things*, 24, 100910. doi: https://doi.org/10.1016/j.iot.2023.100910.

Mohamed, N. B., Hassan, M. Z. & Kaddoum, G. (2023). Spectral Efficiency Improvement in Downlink Fog Radio Access Network With Deep-Reinforcement-Learning-Enabled Power Control. *IEEE Internet of Things Journal*, 10(17), 15044-15059. doi: 10.1109/JIOT.2023.3263756.

Mohamed, N. B., Hassan, M. Z. & Kaddoum, G. (2024). RSMA-Enabled Interference Management for Industrial Internet of Things Networks With Finite Blocklength Coding and Hardware Impairments. *IEEE Transactions on Machine Learning in Communications and Networking*, 2, 1319-1340. doi: 10.1109/TMLCN.2024.3455268.

Mohammadi, H. & Marojevic, V. (2021). Artificial Neuronal Networks for Empowering Radio Transceivers: Opportunities and Challenges. *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*, pp. 1-5. doi: 10.1109/VTC2021-Fall52928.2021.9625494.

Mohammadi, H., AlQwider, W., Rahman, T. F. & Marojevic, V. (2022). AI-Driven Demodulators for Nonlinear Receivers in Shared Spectrum with High-Power Blockers. *2022 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 644-649. doi: 10.1109/WCNC51071.2022.9771613.

Mota, M. P., Araujo, D. C., Costa Neto, F. H., de Almeida, A. L. F. & Cavalcanti, F. R. (2019). Adaptive Modulation and Coding Based on Reinforcement Learning for 5G Networks. *2019 IEEE Globecom Workshops (GC Wkshps)*, pp. 1-6. doi: 10.1109/GCWkshps45667.2019.9024384.

Mu, X., Zhao, X. & Liang, H. (2020). Power Allocation Based on Reinforcement Learning for MIMO System With Energy Harvesting. *IEEE Transactions on Vehicular Technology*, 69(7), 7622-7633. doi: 10.1109/TVT.2020.2993275.

Muhammad Fahad Noman, H., Dimyati, K., Ariffin Noordin, K., Hanafi, E. & Abdrabou, A. (2024). FeDRL-D2D: Federated Deep Reinforcement Learning- Empowered Resource Allocation Scheme for Energy Efficiency Maximization in D2D-Assisted 6G Networks. *IEEE Access*, 12, 109775-109792. doi: 10.1109/ACCESS.2024.3434619.

Munaye, Y. Y., Juang, R.-T., Lin, H.-P., Tarekegn, G. B. & Lin, D.-B. (2021). Deep Reinforcement Learning Based Resource Management in UAV-Assisted IoT Networks. *Applied Sciences*, 11(5). doi: 10.3390/app11052163.

Naderializadeh, N., Sydir, J. J., Simsek, M. & Nikopour, H. (2021). Resource Management in Wireless Networks via Multi-Agent Deep Reinforcement Learning. *IEEE Transactions on Wireless Communications*, 20(6), 3507-3523. doi: 10.1109/TWC.2021.3051163.

Naparstek, O. & Cohen, K. (2019). Deep Multi-User Reinforcement Learning for Distributed Dynamic Spectrum Access. *IEEE Transactions on Wireless Communications*, 18(1), 310-323. doi: 10.1109/TWC.2018.2879433.

Nasir, A. A., Tuan, H. D., Nguyen, H. H., Debbah, M. & Poor, H. V. (2021). Resource Allocation and Beamforming Design in the Short Blocklength Regime for URLLC. *IEEE Transactions on Wireless Communications*, 20(2), 1321-1335. doi: 10.1109/TWC.2020.3032729.

Nasir, Y. S. & Guo, D. (2019). Multi-Agent Deep Reinforcement Learning for Dynamic Power Allocation in Wireless Networks. *IEEE Journal on Selected Areas in Communications*, 37(10), 2239-2250. doi: 10.1109/JSAC.2019.2933973.

News, U. (2015, July). UN projects world population to reach 8.5 Billion by 2030, driven by growth in developing countries. Retrieved from: https://news.un.org/en/story/2015/07/505352-un-projects-world-population-reach-85-billion-2030-driven-growth-developing.

Nguyen, T. T., Nguyen, N. D. & Nahavandi, S. (2020). Deep Reinforcement Learning for Multiagent Systems: A Review of Challenges, Solutions, and Applications. *IEEE Transactions on Cybernetics*, 50(9), 3826-3839. doi: 10.1109/TCYB.2020.2977374.

Nguyen, T.-V., Hua, D.-T., Huong, T. H., Hoang, V. T., Dao, N.-N. & Cho, S. (2024). Intelligent QoE Management for IoMT Streaming Services in Multiuser Downlink RSMA Networks. *IEEE Internet of Things Journal*, 11(7), 12602-12618. doi: 10.1109/JIOT.2023.3334473.

Nguyen, T. (2017, May). Small Cell Networks and the Evolution of 5G (Part 1). Retrieved from: https://www.qorvo.com/design-hub/blog/small-cell-networks-and-the-evolution-of-5g.

Omidkar, A., Khalili, A., Nguyen, H. H. & Shafiei, H. (2022). Reinforcement-Learning-Based Resource Allocation for Energy-Harvesting-Aided D2D Communications in IoT Networks. *IEEE Internet of Things Journal*, 9(17), 16521-16531. doi: 10.1109/JIOT.2022.3151001.

Pandey, G. K., Gurjar, D. S., Yadav, S. & Solanki, S. (2023). UAV-Empowered IoT Network With Hardware Impairments and Shadowing. *IEEE Sensors Letters*, 7(7), 1-4. doi: 10.1109/LSENS.2023.3284089.

Park, J., Choi, J., Lee, N., Shin, W. & Poor, H. V. (2023). Rate-Splitting Multiple Access for Downlink MIMO: A Generalized Power Iteration Approach. *IEEE Transactions on Wireless Communications*, 22(3), 1588-1603. doi: 10.1109/TWC.2022.3205480.

Park, T., Abuzainab, N. & Saad, W. (2016). Learning How to Communicate in the Internet of Things: Finite Resources and Heterogeneity. *IEEE Access*, 4, 7063-7073. doi: 10.1109/ACCESS.2016.2615643.

Parsa, A., Moghim, N. & Salavati, P. (2022). Joint power allocation and MCS selection for energy-efficient link adaptation: A deep reinforcement learning approach. *Computer Networks*, 218, 109386. doi: https://doi.org/10.1016/j.comnet.2022.109386.

Pedro, J. & Maas, S. (2005). A comparative overview of microwave and wireless power-amplifier behavioral modeling approaches. *IEEE Transactions on Microwave Theory and Techniques*, 53(4), 1150-1163. doi: 10.1109/TMTT.2005.845723.

Peng, M., Yan, S., Zhang, K. & Wang, C. (2016). Fog-computing-based radio access networks: issues and challenges. *IEEE Network*, 30(4), 46-53. doi: 10.1109/MNET.2016.7513863.

Polyanskiy, Y., Poor, H. V. & Verdu, S. (2010). Channel Coding Rate in the Finite Blocklength Regime. *IEEE Transactions on Information Theory*, 56(5), 2307-2359. doi: 10.1109/TIT.2010.2043769.

Qadir, Z., Le, K. N., Saeed, N. & Munawar, H. S. (2023). Towards 6G Internet of Things: Recent advances, use cases, and open challenges. *ICT Express*, 9(3), 296-312. doi: https://doi.org/10.1016/j.icte.2022.06.006.

Ranjha, A. & Kaddoum, G. (2021). Quasi-Optimization of Uplink Power for Enabling Green URLLC in Mobile UAV-Assisted IoT Networks: A Perturbation-Based Approach. *IEEE Internet of Things Journal*, 8(3), 1674-1686. doi: 10.1109/JIOT.2020.3014039.

Ranjha, A., Kaddoum, G. & Dev, K. (2022). Facilitating URLLC in UAV-Assisted Relay Systems With Multiple-Mobile Robots for 6G Networks: A Prospective of Agriculture 4.0. *IEEE Transactions on Industrial Informatics*, 18(7), 4954-4965. doi: 10.1109/TII.2021.3131608.

Ratasuk, R., Mangalvedhe, N., Lee, G. & Bhatoolaul, D. (2021). Reduced Capability Devices for 5G IoT. *2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1339-1344. doi: 10.1109/PIMRC50174.2021.9569595.

Rawat, M., Pagin, M., Giordani, M., Dufrene, L.-A., Lampin, Q. & Zorzi, M. (2023). Minimizing Energy Consumption for 5G NR Beam Management for RedCap Devices. *GLOBECOM 2023 - 2023 IEEE Global Communications Conference*, pp. 7091-7096. doi: 10.1109/GLOBECOM54140.2023.10437644.

Ren, H., Pan, C., Deng, Y., Elkashlan, M. & Nallanathan, A. (2019a). Resource Allocation for URLLC in 5G Mission-Critical IoT Networks. *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pp. 1-6. doi: 10.1109/ICC.2019.8761334.

Ren, H., Pan, C., Deng, Y., Elkashlan, M. & Nallanathan, A. (2020). Joint Pilot and Payload Power Allocation for Massive-MIMO-Enabled URLLC IIoT Networks. *IEEE Journal on Selected Areas in Communications*, 38(5), 816-830. doi: 10.1109/JSAC.2020.2980910.

Ren, J., Wang, Z., Xu, M., Fang, F. & Ding, Z. (2019b). An EM-Based User Clustering Method in Non-Orthogonal Multiple Access. *IEEE Transactions on Communications*, 67(12), 8422-8434. doi: 10.1109/TCOMM.2019.2945334.

Rezwan, S. & Choi, W. (2021). Priority-Based Joint Resource Allocation With Deep Q-Learning for Heterogeneous NOMA Systems. *IEEE Access*, 9, 41468-41481. doi: 10.1109/ACCESS.2021.3065314.

Romeijn, H. & Morales, D. R. (2000). A class of greedy algorithms for the generalized assignment problem. *Discrete Applied Mathematics*, 103(1), 209-235. doi: https://doi.org/10.1016/S0166-218X(99)00224-3.

Saavedra, E., del Campo, G. & Santamaria, A. (2020). Smart Metering for Challenging Scenarios: A Low-Cost, Self-Powered and Non-Intrusive IoT Device. *Sensors*, 20(24). doi: 10.3390/s20247133.

Salameh, H. A. B., Al-Masri, S., Benkhelifa, E. & Lloret, J. (2019). Spectrum Assignment in Hardware-Constrained Cognitive Radio IoT Networks Under Varying Channel-Quality Conditions. *IEEE Access*, 7, 42816-42825. doi: 10.1109/ACCESS.2019.2901902.

Sandholm, T., Larson, K., Andersson, M., Shehory, O. & Tohmé, F. (1999). Coalition structure generation with worst case guarantees. *Artificial Intelligence*, 111(1), 209-238. doi: https://doi.org/10.1016/S0004-3702(99)00036-3.

Sarkar, M. et al. (2025, June). Joint user association and bandwidth assignment for digital twin-assisted multi-RAT networks. *Proceedings of the IEEE International Conference on Communications (ICC)*.

Sediq, A. B., Gohary, R. H., Schoenen, R. & Yanikomeroglu, H. (2013). Optimal Tradeoff Between Sum-Rate Efficiency and Jain's Fairness Index in Resource Allocation. *IEEE Transactions on Wireless Communications*, 12(7), 3496-3509. doi: 10.1109/TWC.2013.061413.121703.

Seid, A. M., Boateng, G. O., Mareri, B., Sun, G. & Jiang, W. (2021). Multi-Agent DRL for Task Offloading and Resource Allocation in Multi-UAV Enabled IoT Edge Network. *IEEE Transactions on Network and Service Management*, 18(4), 4531-4547. doi: 10.1109/TNSM.2021.3096673.

Shafique, K., Khawaja, B. A., Sabir, F., Qazi, S. & Mustaqim, M. (2020). Internet of Things (IoT) for Next-Generation Smart Systems: A Review of Current Challenges, Future Trends and Prospects for Emerging 5G-IoT Scenarios. *IEEE Access*, 8, 23022-23040. doi: 10.1109/ACCESS.2020.2970118.

Shahiri, V., Kuhestani, A. & Hanzo, L. (2022). Short-Packet Amplify-and-Forward Relaying for the Internet-of-Things in the Face of Imperfect Channel Estimation and Hardware Impairments. *IEEE Transactions on Green Communications and Networking*, 6(1), 20-36. doi: 10.1109/TGCN.2021.3092067.

Shammasi, M. M. & Safavi, S. M. (2012). Performance of a predistorter based on Saleh model for OFDM systems in HPA nonlinearity. *2012 14th International Conference on Advanced Communication Technology (ICACT)*, pp. 148-152.

Shen, K. & Yu, W. (2018). Fractional Programming for Communication Systems—Part I: Power Control and Beamforming. *IEEE Transactions on Signal Processing*, 66(10), 2616-2630. doi: 10.1109/TSP.2018.2812733.

Shenfeld, I., Pari, J. & Agrawal, P. (2025). RL's Razor: Why Online Reinforcement Learning Forgets Less. Retrieved from: https://arxiv.org/abs/2509.04259.

Shi, Q., Razaviyayn, M., Luo, Z.-Q. & He, C. (2011). An Iteratively Weighted MMSE Approach to Distributed Sum-Utility Maximization for a MIMO Interfering Broadcast Channel. *IEEE Transactions on Signal Processing*, 59(9), 4331-4340. doi: 10.1109/TSP.2011.2147784.

Soleymani, M., Lameiro, C., Santamaria, I. & Schreier, P. J. (2019). Improper Signaling for SISO Two-User Interference Channels With Additive Asymmetric Hardware Distortion. *IEEE Transactions on Communications*, 67(12), 8624-8638.

Strehl, A. L., Li, L., Wiewiora, E., Langford, J. & Littman, M. L. (2006). PAC model-free reinforcement learning. *Proceedings of the 23rd International Conference on Machine Learning*, (ICML '06), 881–888. doi: 10.1145/1143844.1143955.

Sun, C., She, C., Yang, C., Quek, T. Q. S., Li, Y. & Vucetic, B. (2019). Optimizing Resource Allocation in the Short Blocklength Regime for Ultra-Reliable and Low-Latency Communications. *IEEE Transactions on Wireless Communications*, 18(1), 402-415. doi: 10.1109/TWC.2018.2880907.

Sun, H., Liu, Y., Al-Tahmeesschi, A., Chetty, S., Zaidi, S. A. R., Nag, A. & Ahmadi, H. (2025). An Explainable AI Framework for Dynamic Resource Management in Vehicular Network Slicing. Retrieved from: https://arxiv.org/abs/2506.11882.

Sun, H., Chen, X., Shi, Q., Hong, M., Fu, X. & Sidiropoulos, N. D. (2018). Learning to Optimize: Training Deep Neural Networks for Interference Management. *IEEE Transactions on Signal Processing*, 66(20), 5438-5453. doi: 10.1109/TSP.2018.2866382.

Sun, Y., Xu, J. & Cui, S. (2022). User Association and Resource Allocation for MEC-Enabled IoT Networks. *IEEE Transactions on Wireless Communications*, 21(10), 8051-8062. doi: 10.1109/TWC.2022.3163809.

Sutton, R. S. & Barto, A. G. (2018). *Reinforcement learning: An introduction* (ed. 3). Cambridge: MIT Press.

Tang, F., Mao, B., Kato, N. & Gui, G. (2021). Comprehensive Survey on Machine Learning in Vehicular Network: Technology, Applications and Challenges. *IEEE Communications Surveys and Tutorials*, 23(3), 2027-2057. doi: 10.1109/COMST.2021.3089688.

Tong, H., Chen, M., Zhao, J., Hu, Y., Yang, Z., Liu, Y. & Yin, C. (2025). Continual Reinforcement Learning for Digital Twin Synchronization Optimization. *IEEE Transactions on Mobile Computing*, 24(8), 6843-6857. doi: 10.1109/TMC.2025.3546507.

Varsier, N., Dufrène, L.-A., Dumay, M., Lampin, Q. & Schwoerer, J. (2021). A 5G New Radio for Balanced and Mixed IoT Use Cases: Challenges and Key Enablers in FR1 Band. *IEEE Communications Magazine*, 59(4), 82-87. doi: 10.1109/MCOM.001.2000660.

Vu, H. V., Farzanullah, M., Liu, Z., Nguyen, D. H. N., Morawski, R. & Le-Ngoc, T. (2022). Multi-Agent Reinforcement Learning for Channel Assignment and Power Allocation in Platoon-Based C-V2X Systems. *2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)*, pp. 1-5. doi: 10.1109/VTC2022-Spring54318.2022.9860518.

Wang, F., Song, L., Han, Z., Zhao, Q. & Wang, X. (2013). Joint scheduling and resource allocation for device-to-device underlay communication. *2013 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 134-139. doi: 10.1109/WCNC.2013.6554552.

Wang, J., Jiang, C., Zhang, K., Hou, X., Ren, Y. & Qian, Y. (2020a). Distributed Q-Learning Aided Heterogeneous Network Association for Energy-Efficient IIoT. *IEEE Transactions on Industrial Informatics*, 16(4), 2756-2764. doi: 10.1109/TII.2019.2954334.

Wang, X., Sheng, Z., Yang, S. & Leung, V. C. M. (2016). Tag-assisted social-aware opportunistic device-to-device sharing for traffic offloading in mobile social networks. *IEEE Wireless Communications*, 23(4), 60-67. doi: 10.1109/MWC.2016.7553027.

Wang, X., Zhang, Y., Shen, R., Xu, Y. & Zheng, F.-C. (2020b). DRL-Based Energy-Efficient Resource Allocation Frameworks for Uplink NOMA Systems. *IEEE Internet of Things Journal*, 7(8), 7279-7294. doi: 10.1109/JIOT.2020.2982699.

Xiao, L., Hong, S., Xu, S., Yang, H. & Ji, X. (2022). IRS-Aided Energy-Efficient Secure WBAN Transmission Based on Deep Reinforcement Learning. *IEEE Transactions on Communications*, 70(6), 4162-4174. doi: 10.1109/TCOMM.2022.3169813.

Xu, J., Dizdar, O. & Clerckx, B. (2023). Rate-Splitting Multiple Access for Short-Packet Uplink Communications: A Finite Blocklength Analysis. *IEEE Communications Letters*, 27(2), 517-521. doi: 10.1109/LCOMM.2022.3226817.

Xu, L., Zhou, X., Li, Y., Cai, F., Yu, X. & Kumar, N. (2022a). Intelligent Power Allocation Algorithm for Energy-Efficient Mobile Internet of Things (IoT) Networks. *IEEE Transactions on Green Communications and Networking*, 6(2), 766-775. doi: 10.1109/TGCN.2022.3144532.

Xu, Y., Mao, Y., Dizdar, O. & Clerckx, B. (2022b). Rate-Splitting Multiple Access With Finite Blocklength for Short-Packet and Low-Latency Downlink Communications. *IEEE Transactions on Vehicular Technology*, 71(11), 12333-12337. doi: 10.1109/TVT.2022.3191085.

Xu, Z., Wang, Y., Tang, J., Wang, J. & Gursoy, M. C. (2017). A deep reinforcement learning based framework for power-efficient resource allocation in cloud RANs. *2017 IEEE International Conference on Communications (ICC)*, pp. 1-6. doi: 10.1109/ICC.2017.7997286.

Yan, Z., Cheng, P., Chen, Z., Vucetic, B. & Li, Y. (2021). Two-Dimensional Task Offloading for Mobile Networks: An Imitation Learning Framework. *IEEE/ACM Transactions on Networking*, 29(6), 2494-2507. doi: 10.1109/TNET.2021.3093452.

Yang, Z., Chen, M., Saad, W. & Shikh-Bahaei, M. (2021). Optimization of Rate Allocation and Power Control for Rate Splitting Multiple Access (RSMA). *IEEE Transactions on Communications*, 69(9), 5988-6002. doi: 10.1109/TCOMM.2021.3091133.

Ye, H., Li, G. Y. & Juang, B.-H. F. (2019). Deep Reinforcement Learning Based Resource Allocation for V2V Communications. *IEEE Transactions on Vehicular Technology*, 68(4), 3163-3173. doi: 10.1109/TVT.2019.2897134.

Zhang, H., Zhang, H., Liu, W., Long, K., Dong, J. & Leung, V. C. M. (2020). Energy Efficient User Clustering, Hybrid Precoding and Power Optimization in Terahertz MIMO-NOMA Systems. *IEEE Journal on Selected Areas in Communications*, 38(9), 2074-2085. doi: 10.1109/JSAC.2020.3000888.

Zhang, J., Liu, Z., Zhu, Y., Shi, E., Xu, B., Yuen, C., Niyato, D., Debbah, M., Jin, S., Ai, B., Xuemin & Shen. (2025). Multi-Agent Reinforcement Learning in Wireless Distributed Networks for 6G. Retrieved from: https://arxiv.org/abs/2502.05812.

Zhang, L., Ijaz, A., Xiao, P., Molu, M. M. & Tafazolli, R. (2018a). Filtered OFDM Systems, Algorithms, and Performance Analysis for 5G and Beyond. *IEEE Transactions on Communications*, 66(3), 1205-1218. doi: 10.1109/TCOMM.2017.2771242.

Zhang, L., Tan, J., Liang, Y.-C., Feng, G. & Niyato, D. (2019a). Deep Reinforcement Learning-Based Modulation and Coding Scheme Selection in Cognitive Heterogeneous Networks. *IEEE Transactions on Wireless Communications*, 18(6), 3281-3294. doi: 10.1109/TWC.2019.2912754.

Zhang, X., Matthaiou, M., Coldrey, M. & Björnson, E. (2015). Impact of Residual Transmit RF Impairments on Training-Based MIMO Systems. *IEEE Transactions on Communications*, 63(8), 2899-2911. doi: 10.1109/TCOMM.2015.2432761.

Zhang, Y., Kang, C., Ma, T., Teng, Y. & Guo, D. (2018b). Power Allocation in Multi-Cell Networks Using Deep Reinforcement Learning. *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pp. 1-6. doi: 10.1109/VTCFall.2018.8690757.

Zhang, Y., Wang, X. & Xu, Y. (2019b). Energy-Efficient Resource Allocation in Uplink NOMA Systems with Deep Reinforcement Learning. *2019 11th International Conference on Wireless Communications and Signal Processing (WCSP)*, pp. 1-6. doi: 10.1109/WCSP.2019.8927898.

Zhang, Z., Huang, Y., Zhang, C., Zheng, Q., Yang, L. & You, X. (2024). Digital Twin-Enhanced Deep Reinforcement Learning for Resource Management in Networks Slicing. *IEEE Transactions on Communications*, 72(10), 6209-6224. doi: 10.1109/TCOMM.2024.3395698.

Zhao, N., Liang, Y.-C., Niyato, D., Pei, Y., Wu, M. & Jiang, Y. (2019). Deep Reinforcement Learning for User Association and Resource Allocation in Heterogeneous Cellular Networks. *IEEE Transactions on Wireless Communications*, 18(11), 5141-5152. doi: 10.1109/TWC.2019.2933417.

Zhou, Z., Dong, M., Ota, K., Wang, G. & Yang, L. T. (2016). Energy-Efficient Resource Allocation for D2D Communications Underlaying Cloud-RAN-Based LTE-A Networks. *IEEE Internet of Things Journal*, 3(3), 428-438. doi: 10.1109/JIOT.2015.2497712.

Zhou, Z., Liu, G. & Tang, Y. (2024). Multiagent Reinforcement Learning: Methods, Trustworthiness, Applications in Intelligent Vehicles, and Challenges. *IEEE Transactions on Intelligent Vehicles*, 9(12), 8190-8211. doi: 10.1109/TIV.2024.3408257.

Zhu, J., Ng, D. W. K., Wang, N., Schober, R. & Bhargava, V. K. (2017). Analysis and Design of Secure Massive MIMO Systems in the Presence of Hardware Impairments. *IEEE Transactions on Wireless Communications*, 16(3), 2001-2016. doi: 10.1109/TWC.2017.2659724.

Şahin, M. M., Dizdar, O., Clerckx, B. & Arslan, H. (2024). OFDM-RSMA: Robust Transmission under Inter-Carrier Interference. *IEEE Transactions on Communications*, 1-1. doi: 10.1109/TCOMM.2024.3395713.