Distributed Learning for 5G and Beyond Network Management and Orchestration

Mohamed Sana

Université Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE L' UNIVERSITÉ GRENOBLE ALPES

Spécialité : **Mathématiques et Informatique**

Arrêté ministériel : 25 mai 2016

Présentée par

Mohamed SANA

Thèse dirigée par **Emilio Calvanese Strinati**

Préparée au sein du **Laboratoire CEA Grenoble - LETI**
dans l'École Doctorale **Mathématiques, Sciences et Technologies de l'Information, Informatique (MSTII)**

Apprentissage Distribué pour la Gestion et l'Orchestration des Réseaux 5G et au-delà

Distributed Learning for 5G and beyond Networks Management and Orchestration

Thèse soutenue publiquement le 26 Octobre 2021,
devant le jury composé de :

Pr. Mérouane DEBBAH
Professeur à CentralSupelec, Paris, Examinateur
Pr. Sergio BARBAROSSA
Professeur à Sapienza Università di Roma, Italie, Rapporteur
Pr. Petar POPOVSKI
Professeur à Aalborg University, Danemark, Examinateur
Pr. Deniz GÜNDÜZ
Professeur à Imperial College of London, Royaume Uni, Rapporteur **Pr. Denis TRYSTRAM**
Professeur à Grenoble-INP, France, Président du Jury
Dr. Emilio CALVANESE STRINATI
Directeur de recherche, CEA-LETI Grenoble, France, Directeur de thèse

À mes parents
Souleymane et Rasmata,
– v17.24

À la mémoire de mon oncle
Ousmane Sana,
Je vous espère à présent reposé,
fier de voir germer les graines que vous avez semé.

Abstract

Wireless communications are experiencing an unprecedented demand for communication bandwidth. It is not only the volume of data traffic exploding, but also the characteristics and nature of communicating objects are diversifying. In addition, new applications and use cases are emerging, each one with stringent requirements, making the management of radio, computing, and storage resources complex, requiring advanced, flexible, scalable, and low complexity solutions.

This thesis focuses on *distributed learning* approaches for effective and efficient radio resource management in the context of 5G networks and beyond. Distributed solutions have the advantage of being flexible, scalable, and robust to environmental artifacts. Furthermore, they reduce signaling overhead and strongly limit cumbersome centralized computations. However, distributed learning faces several challenges, especially in dense 5G networks deployments, due to an uncertain wireless environment and limited radio and computing resources. Motivated by these challenges, we propose new distributed learning frameworks based on multi-agent reinforcement learning, which consider environment dynamics, including radio channel variations, intra- and inter-cell interference, users' traffic, and mobility for dynamic radio resource management. Specifically, our approach models user devices as independent agents, collaborating with (or competing against) each other for radio and/or computing resources to optimize network utility functions. To do so, the agents rely on their local observations (and global observations if available) to make autonomous decisions, thereby significantly reducing signaling and computational overhead.

Following this approach, we propose a fully distributed and decentralized user association framework for the optimal assignment of user equipments to base stations. Then, we extend this study to propose a new architecture, which conveniently combines neural attention mechanisms and multi-agent reinforcement learning to build fully transferable user association policies with *zero generalization capability*. In other words, with the proposed new framework, the knowledge acquired in one specific scenario is transferable to another without requiring any additional training procedure. We show that the proposed mechanism adapts well and by design to variations in the number and positions of users. These conclusive results allow us to address the problem of energy-efficient dynamic computation offloading, where multiple users compete for radio and computing resources to offload data generated dynamically at the user's devices to an edge server. We formulate this problem as a long-term energy minimization problem with end-to-end delay constraints to meet user quality of service. Using Lyapunov stochastic optimization tools, we decouple this problem into a *per slot* frequency allocation problem and a radio resource allocation problem, which we jointly solve with a proposed fast iterative algorithm and the proposed transferable user association solution. The resulting framework exhibits near-optimal performance, improving the network's energy efficiency while significantly reducing its complexity. Finally, to further enhance the system's performance, in the last part of this thesis, we explore the opportunity offered by semantic communications. In this paradigm, whenever communication occurs to convey meaning between two agents, what matters is the receiver's understanding of the transmitted message and not necessarily their correct reconstruction. Transmitting only relevant information sufficient for agents to capture the meaning intended can save significant communication bandwidth. Therefore, we propose a new architecture that enables *representation learning* of semantic symbols. Our preliminary numerical results are promising, making semantic communications a good candidate to improve the efficiency and sustainability of future 6G networks.

Keywords – *Distributed Learning, 5G and beyond Networks, Radio Resource Management, Reinforcement Learning, Wireless Networks, User Association, Handover Management, Mobile Edge Computing, Semantic Communications, Goal-Oriented Communications.*

Résumé

Les communications sans fil connaissent une demande sans précédent de débit et de bande passante. Non seulement le volume du trafic de données explose, mais les spécificités et la nature des objets communicants se diversifient. De plus, l'apparition de nouvelles applications et de nouveaux cas d'utilisation, chacun avec des exigences strictes, complexifie la gestion des ressources radio, de calcul, et de stockage, qui nécessite désormais des solutions avancées, flexibles, évolutives et peu complexes.

Cette thèse se focalise sur les approches d'*apprentissage distribué* pour une gestion efficace et efficiente des ressources radio des réseaux mobiles 5G et au-delà. Les solutions distribuées ont l'avantage d'être flexibles, évolutives et robustes face aux perturbations ambiantes. En outre, elles réduisent la surcharge de signalisation et limitent des calculs centralisés laborieux. Cependant, l'apprentissage distribué fait face à plusieurs défis, notamment dans les réseaux 5G denses, en raison d'un environnement sans fil incertain et des ressources radio et de calcul limitées. Motivés par ces défis, nous proposons de nouveaux cadres d'apprentissage distribué basés sur l'apprentissage par renforcement multi-agent, tenant compte de la dynamique de l'environnement (variations des canaux sans fil, interférences intra et intercellulaires, trafic et mobilité des utilisateurs) pour une gestion dynamique des ressources radio. Plus précisément, notre approche modélise les équipements utilisateur comme des agents indépendants, qui collaborent (ou rivalisent) pour accéder à des ressources radio et/ou computationnelles afin d'optimiser des fonctions d'utilité du réseau. Pour cela, les agents s'appuient sur leurs observations locales (et sur d'éventuelles observations globales) pour prendre des décisions autonomes, réduisant ainsi considérablement les coûts de signalisation et de calcul.

Ce faisant, un cadre d'association d'utilisateurs entièrement distribué et décentralisé est d'abord proposé pour l'affectation optimale des équipements utilisateurs aux stations de base, et pour gérer la mobilité. Nous étendons ensuite cette étude pour proposer une nouvelle architecture combinant judicieusement des mécanismes d'attention neuronale et d'apprentissage par renforcement multi-agent. Les solutions obtenues sont entièrement *transférables et généralisables* : les connaissances acquises dans un scénario spécifique sont applicables à d'autres sans nécessiter de procédure d'apprentissage supplémentaire. Nous montrons que cette solution s'adapte bien aux variations du nombre et des positions des utilisateurs. Cela nous permet ensuite d'aborder le problème du déchargement dynamique des calculs à faible coût énergétique, où plusieurs utilisateurs se disputent des ressources radio et computationnelles pour décharger des tâches sur un serveur périphérique. Il s'agit d'un problème de minimisation d'énergie à long terme sous des contraintes strictes de délai. Avec des outils d'optimisation stochastique de Lyapunov, nous traduisons ce problème en un problème d'allocation conjointe de fréquence et de ressources radio par slot, que nous résolvons de manière quasi-optimale avec un algorithme itératif rapide combiné à notre solution d'association d'utilisateurs transférable. Enfin, la dernière partie de cette thèse explore les communications sémantiques. Dans ce paradigme, lorsqu'une communication a lieu pour véhiculer un sens entre deux agents, ce qui importe est la compréhension par le récepteur du message transmis et non sa reconstruction correcte. Transmettre uniquement les informations pertinentes suffisantes pour que les agents saisissent le sens voulu permet d'énormes économies de bande passante. Nous proposons donc une méthode permettant l'*apprentissage de la représentation* des symboles sémantiques. Nos résultats numériques préliminaires sont prometteurs et montrent le potentiel des communications sémantiques pour des futurs réseaux 6G efficaces et durables.

Mots-clés – *Réseaux sans Fils, Réseaux 5G mobile, Apprentissage Distribué, Apprentissage par Renforcement Multi-Agents, Gestion de Ressources Radio, Association d'Utilisateurs, Gestion de la Mobilité, Informatique mobile de Périphérie, Communications Sémantiques, Communications axées sur les Objectifs.*

List of Author's Publications

The content of this manuscript is based on the following patents, conferences and journal publications.

Patents

[P1] **M. Sana**, A. De Domenico, *"Method for associating user equipment in a cellular network via multi-agent reinforcement learning,"* Issued in May 20, 2021, US17099922.

[P2] **M. Sana**, N. di Pietro, E. Calvanese Strinati, and B. Miscopein, *"Method for associating user equipment in a cellular network according to a transferable association policy,"* Filed in September 30, 2020, FR2009989.

International Journal Communications

[J1] **M. Sana**, A. De Domenico, W. Yu, Y. Lostanlen, and E. Calvanese Strinati, *"Multi-Agent Reinforcement Learning for Adaptive User Association in Dynamic mmWave Networks,"* IEEE Transactions on Wireless Communications, 19 (10):6520–6534, 2020.

International Conference Communications

[C1] **M. Sana**, A. De Domenico, and E. Calvanese Strinati, *"Multi-Agent Deep Reinforcement Learning based User Association for Dense mmWave Networks,"* In Proc. IEEE Global Communications Conference (GLOBECOM), HI, USA, pages 1–6., Dec 2019.

[C2] **M. Sana**, A. De Domenico, E. Calvanese Strinati, and A. Clemente, *"Multi-Agent Deep Reinforcement Learning for Distributed Handover Management In Dense MmWave Networks,"* In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Madrid, Spain, pages 8976–8980., May 2020.

[C3] **M. Sana**, N. di Pietro, and E. Calvanese Strinati, *"Transferable and Distributed User Association Policies for 5G and Beyond Networks,"* IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Virtual, Sept. 2021.

[C4] **M. Sana**, M. Merluzzi, N. di Pietro, and E. Calvanese Strinati, *"Energy Efficient Edge Computing: When Lyapunov Meets Distributed Reinforcement Learning,"* in Proc. IEEE International Conference on Communications (ICC) Workshops, Virtual, Montreal, Canada, June 2021.

[C5] **M. Sana** and E. Calvanese Strinati, *"Learning Semantics: An Opportunity for Effective 6G Communications,"* in Proc. IEEE Consumer Communications and Networking Conference (CCNC), Virtual, Las Vegas, January 2022.

Extra-thesis collaborations

These are the results of some collaborations done during the thesis but not directly related to it.

[C6] F. Wolf, **M. Sana**, S. de Rivaz, F. Dehmas, and J.-P. Cances, *"Comparison of Multi-Channel Ranging Algorithms for Narrow band LPWA Network Localization,"* In Proc. International Symposium on Ubiquitous Networking (UNet), Limoges, France, pages 3-17., Nov. 2019 (**Best Paper Award**).

[P3] F. Wolf, S. de Rivaz, F. Dehmas, and **M. Sana**, *"Method of estimating a distance in a network lpwa and method of estimating the associated position,"* Issued March 18, 2021, US17018267.

Contents

List of Figures

List of Tables

List of Acronyms

OSI	Open Systems Interconnection
3GPP	Third Generation Partnership Project
LTE	Long Term Evolution
RRC	Radio Resource Control
RRM	Radio Resource Management
5G	Fifth Generation
6G	Sixth Generation
BSs	Base Station
APs	Access Points
ES	Edge Server
ESs	Edge Servers
UEs	User Equipments
SBS	Small cell Base Station
MBS	Macro Base Station
HetNets	Heterogeneous Networks
ACK	Acknowledgment
AoA	Angle of Arrival
CSI	Channel-State Information
MIMO	Multiple-Input/Multiple-Output
CCDF	Complementary Cumulative Distribution Function
mmWave	millimeter-wave
PPP	Poisson Point Process
DNN	Deep Neural-Network
NN	Neural-Network
PNA	Policy Network Architecture
SGD	Stochastic Gradient Descend
SDMA	Spatial Division Multiple Access
SNR	Signal-to-Noise Ratio

SINR	Signal-to-Noise-plus-Interference Ratio
URLLC	Ultra-Reliable Low-Latency Communications
eMBB	enhanced Mobile Broadband
RAN	Radio Access Network
CN	Core Network
mMTC	massive Machine-Type Communication
MEC	Multi-Access Edge Computing
ML	Machine Learning
DL	Deep Learning
MARL	Multi Agent Reinforcement Learning
MTRL	Multi-Task Reinforcement Learning
MDP	Markov Decision Process
TD	Temporal Difference
LSTM	Long Short-Term Memory
KL	Kullback-Leibler
GRU	Gated Recurrent Unit
MLP	Multi Layer Perceptron
NLP	Natural Language Processing
BLEU	Bilingual Evaluation Understudy
RNN	Reccurent Neural Network
AI	Artificial Intelligence
AoI	Age of the Information
PPO	Proximal Policy Optimization
POMDP	Partially Observable Markov Decision Process
DRL	Deep Reinforcement Learning
RL	Reinforcement Learning
DRQN	Deep Recurrent Q-Network
HDRQN	Hysteretic Deep Recurrent Q-Network
DQN	Deep Q-Network
HO	Handover
RSS	Received Signal Strength

QoS	Quality of Service
QoE	Quality of Experience
RSRP	Reference Signal Received Power
RIS	Reflective Intelligent Surfaces
RSRQ	Reference Signal Received Quality
WEI	Word Error Indicator
TTT	Time-to-Trigger
UAVs	Unmanned Aerial Vehicles
IoT	Internet of Things
XR	eXtended Reality
VR	Virtual Reality

List of Symbols

System parameters

T_s	Symbol duration
B	System bandwidth
f_c	Carrier frequency
G^{Tx}, G^{Rx}	Transmitted/received Antenna gain
P^{Tx}, P^{Rx}	Transmitted/received power
h	Channel coefficient
N_0	Noise power spectral density
d	Propagation distance between transmitter and receiver
$R(t)$	Instantaneous network sum-rate

Mathematical Notations

$\mathcal{N}(\mu, \sigma^2)$	Normal distribution with mean μ and variance σ^2
$\mathbb{C}\mathcal{N}(\mu, \sigma^2)$	Complex normal distribution with mean μ and variance σ^2
$\mathbb{E}[\cdot]$	Expectation operator
$\lfloor \cdot \rfloor$	Floor operator
$\arg\max_x \{g(x)\}$	Argument of the maximum of $g(x)$
$\arg\min_x \{g(x)\}$	Argument of the minimum of $g(x)$
$\text{softmax}(\cdot)$	Softmax function or normalized exponential function
$\mathcal{L}(\cdot)$	Loss function
$\mathbf{W}, \boldsymbol{\theta}, \boldsymbol{\vartheta}$	Weight matrix (in a neural network)
$Q(a, s)$	Q-values in Q-learning: $Q(a, s)$ denotes the Q-values associated with state-action pair (a, s)
$A(a, s)$	Advantage of taking action a in a state s
$\|\cdot\|$	Euclidean norm
$P(\cdot), p(\cdot)$	Probability of an event and associated probability density function
$\mathcal{U} = \{1, \ldots, N\}$	Set of size N
$\|\mathcal{A}\|$	Cardinal of set $\mathcal{A}$, *i.e.* , the number of elements in $\mathcal{A}$
$\bigtimes_{j=1\ldots N} \mathcal{A}_j$	Cartesian product of ensemble $\mathcal{A}_j,\ j = 1 \ldots N = \mathcal{A}_1 \times \mathcal{A}_2 \times \cdots \times \mathcal{A}_N$
π	An agent policy (or strategy). $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ specifies a probability distribution over joint state space $\mathcal{S}$ and action space $\mathcal{A}$. For a given state s and action a, $\pi(s, a) = P(a\|s)$. We use the notation $\pi_{\boldsymbol{\theta}}$, when the policy is parameterized by $\boldsymbol{\theta}$
γ	Agent reward discounting factor
ϵ	Probability that an agent takes an action a given state s in an ϵ-greedy policy
$\mathbb{1}_{\text{Cond(x)}}$	Indicator function: equals 1 if x satisfies condition "Cond" and 0 otherwise
$\delta(x)$	Dirac function
$\text{KL}(p\|\|q)$	Kullback-Leibler divergence between the densities of probability p and q
$H(X)$	Entropy of a random variable X
$I(X; Y)$	Mutual information between the random variables X and Y

Chapter 1

Introduction

"La vie est une souffrance continuelle pour celui qui l'affronte avec son coeur et un passe temps agréable pour celui qui l'affronte avec son intelligence."

"Life is a continuous suffering for the one who faces it with his heart and a pleasant pastime for the one who faces it with his intelligence."

– Proverbe Africain

Contents

1.1 5G Networks: A Technological Breakthrough

With the proliferation of smart and connected devices, the cyber and physical spaces are fusing, turning humans, objects, and events into an exponentially growing source of digital information [3]. As a result, wireless networks are witnessing an unprecedented demand for communication bandwidth, and an explosion of connected devices. At the same time, new applications and services are emerging with stringent requirements in terms of reliability and/or latency. Examples of these range from eXtended Reality (XR), including augmented, virtual, and mixed reality to telemedicine, autonomous vehicles, flying vehicles, Internet of Things (IoT) high-precision manufacturing, smart cities. This pushes the wireless network toward constant evolution and revolution to address these challenges. The recent introduction of Fifth Generation (5G) networks is a perfect example [4]. 5G technology represents a breakthrough in the design of communication networks. It provides a communication infrastructure able to deliver, simultaneously, high reliability, low latency, and high data rates, thus supporting a variety of services or verticals. These services are usually split into three mains categories:

1. **enhanced Mobile Broadband (eMBB)**: driven by the need to provide higher data throughput, eMBB services aim at enhancing the network capacity to support stable connections with very high peak data rates (up to 20 Gbps in downlink [5]) as well as moderate data rates for cell-edge users (overall providing a user's experienced data rate of 100 Mbps anytime, anywhere).

2. **massive Machine-Type Communication (mMTC)**: this service aims to support a massive number of connected devices with sporadic communications (sending small data payloads) and low energy consumption such as IoT devices. Other use cases include smart grids, tactile internet as well as services involving machine-to-machine communications.

3. **Ultra-Reliable Low-Latency Communications (URLLC)**: this service aims to support applications with low-latency short-packets transmission and extremely high reliability ($10^{-5} - 10^{-9}$ packet error rate). Such applications range from telesurgery to autonomous vehicles and high precision manufacturing.

1.1.1 5G keys enablers: why these choices?

To accommodate all these stringent requirements, 5G adopts mainly millimeter-wave (mmWave) communications together with massive Multiple-Input/Multiple-Output (MIMO) and (ultra) dense deployment of network Access Points (APs) [6]. The reasons behind these choices are easy to understand. Historically, wireless networks evolution has been driven by the need for higher and higher data rates. Back in 1948's, Shannon-Hartley Theorem (named after Claude Shannon and Ralph Hartley) has established the link between the maximum achievable rate C on a given communication link experiencing an interference I and the received signal power P^{Rx} as well as the communication bandwidth B:

$$C = B \log_2 \left(1 + \frac{P^{\mathrm{Rx}}}{N_0 B + I}\right), \tag{1.1}$$

where N_0 denotes the noise power spectral density. Using Friis formula for free-space model, the received power can be expressed as follows:

$$P^{\mathrm{Rx}} = G^{\mathrm{Tx}} G^{\mathrm{Rx}} \left(\frac{c}{4\pi f_0 d}\right)^2 P^{\mathrm{Tx}}. \tag{1.2}$$

Here, P^{Tx} is the transmit power, f_0 is the signal carrier frequency, d is the distance between transmitter and receiver, c is the light speed, and G^{Tx}, G^{Rx} the transmit and received antenna gain respectively.

From Eqn. (1.1), an immediate solution to increase the maximum achievable rate C is to increase the communication bandwidth. This is the idea behind the adoption of mmWave bands, which offer large spectrum resources. Meanwhile, adopting mmWave bands means going to higher frequencies, which implies increasing the signal carrier frequency f_0. However, transmissions at higher frequencies suffer from severe attenuation due to rain, atmospheric, and molecular absorption, thus, limiting the range of communication [7]. One solution to compensate for signal loss due to attenuation is to increase the antenna directivity gains. This is the idea introduced by massive MIMO, which consists in increasing the number of antenna elements to provide high directivity antenna gain. Furthermore, the short-wavelength characteristics of mmWave allow for compact design of MIMO antenna array as the size of the antenna element is reduced. Therefore, massive MIMO can help improve coverage performance with directional beamforming techniques. Another solution to combat path loss and increase channel capacity is to reduce the distance between transmitter and receiver. This can be achieved by densifying network APs. Indeed, by increasing the number of APs in a given geographical area, the distance to the end-users eventually gets reduced. However, network capacity does not systematically increase with densification of APs due to *e.g.*, (co-)channel interference I and inefficient resource allocation. Hence, despite their enormous potential, these key enablers also pose new challenging problems, which need to be addressed for efficient and effective 5G and beyond communications.

Remark 1. *Although the transmission with more power can also increase the network capacity, it has the main drawback of increasing the energy consumption.*

Remark 2. *Another way to increase the network capacity consists to act directly on the channel pathloss. This alternative implies being able to shape the wireless propagation channel, which can be achieved with the recently introduced Reflective Intelligent Surfaces (RIS) and meta-surfaces technologies [8]. The fundamental idea behind these technologies is to turn the wireless environment into a smart reconfigurable and controllable space capable of actively transferring and processing information [9].*

1.1.2 New challenges for radio resource management

In wireless communications, Radio Resource Management (RRM) involves all strategies, procedures, and algorithms used to manage radio resources (beamforming, power allocation, modulation and channel coding scheme, etc.). An efficient RRM adjusts network parameters to system dynamics including base stations density and load, users traffic loads, users positions and mobility, as well as their Quality of Service (QoS) to optimize network spectral efficiency. However, with the adoption of the aforementioned advanced technologies *i.e.*, mmWave communications, massive MIMO and network densification in 5G, RRM is becoming more and more complex. This complexity is further accentuated by an exponentially growing number of users or smart devices in wireless networks, with heterogeneous service requirements and variable traffic loads, making RRM even more challenging. Some of these challenges are listed below:

Interference management. Intra- and inter-cell interference are detrimental to wireless networks. They are exacerbated in large-scale networks with the dense deployment of APs. Dynamic management of interference *w.r.t.* varying network topology, traffic, as well as channel dynamics is a very challenging task, yet crucial for efficient RRM.

User association or cell selection and handover management. User association is the process of associating User Equipments (UEs) with network APs. It is a fundamental task, which is also crucial in mobile communications as it directly affects the network spectral efficiency as well as the users' perceived QoS. Efficient user association can help mitigate interference. Conversely, a wrong user association can lead to significant interference, which can be detrimental to wireless system performance. User association performance may also vary *w.r.t.* wireless channel dynamics (fading, shadowing), base stations load, users mobility (handover), as well as their QoS requirements.

Heterogeneous QoS. One major innovation of 5G is its ability to support on the same communication infrastructure, different services, or verticals (*e.g.*, autonomous driving vehicles, smart industry, etc.). Consequently, users in wireless networks are becoming heterogeneous, each with its characteristics and communications requirements, thus, requiring specialized and customized radio resources.

Energy management. Energy efficiency in wireless communication is primordial to reduce network energy consumption. In the context of IoT with limited battery lifetime devices, this becomes a must. Yet, designing RRM algorithms taking into account both radio resource allocation and network energy consumption is challenging.

Multi Access Edge computing. Today, many mobile applications (*e.g.*, surveillance and video analytics in IoT) rely on cloud services (with a virtually infinite capacity) to process data generated on users' devices. As the amount of generated data is becoming more and more important, offloading it to the cloud through Radio Access Network (RAN) becomes intractable as it can lead to excessive network congestion and significant communication overhead. A solution to handle this is to process data close to

end-users at the network edge, which leads to a brand-new paradigm: mobile edge computing or Multi-Access Edge Computing (MEC) in its standardized version [10]. By bringing native cloud functionalities (storage and computing capabilities) to the network edge *e.g.*, within the RAN or the Core Network (CN), MEC promises low communication delay, reduced backbone congestion as well as distributed computing and storage. However, all these benefits do not come for free. RRM is becoming extremely complex as now, radio resources need to be jointly optimized with *limited* computing resources at the edge.

1.2 Distributed Learning for Radio Resource Management

Conventionally, solutions to resource allocation problems are obtained by solving complex optimizations based on, *e.g.*, (instantaneous) Channel-State Information (CSI) and traffic load, QoS requirements of the users, base stations load, and under specific constraints on, *e.g.*, users energy consumption, end-to-end (E2E) latency, etc. In general, these optimization problems are integer (or mixed-integer) programming problems, which are non-convex and NP-hard. Therefore, traditional solutions generally work in a centralized manner. Indeed, centralized approaches yield better results as information from multiple nodes are collected and processed in a unified way. However, they lead to significant signaling overhead and require excessive computation, impractical for 5G networks due to dense deployment of UEs and Base Station (BSs). In addition, as aforementioned, RRM involves many optimization variables not always well-defined mathematically (*e.g.*, due to the dynamic nature of the wireless environment, the mobility patterns of the users), making it difficult to formulate the optimization problem. This motivates the exploration of more advanced solutions for RRM.

Among different solutions under consideration, a pervasive introduction of Artificial Intelligence (AI) at the network edge (*edge intelligence*) is envisioned [11]. In this context, multiple distributed AI-powered devices *can learn and possibly share their knowledge* to optimize some network utility functions and achieve common goals [3, 12]. This approach is currently made possible by endowing mobile devices with AI algorithm computing capabilities [13, 14]. Hence, this thesis focuses in adopting *distributed artificial intelligence*, namely distributed Machine Learning (ML) techniques to solve RRM problems.

1.2.1 Machine learning for communications and networking

With its ability to infer knowledge from randomly distributed data or observations, ML, especially Deep Learning (DL) has gained popularity and widespread interest in wireless communications [15], in particular for RRM problems. This includes optimal power allocation, beamforming or beam selection, interference mitigation, joint source and channel coding, etc. One reason for this craze towards data-driven RRM solution is the growing complexity of wireless networks and the difficulty of deriving accurate and tractable mathematical models [16]. Moreover, when no expert database is available for training DL algorithms, Reinforcement Learning (RL) appears as a good option, since it enables learning through trial-and-error, *i.e.* by interaction with the wireless environment. One particular advantage of RL methods is that there is no need for *a priori* knowledge about environment dynamics, which can be stochastic and/or non-linear [17]. Combined with DL, Deep Reinforcement Learning (DRL) becomes a powerful tool, which is particularly suitable for solving complex problems in wireless communications, especially when no tractable theoretical model of the environment dynamics is available [18].

Our focus in this thesis is on distributed (possibly decentralized) learning approach. Adopting such an approach offers several potential benefits.

Speeding up computation. By distributing learning, computation can be speed up. Moreover, each agent decisions can be made locally, avoiding excessive communications between *e.g.* users and a central orchestrator.

Scalability. In general, distributed approaches are scalable, with linear complexity. This aspect is particularly important in dense networks.

Accuracy and robustness. Another practical feature of distributed learning is robustness against environment artifacts. Each agent has a local perspective (or database) of the environment, thus enriching the learning procedures.

Important note 1 (Communications for Machine Learning)**.** *Recently, many studies have started to explore how to perform efficiently distributed training over wireless networks [19, 20]. One prominent example is* Federated Learning*, which enables a group of agents to collaboratively execute a common learning task (e.g., image classification) by exchanging only their model parameters, rather than their raw data [21, 22, 23]. Note that this distributed learning setting is different from the one covered in this thesis. Our focus is on how to leverage distributed learning for solving problems directly related to RRM rather than how to optimize RRM to perform distributed learning.*

1.2.2 Related challenges and complexity

Unfortunately, there are still many challenging issues related to distributed learning, especially in the context of wireless communications. The first challenge is the loss of theoretical guarantees of convergence. Indeed, in the general setting of distributed learning, multiple agents cooperate with (or compete against) each other for radio resources to optimize predefined network utility functions. Such cooperation (or competition) can lead to the non-stationarity of the environment from a single agent's perspective. That is particularly true for multi-agent systems and is known to be a difficult task [24]. The second challenge is that RRM problems are generally NP-hard with non-convex objective function and multiple constraints. Hence, it becomes difficult to define a good learning goal for multiple distributed agents, enabling efficient coordination amongst them. Another challenge is the information exchange bottleneck. While inter-agent communications can reduce some undesirable effects of locality and help ensure coordination of distributed agents, this is not always possible (or tolerable) due to limited communication bandwidth and communication constraints (*e.g.*, latency, energy consumption, privacy). Moreover, even when information exchange between agents is required, it must be relevant and efficient for both learning and communications. Finally, the limited computation capability of edge devices is also challenging and requires consideration when designing distributed learning mechanisms and communication frameworks.

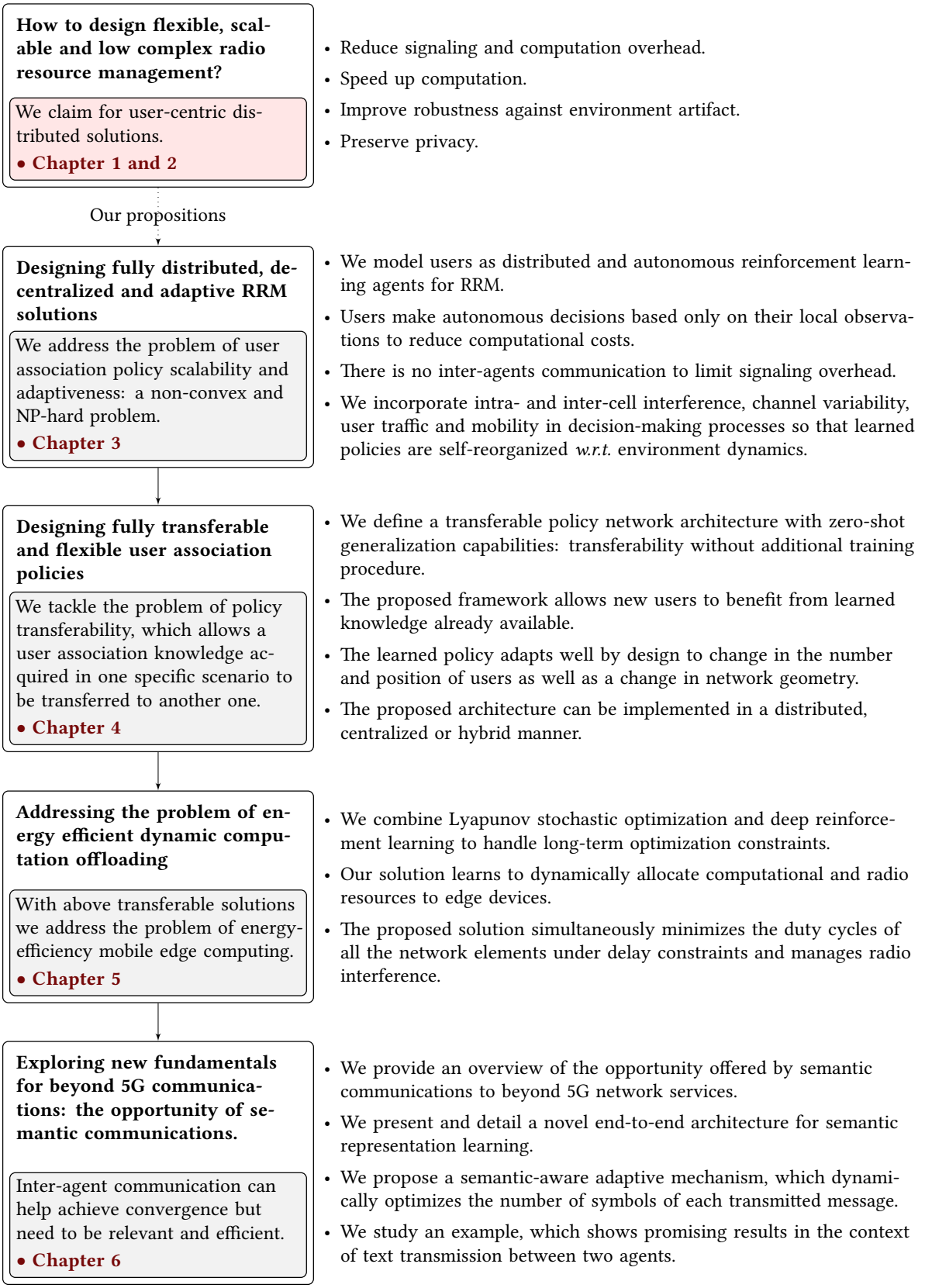


Figure 1.1: Thesis summary, starting from the initial question on how to build flexible, scalable and low complex radio resource management solutions.

1.3 Main Contributions and Outline

This thesis aims to address the above challenges associated with both radio resource management and distributed learning. We start by focusing on *user association problems*. The reason is that many RRM problems take a similar form as user association problems. Accordingly, in Chapter 2, we first review the user association problem and the associated challenges and complexities. In particular, in this chapter, we motivate the need for user-centric distributed approaches for flexible, scalable, and low complex RRM. Starting from this, our research work follows the roadmap of Figure 1.1, where our technical contribution begins in Chapter 3.

Chapter 3: Designing fully distributed and decentralized user association policies. In Chapter 3, we propose a novel distributed algorithm based on Multi Agent Reinforcement Learning (MARL), which enables fully distributed and decentralized user association. More specifically, we model each user as an autonomous agent that, at each time step, maps its *local observations* of the radio environment to an action, corresponding to an association request towards a base station in its coverage range. The novelty of the proposed solution also lies in the fact that there is no information exchange amongst users. Thus, we limit inter-agent communications, hence signaling overhead, while still being able to ensure coordination between users. In addition, our proposed solution incorporates the environment dynamics (channel interference, fast fading, and network traffic) during the learning phase so that the user association is self-reorganized toward the optimal association when a relevant change occurs in the environment. Therefore, we further reduce signaling overhead as well as computational complexity. This is in contrast to current state-of-the-art solutions, which do not consider the dynamic nature of wireless networks, thus, requiring to re-compute periodically or whenever a notable change has occurred in the environment to correct possible drifts from the optimal association. The proposed approach is validated in the context of user association in dense 5G networks with mmWave communications subject to severe path-loss, blockage, and deafness, which make the problem even more complex. We also propose an application of the proposed scheme to *distributed handover management* by considering users' mobility. Overall, the novelty of this chapter is validated in the following contributions.

[C1] **M. Sana**, A. De Domenico, and E. Calvanese Strinati, *"Multi-Agent Deep Reinforcement Learning based User Association for Dense mmWave Networks,"* In Proc. IEEE Global Communications Conference (GLOBECOM), HI, USA, pages 1–6., Dec 2019.

[C2] **M. Sana**, A. De Domenico, E. Calvanese Strinati, and A. Clemente, *"Multi-Agent Deep Reinforcement Learning for Distributed Handover Management In Dense MmWave Networks,"* In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Madrid, Spain, pages 8976–8980., May 2020.

[J1] **M. Sana**, A. De Domenico, W. Yu, Y. Lostanlen, and E. Calvanese Strinati, *"Multi-Agent Reinforcement Learning for Adaptive User Association in Dynamic mmWave Networks,"* IEEE Transactions on Wireless Communications, 19 (10):6520–6534, 2020.

[P1] **M. Sana**, A. De Domenico, *"Method for associating user equipment in a cellular network via multi-agent reinforcement learning,"* Issued in May 20, 2021, US17099922.

Chapter 4: Designing transferable policies for dynamic and scalable user association. One major limitation of RRM algorithms is that they are often grounded on quite rigid assumptions, such as pre-sized and fixed sets of BSs and static UEs, favorable channel conditions, absence of intra- or inter-cell interference, full-buffer network traffic. Yet, in dynamic mmWave networks, especially in dense networks, the number of UEs, their position *w.r.t.* each other and BSs, and the performance requirements of the services they access are likely to change over time and are characterized by a

high dynamicity. Even in relatively stable scenarios from the radio channel and data traffic points of view, the arrival in the network or the departure from the network of one or more users has an impact on the overall network performance, which requires a constant adaptation of the user association to dynamically guarantee the best possible quality of service. To address these issues, in Chapter 4, we propose a scalable and easily manageable user association policy. Specifically, our solution focuses on the central aspect of *transferability*. It allows applying a user association's strategy or policy acquired in a specific scenario (*e.g.*, a network deployment) to a distinct but related one without needing a substantial redesign, recomputation, or relearning of a new policy. Moreover, our proposed solution has *zero shot generalization* capability: it adapts well by design to variations in the number of users and their positions without requiring additional training. This feature significantly reduces the computational complexity of user association during the network operations and makes the policy suitable to distributed and dynamic scenarios. Overall, the novelty of this chapter is validated in the following contributions.

[C3] **M. Sana**, N. di Pietro, and E. Calvanese Strinati, *"Transferable and Distributed User Association Policies for 5G and Beyond Networks,"* In Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Virtual, Sept. 2021.

[P2] **M. Sana**, N. di Pietro, E. Calvanese Strinati, and B. Miscopein, *"Method for associating user equipment in a cellular network according to a transferable association policy,"* Filed in September 30, 2020, FR2009989.

Chapter 5: Addressing the problem of energy efficient dynamic computation offloading. So far, we have studied user association mechanisms to improve network spectral efficiency. We propose now to consolidate all these achievements to solve the problem of energy-efficient computation offloading enabled by edge computing. Indeed, with the deployment of computing and storage capabilities at the network edge, Edge Computing (also known as Multi-Access Edge Computing (MEC)) was conceived to enable energy-efficient, low-latency, highly reliable services by bringing cloud resources close to end-users. In this context, *dynamic computation offloading* allows resource-poor devices to transfer application execution to Edge Servers (ESs) to reduce energy consumption and latency. In the considered scenario, multiple users simultaneously compete for limited radio and edge computing resources to get offloaded tasks processed under a delay constraint, with the possibility of exploiting low-power sleep modes at all network nodes to reduce energy consumption. From a network management perspective, this task is complex and requires jointly optimizing radio and computation resources. In Chapter 5, we formulate the underlying problem as a *dynamic long-term optimization* aiming to reduce long-term energy consumption under strict delay constraints. Then, based on Lyapunov stochastic optimization tools, we show that this problem can be decoupled into *a per-slot* CPU scheduling problem and a radio resource allocation problem, namely a user association problem. Hence, we propose a fast iterative algorithm, particularly efficient to solve the first problem and hinge on the previously proposed user association scheme to solve the second one. Overall the originality of the resulting framework lies in its capacity to *simultaneously*: *i*) minimize the duty cycles of all the network elements under delay constraints; *ii*) effectively manage radio interference; *iii*) be low-complexity; *iv*) combine Lyapunov optimization with MARL; *v*) be distributed and compatible with UE's mobility. The novelty of this work has been validated by the following conference paper.

[C4] **M. Sana**, M. Merluzzi, N. di Pietro, and E. Calvanese Strinati, *"Energy Efficient Edge Computing: When Lyapunov Meets Distributed Reinforcement Learning,"* in Proc. IEEE International Conference on Communications (ICC) Workshops, Virtual, Montreal, Canada, June 2021.

Chapter 6: Exploring the opportunity of semantic and goal communications. We have shown in Chapter 4 and 5 that inter-agent communication, although limited, may be necessary for some scenarios to guarantee convergence. Back to Shannon's information theory, the goal of communication has long been to ensure the correct reception of transmitted messages irrespective of their meaning. However, in general, whenever communication occurs to convey a meaning, what matters is the receiver's understanding of the transmitted messages and not necessarily their correct reconstruction. This paradigm refers to *semantic communications*: transmitting only relevant information sufficient for agents to capture the intended meaning (the targeted objective) can notably reduce communication bandwidth. Therefore, in the last contribution of this thesis, we propose to explore the opportunity offered by semantic communications to beyond 5G networks services. To this end, in this preliminary work, we focus on semantic compression. In our study, we refer to *semantic* as a "meaningful" message (a sequence of well-formed symbols, which are possibly learned from data) that has to be interpreted at the receiver. This requires a reasoning unit, here artificial, based on a knowledge base, *i.e.*, a symbolic knowledge representation of the specific application. Thus, in Chapter 6, we propose and detail a novel E2E architecture that enables representation learning of semantic symbols for effective semantic communications. We discuss theoretical aspects and successfully design objective functions, which help learn effective semantic encoders and decoders. Also, we propose an adaptive mechanism, which dynamically optimizes the number of symbols of each transmitted message. Finally, we present some preliminary numerical results for a scenario of text transmission. In this scenario, a sender - an AI agent - transmits sentences in a given language by mapping each word to a sequence of semantic symbols that the receiver - another AI agent - must decode and understand in another language. We show that our proposed E2E framework can effectively address this problem, providing significant semantic compression gain. The novelty of this work has been validated by the following conference paper.

[C5] **M. Sana** and E. Calvanese Strinati, *"Learning Semantics: An Opportunity for Effective 6G Communications,"* in Proc. IEEE Consumer Communications and Networking Conference (CCNC), Virtual, Las Vegas, January 2022.

Part I

Approach and Methodology

Chapter 2

The User Association Problem

"On ne tire pas sur une fleur pour la faire pousser. On l'arrose et on la regarde grandir... patiemment."

"You don't pull on a flower to make it grow. You water it and watch it grow... patiently."

– Proverbe Africain

Contents

2.1 Introduction

User association is the process of assigning user equipment to network access points. It is a fundamental task, which is crucial in mobile communications as it directly affects the network spectral efficiency as well as the users' perceived QoS. However, the user association is a difficult task as it usually involves non-convex and NP-hard optimizations. In addition, optimal user association may require joint consideration of radio resources (*e.g.*, bandwidth, spectrum, power), computing resources (*e.g.*, computation power at a server) as well as learning resources (*e.g.*, distribution of data across users' devices in Federated Learning). This chapter aims to provide a global overview of the user association problem. We first present a general review of the literature on this problem. Then, focusing on the main characteristics of 5G networks, we formulate the user association problem in the context of mmWave networks and discuss its challenges and complexity. Finally, we explain our motivations and approach to address this problem using *distributed learning* mechanisms.

2.2 User Association Taxonomy

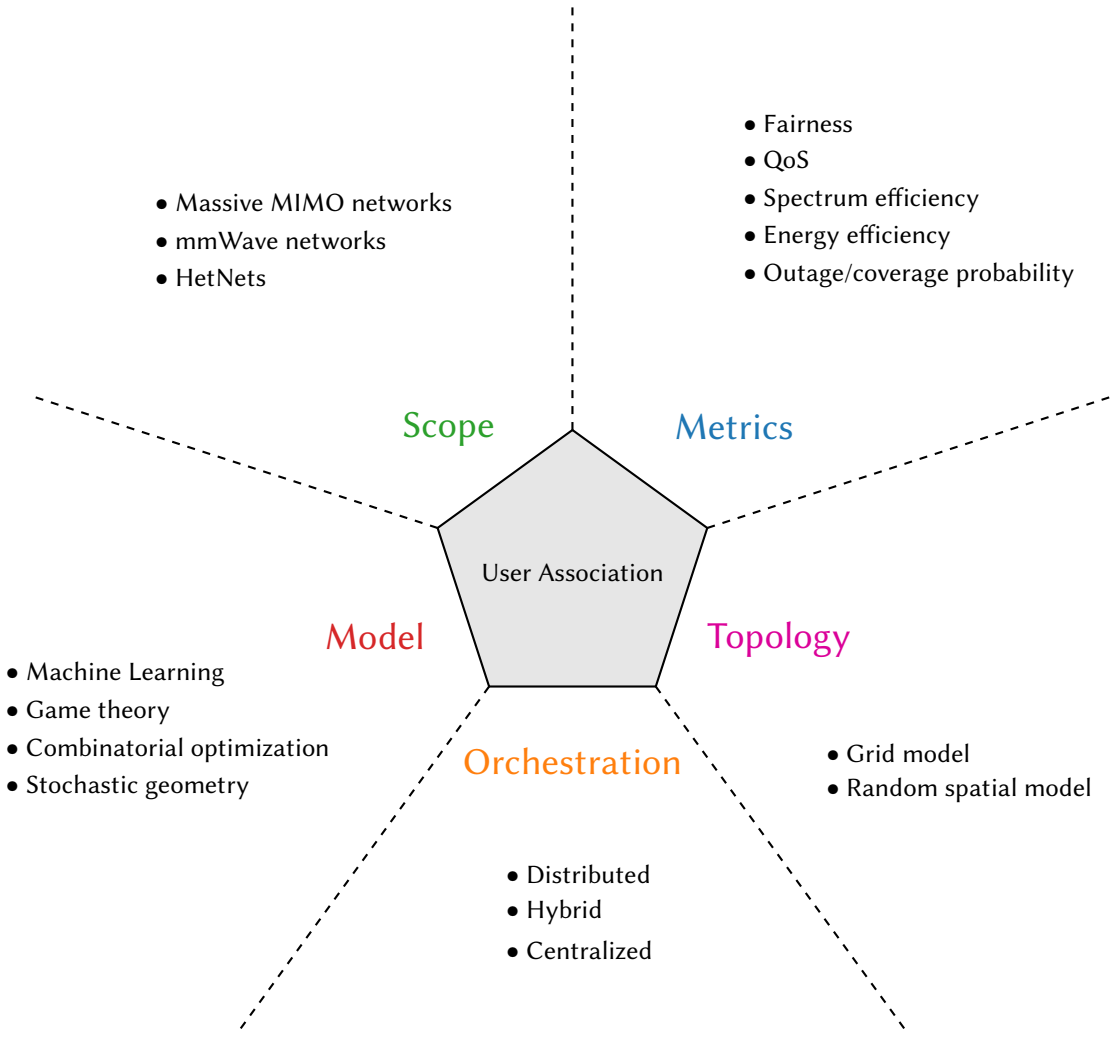


Figure 2.1: General (non-exhaustive) user association taxonomy. This taxonomy is classified given the scope (see subsection 2.2.1), the used metrics (see subsection 2.2.2), the network topology (see subsection 2.2.3), the orchestration mechanism (see subsection 2.2.4) and the used model (see subsection 2.2.5).

In the literature, numerous solutions have been proposed to address the problem of user association. These solutions utilize different *models*, *i.e.* approaches, depending on the *scope*, the used *metrics*, the network *topology* as well as the *orchestration* mechanism. In Figure 2.1, we summarize the different facets of the user association problem based on its taxonomy.

2.2.1 Scope

The scope refers to the type of network on which the user association is to be implemented. A non-exhaustive list of these networks spans from Heterogeneous Networks (HetNets) to massive Multiple-Input/Multiple-Output (MIMO) and millimeter-wave (mmWave) networks or combination of them. The challenges and complexities of the user association vary accordingly.

HetNets. They are characterized by the deployment of small cell base stations (*e.g.*, picocells, femtocells, relays) together with overlaying macro cell base stations, all possibly operating at different frequencies. In HetNets, cell densification is also considered to boost the network capacity by spatially reusing the spectrum across a geographical area, thereby improving coverage quality and the performance of cell-edge users [25]. However, densification poses a serious challenge to user association. As the number of network nodes increases, the orchestration of radio resources becomes complex. Moreover, network capacity does not increase systematically with the densification of network access points as this also ultimately leads to co-channel interference. In addition, in HetNets, backhaul links typically connect small cells and macro cells to the core network, which can limit the user association performance when they are not provisioned sufficiently [26, 27]. Therefore, efficient resource allocation is required to take full advantage of HetNets.

Massive MIMO networks. Thanks to beamforming techniques, massive MIMO allows base stations with large antenna arrays to support multiple UEs simultaneously over the same time and frequency range. It can achieve high multiplexing gain, thus, substantially improving spectrum efficiency. Moreover, massive MIMO achieves high antenna gain, significantly increasing received signal power or equivalently reducing transmit power to meet a targeted QoS. In addition, thanks to the extra diversity afforded by massive MIMO large antenna arrays, channel estimation errors, small-scale fading effects are averaged out, vanishing undesirable instantaneous fluctuations [28]. However, from a RRM point of view, user association in massive MIMO networks is difficult due to the complex design of channel precoding and complex beam management. For example, there are 35960 possibilities to set up 4 beams out of 32 possible beams. Determining the optimal set is very challenging [29, 30].

mmWave networks. Due to the large spectrum resources available between $28-90$ GHz, the adoption of mmWave communications in 5G, enables significant improvement of the network capacity [6]. Indeed, mmWave technology enables highly directional communications using narrow beams, thus achieving high beamforming gain. In addition, due to the short wavelength characteristics of mmWave, the size of the antenna element is reduced, thus allowing the compact design of antenna array [6]. However, mmWave transmissions suffer from severe path-losses and are highly sensitive to blockages [31], which challenges user association especially in the context of mobility (*e.g.* for handover management [32]).

2.2.2 Metrics

Different metrics are used to assess the performance of the user association. Here we list the main metrics:

Fairness. Fairness here, refers to how the user association strategy treats different UEs depending on *e.g.* their QoS. Main fairness criteria are:

- Max-min fairness, where the optimization of the user association is meant to maximize for *e.g.* the lowest achievable rate amongst users.

- Proportional fairness, which maintains a balance between maximizing the network throughput and allowing all users a chance to be connected instead of prioritizing best users (*e.g.* users with good Signal-to-Noise-plus-Interference Ratio (SINR) or high data request) at the expense of the others.

QoS. The QoS gives an indication of the service quality experienced by users in the network. It is often quantitatively expresses in terms of latency, user throughput or SINR, packet loss, etc.

Spectrum and Energy efficiency. Spectrum efficiency is an important performance indicator. It measures the total throughput achievable in the network for a given allocated bandwidth. For example, one of the main targets of 5G is to provide eMBB services, which are characterized by high data rate requirements [5]. On the other hand, energy efficiency measures the energy-saving capability of a given user association algorithm.

Coverage/Outage probability. The coverage (outage) probability defines the probability that the SINR of a randomly chosen user in the network goes above (drops below) a certain threshold [31]. It is often used to characterize the probability of satisfying users QoS.

2.2.3 Topology

The two mains topologies used in the literature are the Grid model and the random spatial model. In the Grid model, the APs are assumed to be uniformly distributed in the center of regular grids. In a random spatial model, the APs are randomly distributed in the network (usually according to a Poisson Point Process (PPP)). The later model is often used in conjunction with stochastic geometry analysis to capture the randomness of network geometry [33].

2.2.4 Orchestration

In general, user association algorithms can be classified into three main categories: i) *centralized algorithms*, which usually provide near-optimal solutions. However, they require to collect and process information (such as CSI) from multiple network nodes in a unified way, which induces a large amount of signaling, ii) *distributed algorithms*, which generally lead to sub-optimal solutions but have low computational complexity and low signaling overhead due to local decisions, and finally, iii) *hybrid algorithms*, which exploit the advantages of both centralized and distributed algorithms.

2.2.5 Model

Solutions to user association problems are diverse and range according to the above taxonomy. Some works investigate combinatorial optimization using Lagrangian tools [34] or fractional programming [35]. Other approaches include game theory [36, 37] and stochastic geometry [38]. Most recent works on user association involve Machine Learning and Reinforcement Learning to cope with user association complexities and the radio environment dynamics [39, 40, 41].

2.3 User Association in HetNets with mmWave Communications

To better understand the user association problem, we propose to formulate the underlying optimization problem. This problem formulation will help understand its central complexities and why this requires further research. To this end, in this chapter, we focus on downlink mmWave communications for eMBB services, which are characterized by high data rate and are at the core of the performance improvement expected in 5G [5]. Accordingly, we focus on the objective to maximize the total network sum-rate. This objective also considers the data requirement of different eMBB UEs to devise optimal user association strategies. However, in Chapter 5, we will show how the proposed solutions can be leveraged for

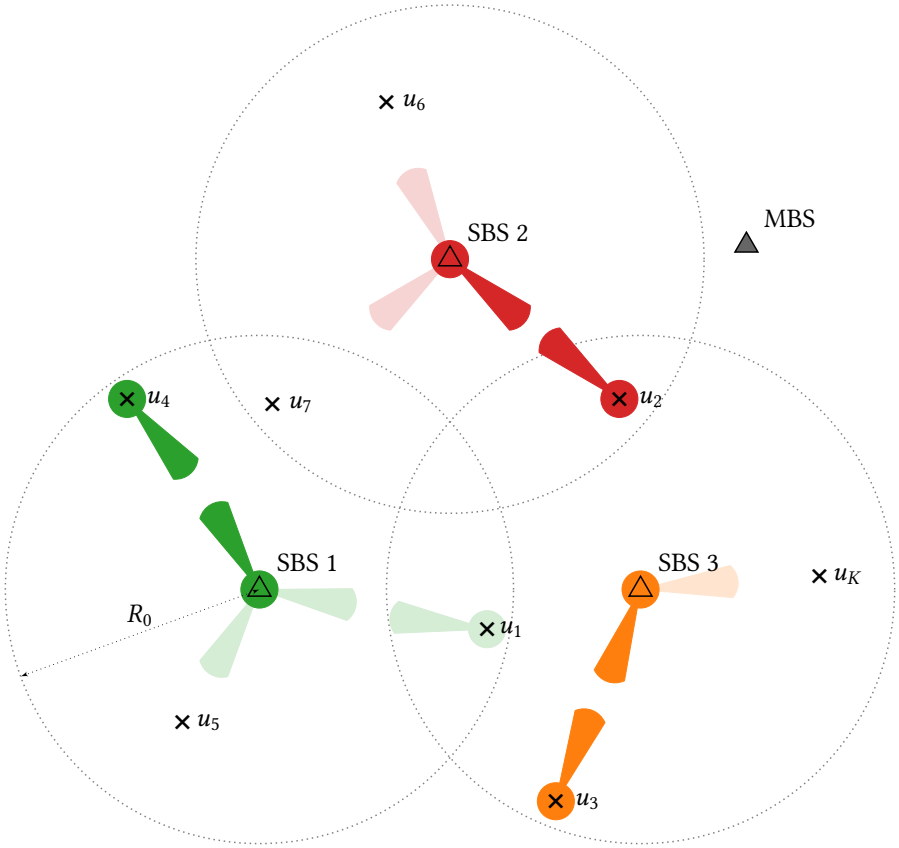


Figure 2.2: A downlink heterogeneous network with $N_s = 3$ SBSs operating a mmWave frequencies, one sub-6 Ghz MBS, and K UEs. Here, as a example, the number of UEs under SBS 1 coverage is $\mathcal{U}_1 = \{1, 4, 5, 7\}$, and UE 1 action space is $\mathcal{A}_1 = \{1, 3\}$.

energy-efficient uplink communications in the context of dynamic computation offloading enabled by edge computing.

2.3.1 General system model

We consider a downlink network consisting of N_s mmWave small cells and one macro cell[1] jointly providing services to K randomly deployed UEs as shown in Figure 2.2. We denote by $\mathcal{A} = \{0, 1, 2, \ldots, N_s\}$ the set of $N_s + 1$ BSs in the network where 0 indexes the Macro Base Station (MBS), which uses sub-6 GHz technology to enable ubiquitous network coverage. Also, we use $\mathcal{U}_i$ to indicate the set of UEs under coverage of the i-th BS; hence, $\mathcal{U} = \bigcup_{i=0}^{N_s} \mathcal{U}_i = \{1, 2, \ldots, K\}$ represents the set of all UEs in the network.

In this architecture with multi-radio access technologies, a UE may receive control signals from multiple BSs. Therefore, we define $\mathcal{A}_j = \{i, \quad d_{i,j} \leq \phi_i/2, \ i \in \mathcal{A}\} \subseteq \mathcal{A}$[2] as the set of BSs the UE j could connect to, where $\phi_i/2$ is the cell radius of the BS i and $d_{i,j}$ is the distance between BS i and UE j. Note that $\mathcal{A}_j \neq \emptyset, \forall j$ as a UE can always be associated with the MBS. Let $x_{i,j} \in \{0, 1\}$ be the binary association variable such that $x_{i,j} = 1$ when UE j is served by the BS i and $x_{i,j} = 0$ otherwise. Here we assume that each UE can only receive data from one BS at a time. Moreover, due to limited resource and hardware complexity, we consider that each mmWave Small cell Base Station (SBS) cannot serve

[1] The limitation to one macro cell is made for simplification reasons. The extension to several macro cells is however trivial.

[2] $\mathcal{A}_j$ can also be derived based on links quality, *e.g.*, the received signal strength indicator between UE j and BS i ($\text{RSSI}_{i,j}$) should be greater than a predefined threshold ζ_j, *i.e.*, $\mathcal{A}_j = \{i, \ \text{RSSI}_{i,j} \geq \zeta_j\}$.

more than N_i UEs simultaneously, where N_i is the maximum number of beams available at the SBS i.

Assumptions. In our system model, we consider that the SBSs allocate all the available mmWave's band to each served UE using Spatial Division Multiple Access (SDMA); in contrast, the MBS equally shares its band across the served UEs. Finally, we consider that the SBSs and the UEs have already performed beam training and alignment mechanisms in advance and therefore are able to configure the appropriate beams when a data connection is set up. For instance, an initial access protocol based on the SINR can be used to complete this task [42].

2.3.2 Channel model

For simplicity of analysis, and since we consider a dense regime, following [31], we denote with R_0 the size of the coverage range of SBSs. Thus, a UE can only be associated with a SBS located at most at a distance R_0. Moreover, we consider that each communication link experiences a small scale m-Nakagami fading. We use h to denote the fading coefficient, which follows a normalized Gamma distribution $\Gamma(m, \frac{1}{m})$. We assume Rayleigh fading for UE-MBS links, which is a special case of m-Nakagami fading by taking $m = 1$. In addition, we adopt the commonly used Friis propagation loss model [43], where the received power P^{Rx} is given as a function of the distance d between the UE and its serving BS:

$$P^{\mathrm{Rx}}(d) = hP_s^{\mathrm{Tx}}G_s^{\mathrm{Tx}}G_s^{\mathrm{Rx}}C_s d^{-\eta_s}, \quad s \in \{\mathrm{MBS}, \mathrm{SBS}\}. \tag{2.1}$$

Here, C_s denotes the path-loss constant, η_s is the path-loss exponent, and, P_s^{Tx} is the transmit power *w.r.t.* BS s. Later, we denote with $G_s^{\mathrm{Ch}}(d) = hC_s d^{-\eta_s}$ the channel gain. The transmitter and receiver antennas' gain *w.r.t.* BS s are G_s^{Tx} and G_s^{Rx} respectively. In addition, we assume that the UEs and the BSs perform beam steering in advance such that when a communication is set up, the useful received power in absence of interference is maximized, *i.e.*, $G_s^{\mathrm{Tx}} = G_{\max}^{\mathrm{Tx}}$ and $G_s^{\mathrm{Rx}} = G_{\max}^{\mathrm{Rx}}$, where $G_{\max}^{\mathrm{Tx}}$ and $G_{\max}^{\mathrm{Rx}}$ are the maximum antenna gain at the transmitter and the receiver, respectively.

2.3.3 Cell interference

Since we assume the presence of a single MBS, which orthogonalizes the UEs it serves by sharing its band across them, the communication links between the MBS and its served UEs experience neither intra-cell nor inter-cell interference. Therefore, interference is only due to the communications between mmWave SBSs and UEs, as a result of overlapping beams. Indeed, let us consider a typical UE (say UE j_0) placed at a distance d_0 from its serving SBS (say SBS i_0). Given an interfering SBS i that is located at a distance d_i with a relative angle ψ_i *w.r.t.* the typical UE, which is serving $n_i \leq N_i$ other UEs in n_i random directions defined by their relative angle $\phi_{i,j}$ (see Figure 2.3), we use $I_{i,j}$ to denote the interference caused by its j-th beam towards the typical UE. Thus, $I_i = \sum_{j=1}^{n_i} I_{i,j}$ is the total interference engendered by this SBS on the typical UE. For sake of simplicity, we assume that the receiver and the transmitter use the same antenna radiation pattern denoted by $G(\theta, \alpha)$, where θ is the beam width and α is the azimuthal angle to the main lobe (see Figure 2.3). Hence, the interference induced by the communication between the i-th SBS and its j-th UE is:

$$I_{i,j} = P^{\mathrm{Tx}}h_i G(\theta, \psi_i)G(\theta, \phi_{i,j})C d_i^{-\eta}. \tag{2.2}$$

The total interference induced by SBS i on the typical UE can be classified into two categories:

Inter-cell interference. If $i \neq i_0$, meaning the typical UE experiences interference coming from a BS different from its serving BS, then

$$I_i = P^{\mathrm{Tx}}h_i C \sum_{j=1}^{n_i} G(\theta, \psi_i)G(\theta, \phi_{i,j})d_i^{-\eta}, \quad i \neq i_0 \tag{2.3}$$

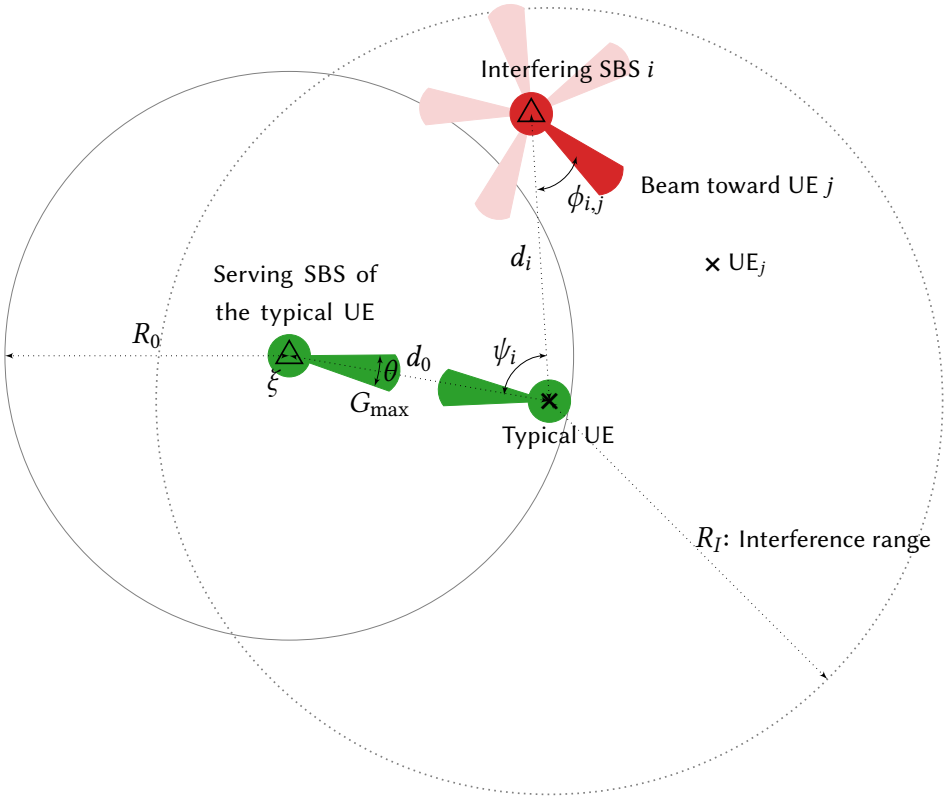


Figure 2.3: Cell interference illustration

As we assume spatial reuse of the spectrum across the different small cells, inter-cell interference mainly affects cell edge users, which can be detrimental to their performance, especially in dense networks.

Intra-cell interference. If $i = i_0$, meaning that the interference perceived by the typical UE comes from its serving BS resulting from the communications of the other UEs associated with this BS.

$$I_{i_0} = P^{\mathrm{Tx}} h_{i_0} C G_{\max}^{\mathrm{Rx}} \sum_{j=1,\ j\neq j_0}^{n_{i_0}} G(\theta, \phi_{i_0,j}) d_0^{-\eta}. \tag{2.4}$$

Although additional beamforming techniques can help to mitigate intra-cell interference, we do not assume specific numerical beamforming. Therefore, both intra- and inter-cell interference result from the overlapping of different beams serving different UEs. Hence, the signal-to-interference-plus-noise ratio SINR_{i_0,j_0} between the typical UE j_0 and its serving BS i_0, which comprises both intra-cell and inter-cell interference is defined as:

$$\mathrm{SINR}_{i_0,j_0} = \frac{P^{\mathrm{Tx}} h_0 G_{\max}^{\mathrm{Tx}} G_{\max}^{\mathrm{Rx}} C d_0^{-\eta}}{\sum_{i\in\mathcal{A}} I_i + N_0 B_{i_0,j_0}}, \tag{2.5}$$

where $B_{i,j}$ is the bandwidth allocated to the UE j by BS i, and N_0 is the noise power spectral density.

The generic nature of our system model makes it easily transposable to other radio technologies. Eventually, to improve the network spectral efficiency, we focus on optimizing the user association.

2.3.4 The user association problem: challenge and complexity

From the above definitions, the achievable communication rate between BS i and UE j is given by the Shannon capacity:

$$R_{i,j}(t) = B_{i,j}\log_2\left(1 + \text{SINR}_{i,j}(t)\right). \tag{2.6}$$

In our model, we take into account the UEs traffic request to devise user association strategies. Accordingly, let denote with $D_j(t)$, the *data rate demand* of UE j at time step t. We assume that follows $D_j(t)$ a Poisson distribution with intensity $\overline{D}_j = \mathbb{E}\left[D_j(t)\right]$. Therefore, given a UE j with a traffic demand $D_j(t)$, the effective data rate exchanged with BS i at the time t is $\min\Big(D_j(t), R_{i,j}(t)\Big)$. Next, let $R(t)$ be the α-fair network utility function, which is defined as follows:

$$\begin{aligned} R(t) &= \sum_{i\in\mathcal{A}}\sum_{j\in\mathcal{U}} x_{i,j}U_\alpha\left(\min\left(R_{i,j}(t), D_j(t)\right)\right), \\ &= \sum_{i\in\mathcal{A}}\sum_{j\in\mathcal{U}} x_{i,j}U_\alpha\left(\min\left(1, \frac{R_{i,j}(t)}{D_j(t)}\right)D_j(t)\right), \\ &= \sum_{i\in\mathcal{A}}\sum_{j\in\mathcal{U}} x_{i,j}U_\alpha\left(\kappa_{i,j}(t)D_j(t)\right), \end{aligned} \tag{2.7}$$

where $x_{i,j} = 1$ indicates that UE j is associated with BS i; otherwise $x_{i,j} = 0$ and $\kappa_{i,j}(t) = \min\left(1, \frac{R_{i,j}(t)}{D_j(t)}\right) \in [0,1]$ indicates the QoS satisfaction of UE j *w.r.t.* its associated BS i, which is fully satisfied when $\kappa_{i,j} = 1$. Here, $U_\alpha(\cdot)$ is the α-fair utility function given in [44] as follows:

$$U_\alpha(x) = \begin{cases} (1-\alpha)^{-1}x^{1-\alpha}, & \text{for some } \alpha \geq 0 \text{ and } \alpha \neq 1, \\ \log(x), & \alpha = 1. \end{cases} \tag{2.8}$$

User association problem. Following above definitions, we formulate the user association problem to maximize the network utility as follows:

$$\underset{\{x_{i,j}\}}{\text{maximize}} \quad R(t) = \sum_{i\in\mathcal{A}}\sum_{j\in\mathcal{U}} x_{i,j}U_\alpha\left(\kappa_{i,j}(t)D_j(t)\right), \tag{2.9}$$

$$\text{subject to} \quad x_{i,j} \in \{0,1\}, \qquad \forall i,j, \tag{2.10}$$

$$\sum_{j\in\mathcal{U}_i} x_{i,j} \leq N_i, \qquad \forall i \in \mathcal{A}\backslash\{0\}, \tag{2.11}$$

$$\sum_{i\in\mathcal{A}_j} x_{i,j} = 1, \qquad \forall j \in \mathcal{U}. \tag{2.12}$$

The constraint (2.10) ensures that the decision variables $x_{i,j}$ are binary. The constraint (2.11) highlights that a given SBS i can serve at most N_i UEs at the same time. Finally, constraint (2.12) indicates that, in our setting, each UE is associated with exactly one BS. Note that additional constraints can also be considered, such as limited transmit power, strict end-to-end delay constraints, which we will discuss in Chapter 5.

User association fairness. Depending on the value of α, this optimization problem guarantees different fairness criteria in the user association. Indeed, we have the following Lemma:

Lemma 1 (from [44], Section 2.2.1). *When $\alpha = 0$, $U_\alpha(x) = 1$ and problem (2.9)-(2.12) is equivalent to the network sum-rate maximization problem. When $\alpha \to 1$, $U_\alpha(\cdot)$ converges to the proportional fair utility function as $\lim_{\alpha\to 1} U_\alpha(x) = \log(x)$ and problem (2.9)-(2.12) falls into network sum-log-rate maximization, also known as proportional fairness user association. As $\alpha \to \infty$, $U_\alpha(x)$ approaches max-min fairness utility function and problem (2.9)-(2.12) is equivalent to max-min fairness user association.*

Complexity analysis. The formulated problem (2.9)-(2.12) is non-convex. Indeed, the association of a given UE j with a given BS i depends on its $\text{SINR}_{i,j}$ value. However, by observing Eqn. (2.5), the expression of the SINR also depends on the association of other users through the interference terms in the denominator. These cross-dependencies combined with the binary decision variables make the optimization problem non-convex and NP-hard, hence, difficult to solve with conventional optimization frameworks [45]. The difficulty is exacerbated when considering the UEs traffic as it introduces a non-linearity through the $min(.,.)$ function. A naive algorithm, which may find the optimal solution of problem (2.9)-(2.12) through an *exhaustive search*, has a complexity equal to $O\left(N_s K(1+N_s)^K\right)$.

Proof. For UE j there are $\text{card}(\mathcal{A}_j)$ possible choices of BSs. The optimal association $\{i,\ \text{s.t. } x_{i,j} = 1\ \forall i \in \mathcal{A}\}$ is an element of $\underset{j\in\mathcal{U}}{\times} \mathcal{A}_j$. That is, for all UEs, there are $\prod_{j\in\mathcal{U}} \text{card}(\mathcal{A}_j)$ possible combinations in which only some of them satisfy the constraint (2.11). For each combination, checking if constraint (2.11) is satisfied required $O\left(\sum_{i\in\mathcal{A}} \text{card}(\mathcal{U}_i)\right)$ iterations. In the worst case, when each UE can associate with any BS, $\text{card}(\mathcal{A}_j) = N_s + 1$. Hence, noting that $\sum_{i\in\mathcal{A}} \text{card}(\mathcal{U}_i) \leq N_s K$, the complexity of running this naive algorithm will be therefore

$$O\left(\sum_{i\in\mathcal{A}} \text{card}(\mathcal{U}_i) \prod_{j\in\mathcal{U}} \text{card}(\mathcal{A}_j)\right) = O\left(N_s K(1+N_s)^K\right). \tag{2.13}$$

□

This complexity is function of the number of UEs K and BSs N_s; and in particular, it has a polynomial complexity *w.r.t.* the number of BSs and an exponential complexity *w.r.t.* the number of UEs. Therefore, such an approach based on exhaustive search is infeasible especially in 5G context due to dense deployment of network access points and UEs with different service requirements. In this heterogeneous ecosystem, there is a need for flexible, scalable and adaptive network design and orchestration mechanism to meet challenges and requirements of 5G and beyond networks.

2.4 On Distributed Approach for Efficient User Association

In existing 5G networks (as well as in Long Term Evolution (LTE) networks), the user association takes place at the Radio Resource Control (RRC) sub-layer, which decides how users are associated depending on their QoS requirements, their priority or the availability of radio resources to maximize the radio exploitation [46]. In conventional cellular systems, the user association is centralized and based on the max-SNR or the max-RSS solution, *i.e.*, a UE is associated with the BS providing either the maximum Signal-to-Noise Ratio (SNR) or the maximum Received Signal Strength (RSS). While these rudimentary solutions have the advantage of low computational complexity, they do not take cellular interference into account and are therefore inefficient in dynamic 5G networks with mmWave communications. Moreover, in a mobility context, solutions based on max-SNR or max-RSS are inefficient due to frequent

handovers. In addition, network-centric solutions require a periodic collection and processing of information (*e.g.*, SNR, RSS, CSI) in a unified way, ultimately leading to significant signaling overhead. Also, with network densification, it becomes infeasible for one central orchestrator to find an optimal association among multiple deployed APs and UEs due to the aforementioned complexities of the user association problem. Hence, as 5G and beyond technologies become more and more sophisticated, the range of services to be supported increases, the QoS requirements become more stringent with a variety of services that needs to coexist on the same network infrastructure, the user association problem calls for more advanced solutions.

Among the various solutions under consideration, distributed user-centric solutions can overcome excessive communications and computation by implementing RRM algorithms at the user side [45, 47]. In particular, the adoption of *distributed* AI at the network edge (*edge intelligence*) is envisioned. In this scenario, multiple distributed AI-powered devices *can learn and possibly share their knowledge* to optimize some network utility functions and achieve some common goals [3, 12]. This approach is currently made possible by endowing mobile devices with AI algorithm computing capabilities. Although training a deep neural network on mobile devices in a computation and energy efficient way is an ongoing research topic, notable efforts have already been made both in terms of hardware design and software accelerators (see [13, 14] and references therein). This makes it possible to move part of the optimization process to the user devices. Therefore, this thesis adopts a user-centric approach and aims to investigate *distributed learning* approaches to address radio resource management problems. Very recently, the work [48] has surveyed user-centric radio access technology selection. They have focused on the user association problem and have highlighted multi-agent learning together with game-theoretical approaches as promising tools to address this problem. Also, the work [49] have recently investigated applications of machine learning to handover management problem in 5G and beyond. All these recent works further support our motivation towards distributed radio resource management.

2.5 Conclusion

In this chapter, we introduced the user association problem. We reviewed its general taxonomy and formulated the main problem in an HetNet with millimeter-wave communications enabled. We also highlighted the central challenges and complexities of the user association problem, which we showed to be non-convex and NP-hard. This leads us to look for scalable, flexible, and low complexity solutions. In particular, we call for *distributed* user-centric solutions instead of cumbersome centralized algorithms, which become infeasible in dense networks such as in 5G networks.

Now that we have motivated the need for distributed user-centric solution, in the next chapter, we will discuss our proposed solution to address the user association problem based on *distributed* multi-agent reinforcement learning approach.

Chapter 3

Distributed Learning of User Association Policies

"Do the best you can until you know better. Then when you know better, do better!"

"Faites du mieux que vous pouvez jusqu'à ce que vous ayez une meilleure connaissance. Et quand vous aurez appris suffisamment, faites mieux !"

– Maya Angelou (1918 – 2014)

Contents

3.1 Introduction

To solve the user association problem in a distributed way, our approach focuses on distributed Multi Agent Reinforcement Learning (MARL). In this chapter, we describe our proposed solution to solve the user association problem in a static and dynamic environment. First, to limit the complexity of the proposed solution, we cast this problem to a MARL framework, where each user independently learns the optimal policy. The proposed solution is distributed, which alleviates computation burdens. In addition, we do not allow inter-agent communications, thus limiting signaling overhead, which we characterize with the signaling messages required to implement the solution in a practical system.

3.1.1 Motivations

In the previous chapter, we have motivated the need for distributed solutions for efficient radio resource management. In particular, we argue that the (entirely) network-centric approaches used so far are no longer suitable in the current generation of wireless networks due to the signaling and computation complexity involved in centralized orchestration. Moreover, 5G and beyond technological solutions (*e.g.*, adoption of mmWave communications, massive MIMO and network densification technologies) are becoming extremely sophisticated, with stringent QoS requirements and a variety of services, which must coexist together. This requires the search for advanced solutions for efficient Radio Resource Management (RRM). In this thesis, our approach focuses on distributed MARL. By using MARL framework, there is no need for an expert database or modeling of the radio environment. Moreover, MARL can be used to model environments with complex interactions where it is difficult to obtain tractable mathematical models. To show this, consider the following well-known riddle example [50].

Figure 3.1: The Dalton is a French animated television series, prisoners of a penitentiary in the Nevada desert, the Dalton brothers try to escape from the penitentiary... but without achieving their ends (src. Wikipédia).

Example 1 ($n = 100$ prisoners and a light bulb)**.** *One hundred prisoners have been newly ushered into prison (see Figure 3.1 for illustration). The warden tells them that starting tomorrow, each of them will be placed in an isolated cell, unable to communicate amongst each other. Each day, the warden will choose one of the prisoners uniformly at random with replacement, and place him in a central interrogation room containing only a light bulb with a toggle switch. The prisoner will be able to observe the current state of the light bulb. If he wishes, he can toggle the light bulb. He also has the option of announcing that he believes all prisoners have visited the interrogation room at some point in time. If this announcement is true, then all prisoners are set free, but if it is false, all prisoners are executed. The warden leaves and the prisoners huddle together to discuss their fate. Can they agree on a protocol that will guarantee their freedom?*

Although this problem does not appear at first as directly linked to a wireless communication problem, there exist some similarities. Indeed, to make the parallel, let consider the one hundred prisoners as *one hundred deployed user devices* in a wireless network, which aim to collaborate to optimize an objective function, here, *to get freed.* For this, they share a common communication resource, the *interrogation room.* Also, they are allowed to communicate through a light bulb (by observing its state and being able to switch it off/on), which is a one-bit communication means, without direct exchange amongst users. Finding the optimal protocol that guarantees prisoners' freedom as fast

as possible is a difficult task, which becomes extremely complex when the number of participating users (the prisoners) in the protocol establishment increases. Additional complexity is that the decision of one user can be detrimental to other users (either they get freed or executed). Same constraints also exist in wireless communications, where *e.g.* the interference resulting from one user's wrong association can severely affect the throughput of other users. Whereas it is difficult to come out with a mathematical formulation of such a complex problem, it can be cast to and successfully solved using MARL [51]. In the sequel, we propose to use MARL approaches to address RRM problems, namely user association.

3.1.2 Related work

Several works have investigated distributed learning for the user association problem [45, 48]. In [34], Athanasiou *et al.* have designed a distributed algorithm to manage the user association using Lagrangian tools. Their solution is sub-optimal as it intentionally ignores interference and does not consider the environment dynamics. Similarly, Lui *et al.* have formulated a decentralized non-cooperative game with local interactions to manage the beam pair selection between UEs and BSs to maximize the network sum-rate [37]. However, this proposal requires information exchange among UEs, thus, inducing large signaling overhead. Moreover, this work also does not consider the environment dynamics. A load-balancing user association is proposed in [52, 36] to balance the radio resources across BSs. Leveraging a game-theoretical approach, the user association is formulated as a matching game in [53] and as a multi-armed bandit problem in [54]. These studies share a common point: the proposed user association solutions are sensitive to the deployment of users and BSs as well as the environment dynamics (fading, interference, traffic) and need to be run continuously to keep tracking of relevant changes in the network. This introduces a significant signaling and computational overhead. Recently, advances in machine learning and reinforcement learning [55, 24] have enabled the design of more flexible algorithms for optimizing the user association. In this context, a Deep Neural-Network (DNN) architecture is introduced in [39] that predicts the user association and power allocation. Similarly, authors in [41] have formulated the problem of user association with multi-connectivity as a multi-label classification problem. All these works are based on cumbersome databases, which in practice, are difficult to acquire. To address this problem, Zhao *et al.* have proposed a distributed user association based on deep MARL algorithm [40]. Nevertheless, [40] has not focused on mmWaves networks and has considered a fully observable environment. Besides that, the solution proposed in [40] is not scalable as the architecture of the proposed DNN depends on the total number of interacting UEs. In contrast, the main goal of the investigation conducted in this chapter focuses on the design of scalable and dynamic user association strategies able to self-reorganize *w.r.t.* the network dynamics (fading, traffic, and interference).

3.1.3 Contributions

The contributions of this chapter can be summarized as follows:

- *Sum-rate maximization in dense mmWave networks*: we first formulate a user association problem to maximize the sum-rate of mmWave networks. In contrast to the existing works, we take into consideration both inter-cell and intra-cell interference and environment dynamics, which are characterized by the time-varying nature of the mmWave channels and the evolving data rate demand of UEs by using only local observations at each UE.

- *Multi-agent reinforcement learning based user association scheme*: we cast the formulated user association problem into a multi-agent reinforcement learning task, where UEs, modeled as agents, collaborate to maximize the network sum-rate. To limit both signaling and computational complexity, the agents act as independent learners *i.e.*, their decisions are independent of each other. We force UEs to act based only on partial observations and perceived rewards by avoiding

inter-agent communications. Such a constraint brings the benefit that a UE does not need to collect and process information related to other users. In this setting, we propose a Deep Recurrent Q-Network (DRQN) architecture and the associated signaling protocol, which enable UEs to learn an efficient association policy for network sum-rate maximization.

- *Mobility management*: we further show that the proposed framework can also be extended to account for mobility in applications for handover management. In these scenarios, the learning goal is to minimize the handover frequency while maximizing the total network sum-rate.

The technical content of this chapter is based on the published journal paper [56], conference papers [57, 58], and patent [59].

The remainder of this chapter is organized as follows. Section 3.2 briefly introduces MARL framework. Section 3.3 details the proposed distributed algorithm and the associated signaling protocol. Section 3.4 discusses the extension of the proposed framework to handover management. Numerical results are provided in each section and conclusions are drawn in Section 3.5.

3.2 Background on Multi-Agent Reinforcement Learning

3.2.1 Markov Decision Processes

In a fully observable environment, single agent decisions making can be formalized as a Markov Decision Process (MDP). Basically, an MDP is defined as tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R})$, in which $\mathcal{S}$ denotes the state space, $\mathcal{A}$ is the action space, $\mathcal{T}(s, a, s') = P(s'|s, a)$ the probability of transitioning from state s to state s' after taking action a, which results in an immediate reward $\mathcal{R}(s, a)$. The problem for agent in an MDP is to find the optimal policy $\pi^* : \mathcal{S} \rightarrow \mathcal{A}$ that maximizes the expected sum of the perceived rewards (possibly discounted), namely the action-value (or Q-value), which is defined as follows:

$$Q^{\pi}(s, a) = \mathbb{E}\left[R_t | s(t) = s, a(t) = a\right]. \tag{3.1}$$

Here, $R_t = \sum_{\tau=t}^{T} \gamma^{\tau-t} r(t)$ is the γ-discounted return from time t, and $r(t) = \mathcal{R}(s(t), a(t))$ is the instantaneous reward perceived by the agent. Hence, the optimal policy is such that $Q^{\pi^*}(s, a) = \max_{\pi} Q(s, a)$.

3.2.2 Partially Observable Processes

In real environment (as in wireless networks), agent has only access to observations o of the latent state s. In this case, we speak of a Partially Observable Markov Decision Process (POMDP), which is formalized as $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \Omega, \mathcal{O})$, in which $\mathcal{O}(o, a, s') = P(s'|o, a)$ denotes now the transition probability to state s' after observing $o \in \Omega$ and taking action a. Hence, an agent in POMDP learns to map observations to actions that yield the best (long-term) rewards.

3.2.3 Reinforcement learning

A RL agent learns by interacting with an environment following a MDP to devise the optimal policy π^*, without an explicit provision of the environment model. In model-based RL (*e.g.* multi-armed bandits), the transition probability $\mathcal{T}$ and the reward function $\mathcal{R}$ are first estimated and then used to derive Q^{π^*}. In contrast, model-free approaches directly estimate the Q-values Q^{π} (*value-based* approach) or the policy π (*policy gradient-based* approach), which can be memory and computation efficient. In the latter case, Q-learning is a widely use model-free value-based approach particularly efficient for problems with small states/actions space [17]. Coupled with Neural-Network (NN), Q-learning allows to address complex problems using Deep Q-Network (DQN): $Q(s, a) \approx Q(s, a; \theta)$, where θ is the set of neural network parameters used to approximate the Q-function [18]. DQN relies on experience replay to speed up and stabilize the training process [18]. At each time t, from a state $s(t)$, agent takes an action $a(t)$

following a policy (*e.g.*, ϵ-greedy), which brings it to a new state $s(t+1)$ with an immediate reward $r(t)$. The resulting experience $e(t) = \{s(t), a(t), r(t), s(t+1)\}$ is stored into an experience replay memory $\mathcal{M}$ from which a mini batch of experiences $\mathcal{B}$ is sampled every iteration to perform the learning phase. In this phase, the weights of the DQN are iteratively updated using Stochastic Gradient Descend (SGD) on mini batches in order to minimize the following loss function:

$$\mathcal{L}(\theta) = \mathbb{E}_{e(t)\sim\mathcal{B}}\left[\delta(t)^2\right]. \tag{3.2}$$

In Eqn. (3.2), $\delta(t) = y(t) - Q(s(t), a(t); \theta)$ denotes the Temporal Difference (TD) error where the γ-discounted target value is computed as follows:

$$y(t) = r(t) + \gamma \max_{a'} Q(s(t+1), a'; \theta). \tag{3.3}$$

Finally, knowing the optimal parameters θ^*, the optimal policy is given by:

$$\begin{aligned} \pi^* : \quad & \mathcal{S} \rightarrow \mathcal{A} \\ & s \rightarrow \arg\max_{a\in\mathcal{A}} Q(s, a; \theta^*). \end{aligned}$$

In general, there can be some states where the outcome is the same regardless of the action the agent could take; therefore, it is not always necessary to determine the state action value at a given state s, $Q(s, a; \theta)$, for every action. For instance, when playing a video game consisting in moving left or right to avoid objects, trying to decide whether the optimal action is to move left or right is totally useless if there is no threatening object in sight. Another example is when a UE is located at the same distance from two BSs that can provide it with the same throughput. In that case, there is not a single optimal action as the result will be the same whatever BS is selected Based on this intuition, Wang *et al.* have introduced the notion of dueling network where $Q(s, a; \theta)$ is decomposed into a state value $V(s; \theta) = \mathbb{E}[Q(s, a; \theta)]$ and the *advantage* of the corresponding action $A(s, a; \theta)$ [60]. That is,

$$Q(s, a; \theta) = V(s; \theta) + A(s, a; \theta). \tag{3.4}$$

The first term is action-less and is inherent to the state while the second measures the goodness of the action in that state. Dueling network shows that learning the DQN by estimating separately the state value and the advantage values can enable notable improvement in the agent policy.

3.2.4 Multi-agent reinforcement learning

In MARL, agents learn by interacting with a shared environment. In particular, usually in distributed MARL, each agent maintains its own policy, while sharing its environment with other agents. Typically, in this context, either each agent acts in a selfish way (concurrent MARL), learning a policy that optimizes its own performance, or aims to determine a global optimal policy, which maximizes the system performance (cooperative MARL). One major issue that arises with MARL is the problem of non-stationarity due to multiple agents interacting simultaneously with the environment. This is especially true in the case of *independent learners*, where agents see each other as part of the environment, which becomes non-stationary from an agent's point of view, as the actions of its teammates change over time. In addition, environment non-stationarity can lead to *shadowed equilibria*.

> **Definition 1** (From [61]). *An equilibrium is shadowed by another one if there exists one agent i which receives a very low gain by unilaterally deviating from this equilibrium and if this gain is lower than the minimal gain when deviating from the other equilibrium.*

In other words, in the presence of *shadowed equilibria*, an agent's locally optimal action could end up being globally sub-optimal [61]. Moreover, during the learning process, an RL agent may face two

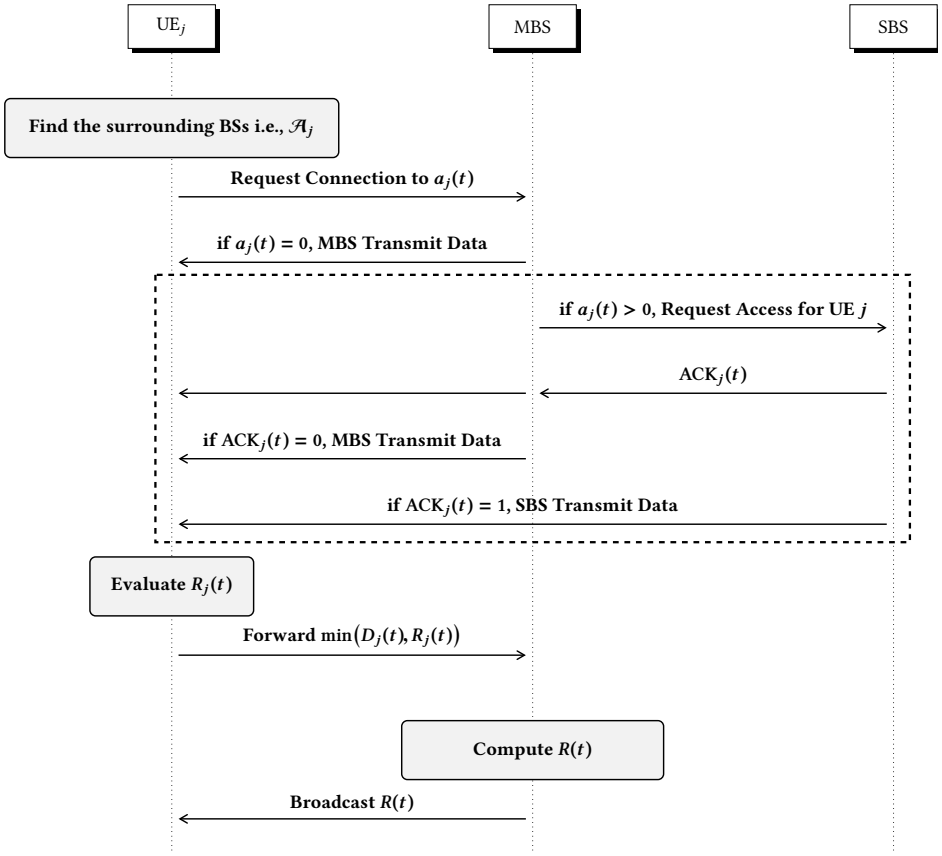


Figure 3.2: Message sequence chart of the proposed mechanism for user association.

conflicting interests: either it exploits an action knowing the (expected) return or reward based on the knowledge acquired so far or it explores new actions with uncertain outcomes but which can help it improve or consolidate its current knowledge. The trade-off between exploration and exploitation is crucial for learning efficient policies in RL, especially in MARL [17]. However, in MARL, the exploration of one agent induces noise on the other agents exploiting their policy. This noise may cause other agents to deviate from their current, albeit optimal, knowledge. Such a behavior is called alter-exploration and can be quantified using the notion of *global exploration* [61]. The global exploration measures the probability ψ that during learning process, at least one agent explores. It can be formulated using the individual exploration rate of each agent.

Lemma 2 (from [61]). *Let a K-agents system in which each agent explores according to a probability $\epsilon \in [0, 1]$. Then the probability that at least one agent explores is $\psi = 1 - (1 - \epsilon)^K$.*

In particular, note that as K increases, ψ converges to 1 ($\psi \longrightarrow 1$): alter-exploration impact becomes worst as the number of agents increases.

In the following, we focus on cooperative MARL, meaning that agents also share a common joint reward and propose a solution to deal with *shadowed equilibria* as well as *alter-exploration.*

3.3 Proposed Dynamic User Association

3.3.1 Proposed solution via multi-agent reinforcement Learning

In this section, we define the proposed MARL framework to solve the optimization problem (2.9)-(2.12) defined in previous chapter. Here, following enhanced Mobile Broadband (eMBB) service requirements,

we focus on network sum-rate maximization. Accordingly we set $\alpha = 0$ in Eqn. (2.7). Thus, the utility function $R(t)$ defined in Eqn. (2.7) corresponds to total network sum-rate:

$$R(t) = \sum_{i \in \mathcal{A}} \sum_{j \in \mathcal{U}} x_{i,j} \min \left(R_{i,j}(t), D_j(t) \right), \tag{3.5}$$

where $D_j(t)$ is the data-demand of UE j and $R_{i,j}(t)$ is its experienced *w.r.t.* BS i.

In our proposed framework, we model UEs as agents and assign them a common objective to maximize the network throughput. In our setting, a UE, based on its local observations, selects and requests service from a target BS, which accepts or rejects the connection request by sending an Acknowledgment (ACK) signal depending on the available resources.

As described in Figure 3.2, each UE j starts by identifying the set of BSs $\mathcal{A}_j$ it may connect to. Note that, in practical systems, the size of this set is limited to reduce complexity on mobile devices. Here, $\mathcal{A}_j$ also defines the UE action space, meaning that the action $a_j(t) \in \mathcal{A}_j$ denotes the index of the BS to which the UE j requests a connection at time t. Then, in every time step, each UE j takes an action $a_j(t)$ and informs the MBS of its choice. If the UE is requesting a connection from the MBS, *i.e.*, $a_j(t) = 0$, the request is automatically granted[1] and the communication is set up. Otherwise, the MBS forwards the connection request to the corresponding SBS. Depending on the overall received requests and the constraint (2.12), the SBS notifies both the UE and the MBS with an $\text{ACK}_j(t)$ signal. If $\text{ACK}_j(t) = 1$, the SBS grants a connection to the UE; otherwise, the MBS establishes the default data link with the UE j. Next, each UE j evaluates the perceived data rate, *i.e.*, $\min \left(D_j(t), R_{a_j(t),j}(t) \right)$ and forwards this value to the MBS. Then, the MBS computes the network sum-rate $R(t)$. Finally, the MBS broadcasts $R(t)$ to each UE, which uses it to evaluate the goodness of its policy $\pi_j(t)$ and to update it accordingly.

Following this process, we define the history $\mathcal{H}_j(t)$ of UE j as the set of all actions, observations, and measurements collected up to time t [62] :

$$\mathcal{H}_j(t) = \left\{ a_j(\tau),\ \text{ACK}_j(\tau),\ \text{RSS}_{a_j(t),j}(\tau),\ D_j(\tau),\ R_{a_j(t),j}(\tau),\ R(\tau) \right\}_{\tau=1}^{t}. \tag{3.6}$$

Hence, the policy of UE j at time t, $\pi_j(t)$, is a mapping from its history $\mathcal{H}_j(t-1)$ to a probability mass function over its action space $\mathcal{A}_j$. Therefore, each UE takes its actions following its own strategy without being aware of the actions taken by the other UEs.

A key feature of the proposed approach is that in contrast to MDPs, here, the decision of the j-th UE is based only on its *local* state observation:

$$\mathbf{o}_j(t) = \left\{ a_j(t-1), R_{a_j(t-1),j}(t-1), R(t-1), \text{ACK}_j(t-1), \text{RSS}_{a_j(t-1),j}(t), D_j(t) \right\}. \tag{3.7}$$

It is worth to note that $\mathbf{o}_j(t)$ carries information related to the previous action/reward, already available at the UE side, and new local information (the RSS and the data demand $D_j(t)$). Specifically, each UE makes association decisions based on how well its previous actions performed. The only observation that implicitly coordinates the actions of the multiple UEs is the network sum rate, which serves as a signal to each UE as to whether their local actions are beneficial to the overall network objective. Note that the overall network objective may increase or decrease due to the actions of multiple UEs, thus it is not a perfect signal in the sense that it does not tell each UE exactly the consequence of its own specific action. Yet, our goal is that, using DRL, each UE is able to learn over time its optimal policy.

It is noteworthy that the size of the state observation of a given UE does not scale with the number of UEs in contrast to other works in the literature, as [40]. This allows us to build general DQNs that can be used in different network scenarios; that is, if a UE leaves or joins the network, there is no need to change the DQN architecture. Moreover, $\mathbf{o}_j(t)$ is a partial observation of the true state $s(t)$, which includes all the observations of other agents. In the literature, the optimization in partially observable environments is addressed as a multi-agent POMDP [63]. Partial observability, in addition

[1]Note: we assume that the MBS is able to simultaneously serve all the active UEs by equally sharing its band across them.

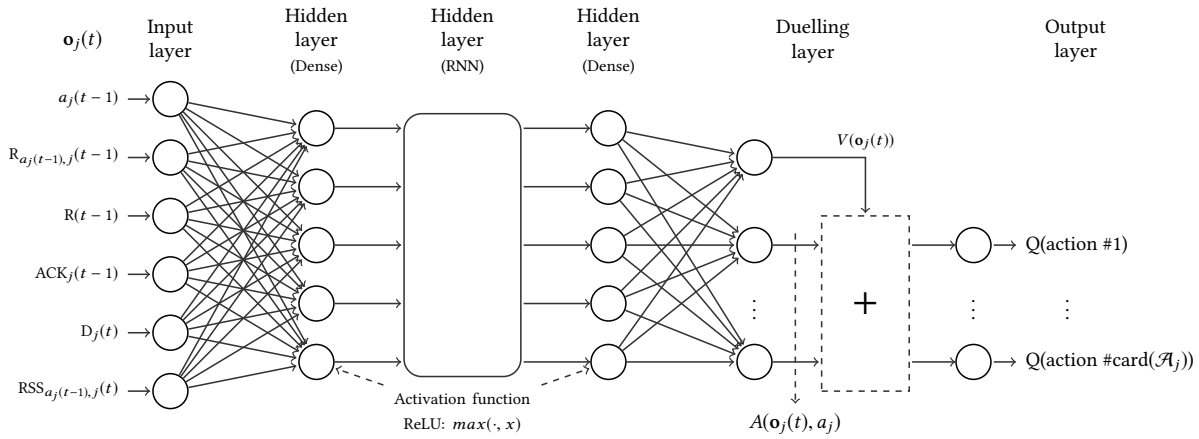


Figure 3.3: Illustration of the architecture of the proposed DRQN.

to non-stationarity issues, make MARL an even more complex task. To tackle this problem, Omidshafiei *et al.* successfully applied hysteretic Q-learning (first introduced by Matignon *et al.* [64]) with partial observability [63]. They empowered the DQNs with Reccurent Neural Network (RNN) to obtain deep recurrent Q-networks (DRQNs), which serves as a basis for our proposed algorithm.

3.3.2 Hysteretic deep recurrent Q-network

In the Hysteretic Deep Recurrent Q-Network (HDRQN) algorithm[2], each UE j acts as an *independent learner* and maintains its own DRQN $Q_j(\mathbf{o}_j(t), h_j(t-1), a_j(t); \theta_j)$. Figure 3.3 describes the proposed DRQN, which is composed of one input layer, two fully connected hidden layers, one RNN hidden layer, a dueling layer, and an output layer. The UE's local state information $\mathbf{o}_j(t)$ and the estimated state action value $Q_j(\cdot;\cdot)$ define respectively the input layer and the output layer of the DRQN (Section 3.3.4 provides more details on the proposed DQRN). We use $h_j(t-1)$ to represent the internal state of the RNN hidden layer and θ_j to define the UE's local DRQN weights. The use of RNNs allows to aggregate past information (previous observed states, *i.e.*, the history $\mathcal{H}_j(t)$) in the agent decision-making process, which is shown to improve the average reward perceived when dealing with partial observability [65]. Indeed, in a partially observable environment, each agent makes its decision relying on the observation $\mathbf{o}_j(t)$ instead of the true state $s_j(t)$, which is unknown. From $\mathbf{o}_j(t)$ solely, the agent may have a partial perspective of the environment. In this case, the commonly used Vanilla DQN may not be effective [65], specifically in multi-agent scenarios, where each agent is unaware of the behavior of its teammate. Hence, we extend the baseline Vanilla DQN with RNN to infer the underlying state $s_j(t)$ from agent past observations, *i.e.*, its history $\mathcal{H}_j(t)$ [65].

The experience of the j-th UE $e_j(t) = \left\{\mathbf{o}_j(t), a_j(t), r_j(t), \mathbf{o}_j(t+1)\right\}$ is stored into a local memory buffer $\mathcal{M}_j$. In order to further stabilize the learning process, synchronized sampling strategy (called *concurrent experience replay trajectories* (CERTs)) is adopted [63]. In other words, during the training, mini batches of experiences of the same time steps are sampled across agents to update the local DRQN weights in order to minimize the hysteretic loss function:

$$\mathcal{L}_j(\theta_j) = \mathbb{E}_{e_j^b(t)\sim\mathcal{B}_j}\left[\left(w_j^b \delta_j^b(t)\right)^2\right], \tag{3.8}$$

where b indexes an entry in the mini batch of experiences $\mathcal{B}_j$, $\delta_j^b(t) = y_j^b(t) - Q_j(\mathbf{o}_j^b(t), h_j^b(t-1), a_j^b(t); \theta_j)$ is the TD error with respect to the target value

$$y_j^b(t) = r_j^b(t) + \gamma \max_{a'} Q_j(\mathbf{o}_j^b(t+1), h_j^b(t), a'; \hat{\theta}_j). \tag{3.9}$$

[2]In the following, we use HDRQN to refer to the proposed algorithm, and DRQN to refer to the related NN architecture.

Here, $\hat{\theta}_j$ represents the weights of the target DRQN, which are updated less frequently to improve the learning stability [18].

In MARL, the agents' reward is the result of their joint actions. Accordingly, an agent experience $e_j(t)$ is *positive*, if the associated TD error $\delta_j^b(t)$ in Eqn. (3.8) is positive, *i.e.*, the perceived global reward is better than the previous rewards independently of the optimality of the agent local action. A positive experience does not necessarily imply that the agent's strategy is converging toward the optimal solution, but the network performance is improving over time. In contrast, a *negative* experience results in an agent receiving a lower reward after taking an action that was fruitful in the past. A negative experience can be caused by the agent's action being non-optimal or more likely by the others agents' behavior. That is, an agent that has taken a local optimal action may receive a lower reward because of the bad choices of other agents. Such events may be exacerbated by the increase in the number of agents. Therefore, negative experiences can be very detrimental in MARL as they may mislead the agent to change its optimal strategy. Consequently, an agent may stabilize its strategy by paying less attention to negative experiences.

This is the idea introduced by hysteretic Q-learning: the neural network weights are updated via SGD with two distinct learning rates $\alpha\mu$ and $\beta\mu$ ($\beta \ll \alpha \leq 1$), where μ is a based learning rate and α and β are control factors. When the TD error is positive, the learning rate $\alpha\mu$ is used; otherwise, $\beta\mu$ is considered. This leads to optimistic updates that give more importance to positive experiences [64]. To implement the hysteretic learning in conventional machine learning libraries, we set μ as the fixed learning rate and scale the TD error $\delta_j^b(t)$ in Eqn. (3.8) as follow:

$$w_j^b = \begin{cases} \alpha, & \text{if } \delta_j^b(t) \geq 0 \\ \beta, & \text{otherwise.} \end{cases} \tag{3.10}$$

3.3.3 Definition of the reward function

The maximum value of the network sum-rate, and hence, the optimal user association is unknown to the agents at the beginning of the learning phase. In other words, there is no explicit or predefined terminal state that agents are aware of and toward which they have to converge to. Accordingly, we treat this learning problem as a continuing task over a time horizon T_e. That is, the agents keep updating their policies as long as it improves the perceived reward.

> **Definition 2.** *We define the beam collision as the event corresponding to a given SBS i receiving more requests than the number of beams N_i it can set up i.e., there is a beam collision if $\sum_{j \in \mathcal{U}_i} x_{i,j} > N_i$.*

Requests collision may occur since all UEs are requesting connections simultaneously. However, our proposed framework aims to effectively train agents to distribute the network load and to properly leverage the advantages of network densification. Consequently, when a collision happens during the training phase, we punish all UEs by setting the instantaneous reward to zero, which discourages agents from colliding. However, during execution time, a practice implementation of this framework may choose between the colliding UEs, which UEs to serve. This selection can be made either randomly or based on RSS. As a result, we define the reward function of UE j in Eqn. (3.9) as:

$$r_j(t) = \begin{cases} R(t), & \text{if there is no collision} \\ 0, & \text{otherwise.} \end{cases} \tag{3.11}$$

During the learning, each UE j builds its policy π_j depending on its data rate requirement, the experienced SINR, the network sum-rate, and whether its requests cause a collision to maximize the accumulated discounted reward:

$$G_j(t) = \sum_{\tau=t+1}^{T_e} \gamma^{\tau-t-1} r_j(\tau), \tag{3.12}$$

where the discounting factor γ is such that $0 \leq \gamma < 1$. Taking $\gamma = 0$ leads to myopic (instantaneous) network throughput maximization. In the case of dynamic scenarios, it is better to consider $\gamma \neq 0$ to take into account the dynamic nature of the environment: there is no need to change the current user association at time step t due to a low reward perceived because of the environment dynamics if at the next time step the system will recover its *equilibrium*. This consideration also makes sense in a practical system where changing the association too often can also induce excessive overhead.

As defined, the reward perceived by the agents continuously varies with the environment stochasticity viz. fading, shadowing, interference, traffic, and noise. Accordingly, this reward setting can lead to many optimal or quasi-optimal *equilibria*, which is a major issue as it results in agents laboriously trying to converge [61]. Algorithm 1 presents the proposed training procedure to deal with these challenges. Note that parts of this algorithm (highlighted in gray) can be executed in parallel across all UEs.

Algorithm 1: User Association: Training Procedure

while $t < T_e$ **do**
  for $j \in \mathcal{U}$ **do**
    Observe state $\mathbf{o}_j(t)$.
    $a_j(t) \leftarrow \arg\max_{a' \in \mathcal{A}_j} Q_j(\mathbf{o}_j(t), h_j(t-1), a'; \theta_j)$ following the ϵ-greedy policy.
    if $a_j(t) \neq 0$ *and connection granted* **then**
      $\text{ACK}_j(t) \leftarrow 1$. // the UE is requesting a connection to a SBS.
    else
      $\text{ACK}_j(t) \leftarrow 0$.
      Automatically redirect to the MBS.
    end
    Measure $R_{a_j(t),j}(t)$.
  end
  $R(t) \leftarrow 0$.
  for $i \in \mathcal{A}$ **do**
    if $\sum_j \mathbb{1}_{a_j(t)=i} > N_i$ **then**
      $R(t) = 0$. // collision.
      Break.
    else
      $R(t) = R(t) + \sum_{j \in \mathcal{U}_i} 1_{a_j(t)=i} \min(R_{i,j}(t), D_j(t))$.
    end
  end
  for $j \in \mathcal{U}$ **do**
    Observe the new state $\mathbf{o}_j(t+1)$.
    Store experience $e_j(t)$ into $\mathcal{M}_j$.
    Samples a batch of experiences from $\mathcal{M}_j$.
    Compute the target value $y_j^b(t)$ in Eqn. (3.9).
    Performs a gradient descent step on $\delta_j^b(t)$ in Eqn. (3.8) with respect to θ_j.
    Periodically reset $\hat{\theta}_j \leftarrow \theta_j$.
  end
  $t = t + 1$.
end

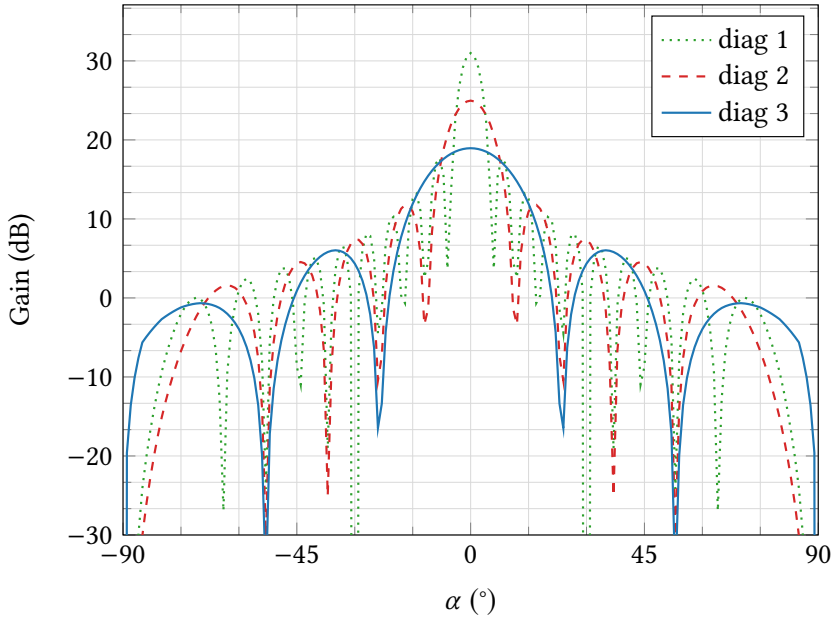


Figure 3.4: Simulated TX/RX antenna gain radiation pattern for an array of 20×20 (`diag 1`), 10×10 (`diag 2`), 5×5 (`diag 3`) elements operating at 28 GHz [1].

3.3.4 Numerical results

In this section, we demonstrate the effectiveness of the proposed HDRQN-based user association by comparing its performance with the exhaustive search algorithm obtained via brute force and two other centralized benchmarks of the literature:

- **Max-SNR:** Each UE is associated with the BS, which provides the maximum SNR taking into account the constraint on the number of beams per BS (see Eqn. (2.12)). Since this method does not consider interference, it has limited performance, especially in dense networks.
- **Heuristic:** Proposed in [39], this algorithm starts by ordering all the possible associations according to their respective SNR values, which do not consider interference. Then, following this order, the algorithm goes from one potential association to the following one and validates it if it increases the network sum-rate $R(t)$ in Eqn. (3.5). Although the evaluation of $R(t)$ takes the interference and UEs traffic into account, the performance of this algorithm depends mainly on the SNRs ordering, which may prevent reaching a global optimum. This approach is recalled in Algorithm 2 with minor modifications compared to the original one since power and beamwidth constraints are not considered in this study.

In the following, we start by analyzing the complexity of the proposed method compared to the two baselines. Then, we study the effect of the hysteretic parameter on both convergence speed and achievable sum-rate. Also, we evaluate the effectiveness of collision cost in limiting collision events and improving network sum-rate. We continue assessing the performance of our scheme in both static and dynamic scenarios. Finally, we conclude the evaluation by demonstrating the adaptive property of the proposed algorithm.

We consider that UEs and SBSs communicate in the mmWave band at a carrier frequency of 28 GHz using the same phased array antenna. To evaluate three different interference scenarios, we consider distinct antenna gain radiation patterns (see Figure 3.4), which correspond to a distinct number of antenna elements in the phased array. Larger the array, thinner the beam[3]. In all tests, three small cells

[3]Note that increasing the number of antenna elements also increases the antenna's size, thus, the hardware complexity.

Algorithm 2: Heuristic scheme: Centralized User Association

Set $x_{i,j} = 0,\ \forall j \in \mathcal{U}, i \in \mathcal{A}_j$.
Get the $\text{SNR}_{i,j}$ and sort it in descending order into $\mathcal{Z} = \{z_1, z_2, \ldots, z_P\}$ with $P = \sum_j \text{card}(\mathcal{A}_j)$.
Let δ be the transformation (defined by the sort) such that $\delta(i,j) = p$: $z_p = z_{\delta(i,j)} \leftarrow \text{SNR}_{i,j}$.
Set $R^1(t) = 0$.
while $p \leq P$ **do**
    Set $x_p = 1$.
    Compute $R^p(t)$. `// R^p(t) is the sum-rate at iteration p.`
    if $R^p(t) > R^{p-1}(t)$ *and a beam is available* **then**
        Let x_p unchanged. `// means that activating this link improves the sum-rate.`
    else
        Reset $x_p = 0$.
    end
end
Apply δ^{-1} to recover which links (i,j) are active.

Table 3.1: Simulations parameters.

	Macro cell [66]	Small cell [67]
Parameters	Values	
Carrier frequency, f_s	2.0 GHz	28 GHz
Bandwidth, B	10 MHz	500 MHz
Thermal noise, N_0	-174 dBm/Hz	-174 dBm/Hz
Noise figure	5 dB	0 dB
Shadowing variance, σ_s^2	9 dB	12 dB
TX power, P^{Tx}	46 dBm	20 dBm
Antenna gain, $G^{\text{Tx}}/G^{\text{Rx}}$	17 dBi / 0 dBi	Fig.5
Radius, r		35 m
Back-lobe gain		-20 dBi
Path-loss coefficient, η_s	3.76	2.5
Inter-cell distance		$1.2 \times r$
Reference distance, $d_{0,s}$	20.7 m$^{(1)}$	5 m
Beam number, N_i		$N_1 = 2$; $N_2 = N_3 = 3$

(1) We use as a path loss model, $G^{\text{Ch}}_{i,j}(\text{dB}) = 128.1 + 37.6\log_{10}(d_{i,j})$, $d_{i,j}$ in Km from Table A.2.1.1.2-3 in [68]. Then, we compute the equivalent reference distance in meter for equation (3).

are deployed inside the macro cell. UE and small cell locations follow the 3GPP recommendations [66]. Table 3.1 summarizes the network parameters.

To learn the user association policy, we use the DRQN described in Figure 3.3. This architecture comprises 2 Multi Layer Perceptron (MLP) of 32 hidden units, one RNN layer (a Long Short-Term Memory (LSTM) layer[4]) with 64 memory cells followed by another 2 MLPs of 32 hidden units. The network then branches off in two MLPs of 16 hidden units to construct the dueling network. All layers use a rectifier linear unit (ReLU) except the final layer, which has a linear activation function. For the hysteretic learning, we set the base learning rate $\mu = 0.001$ and $\alpha = 1$, and then we optimize $\beta \in [0, 1]$ to

[4]Note that it is also possible to use a Gated Recurrent Unit (GRU) layer. During simulations, both LSTM and GRU layer have shown similar performance.

Table 3.2: Deep Recurrent Q-networks training parameters

Discount factor, γ	0.9
Time horizon, T_e	7000
Batch size, $\lvert\mathcal{B}\rvert$	32
CERTs memory size, $\lvert\mathcal{M}\rvert$	500
ϵ (follows a negative Gompertz function[(1)])	$1 \longrightarrow 0.1$
Target network update frequency	10
Number of Monte-Carlo simulations, N	400

[(1)] $\epsilon(t) = 1 - ae^{-e^{-b(t-c)}}$, with $a = 0.9$, $b = 10^{-3}$, $c = 800$.

strike a balance between convergence speed and network sum-rate. The DRQNs are trained offline using an ϵ-greedy policy. The hyper-parameters values summarized in Table 3.2 are selected via informal search. Finally, unless specified, all results are average over N runs of Monte-Carlo simulations. At each run, UE positions are randomly reset.

We evaluate the performance of the proposed solution and the related baselines using either the network sum-rate or the sum-rate ratio *w.r.t.* the brute force approach. Specifically, for these metrics, we compute the average and the standard deviation as follows:

$$\overline{R} = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{T_e}\sum_{t=1}^{T_e} R^{(n)}(t), \tag{3.13}$$

$$\sigma_R^2 = \frac{1}{N}\sum_{n=1}^{N}\left(\frac{1}{T_e}\sum_{t=1}^{T_e} R^{(n)}(t) - \overline{R}\right)^2, \tag{3.14}$$

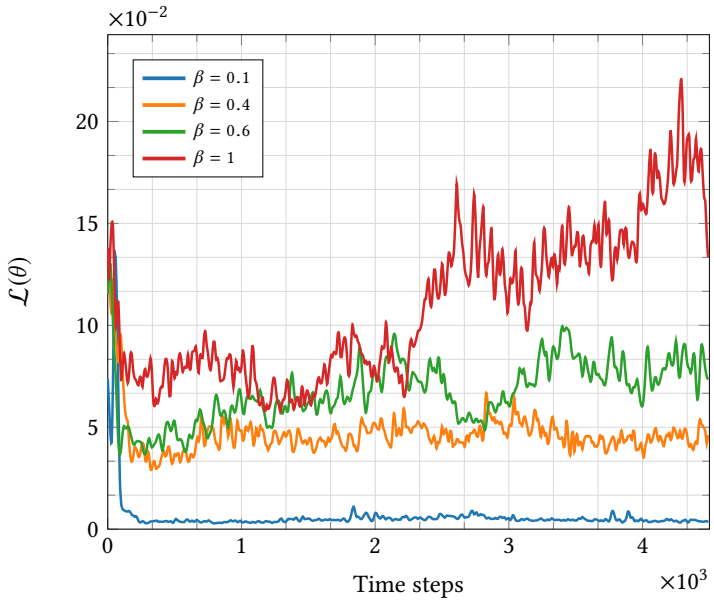
where $R^{(n)}(t)$ is either the sum-rate or the sum-rate ratio at the time step t of run n.

Complexity analysis. We analyze both the computational and signaling complexity of the proposed algorithm and compare it to the two baselines. Since our framework is based on deep Q-learning, a practical implementation completely conducts the learning offline as with the Vanilla DQN initially proposed for Atari games [18], and then, it transfers to each UE the corresponding weights. In this scenario, UEs simply conduct the inference on their local states to find the optimal action, alleviating the computational and power burdens. That is to say, the computational complexity of the proposed framework during its execution is limited to the inference complexity of each local DQRN. Let L_h be the size of hidden layers and L_c the number of cells in the LSTM layer. Each DQRN has six inputs[5], thus the complexity is in the order of $O\left(6L_h + 2L_h^2 + L_hL_c + 2L_h^2 + L_h(\text{card}(\mathcal{A}_j))\right) \approx O\left(6L_h^2 + L_hL_c\right)$. This is very straightforward compare to a naive algorithm, which may find the optimal solution of problem (2.9)-(2.12) through an exhaustive search, which has a complexity $O\left(N_sK(1+N_s)^K\right)$ as shown in Eqn. (2.3.4) of Chapter 2.

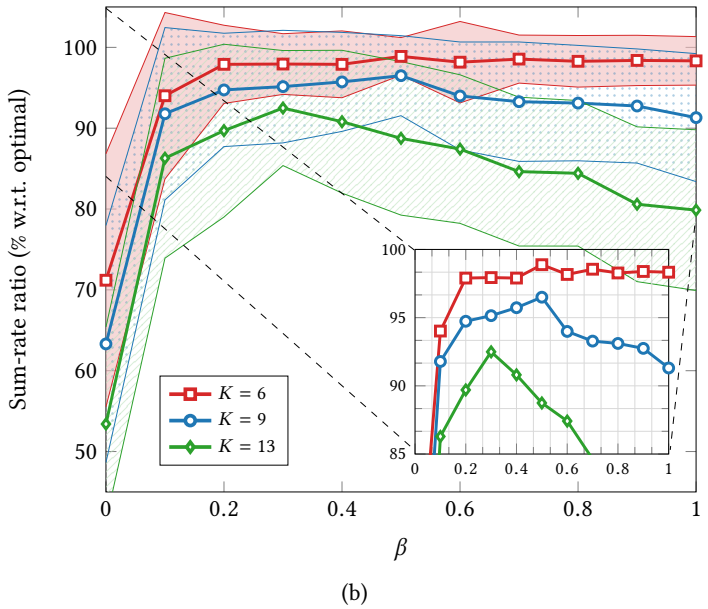
The complexity of both max-SNR and heuristic algorithms during execution is related to sorting the SNR values. Considering a *quicksort* algorithm, this complexity in the worst case ($\text{card}(\mathcal{A}_j) = N_s + 1$) is around $O(n\log(n))$ for max-SNR and $O(n + n\log(n))$ for the heuristic algorithm[6] where $n = K(1+N_s)$. However, the need to collect the SNR values globally is the most notable disadvantage of these centralized approaches. In terms of signaling overhead, compared to the existing standard (*e.g.* 5G), the additional complexity introduced by our framework is due to the broadcasting of the total network sum-rate. The rest of the information used by a UE to take a decision is already either measured by the UEs $\left(R_j(t), \text{RSS}_j, a_j(t)\right)$ or sent by its serving BS $\left(\text{ACK}_j\right)$. Specifically, the number of messages exchanged in

[5] For practical implementation, we encode the entry $a_j(t)$ in Figure 3.3 as a one-hot vector leading to $5 + \text{card}(\mathcal{A}_j)$ inputs.
[6] One pass to sort the SNR values and another to find the association.

the sequence chart of Figure 3.2 is a function of the UE's action $a_j(t)$. If $a_j(t) = 0$, the association is set up in two messages with the MBS. Otherwise, four messages are required to connect to either a SBS or a MBS, depending on the ACK signal. Overall, for each UE to connect to the serving BS, the system needs to exchange at most four messages. Then, two additional messages are required to get the total network sum-rate from the MBS. Therefore, at most six messages are needed to complete one training step.



(a)



(b)

Figure 3.5: Convergence speed and effect of the hysteretic parameter β (using diagram 1). Figure (a) shows loss function for different values of β and for $K = 9$. For the sake of readability, a 20-sized moving average window is applied on plotted data. Figure (b) shows the sum-rate ratio and the associated variance between the proposed scheme and the optimal UE association for different values of β.

Convergence and effect of hysteretic parameter β. Here, we study the impact of the hysteretic parameter β on the performance of the proposed solution in terms of network sum-rate and convergence speed. Specifically, Figure 3.5a shows the evolution of the loss function during the training process

for different values of β, and Figure 3.5b describes the sum-rate ratio of the proposed scheme *w.r.t.* the optimal solution as a function of β.

First, Figures 3.5a and 3.5b show that despite the few pieces of information available locally to each agent, they can successfully learn a user association policy that performs close to the optimal strategy in less than $5 \cdot 10^3$ iterations/associations if $\beta \leq 0.6$. In addition, Figure 3.5a shows that lowering β increases the convergence speed of the algorithm. However, this also results in limited sum-rate performance. For instance, when $\beta = 0$, the proposed scheme achieves only 70% of the optimal performance (see Figure 3.5b). This is because, from Eqn. (3.10), we know that selecting very low values of β makes the agents too optimistic *i.e.*, they tend to neglect actions that produce negative TD errors. This leads agents to potentially select sub-optimal actions. In contrast, when $\beta = 1$, the agents give equal importance to positive and negative TD errors, *i.e.*, they become pessimists. In this setting, a UE may change its (optimal) strategy after taking an action that results in a negative error, although this error is simply the result of the other agents' behaviors. These continuous changes limit the learning performance, and, in fact, Figure 3.5a shows that the loss function diverges for $\beta = 1$. Hence, there is a trade-off between convergence speed, successful coordination of the agents, and network sum-rate.

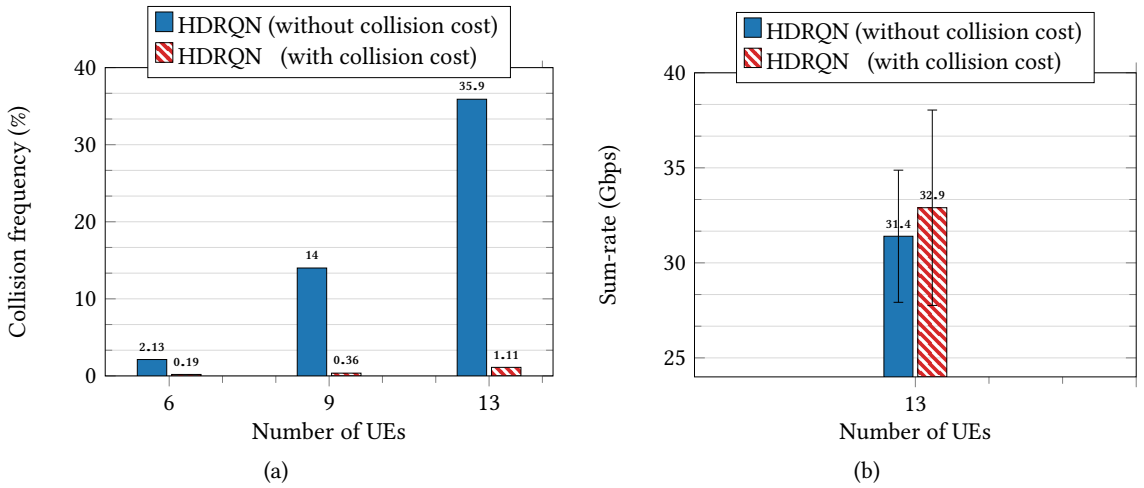
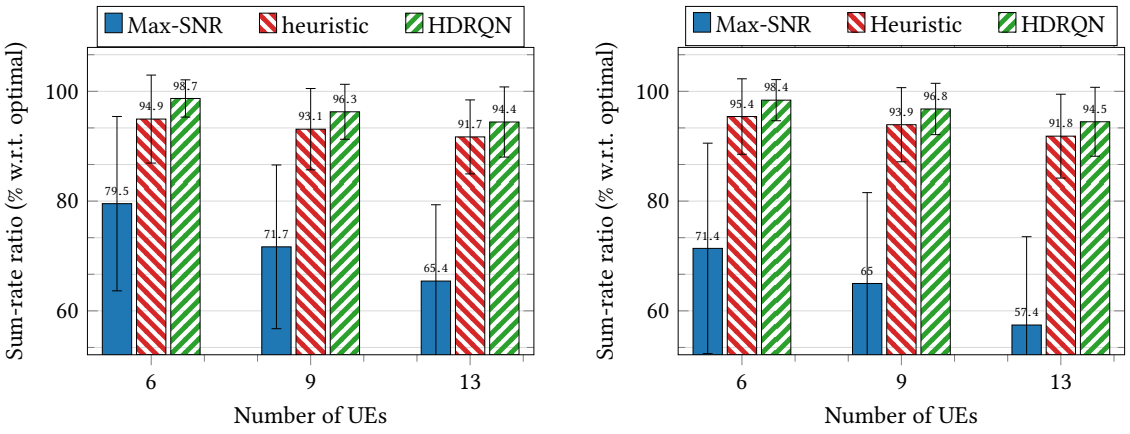


Figure 3.6: Impact of the collision cost on network performance in static scenario (using `diag 1`).

Impact of the collision cost on network performance. Here we assess the effectiveness of the collision cost in Eqn. (3.11), to limit the collision events. For this purpose, we consider a setting in which there is no collision cost. In this case, during the training phase, if a SBS receives more requests than the ones that it can accept, it randomly chooses the serving UEs among the received requests; the remaining UEs are therefore associated with the MBS. Figure 3.6a shows the frequency of the collision event during the test phase. We can observe that the collision frequency increases with the number of UEs as the cell load increases. However, we can see that by introducing the collision penalty, we significantly reduce the collision events up to 97%, which leads to an improvement of the overall network throughput[7] by 4.7% (see Figure 3.6b). This demonstrates that, with the proposed solution, UEs learn to distribute their association requests among the different BSs, balancing the cell load and maximizing the network sum-rate.

Performance of the proposed algorithm in static scenario. We now compare the performance of the proposed user association solution with the one achieved by the two baselines in a static scenario where there is no fading (*i.e.*, $\alpha_{i,j} = 1$) and with full buffer traffic. Consequently, in Eqn. (3.5), we

[7]We have considered the case of $K = 13$ UEs to highlight how, in networks with a large number of users, the collision events impact the network sum-rate.



(a) Case 1: Using diagram 1

(b) Case 2: Using diagram 3

Figure 3.7: Performance comparison in static scenario using diagrams 1 and 3.

set $D_j = +\infty, \forall j$ and disable the corresponding input in the DRQNs[8]. Figures 3.7a and 3.7b show the performance of the different approaches compared to the optimal user association in terms of network sum-rate, using antenna diagrams `diag 1` and `diag 3` respectively. We first note that the sum-rate ratio performance of our solution, as well as the heuristic approach, barely changes between the two antenna diagrams (less than 0.5% change), in contrast to the max-SNR algorithms, which does not consider interference. Specifically, when $K = 13$ the performance of the max-SNR decreases by 12.2% when switching from diagram `diag 1` to `diag 3`, which has lower directivity and thus results in a lower SINR. In addition, we note that, on average, our proposed scheme achieves up to 98.7% of the optimal sum-rate, hence outperforming both the max-SNR and the heuristic approaches. For example, when $K = 6$, by using `diag 3`, the proposed solution exhibits a performance gain of 3.1% and 37.8% over the heuristic and the max-SNR algorithm, respectively.

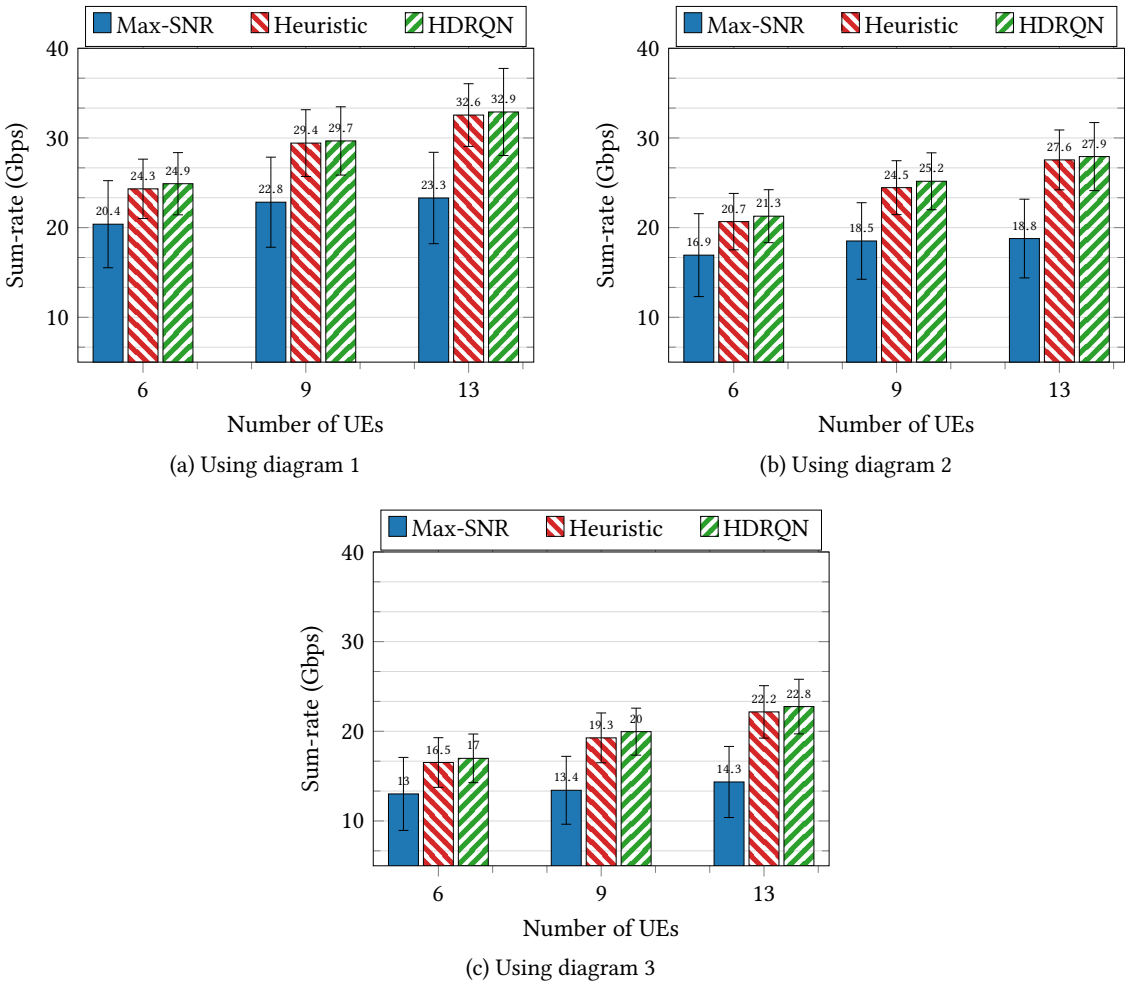
As soon as the number of UEs increases, the performance of our scheme slightly decreases. This is because ensuring coordination becomes more complex when the number of interacting agents increases. For instance, with `diag 3`, our solution only achieves 94.5% of the optimal performance for $K = 13$. However, it still outperforms the two baselines showing now a gain of 3% and 64.6% over the heuristic and the max-SNR approaches, respectively. Although the gain of the proposed solution over the heuristic scheme is small, our framework is distributed while the heuristic approach is centralized.

Performance in dynamic scenarios. We now evaluate the performance of the proposed scheme in dynamic environments and considering the three different antenna diagrams in Figure 3.4. For this purpose, we define two cases: 1) dynamic channels with small scale fading and full buffer traffic, 2) dynamic channels with small scale fading and dynamic traffic. As the optimal user association obtained via exhaustive search requires extensive computation, in the following, unless otherwise stated, we compare only the performance of the proposed scheme with the aforementioned two baselines. To achieve a fair comparison, every time step, we recompute the association solution of the two baselines as this may change due to the environment dynamics.

First, we can highlight from Figures 3.8 and 3.9 that, as expected, the network sum-rate decreases as the antenna diagrams become less and less directive (from `diag 1` to `diag 3`). Also, the gap between our scheme and the two baselines decreases when the antennas are more directive, which is due to the smaller interference perceived at the UE side. Indeed, the two baselines perform better in limited interference scenarios:

- **Case 1**: in this scenario, we have full buffer traffic, $D_j = +\infty, \forall j$ and dynamic channels with Nakagami

[8]To disable an input, we simply set the corresponding entry in $o(t)$ to zero.



(a) Using diagram 1

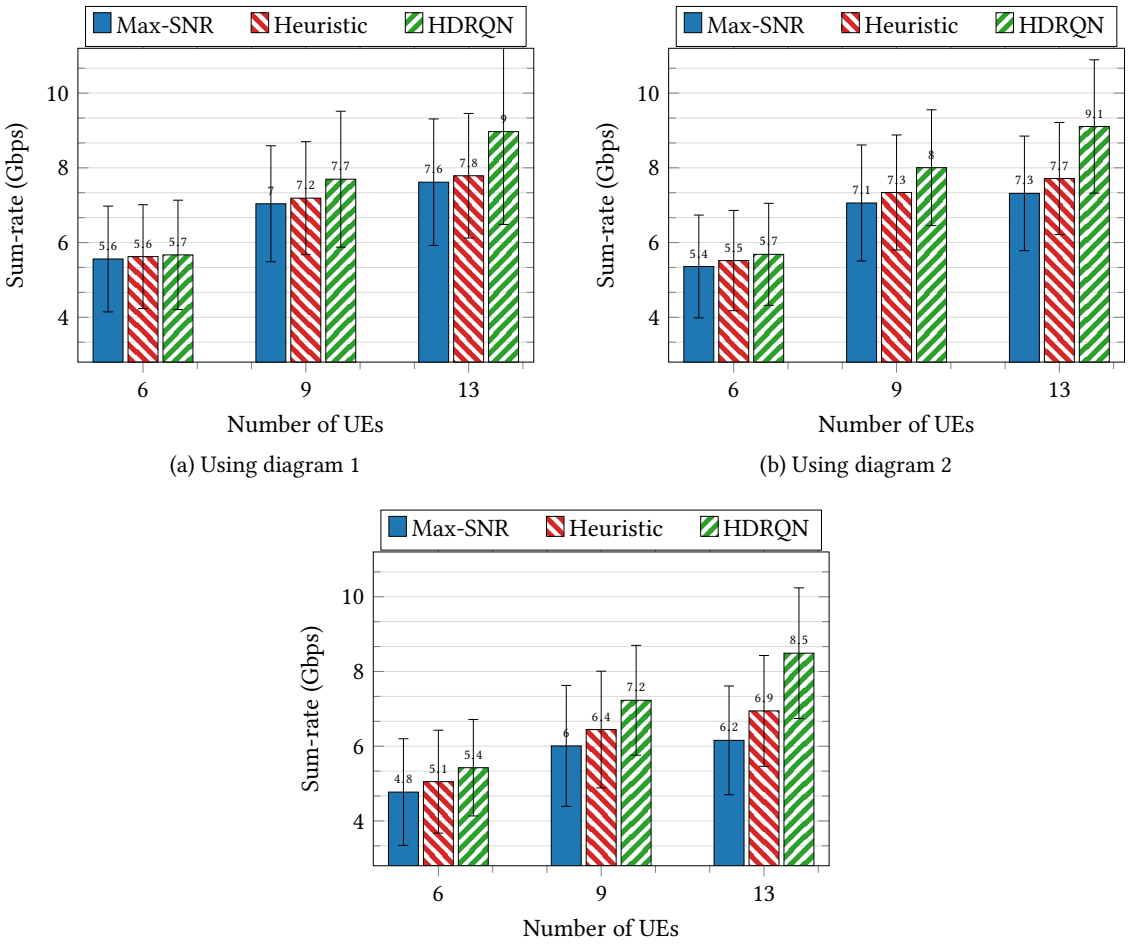
(b) Using diagram 2

(c) Using diagram 3

Figure 3.8: Performance comparison when considering only dynamic channels with fast fading.

small scale fading, characterized by a scale factor $m = 3$ [69]. Figure 3.8 plots the sum-rate achieved by the different algorithms for a different number of UEs. We remark that our distributed solution performs better than the two centralized baselines. Specifically, when the number of UEs is equal to 9, the HDRQN improves the network sum-rate by about 1% and 30.3% when using `diag 1`, 2.8% and 36.2% when using `diag 2`, and 3.6% and 49.2% when using `diag 3`, compared respectively to the heuristic and max-SNR schemes. As in the static case, the gain *w.r.t.* the heuristic is limited when considering only the fast fading effect.

- **Case 2**: we evaluate on Figure 3.9, the performance of our framework considering both fast fading and UE traffic. Here, for each UE, the intensity of its traffic Poisson distribution is uniformly chosen between $[0, 2]$ Gbps at the beginning of each Monte Carlo run. Overall, as expected, the effect of the traffic variations on the rate (see Eqn. (3.5)) is larger than the one related to the fast fading, which leads to small variations on the user-perceived SINR (see Eqn. (3.5)). In addition, our algorithm yields a large performance gain over the two benchmarks. For instance, for $K = 13$ UEs, the proposed solution improves the sum-rate by 19.4% and 18% when using `diag 1`, by 19.7% and 28.2% when using `diag 2`, and by 23.2% and 37.1% with `diag 3`, compared to heuristic algorithm and the max-SNR algorithms, respectively.



(a) Using diagram 1

(b) Using diagram 2

(c) Using diagram 3

Figure 3.9: Performance comparison when considering both dynamic channels with fast fading and dynamic traffic.

3.3.5 Concluding remarks

We have presented a novel and distributed approach for solving user association problems based on Multi Agent Reinforcement Learning (MARL). With the proposed MARL algorithm, agent decisions are based on partial and local observations, which limits the signaling overhead and reduces the computational complexity *w.r.t.* centralized approaches. Our analysis shows that, in the case of full buffer traffic, the proposed scheme achieves up to 98.7% of the optimal performance obtained through exhaustive search. When considering dynamic fading, the proposed solution outperforms centralized baselines, which require to continuously recompute the user association, leading to excessive complexity. In addition, the proposed approach results in large sum-rate gains when we consider dynamic traffic, achieving nearly 40% of performance gain *w.r.t.* baseline solutions from the literature. In the next section, we explore how the proposed solution can be leveraged to solve another challenging problem related to user association, namely *handover management*.

3.4 Application to Distributed Handover Management

A close problem to user association is handover or handoff management also known as user re-association. In dynamic environments characterized by mobile users, a UE to maintain or improve its QoS may need to change its current BS association when moving through the network. This process is called Handover (HO). Performing an HO procedure requires signaling between the UE, the serving BS, and the target BS, which induces overhead and energy consumption, thus decreasing the network performance. In 5G network with mmWave communications, the frequency of handover procedures is even accentuated due to severe pathloss, blockage, and deafness. This leads to a deterioration of mobile users' throughput as well as their battery lifetime. In the literature, the HO management problem has received wide attention, and multiple HO algorithms exist, each trying to limit the impact of frequent HOs in UEs Quality of Experience (QoE). In general, HO decisions are based on measurement signals such as RSS, Reference Signal Received Power (RSRP), Reference Signal Received Quality (RSRQ), or Word Error Indicator (WEI) [70]. 3GPP standard suggests that a UE triggers an HO process when the RSS of the target BS exceeds the one of the serving BS by a certain amount to avoid ping pong effect [70]. This procedure may induce large signaling overhead, which prevents from meeting the latency requirements of future wireless communication services [71]. To improve the HO performance, Yan *et al.* have proposed to limit the time consumed in the HO process by designing a machine learning algorithm that predicts HO decisions [72]. Koda *et al.* have proposed to limit HO frequency by designing a reinforcement learning (RL) framework that uses a Q-learning algorithm to maximize the network throughput [73]. In the same vein, Wang *et al.* have extended this approach using deep RL with actor-critic methods to avoid state discretization and for better scalability [74]. Not all these works consider cell load and limited resource availability when optimizing the HO strategy.

3.4.1 Handover management: system model and problem formulation

We recall the system model from Section 2.3 of Chapter 2. We do not consider UE traffic request, *i.e.* $D_j(t) = +\infty,\ \forall j$, and focus only on UE mobility. Therefore, given a BS i, the set of UEs in its coverage area $\mathcal{U}_i(t)$ changes over time as well as the set $\mathcal{A}_j(t)$ of BSs a UE j could associate with.

As UEs move around the network, they may be subject to multiple handovers to maintain or improve their QoE (see Figure 3.10). However, unnecessary HOs lead to large signaling overhead, which increases the energy consumption, lowers the spectral efficiency, and affects UEs latency. To account for this, we directly introduce a penalty due to the handover in the evaluation of the network performance. Indeed, let $\Delta\tau$ be the time between two possible handovers, also known as Time-to-Trigger (TTT) interval [70]. That is, a handover process can be triggered every time $\tau_p = \tau_0 + p\Delta\tau$, where τ_0 is an initial system delay. If UE j want to perform a handover at time τ_p, then, a time $\beta\Delta\tau$ is dedicated to the handoff procedure while the time $(1-\beta)\Delta\tau$ is used to communicate data (see Figure 3.11). The coefficient $\beta \in [0,1]$ allows controlling the cost of an HO process, which depends on the type of implemented handover (soft or hard handover) [75]. Accordingly, the effective data received by UE j from BS i between time τ_p and τ_{p+1} is

$$\overline{R}_{i,j}(\tau_p, \beta) = \int_{\tau_p}^{\tau_p + (1-\beta\lambda_j(\tau_p))\Delta\tau} R_{i,j}(t)dt, \tag{3.15}$$

where $\lambda_j(\tau_p) = 1$ indicates that UE j has handed over at time τ_p, and $\lambda_j(\tau_p) = 0$ otherwise. Hence, we define the network throughput $R(\tau_p)$ measured between time τ_p and τ_{p+1} as follows:

$$R(\tau_p, \beta) = \frac{1}{\Delta\tau} \sum_{i\in\mathcal{A}} \sum_{j\in\mathcal{U}} \overline{R}_{i,j}(\tau_p, \beta). \tag{3.16}$$

Let $\lfloor . \rfloor$ be the floor operator and $P = \left\lfloor \frac{T}{\Delta\tau} \right\rfloor$ be the number of TTTs over a time period T.

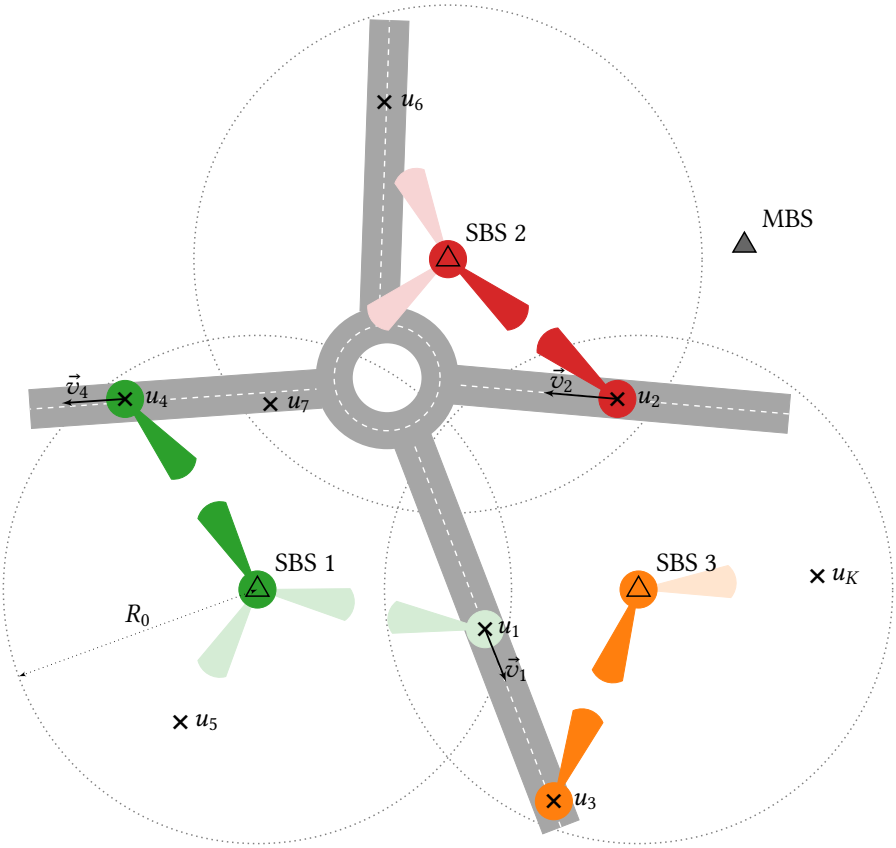


Figure 3.10: A downlink network with $N_s = 3$ SBSs, one MBS, and K UEs taking straight motion.

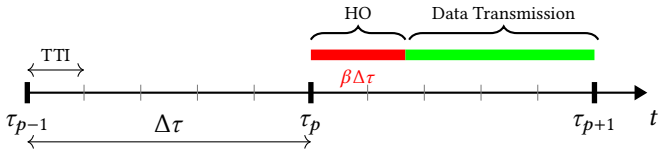


Figure 3.11: HO process timeline. TTI is the Transmission Time Interval.

Handover problem. We aim to find the HO strategy that maximizes the average network throughput $R_T(\beta) = \frac{1}{T}\sum_{p=1}^{P} R(\tau_p, \beta)$ taking into account the cost associated to handoff events. Hence, we formalize this problem as follows:

$$\underset{\{x_{i,j}(t)\}}{\text{maximize}} \quad R_T(\beta) \tag{3.17}$$

$$\text{s.t.} \quad x_{i,j}(\tau_p) \in \{0,1\}, \qquad i \in \mathcal{A},\ j \in \mathcal{U}(t),\ p \in [|1,P|], \tag{3.18}$$

$$\sum_{j \in \mathcal{U}_i} x_{i,j}(\tau_p) \leq N_i, \qquad i \in \mathcal{A}(t)\backslash\{0\},\ p \in [|1,P|], \tag{3.19}$$

$$\sum_{i \in \mathcal{A}_j} x_{i,j}(\tau_p) = 1, \qquad j \in \mathcal{U},\ p \in [|1,P|]. \tag{3.20}$$

The constraint (3.18) ensures that the decision variables are binary. The constraint (3.19) indicates that the maximum number of UEs that a SBS can simultaneously support is limited to N_i. Finally, the constraint (3.20) indicates that a UE is always associated with a BS. The optimization problem (3.17)-(3.20) is a non-convex integer programming problem. In addition to the complexity of such a

problem, the optimal association at time τ_p also depends on the association at time step τ_{p-1} through the handover variable λ_j, making the problem (3.17)-(3.20) intractable with conventional optimization frameworks. In the following, we hinge on our proposed multi-agent reinforcement learning framework to solve this problem.

3.4.2 Proposed handover framework

In this subsection, we depict the proposed HO solution. We formalize the optimization problem (3.17)-(3.20) as a multi-agent reinforcement learning (MARL) task where each UE is modeled as an independent agent that learns in a distributed way its handover strategy with the goal of optimizing the network throughput.

UEs action space. At each time step τ_p, each UE j takes an action $a_j(\tau_p)$ to associate with one BS in the network. If the connection request is addressed to the MBS, this is automatically granted. Otherwise, if the requested SBS is able to support the association, an acknowledgment signal is sent (ACK = 1), otherwise ACK = 0 (see the constraint (3.19)). Finally, if UE j's BS at time step τ_p differs from the one at time step τ_{p-1}, the UE initiates a handover procedure. Later, the MBS collects information from each BS to compute the overall network throughput $R(\tau_p, \beta)$, which is broadcast to all UEs to evaluate the goodness of their policy.

UEs state space. To learn their optimal strategy, UEs continuously collect information about their surrounding environment. We assume that at each time step, each UE can measure the RSS of the surrounding BSs *i.e.*, $\{\text{RSS}_i, \forall i \in \mathcal{A}\}$. In addition, each UE uses the previously perceived data rate $R_{a_j(\tau_p),j}(\tau_{p-1}, \beta)$ and network sum-rate $R(\tau_{p-1}, \beta)$. Hence, at time τ_p, UE j acts based on its local observations:

$$\mathbf{o}_j(\tau_p) = \left\{ v_j^x(\tau_p), v_j^y(\tau_p), a_j(\tau_{p-1}), \overline{R}_j(\tau_{p-1}, \beta), R(\tau_{p-1}, \beta), \text{ACK}_j(\tau_{p-1}), \{\text{RSS}_i(\tau_p)\}_{\forall i \in \mathcal{A}} \right\}, \tag{3.21}$$

where $v_j(\tau_p) = (v_j^x(\tau_p), v_j^y(\tau_p))$ is the corresponding UE's speed.

UEs reward. To optimize the network performance, UEs must learn how to perform association requests, which limit handovers and avoid *collisions* across service requests. Let $c(\tau_p)$ denotes the request *collision* event. There is a request *collision* at time step t, *i.e.*, $c(\tau_p) = 1$, if $\exists\ i$ such that $\sum_{j \in \mathcal{U}} x_{i,j}(\tau_p) > N_i$. Otherwise, we set $c(\tau_p) = 0$. To optimize the handover procedure, we have designed two reward functions taking into account the collision events.

- **RHando-F (Fully cooperative RHando)**: in this strategy, UEs receive the same reward, which favors global network optimization:

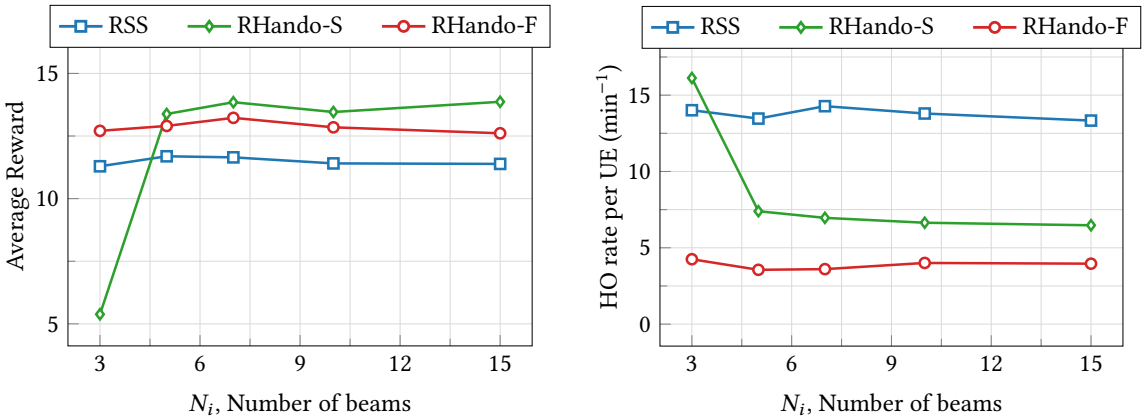
$$r_j(\tau_p) = (1 - c(\tau_p))\Delta\tau R(\tau_p, \beta). \tag{3.22}$$

- **RHando-S (Self interest RHando)**: here, each UE instantaneous reward only considers the data rate it perceived. Hence,

$$r_j(\tau_p) = (1 - c(\tau_p))\overline{R}_{a_j(\tau_p),j}(\tau_p, \beta). \tag{3.23}$$

  It is noteworthy that even in RHando-S, the reward of each UE still depends on other UEs because of the interference and the collision events.

Next, we use the HDRQN architecture proposed in section 3.3.2 to train users' HO policies.



(a) Average reward, $R_T(\beta)$

(b) HO frequency

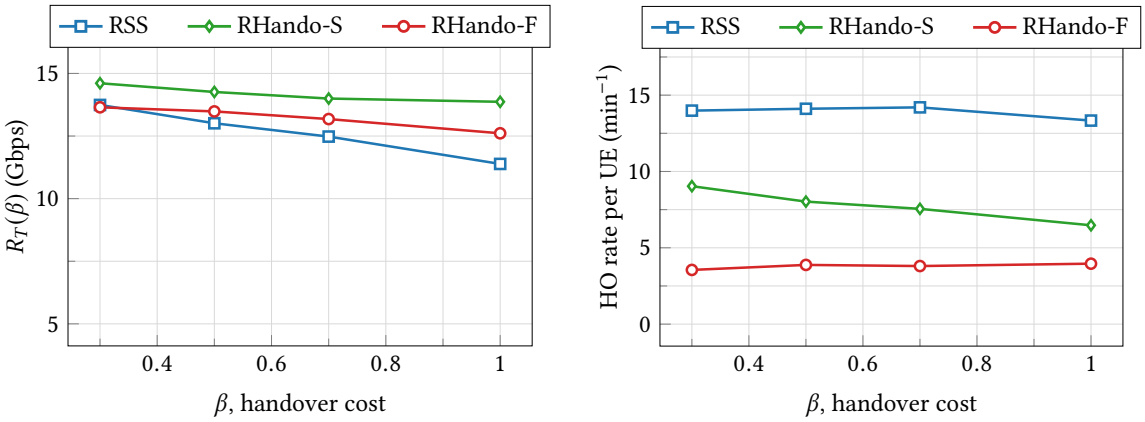
Figure 3.12: Average reward *w.r.t.* to number of beams N_i. Here, $K = 15$, $m = 0.5$, $\beta = 1$.

3.4.3 Performance comparison

To assess the performance of the proposed framework, we consider as a benchmark a simplified version of the HO procedure proposed in 3GPP [70] in which each UE is associated to the BS providing the strongest RSS. In case of request collision, each SBS selects the best UEs in terms of RSS while the MBS serves the other UEs. In all tests, five mmWave SBSs are deployed inside the macro cell. UEs' locations are randomly initialized. To account for heterogeneous mobility, each UE randomly picks a speed between 0 and 10 ms^{-1} and takes a straight motion with a random direction. In addition, without loss of generality, we suppose that users turn back once they reach the macro cell edge. We set $\Delta\tau = 1$s, TTI = 10 ms. The simulation lasts $T = 2000$ s. For a given UE i associated to a given BS j, we evaluate $\overline{R}_{i,j}(\tau_p, \beta)$ by aggregating the data received during each TTI (see Eqn. (3.15)). We use the antenna radiation pattern `diag 3` of Figure 3.4. Additional simulation parameters can be found in Table 3.1.

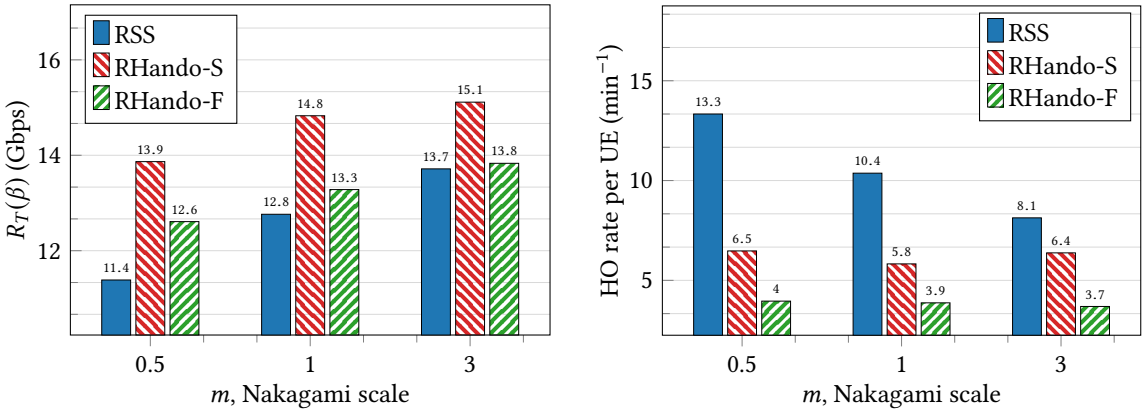
Collision avoidance. As aforementioned, request collisions may happen when BSs do not have enough beams to support, simultaneously, all the service requests. Figures 3.12a and 3.12b show the performance comparison of the two RHando configurations compared to the benchmark solution. Unsurprisingly, for lower values of N_i, RHando-S exhibits poor performance than RHando-F both in terms of average reward (*i.e.*, $(1/P)\sum_{p=1}^{P}\sum_{j\in\mathcal{A}} r_j(\tau_p)$) and HO's frequency. This is because UEs in RHando-F fully cooperate through the common reward they perceive and, as a result, they effectively learn to avoid request collisions. In contrast, with RHando-S, each UE learns a policy based on a local reward, which does not provide sufficient information on the effect of its action on the other UEs' reward. Inversely, when N_i is sufficiently large (> 7), RHando-S outperforms both RHando-F and RSS-based HO in terms of average reward. The throughput is increased by about 17.89% by RHando-S and only 10% by RHando-F compared to the benchmark. Regarding the HO events, RHando-F decreases the HO frequency by about 70% and RHando-S by 54% compared to the baseline. Overall, we can observe that the fully cooperative approach limits the handover rate at the cost of lower reward when N_i is large.

The handover cost factor β has an impact on performance. Now we evaluate the performance of the proposed solutions *w.r.t.* the handover cost factor β. Figure 3.13b shows that when the HO cost increases, the network average throughput decreases. The RSS-based solution is characterized by the worst performance as it does not consider the handover cost. Figure 3.13a shows that when the HO becomes more and more costly, the HO rate decreases with Rhando-S while remaining almost constant with Rhando-F. This is because the HO cost variation has a limited impact on the global reward perceived by the agents in RHando-F: after a handoff decision, an agent can still perceive a large global reward as this is defined as the sum of all the other agents' reward.



(a) Average reward, $R_T(\beta)$ (b) HO frequency

Figure 3.13: Impact of the cost factor β on network performance. Here, $N_i = K = 15$, $m = 0.5$.



(a) Average reward, $R_T(\beta)$ (b) HO frequency

Figure 3.14: Impact of the fading on system performance. Here, $N_i = K = 15$, $\beta = 1$.

The average throughput varies with Nakagami fading scale factor m**.** HO events highly depend on the channel conditions viz. path loss and fading. Here, we evaluate the performance gain of the different algorithms *w.r.t.* the Nakagami scale factor m. Figures 3.14b and 3.14a show that the more severe the fading ($m \to 0$), the more pronounced the gain of the proposed solution compared to the benchmark both in terms of average throughput and number of HOs. The performance of the RSS-based HO strongly deteriorates with the fading while RHando-F and RHando-S adapt their policies to the fading characteristics demonstrating therefore the robustness of the proposed framework.

3.4.4 Concluding remarks

We have shown in this Section an application of our proposed MARL framework for handover management, a different problem, but related to user association. In particular, in this problem, we have optimized the network sum-rate considering the delay induced by handover events. The proposed solution is also distributed among mobile users with limited signaling overhead. We have shown its ability to reduce the handover events by 50% and increase the sum-rate by 10% compared to the baseline solution based on maximum-RSS.

3.5 Conclusion and Perspectives

In this chapter, we have presented our main framework based on distributed MARL, which allows successfully solving the user association problem. In particular, we modeled each user equipment as an independent agent, which takes autonomous decisions based on its local observations. Despite the proposed solution is distributed, we have shown that by observing only a few local parameters, our solution is able to achieve near-optimal performance. Moreover, our proposed solution incorporates environment dynamics viz. fading, user traffic and mobility, intra- and inter-cell interference, so that the optimal solution is self-reorganized when a relevant change occurs. Finally, we have shown that the proposed solution can successfully address handover management, a close problem to user association with additional complexity as it involves user mobility.

Despite all these appreciable features, our proposed solution still lacks, to some extent, flexibility and adaptability. Indeed, our proposed framework, as present solutions in the literature, optimizes the user association by considering either 1) a fixed set or position of users, 2) a fixed distribution of traffic, or 3) in the context of mobility, fixed directions (predefined trajectories). In other words, whenever i) the number of users changes due to the arrival or departure of UEs, or ii) their positions arbitrarily change *e.g.* due to random mobility, or iii) their service requirements change due to *e.g.* a UE switching from a vocal call to online gaming, the solution of the user association has to be recomputed. This involves frequent signaling to report changes in the radio environment and frequent learning processes to adapt to these changes. Therefore, the following questions are still open: *can we build user association policies able to accommodate all these changes? can we come out with transferable user association policies in which knowledge gained in a given scenario can be transferred to another scenario, preventing frequent learning processes?* Solving these problems opens new perspectives to build fully transferable user association knowledge or policies.

In the next chapter, we will investigate solutions to address these issues, which require rethinking the architecture design of the user association policies as well as the associated learning mechanism.

The technical contributions of this chapter have been validated by the following conference papers, journal paper, and patent.

[C1] **M. Sana**, A. De Domenico, and E. Calvanese Strinati, *"Multi-Agent Deep Reinforcement Learning based User Association for Dense mmWave Networks,"* In Proc. IEEE Global Communications Conference (GLOBECOM), HI, USA, pages 1–6., Dec 2019.

[C2] **M. Sana**, A. De Domenico, E. Calvanese Strinati, and A. Clemente, *"Multi-Agent Deep Reinforcement Learning for Distributed Handover Management In Dense MmWave Networks,"* In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Madrid, Spain, pages 8976–8980., May 2020.

[J1] **M. Sana**, A. De Domenico, W. Yu, Y. Lostanlen, and E. Calvanese Strinati, *"Multi-Agent Reinforcement Learning for Adaptive User Association in Dynamic mmWave Networks,"* IEEE Transactions on Wireless Communications, 19 (10):6520–6534, 2020.

[P1] **M. Sana**, A. De Domenico, *"Method for associating user equipment in a cellular network via multi-agent reinforcement learning,"* Issued in May 20, 2021, US17099922.

Part II

Design of Distributed and Transferable Intelligence

Chapter 4

On the Transferability of User Association Policies

"La connoissance de certains principes supplée facilement à la connoissance de certains faits."

"The knowledge of certain principles easily compensates the lack of knowledge of certain facts."

– Claude Adrien Helvétius, *De l'esprit* (1715 – 1771)

Contents

4.1 Introduction

This chapter addresses the problem of *transferability* of the user association policy. Here, we focus on a solution of user association, which allows the learned policy to cope with environment dynamics, including changes in the number and/or position of users, variation in traffic dynamics as well as variation in wireless channels. To this end, we first propose a *policy distillation* mechanism that builds on the user association solution previously proposed in Chapter 3 to consolidate the knowledge gained in different scenarios into one global knowledge. Although this solution is able to cope with the variation of network traffic, it is limited in terms of scalability. Therefore, we propose a novel Proximal Policy Optimization (PPO) and learning mechanism to derive a transferable user association strategy based on Multi Agent Reinforcement Learning (MARL) and neural attention mechanisms. The resulting framework is able to address changes in the radio environment, including channel dynamics, mobility of UEs as well as the variability of the number of UEs over time.

4.1.1 Motivations

Current state-of-the-art solutions for user association are, in general, not scalable and tangibly lack adaptability. In particular, they are often grounded on quite rigid assumptions, such as pre-sized and fixed sets of BSs and static UEs, favorable channel conditions, absence of inter-cell or intra-cell interference, full-buffer network traffic. Yet, in dynamic mmWave networks, especially in dense networks, the number of UEs, their position to each other and BSs, and the performance requirements of the services they access are likely to change over time and are characterized by a high dynamicity. Even in relatively stable scenarios, from the radio channel and data traffic points of view, the arrival in the network or the departure from the network of one or more users has an impact on the overall network performance, which requires a constant adaptation of the user association to dynamically guarantee the best possible quality of service. To tackle these problems, we propose transferable user association policies. Transferability is an important key feature. It allows transferring the user association knowledge acquired in one specific scenario to another one [76], thus, resulting in a significant gain in terms of signaling and computation overhead.

4.1.2 Related work

Very few works in the literature have addressed knowledge transfer for user association [77]. In [78], authors propose a transfer learning scheme, which enables base stations to share learning knowledge to improve system QoS. A transfer learning algorithm is developed in [79], which allows transferring the expertise knowledge learned from spectrum assignment to formulate a knowledge base for user association. Similarly, [80] proposes to apply transfer learning for spectrum sensing. In [81], an apprenticeship learning mechanism is proposed for spectrum decision, namely for channel selection and handoff. None of these works apply to user association in 5G networks or to the distributed multi-agent system. In this new chapter, we address the problem of *transferability* of the user association policy. We propose a novel Policy Network Architecture (PNA) and learning mechanism to derive a transferable user association strategy able to address changes in the radio environment, including channel dynamics, mobility of UEs as well as the variability of the number of UEs over time.

4.1.3 Contributions

The contribution of this chapter can be summarized as follows:

- *Policy distillation in small-scale dynamics*: we design an offline *distillation procedure* consisting of integrating experiences related to different scenarios in a single one so that the users can adjust their association policy to abrupt changes in the radio environment. In particular, this is the case

when the dynamic of the UE traffic requests changes in time and that the user association must be updated accordingly to avoid performance losses.

- *knowledge transferability*: unlike approaches in the literature [39, 40], which require reconstructing the PNA (*i.e.*, the NN architecture) and a completely new learning process each time the number or the position of UEs changes, our new proposed solution has the advantage of being transferable. In other words, both the PNA and the learned association skills can be transferred to a new scenario or to a new UE that joins the coverage area without any additional changes. To do this, instead of having one specific policy per UE as in the previous chapter, we come out with a single global PNA based on neural attention mechanisms, which can be trained efficiently with the experiences of all UEs. Thanks to the attention mechanism, our proposed architecture is transferable without any additional loss in performance.

- *hysteretic proximal policy optimization*: to optimize the proposed user association PNA, we use a Multi Agent Reinforcement Learning (MARL) framework with policy gradient algorithm, in particular, the PPO framework. However, dynamic channels and network traffic combined with the simultaneous interaction of agents make the radio environment highly non-stationary, which challenges MARL systems. Therefore, as in the previous chapter, to stabilize the learning process and improve the convergence, we rely on the concept of the hysteretic Q-learning [64, 63]. We modify the PPO algorithm by introducing two clipping factors that induce a hysteretic behavior in policy updates. By doing so, agents become optimistic by giving less importance to the low reward received (*e.g.*, because of environment noise) from actions that were successful in the past. We show through numerical simulations the benefit of such a method both on the convergence and the system performance.

- *zero-shot generalization*: in addition and in sharp contrast with existing solutions, the proposed mechanism has zero-shot learning capability, *i.e.*, it can actively adapt to the variations due to the departure or arrival of UEs without requiring additional training iterations. For this purpose, we introduce a *UE dropout mechanism*, which consists in masking some UEs during the learning process to enable robustness of the learned policy *w.r.t.* the variation of the number of UEs in the network. We show that the *dropout mechanism* further stabilizes the learning process and enables better knowledge transferability.

- *adaptability w.r.t. to channel and traffic dynamic*: our learning mechanism also incorporates channel dynamics (fast fading, shadowing) and network traffic dynamic allowing the proposed solution to quickly adapt to fluctuations of these parameters in practical implementations.

- *distributed, centralized or hybrid architecture*: as we come out with a solution involving only one global model shared by all UEs, another salient feature of the proposed architecture is that both the learning process and the execution can be either distributed or centralized or even be implemented in an hybrid way. In the case of a centralized implementation, the PNA may be located at a central controller, which assigns BSs to UEs based on their feedback. In a distributed setting, instead, each UE has a copy of the PNA and can take its association decisions locally. Finally, for a hybrid implementation, we show that parts of the PNA can be located at the UEs and at the central controller to leverage the advantage of both the centralized and distributed solutions.

The technical content of this chapter is based on the published journal paper [56] and conference paper [82].

The remainder of this chapter is organized as follows. Section 4.2 presents the proposed adaptive user association based on policy distillation mechanism. Section 4.3 details our transferable user association solution. We provide numerical results in Section 4.5 and draw conclusions in Section 4.6.

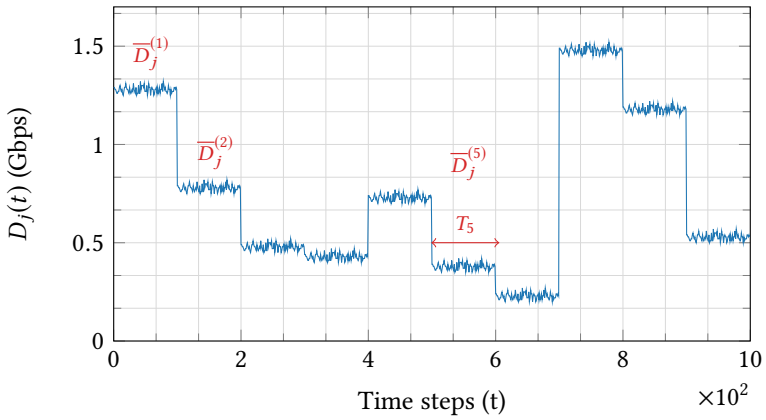


Figure 4.1: Example of the variation of UE j service request with time.

4.2 Proposed Adaptive solution via Policy Distillation

4.2.1 Policy distillation

Let consider the system model defined in Section 2.3 of Chapter 2 (see Figure 2.2). We now focus on a more realistic scenario, where the service requests of the UEs can change over time, *e.g.*, from video streaming to Virtual Reality (VR) applications. We model this change by abruptly modifying the intensity of the Poisson distribution that characterizes the UE traffic, *i.e.*, for UE j, $\overline{D}_j(t)$ is now time-dependent (see Figure 4.1). This increases the non-stationarity of our system and makes the learning process more challenging. To deal with this, the agents may keep updating their policies online, to adapt them to an eventual drastic change in the environment's dynamics. This approach may lead to good performance if the convergence time of the algorithm is sufficiently shorter than the time during which the system is stationary. However, in a multi-agent system, this condition is unlikely satisfied and thus, we design an offline training strategy that allows the agents to perform well during the execution time even in strong non-stationary conditions.

Let us assume that the time horizon T_e can be divided into P time intervals T_p such that $\sum_{p=1}^{P} T_p = T_e$, where the intensities $\overline{D}_j(t)$, $\forall j \in \mathcal{U}$ remain constant. Accordingly, we denote by $\overline{D}_j^{(p)}$ the average data rate requested by UE j in the time interval p. Then, we define a task $\mathcal{T}_p$ as the set of the UEs' traffic requests during the time interval p:

$$\mathcal{T}_p = \left\{ \overline{D}_1^{(p)}, \overline{D}_2^{(p)}, ..., \overline{D}_K^{(p)} \right\}. \tag{4.1}$$

In our setting, each agent does not have the global knowledge of each task specifications; in fact, a UE is unaware of the data rate demands of the other UEs. However, we aim to derive, for each user, a unique policy that performs well in any task. This problem falls in the context of the so-called Multi-Task Reinforcement Learning (MTRL) [83], where *policy distillation* consolidates multiple task-specific policies into a single policy. Indeed, *policy distillation* enables to transfer one or more action policies (learned knowledge) from expert Q-networks to an untrained network. Specifically, with this mechanism, for every task, we run Algorithm 1 to collect the agents task-specific policies $\pi(\mathcal{T}_p)$; that is, we derive as many policies as there are tasks for any single agent. Then, for every agent j and task p, we execute the related policy for a time T_p and we store all the collected observations/action values $\langle o_j(t), Q_j(o_j(t); \theta_j) \rangle_p$ into a memory $\mathcal{M}_j$ (see Algorithm 3). Later, for each UE j, we conduct supervised learning on the generated database $\mathcal{M}_j$ to learn a distilled policy π_j^D through a single Deep Recurrent Q-Network (DRQN) (having the same architecture as in Figure 3.3 with parameters θ_j^D) trained via a

Algorithm 3: Distillation Procedure for UE j

for $p = 1, \ldots, P$ **do**
  Initialize $o_j = \{0\}$.
  Select a policy $\pi_j(\mathcal{T}_p)$.
  for $t = 0, \ldots, T_p$ **do**
    Observe the new state $o_j(t)$.
    Using the expert policy $\pi_j(\mathcal{T}_p)$ takes $a_j(t)$.
    Get $Q_j(o_j; \theta_j)$.
    Store $\langle o_j(t), Q_j(o_j(t); \theta_j)\rangle$ into a memory $\mathcal{M}_j$.
  end
end
Initialize the distilled DRQN weights θ_j^D.
Perform supervised learning using $\mathcal{M}_j$.

tempered Kullback-Leibler (KL) divergence loss function:

$$\mathcal{L}(\theta_j^D) = \mathbb{E}_{\mathcal{M}_j}\left[\text{softmax}\left(\frac{Q_j}{\tau}\right)\log\left(\frac{\text{softmax}\left(\frac{Q_j}{\tau}\right)}{\text{softmax}\left(Q_j^D\right)}\right)\right], \tag{4.2}$$

where the temperature τ controls the way the knowledge is transferred from the expert policies to the distilled policy [83]. Increasing the temperature softens the Q-values, which may prevent the distilled agent from taking the same actions as the expert. In contrast, when the temperature decreases, Q-values becomes more and more sharpened ensuring more knowledge distillation. Therefore, τ is typically set as a small positive value [83].

4.2.2 Performance comparison

Here, we show the capacity of our scheme to adapt the association policy with respect to time-varying service requests in the network. As the service request (and its corresponding data rate) at each UE change during time, the user association has to adapt to keep optimizing the network performance *i.e.*, balancing the cell load. To achieve this property, we use the aforementioned distillation mechanism.

Let us consider three services rate requirements denoted as SERVICE 1, SERVICE 2, and SERVICE 3, corresponding respectively to an average data rate demand of $D_{s1} = 5$ Mbps, $D_{s2} = 200$ Mbps, and $D_{s3} = 1.5$ Gbps. SERVICE 1 may be related to web browsing or voice call services, SERVICE 2 to online video streaming, and SERVICE 3 to augmented reality or virtual reality applications. In the following, we focus on three time periods during which the UEs randomly change their service requests and we apply the distillation procedure (*i.e.*, Algorithm 3 with $P = 3$). Figure 4.2 shows a sample of the performance of the proposed HDRQN with and without distillation. Specifically, the agent policies without distillation are obtained through a single training phase over the three time periods. The upper part of this figure highlights the data rate changes for each of the 9 UEs in the network. The middle part of the figure describes the corresponding user association[1]. Finally, the lower part shows the evolution of the network sum-rate. Overall, Figure 4.2 shows that the proposed algorithm using distillation mechanism can effectively adapt the user association to service request dynamics thus, outperforming the two baselines. For example, we see that UE 4 is served by SBS 1 in the first two time intervals when it is requiring SERVICE 3; in contrast, in the last interval, when it demands for SERVICE 1, which is characterized by a lower data rate request, its access is provided by the MBS. Meanwhile, in the last interval, UE 5 asks for

[1]Note that the UEs not served by a mmWave beam are receiving data through the MBS.

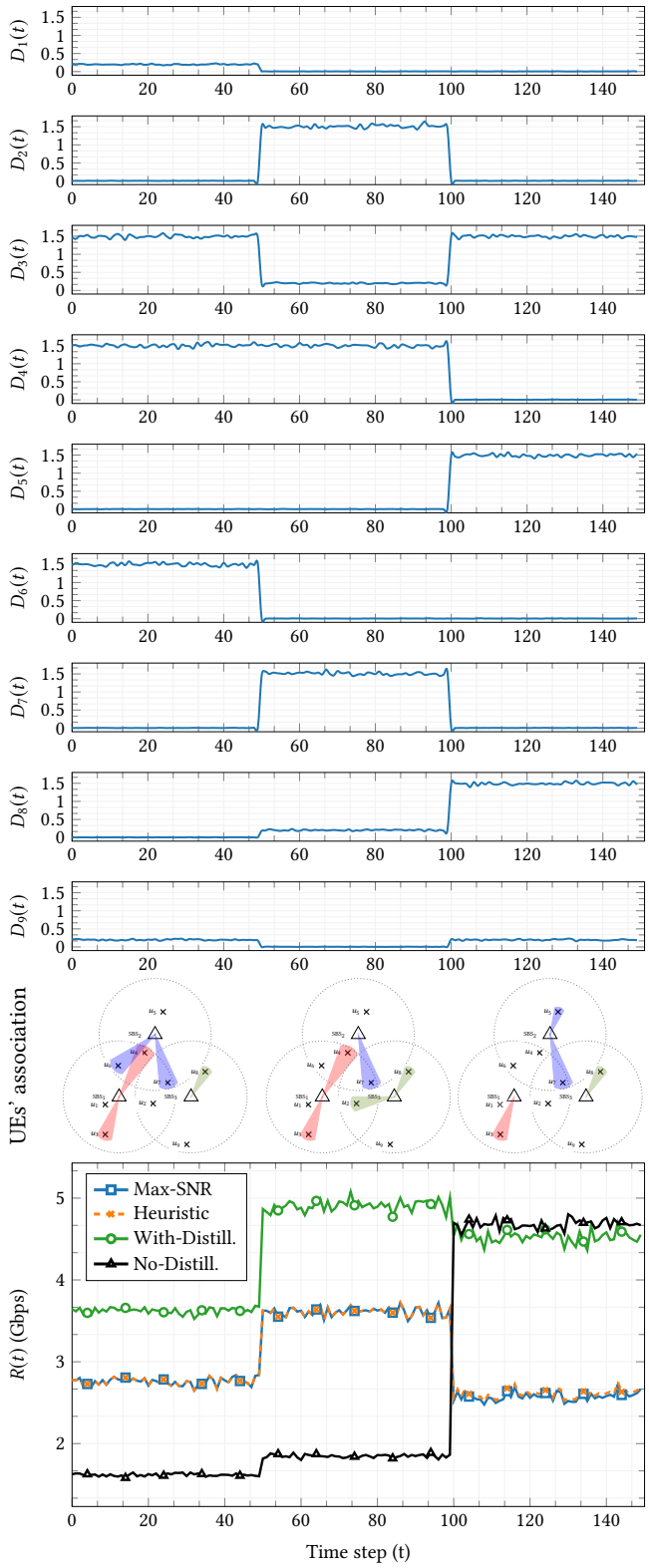


Figure 4.2: Dynamic behavior of the proposed adaptive user association scheme. We set the loss temperature to $\tau = 0.01$ via informal search. Here, $D_j(t)$ is expressed in Gbps.

SERVICE 3; therefore, it hands off from the MBS to SBS 2, which can satisfy its demand for a higher data rate. Moreover, we can highlight that in the absence of distillation, the proposed solution shows poor performance during the first two time periods. This is due to the forgetting effect inherent to neural networks training: at the end of the third period, the agents have forgotten what they have learned in the first two periods. The resulting policy is therefore only appraised to handle the last service for

which it exhibits the best performance.

4.2.3 Concluding remarks

Despite that the solution based on policy distillation can adapt the user association decision to the environment's dynamics, like most of the state-of-the-art mechanisms, it optimizes the user association for a fixed number and/or position of UEs in the network. This has two implications:

- Whenever the position or the number of UEs change, the solution of the user association has to be recomputed.
- Transferring the knowledge (association policy) from a given user already in the network to a new incoming user is not a trivial task. Indeed, each user learns its own DRQN, *i.e.*, it has its own association policy, which is specific to its location and requirements.

4.3 Design of Transferable Policy Network Architecture

Taking into account the targeted optimization objective (2.7), we derive in this section an adaptive association policy capable of solving the user association problem regardless of the location and the number of UEs in the network. The desired policy must be able to adapt to the departure or arrival of UEs from and in the network, as both events have an impact on the optimal user association. To do so, we propose to construct a transferable user association PNA, invariable with the number of UEs, which can be efficiently trained and then transferred to any UE that arrives in the cell. This policy leverages UEs' local information and if available global information to optimize the association decisions using a MARL framework.

4.3.1 System model

Let us recall the system model defined in Section 2.3 of Chapter 2 (see Figure 2.2). Now we assume that the number $K(t)$ of UEs varies over time *e.g.* due to arrival or departure of UEs or change of network deployment. We call a network deployment $\mathcal{D}(t)$, a collection of positions of all UEs in the network:

$$\mathcal{D}(t) = \left\{ \left(x_j(t), y_j(t)\right), j \in \mathcal{U}(t) \right\}, \tag{4.3}$$

where $x_j(t)$ and $y_j(t)$ denote respectively the two coordinates of UE j in deployment $\mathcal{D}(t)$, expressed *w.r.t.* a reference system common to all UEs and BSs. Accordingly, the set of UEs $\mathcal{U}(t) = \{1, 2, \ldots, K(t)\}$ varies with time as well as the action space $\mathcal{A}_j(t)$ of each UE j. Our goal is still to solve the optimization problem (2.9)-(2.12). However, we focus on association policies, which are also transferable and capable of solving problem (2.9)-(2.12) at each time t regardless of the location and the number of UEs in the network, *i.e.*, regardless of the deployment $\mathcal{D}(t)$. This policy must adapt to the departure or arrival of UEs without requiring any additional learning procedure, as both events impact the optimal user association. Thus, a policy learned, *e.g.*, in a scenario of K_1 UEs has to be effectively applicable to a scenario of $K_2 \neq K_1$ UEs without additional training. To achieve this, the architecture of the association policy needs to be transferable, so does the learned policy.

4.3.2 Policy network architecture: general framework

In this section, we provide a general description of the PNA illustrated in Figure 4.3, whose component design details will be specified in Section 4.3.3. For now, let us denote by $\mathbf{o}_j^{\mathrm{L}}(t)$ and $\mathbf{o}_j^{\mathrm{G}}(t)$ the *local* and *global* observation of UE j respectively. $\mathbf{o}_j^{\mathrm{L}}(t)$ comprises the set of measurement signals directly accessible to (or measurable by) the user's device. Instead, depending on the optimization objective and constraints, $\mathbf{o}_j^{\mathrm{G}}(t)$ embeds higher-level information (macro observations), which can be collected

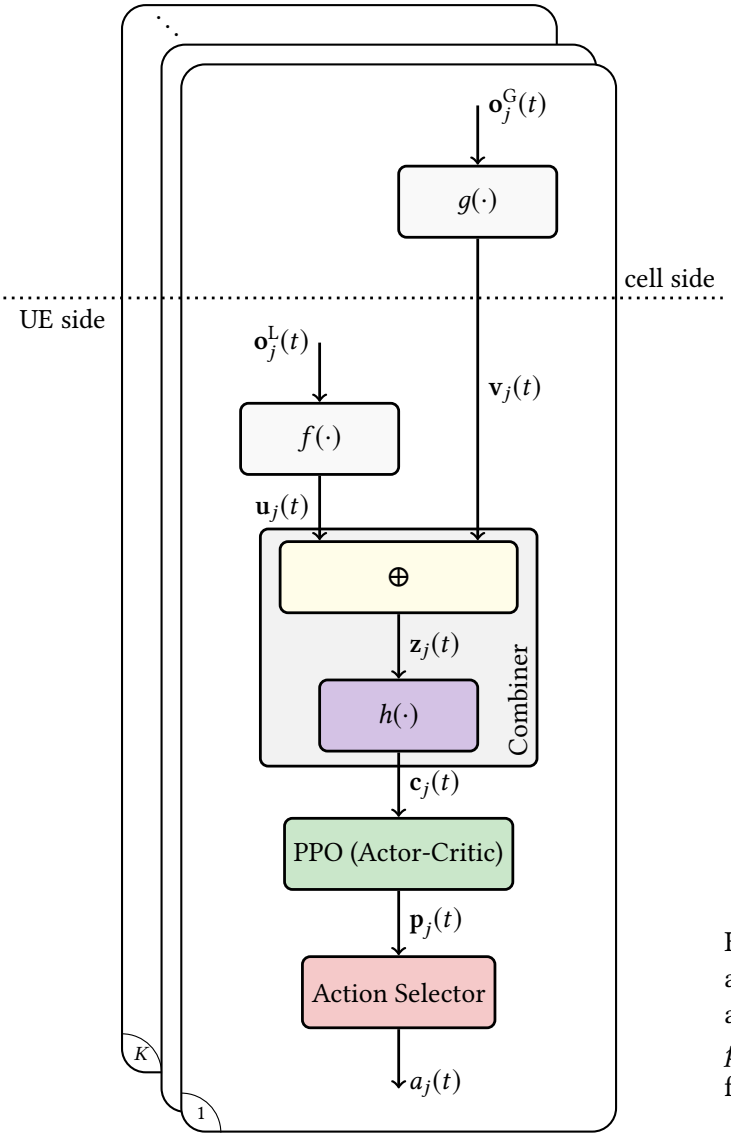


Figure 4.3: UE association policy network architecture. This model is shared across all UEs and is trained using *proximal policy optimization* with an actor-critic framework.

and forwarded to UEs by the central controller. Then, in our proposed framework, each UE starts by building its local state encoding $\mathbf{u}_j(t) = f(\mathbf{o}_j^{\mathrm{L}}(t))$ and global state encoding $\mathbf{v}_j(t) = g(\mathbf{o}_j^{\mathrm{G}}(t))$ using differentiable and learnable functions $f(\cdot)$ and $g(\cdot)$[2] (*i.e.*, functions with learnable parameters like NNs). Next, the local and global state encoding are combined together to form the agent context encoding $\mathbf{c}_j(t)$ using a combiner function $h(\cdot)$, *e.g.*, a concatenation operator, or a neural network. The role of this combiner function is to build UE context understanding vector, as a representation of its local and global observations. Now, given the context vector $\mathbf{c}_j(t)$, the goal of the learning agent j at each time instant t, is to define an *association probability vector* $\mathbf{p}_j(t) = [p_{0,j}, \ldots, p_{N_s,j}] \in [0,1]^{N_s+1}$ with $\sum_{i \in \mathcal{A}} p_{i,j} = 1$ and $p_{i,j} = 0 \; \forall \; i \notin \mathcal{A}_j$. Then, UE's action $a_j(t)$, which corresponds to a connection request towards the BS indexed by $a_j(t)$ in $\mathcal{A}_j$, is sampled from the distribution characterized by the $p_{i,j}$. Thus, the learning problem here consists in deriving an association policy that optimizes the corresponding association probability vector $\mathbf{p}_j(t)$, so that sampling from it maximizes the network utility function (3.5).

Figure 4.3 describes the proposed PNA. Note that in this architecture, UEs' agents share the same model, *i.e.*, $f(\cdot)$, $g(\cdot)$, and $h(\cdot)$ are common to all UEs. This setting does not preclude UEs from taking different actions as they do not observe the same inputs. In contrast, sharing the parameters among UEs enables a better skill transfer since there is only a unique policy (in contrast to having one policy

[2]One can view this process as a filtering stage, which consists in building a state representation of the input observations.

per UE as in the previous chapter), which can be efficiently and simultaneously trained with all UEs' experiences.

4.3.3 On transferable policy architecture: PNA components design

For the policy architecture to be transferable, a proper design of the PNA components is required. Our objective is to construct a policy architecture whose size does not vary with the number of UEs in the network, which is bound to change over time. In the following, we will describe the main components of the proposed PNA, including the contents of local and global observations, as well as the characteristics of encoding functions $f(\cdot)$, $g(\cdot)$ and $h(\cdot)$, which allow the transferability of the policy architecture.

4.3.3.1 UE local observation encoding

In this study, we assume that at each time step, each UE j can estimate the Received Signal Strength (RSS) and the corresponding Angle of Arrival (AoA) *w.r.t.* its surrounding BSs, which enables UEs to have a broad perspective of their environment. We denote with $\text{RSS}_{i,j}$ and $\vartheta_{i,j}$ the estimated RSS and AoA of UE j *w.r.t.* BS i, respectively. Moreover, as in the previous chapter, a UE receives an acknowledgment (ACK/NACK) signal whenever its connection request succeeds ($\text{ACK}_j = 1$) or fails ($\text{ACK}_j = 0$), which may happen due to the limited resources available at each BS (2.11) inducing *requests collision*. Hence, we define the local state of a UE, $\mathbf{o}_j^{\text{L}}(t)$, as follows[3]:

$$\mathbf{o}_j^{\text{L}}(t) = \left\{ a_j(t-1), R_{a_j(t-1),j}, R(t-1), \text{ACK}_j, \left\{\text{RSS}_{i,j}\right\}_{i\in\mathcal{A}_j}, \left\{\vartheta_{i,j}\right\}_{i\in\mathcal{A}_j} \right\}. \tag{4.4}$$

Here, $R_{a_j(t-1),j}$ represents the achievable communication rate when UE j is associated with the BS indexed by $a_j(t-1)$.

Note that the size of $\mathbf{o}_j^{\text{L}}(t)$ does not depend on the number of UEs, in sharp contrast with [40]. Then, we obtain the n-dimensional local encoding vector $\mathbf{u}_j(t) = f(\mathbf{o}_j^{\text{L}}(t))$, where $f : \mathbb{R}^l \to \mathbb{R}^n$ is a neural network, and l is the size of the vector obtained after the concatenation of the elements in $\mathbf{o}_j^{\text{L}}(t)$.

Remark 3 (Collision events handling)**.** *Collisions may occur when a BS receives more connection requests than it can support. In the previous Chapter, we severely discouraged collisions by zero-rewarding UEs when collision events occurred; however, here, as the positions of UEs change over time, the collision management is considerably more complex. Agents must learn that the collision events depend not only on their actions but also on their relative positions. To handle such complexity, we consider a softer solution: when a collision occurs, the BSs send a NACK signal to notify UEs of the collision event, then each BS selects among the colliding UEs the best ones to associate with, according to their association probability. In this way, we do not severely set the reward to zero to punish UEs, and we directly relate the collision events to the training performance.*

4.3.3.2 UE global observation encoding

After an action, $a_j(t)$, the controller can encode for UE j some meaningful information about the global state (*i.e.* macro observations) $\mathbf{o}_j^{\text{G}}(t)$ such as the estimated position of UEs of interfering links, *i.e.*, of active mmWave links, the load of each BS, etc. However, note that incorporating more information does not necessarily imply performance improvement as it also increases the agent's state space, thus requiring more exploration to discover the intrinsic state/action relation at the risk of misleading the agent. In our scenario, we consider the information about the actual rate perceived by each UE j and the

[3] $\mathbf{o}_j^{\text{L}}(t)$ is local in the sense that part of the information in $\mathbf{o}_j^{\text{L}}(t)$ is either local or available to UE in previous time steps (*e.g.*, the total network sum-rate).

position of the potential interferers of UE j, *i.e.*, the set of UEs $\mathcal{N}_j$, susceptible to impact the association decision of UE j through the interference resulting from their communications[4]. Thus, we define $\mathbf{o}_j^{\mathrm{G}}(t)$ as:

$$\mathbf{o}_j^{\mathrm{G}}(t) = \left\{ \varsigma_l = \left[x_l, y_l, R_{a_l(t-1),l} \right], \ \ l \in \mathcal{N}_j \right\}. \tag{4.5}$$

Then, in the sequel, we propose two solutions to construct UE j global state encoding vector $\mathbf{v}_j(t) = g(\mathbf{o}_j^{\mathrm{G}}(t))$.

Fixed-size encoding. A naive solution to construct $\mathbf{v}_j(t)$ is to first concatenate all elements in $\mathbf{o}_j^{\mathrm{G}}(t)$ resulting in a vector of size $m = 3 \times \mathrm{card}(\mathcal{N}_j)$. Then, we obtain the local encoding vector $\mathbf{v}_j(t) = g(\mathbf{o}_j^{\mathrm{G}}(t))$, where $g : \mathbb{R}^m \rightarrow \mathbb{R}^n$ is also a NN. However, such an approach i) has limited scalability as the size of $\mathbf{o}_j^{\mathrm{G}}(t)$ varies with the number of UEs, especially in the neighborhood, and ii) requires ordering elements before concatenation, preventing transferability.

Attention mechanism for order-agnostic and variable-size encoding. An efficient solution to the problem should be agnostic of the ordering in $\mathbf{o}_j^{\mathrm{G}}(t)$. Moreover, to build a scalable and transferable architecture, the size of $\mathbf{v}_j$ should be independent of the length of $\mathbf{o}_j^{\mathrm{G}}(t)$, *i.e.*, the number of UEs in UE j neighborhood. To satisfy these properties, we adopt ideas from the *dot-product attention mechanisms* developed in [84]. Considering this approach, let $\mathbf{k}_j = g_k\left(\varsigma_j\right)$, $\mathbf{q}_j = g_q\left(\varsigma_j\right)$, and $\boldsymbol{\nu}_j = g_v\left(\varsigma_j\right)$, where $g_k,\ g_q,\ g_v : \mathbb{R}^3 \rightarrow \mathbb{R}^n$ are also encoding functions (*e.g.*, neural networks), and $\mathbf{k}_j$, $\mathbf{q}_j$, $\boldsymbol{\nu}_j$ denote the *key*, the *query* and the *value* associated with UE j, respectively. For a given UE j, we compute for each UE in its neighborhood $\mathcal{N}_j$ a weight (or score) $\alpha_{k,j}$

$$\alpha_{k,j} = \mathrm{softmax}\left(\left[\frac{\mathbf{q}_k \mathbf{k}_j^T}{\sqrt{n}} \right]_{k \in \mathcal{N}_j} \right). \tag{4.6}$$

Here, softmax$(\cdot)$ is the softmax function also known as the normalized exponential function. Let $\boldsymbol{\alpha}_j = \left[\alpha_{k,j}, k \in \mathcal{N}_j \right]$. The vector $\boldsymbol{\alpha}_j$ represents the interaction of UE j with its neighbors. Then, we compute the encoding $\mathbf{v}_j$ by aggregating all values' information from the neighborhood as follows:

$$\mathbf{v}_j = \sum_{k \in \mathcal{N}_j} \alpha_{k,j} \boldsymbol{\nu}_k. \tag{4.7}$$

Remark 4. *By construction, the size of $\mathbf{v}_j$ in Eqn. (4.7) is invariable with the size of $\mathcal{N}_j$. That is to say, whenever the number of UEs varies, there is no need to change the PNA.*

Remark 5. *The above process can also be viewed as a message-passing between UEs. In this case, UEs only need to exchange their queries and values with each other in the neighborhood.*

Local and global information combining. Now, once we obtain the UE local and global encoding vector, they are merged to build its context understanding vector $\mathbf{c}_j$, *i.e.*, its perception of the radio environment. This is done thanks to the combiner function $h(\cdot)$ introduced in Section 4.3.2. Here, we propose two design solutions for the combiner function: the *simple combiner* and the *attention-based* combiner.

[4]Note that in this work, for the sake of simplicity, we consider $\mathcal{N}_j$ as the k-nearest neighbors of UE j however, solutions based on local interaction graphs can be considered, where potential interferers can be identified on the basis of an interference threshold following approaches in [37].

Simple combiner. A simple combiner first concatenates $\mathbf{u}_j$ and $\mathbf{v}_j$ to form a $2n$-dimensional embedding vector $\mathbf{z}_j(t) = \mathbf{u}_j(t) \oplus \mathbf{v}_j(t)$, where $\oplus$ denotes the concatenation operation. The agent's context encoding $\mathbf{c}_j(t)$ is finally obtained from $\mathbf{z}_j(t)$ as:

$$\mathbf{c}_j(t) = h\left(\mathbf{z}_j(t)\right), \tag{4.8}$$

where $h : \mathbb{R}^{2n} \rightarrow \mathbb{R}^n$ is also taken here, as a NN.

Self-attention combiner. For the UE to be able to selectively weight the importance of local or global information, the combiner is constructed using a *self-attention mechanism* [84]:

$$\mathbf{c}_j(t) = \boldsymbol{\beta}_j^T \begin{bmatrix} \mathbf{u}_j(t) \\ \mathbf{v}_j(t) \end{bmatrix}. \tag{4.9}$$

Here, the combiner function $h : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^2$ and $\boldsymbol{\beta}_j = \text{softmax}\left(h(\mathbf{u}_j(t), \mathbf{v}_j(t))\right) \in \mathbb{R}^2$. The intuition here is that there may be some scenarios where either local or global information is sufficient for the UE to understand its context. Worse still, the UE can be misled in trying to always consider all the information it receives.

4.4 Proximal Policy Optimization

Our solution relies on MARL. In a MARL system, agents learn by interacting with a shared environment by making decisions following a Markov Decision Process (MDP). In MDP, the action $a_j(t)$ of an agent j in a given state $\mathbf{s}_j(t)$ leads it to the next state $\mathbf{s}_j(t+1)$ and results in a reward $r_j(t)$. From the underlying *experience* $e_j(t) = \{\mathbf{s}_j(t), a_j(t), r_j(t), \mathbf{s}_j(t+1)\}$, the agent learns its policy $\pi_{j,\boldsymbol{\theta}}(\cdot|\cdot)$, parameterized by $\boldsymbol{\theta}$, the set of PNA parameters, where $\pi_{j,\boldsymbol{\theta}}(a_j|\mathbf{s}_j)$ is the probability that agent j takes action a_j in state $\mathbf{s}_j$[5], to maximize an accumulated long-term γ-discounted reward $G_j(t) = \sum_{\tau=t+1}^{T_e} \gamma^{\tau-t-1} r_j(\tau)$ over an *episode* - a new network deployment - of duration T_e:

$$\pi_{j,\boldsymbol{\theta}}^* = \arg\max_{\pi_j} \mathbb{E}_t\left[G_j(t)\right]. \tag{4.10}$$

In our study, we consider the particular case of *cooperative* MARL [85], *i.e.*, UEs share the same reward, hence, they are assigned to the same objective of maximizing the network utility function: $r_j(t) = R(t),\ \forall j$. Moreover, UEs also share the same policy, *i.e.*, $\pi_{j,\boldsymbol{\theta}} = \pi_{\boldsymbol{\theta}},\ \forall j$.

In general MARL, an agent has only access to a partial observation $\mathbf{o}_j(t) = \left\{\mathbf{o}_j^{\text{L}}(t), \mathbf{o}_j^{\text{G}}(t)\right\}$ of the actual state $\mathbf{s}_j(t)$, which is unknown, resulting in Partially Observable Markov Decision Process (POMDP) [63]. Moreover, MARL is subject to non-stationarities due to simultaneous interactions of agents with the environment, which make the learning process more complex. In the literature, *policy gradient* algorithms are used to solve this problem [17], by iteratively updating the policy parameters $\boldsymbol{\theta}$ as follows:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \mu \hat{\boldsymbol{\rho}}_t \tag{4.11}$$

$$\begin{aligned} \hat{\boldsymbol{\rho}}_t &= \mathbb{E}_\pi\left[\nabla \frac{\pi_{\boldsymbol{\theta}_t}(a_j|\mathbf{o}_j)}{\pi_{\boldsymbol{\theta}_{t-\tau}}(a_j|\mathbf{o}_j)} \hat{A}(a_j, \mathbf{o}_j)\right], \\ &= \mathbb{E}_\pi\left[\nabla \zeta(\boldsymbol{\theta}_t) \hat{A}(a_j, \mathbf{o}_j)\right]. \end{aligned} \tag{4.12}$$

Here, the expectation $\mathbb{E}_\pi[\cdot]$ is taken under the stochastic policy π. μ denotes the learning rate, $\hat{\boldsymbol{\rho}}_t$ is the gradient estimator, $\zeta(\boldsymbol{\theta}_t) = \frac{\pi_{\boldsymbol{\theta}_t}(a_j|\mathbf{o}_j)}{\pi_{\boldsymbol{\theta}_{t-\tau}}(a_j|\mathbf{o}_j)}$ is the ratio between the estimate probability at time t and time $t-\tau$, and $\hat{A}(\cdot,\cdot)$ denotes the advantage estimator, which measures the advantage of selecting a given action in a given state. $\hat{A}(a_j, \mathbf{o}_j)$ can be estimated using one step Temporal Difference (TD)

[5] Note that, $\pi_{j,\boldsymbol{\theta}}(a_j|\mathbf{s}_j) = p_{a_j(t),j}$, where $p_{k,l}$ is the probability defined in Section 4.3.2.

error [17] or Generalized Advantage Estimation (GAE) [86]. Hence, at each iteration, the update of $\boldsymbol{\theta}$ is proportional to the advantage estimator to favor actions that yield the highest *advantages* and inversely proportional to the action probability to encourage exploration by enabling actions of lowest probability to be sampled. However, policy gradient updates suffer from high variability as $\hat{\boldsymbol{\rho}}_t$ can take large values from one iteration to another, leading to large updates. To tackle this problem, the PPO approach introduces a constraint in policy updates preventing large discrepancies between iterations [87]. This is done by minimizing the ϵ-clipped surrogate objective function[6]

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{\pi}\left[\min\left(\zeta(\boldsymbol{\theta})\hat{A}, \textbf{clip}\left(\zeta(\boldsymbol{\theta}), 1-\epsilon, 1+\epsilon\right)\hat{A}\right)\right], \tag{4.13}$$

where $\textbf{clip}(x, a, b) = \min\left(\max\left(x, a\right), b\right)$. It is noteworthy that the quantity in the expectation is a lower, hence, pessimist bound of $\zeta(\boldsymbol{\theta})\hat{A}$ so that agent pessimistically ignores updates that will lead to a high change in its policy.

4.4.1 Proposed hysteretic proximal policy optimization

In multi-agent environments, an agent should not be pessimistic in the same way for both "positive" ($\zeta(\boldsymbol{\theta}) > 1$) and "negative" ($\zeta(\boldsymbol{\theta}) < 1$) experience. Indeed, due to the interaction of multiple agents with the environment and the common reward of the cooperative framework, an agent may receive a lower reward because of the bad behavior of its teammates. This may cause the user to change its policy at the risk to misleading it. To overcome this issue, following the concept of hysteretic Q-learning in [64], we introduce *hysteretic proximal policy optimization*, where we modify the surrogate loss as follows:

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{\pi}\left[\min\left(\zeta(\boldsymbol{\theta})\hat{A}, \textbf{clip}\left(\zeta(\boldsymbol{\theta}), 1-\epsilon_1, 1+\epsilon_2\right)\hat{A}\right)\right], \tag{4.14}$$

where $\hat{A}$ is estimated using one step TD error; we use ϵ_1 for negative updates and ϵ_2 for positive updates, where $\epsilon_1 < \epsilon_2$. In this way, an agent gives more importance to updates that improve its policy rather than to ones that worsen it. This setting is particularly important when agents do not have equal contribution to the team's reward and for decentralized learning.

Note that the association policy can be efficiently trained in a centralized way with the experience of all agents or in a decentralized way, *e.g.*, by leveraging the decentralized and distributed PPO approaches presented in [88].

4.4.2 Training with variable number of UEs: proposed UE dropout mechanism

To further enhance the robustness of the learning to the variability of the number of UEs over time, we introduce a *UE dropout mechanism*[7]. Let K_0 be the initial number of UEs in the network. Between episodes of the learning phase, some UEs are randomly selected and masked out (dropped out) to simulate a dynamic environment *w.r.t.* the number of UEs. To mask a UE j at a given time without impacting the learning, we make its agent's observations $\mathbf{o}_j$ correspond to those of a UE located very far from the BSs (*e.g.* "infinitely far"), so that it can be no more associated with any of the SBSs. As a result, its impact on the other UEs (in terms of interference, thus, in terms of association decisions) becomes negligible. In this way, the masked UE seemingly appears as non-existent in the cell for the other UEs.

To this end, during the learning phase, we randomly select the UEs to be masked, by assigning to each UE j an independent Bernoulli variable $\mathrm{B}_j \in \{0, 1\}$. Event $\mathrm{B}_j = 0$ in a given episode represents the masking of UE j and happens with probability $1 - p_j$. As a result, the average number m_K of UEs per

[6] We write $\hat{A}$ instead of $\hat{A}(a_j, \mathbf{o}_j)$ for notation clarity.

[7] This idea is similar to the dropout mechanism in neural networks.

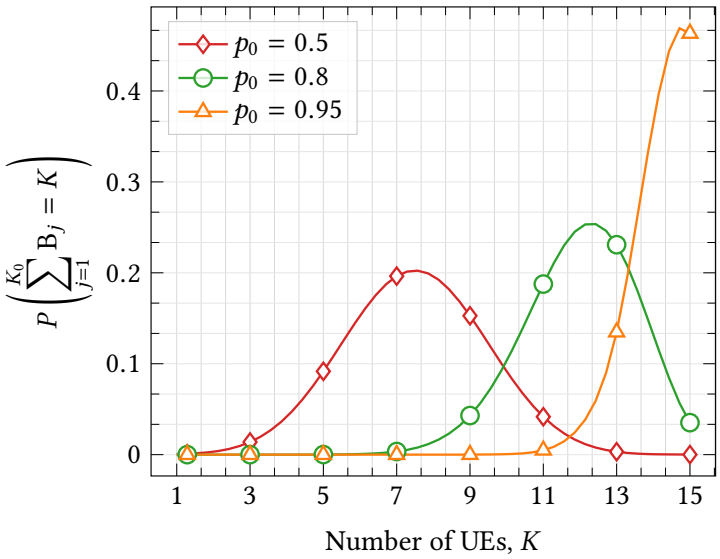


Figure 4.4: Probability density function of $\sum_{j=1}^{K_0} \mathrm{B}_j$ for different values of p_0.

episode that are not masked and the associated variance σ_K^2 are

$$m_K = \mathbb{E}\left[\sum_{j=1}^{K_0} \mathrm{B}_j\right] = \sum_{j=1}^{K_0} p_j. \tag{4.15}$$

$$\sigma_K^2 = \mathbb{E}\left[\left(\sum_{j=1}^{K_0} \mathrm{B}_j - m_K\right)^2\right] = \sum_{j=1}^{K_0} p_j(1-p_j). \tag{4.16}$$

As we assume no difference between UEs, *i.e.* they have the same priority, without lost of generality, we set $p_j = p_0, \forall j$. Accordingly, $\sum_{j=1}^{K_0} \mathrm{B}_j$ follows a Binomial distribution with mean $m_K = K_0 p_0$ and variance $\sigma_K^2 = K_0 p_0(1-p_0)$ (see Figure 4.4). σ_K^2 measures the variability of the number of UEs in the cell between episodes. Although high variability may suggest better generalization, it leads to a large disparity in the number of UEs between episodes, which may prevent the convergence of the learned policy. Therefore, a careful choice of the value of the hiding probability p_0 must be made to achieve the expected improvement in learning robustness via the dropout mechanism.

Important note 2 (Practical implementation of the proposed solution)**.** *In contrast to [40], we come out with a solution with only one global policy shared by all UEs. Consequently, it can be flexibly adapted to the specific design constraints of different practical implementations:*

- *Centralized user association: in a centralized deployment, the PNA can be located at the central controller responsible for assigning a BS to each UE, based on their feedback. This solution reduces the computational complexity at the UEs' side. However, it may result in a large signaling overhead as it requires collecting information from all UEs to take the association decisions.*
- *Distributed user association: in a fully distributed setting, each UE has a full copy of the weights of the PNA, and can take locally its association decisions. Although this solution alleviates the computation burdens due to its distributed nature, it is also subject to an increased downlink signaling overhead, especially when the global information $\mathbf{o}_j^G(t)$ has to be sent to each UEs.*

- *Hybrid user association: to find a hybrid compromise, part of the PNA (like the encoding function $g(\cdot)$) can remain at the central controller and the rest is deployed at the UE's level. In this case, the central controller makes sure to provide each user with the computed vector $\mathbf{v}_j(t)$ to derive its association policy, taking into account local observations $\mathbf{o}_j^L(t)$ (see Figure 4.3). As a result, the signaling overhead is limited, as well as the computation complexity both on the UEs' and the controller's sides.*

In the Appendix B, we provide some details on a practical implementation of the propose mechanism in a distributed setting.

Table 4.1: Transferable policies training parameters

Discount factor, γ	0.6
Time horizon, T_e	250
UE dropout probability, p_0	0.95
Actor and critic learning rate, μ	10^{-4}
Initial number of user K_0	15
Hysteretic parameter ϵ	$\epsilon_1 = 0.01$, $\epsilon_2 = 0.5$
Number of MLP neurons, n	128
Number of Monte-Carlo simulations, N	500

4.5 Simulation Results

In this section, we evaluate the effectiveness of our approach in different simulation settings. We assess both the impact of the training parameters and the dynamics of the radio environment on the system performance. We also evaluate the zero-shot generalization capacity of the proposed framework and, consequently, its transferability.

Radio Environment. In our simulations, we consider $K_0 = 15$ UEs randomly located in a bi-dimensional region, under the coverage of $N_s = 3$ SBSs working at mmWave frequencies with a carrier frequency of 28 GHz, and one MBS communicating at 2 GHz. We assume that when UEs and SBSs communicate together, they use the same antenna radiation pattern obtained through the analog beamforming (see diag 2 in Figure 3.4). In contrast, the MBS transmits via a 17 dBi omnidirectional antenna. In addition, we assume that the error in the estimation of the AoA follows a normal distribution with a mean equal to 2°. Also, in our simulations, we consider three types of service corresponding to an average data rate demand $\overline{D}_j \in \{5, 200, 1500\}$ Mbps . We assume that the traffic request of a UE j is a random variable, which follows a Poisson distribution with intensity $\overline{D}_j = \mathbb{E}\left[D_j(t)\right]$. Simulation parameters are summarized in Table 4.1. Additional simulation parameters can be founded in Table 3.1.

UE action space. Since all UEs share the same policy network, $\mathcal{A}$ coincides with the action space. In this way, we guarantee a fixed action space for all UEs irrespective of their positions. However, a UE j can only be associated with BSs in $\mathcal{A}_j \subseteq \mathcal{A}$. Accordingly, unauthorized actions or connection requests $a_j(t) \notin \mathcal{A}_j$ are redirected towards the MBS, *i.e.*, they appear as connection requests to the MBS.

Learning parameters specification. We fixed the size of the encoding functions $n = 128$. All encoding functions are composed of only one hidden multi-layer perceptron (MLP) of n neurons. The network parameters are optimized using *actor-critic* PPO [87], where both actor and critic comprise also one hidden layers with $2n$ neurons. All layers use a rectifier linear unit (ReLU) activation. We set the learning rate μ to 10^{-4} and the discounting factor $\gamma = 0.6$. Unless specified, we empirically fix the

clipping factors to $\epsilon_1 = 0.01$, $\epsilon_2 = 0.5$, the time horizon to $T_e = 250$ and the dropout probability to $p_0 = 0.95$. Also, we limit the neighborhood of a UE to its k-nearest neighbors, where $k \leq 15$.

Benchmarks. As a comparison, we consider the same benchmarks as in the previous chapter, *i.e.*, the Max-SNR algorithm, which associates UEs based on links with the maximum SNR, and the centralized heuristic algorithm, which consists in associating UEs, starting from the links with the maximum SNR, and in an iterative way as long as it increases the network utility. Originally proposed in [39], the centralized heuristic algorithm is shown to exhibit good performance, specifically in an interference-limited network. Therefore, we use it as a baseline solution in place of the optimal solution, infeasible, due to the network size and complexity.

To assess the convergence performance of the proposed algorithm, we define

$$r_d(t) = \overline{R}^{\text{Trans. RL}}(t) - \overline{R}^{\text{Heur.}}(t), \tag{4.17}$$

which corresponds to the difference of the average reward over an episode reached by the proposed algorithm (denoted Trans. RL) compared to the centralized heuristic approach (denoted Heur.). For sake of clarity, we plot the associated rolling average and standard deviation on a 100-sized window, with a logarithmic scale on the x-axis. Also, unless otherwise specified, we represent on the histograms, the average performance over $N = 500$ random deployments of UEs.

4.5.1 Convergence properties

In this section, we evaluate the algorithm's convergence *w.r.t.* the above learning parameters.

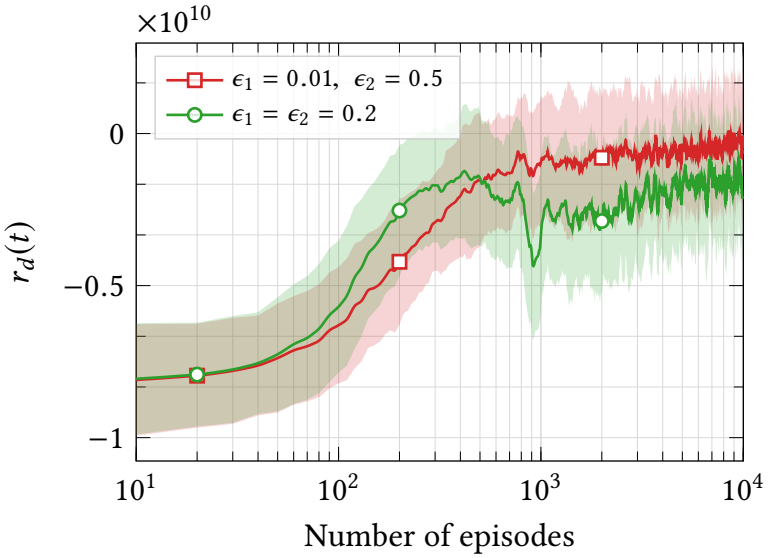


Figure 4.5: Effect of the hysteretic clipping factors on the system's convergence. Here we maximize network sum-rate, *i.e.*, $\alpha = 0$ and $D_j(t) = \infty,\ \forall j$.

4.5.1.1 Effect of hysteretic clipping factors on convergence

Let us start by evaluating the impact of the clipping factors ϵ_1 and ϵ_2 on the convergence. Figure 4.5 shows the evolution of $r_d(t)$ in two settings: $\epsilon_1 = \epsilon_2 = 0.2$, corresponding to the setting of the vanilla PPO proposed in [87], and our empirically optimized hysteretic setting $\epsilon_1 = 0.01,\ \epsilon_2 = 0.5$. We show that by simply introducing a hysteretic effect in the clipping factors, we notably improve the stability and the learning performance, reaching the same performance as the heuristic algorithm (as $r_d(t)$ converges on average to zero).

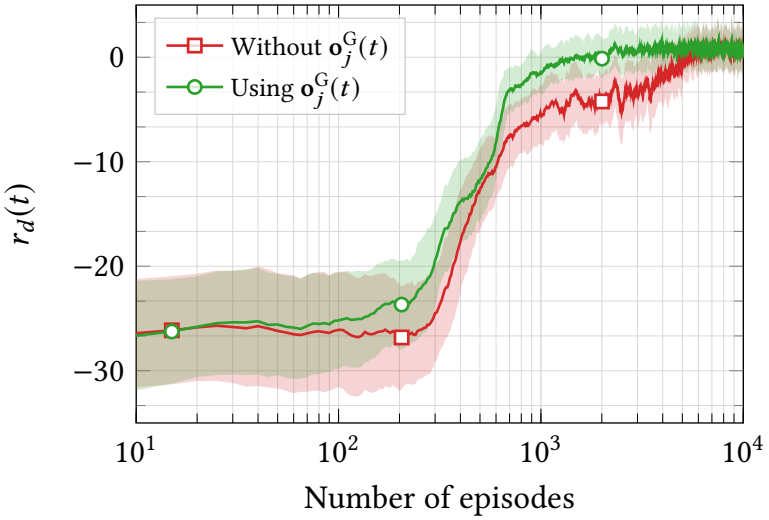


Figure 4.6: Impact of global observations on the system's convergence. Here, we optimize network sum-log-rate, *i.e.*, $\alpha = 1$ and $D_j(t) = \infty,\ \forall j$.

4.5.1.2 Impact of the global information $\mathbf{o}_j^{\mathrm{G}}(t)$ on convergence

Here, we assess the add-on impact of the global information $\mathbf{o}_j^{\mathrm{G}}(t)$ for the learning convergence. Figure 4.6 shows the evolution of $r_d(t)$ when UEs have or do not have access to global information. We remark that $\mathbf{o}_j^{\mathrm{G}}(t)$ can effectively help accelerate the convergence of the algorithm. However, after 5×10^3 episodes, the two curves eventually end up with the same performance. This last result comes from the fact that the information (*i.e.*, $\varsigma_k,\ k \in \mathcal{N}_j$) carried on $\mathbf{o}_j^{\mathrm{G}}(t)$ is also embedded in $\mathbf{o}_j^{\mathrm{L}}(t)$ through the RSSI and $R(t)$, although this information is "drowned". By separating each piece of information in $\mathbf{o}_j^{\mathrm{G}}(t)$, we further improve UEs' context understanding, thus the learning speed.

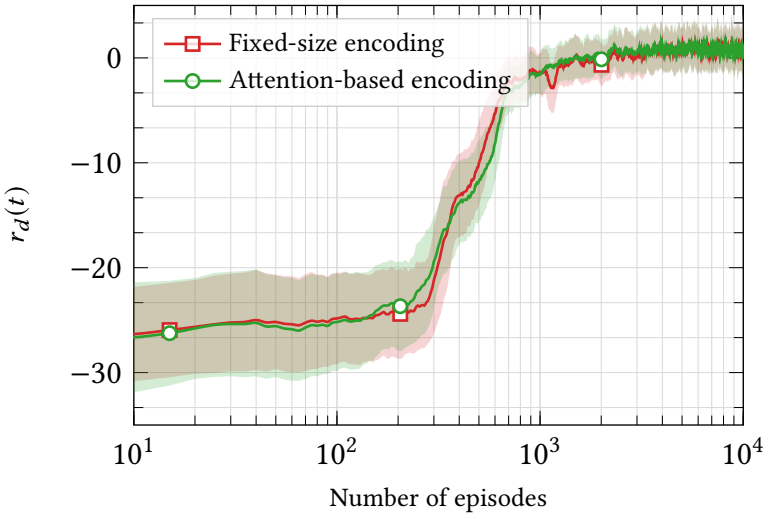
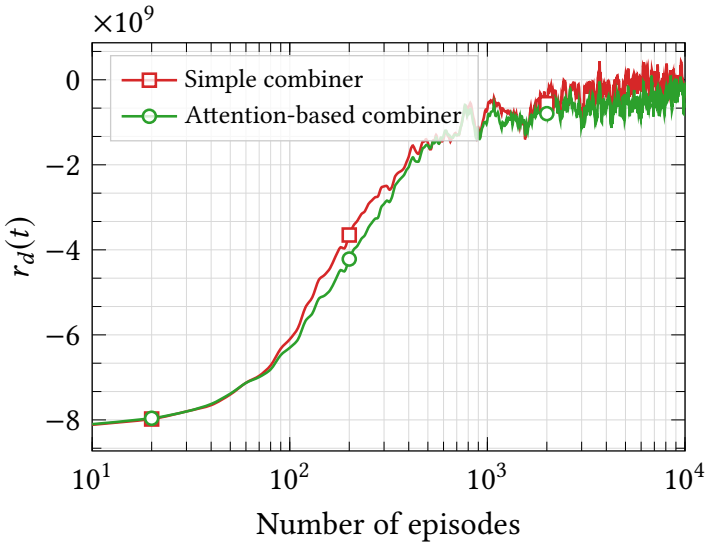


Figure 4.7: Fixed-size encoding Vs. attention-based encoding. We use a simple combiner.

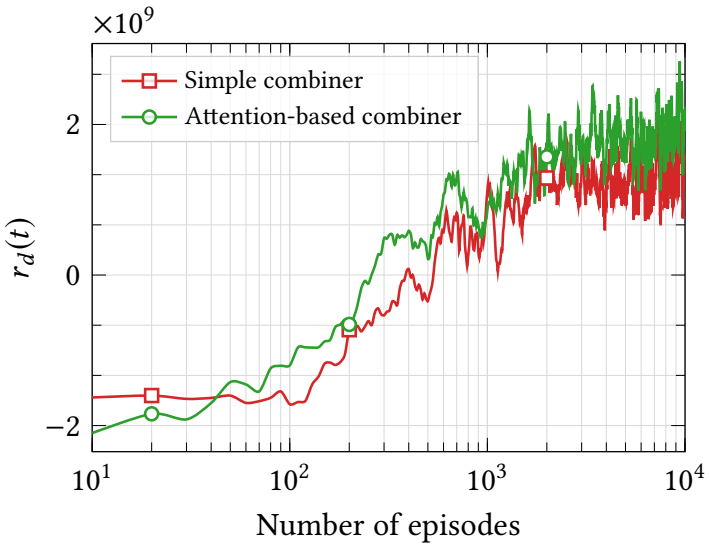
4.5.1.3 Performance of the attention-based mechanisms

Attention-based encoding. In Figure 4.7, we evaluate the performance of the attention-based encoding in being able to capture information of the fixed-sized encoding in addition to favoring transferability.

We can see that with the attention-based encoding, we obtain the same learning curve as the fixed-size encoding, with a slight improvement in learning stability. Therefore, by empowering our proposed architecture with an attention mechanism, we gain transferability without loss in performance.



(a) Full-buffer traffic



(b) Dynamic network traffic

Figure 4.8: Simple combiner Vs. attention-based combiner. We use an attention-based encoding. The learning curves concern the sum-rate maximization problem, *i.e.*, we set $\alpha = 0$ in Eqn. (2.7).

Attention-based combiner. The attention-based combiner has a different role in our framework. It enables each UE to weigh the importance of local or global information. The simple combiner can be viewed as a particular case of the attention combiner, where we set $\boldsymbol{\beta}_j^T = [0.5, 0.5]$ in Eqn. (4.9). Figure 4.8 shows the learning curves for sum-rate maximization in two scenarios corresponding to full-buffer traffic, *i.e.*, $D_j(t) = \infty$ (see Figure 4.8a) and dynamic network traffic, *i.e.*, $D_j(t) \neq \infty$ and the traffic dynamic follows a Poisson Distribution (see Figure 4.8b). Whereas the attention-based combiner and the simple combiner exhibit almost the same performance when there is no traffic (see Figure 4.8a), we can observe in Figure 4.8b, an improvement of the attention-based combiner over the simple one in

the case of dynamic traffic, which is more realistic. This is because, in our particular setting, UE's local observation is more informative than the global one as it embeds the UE's traffic request $D_j(t)$.

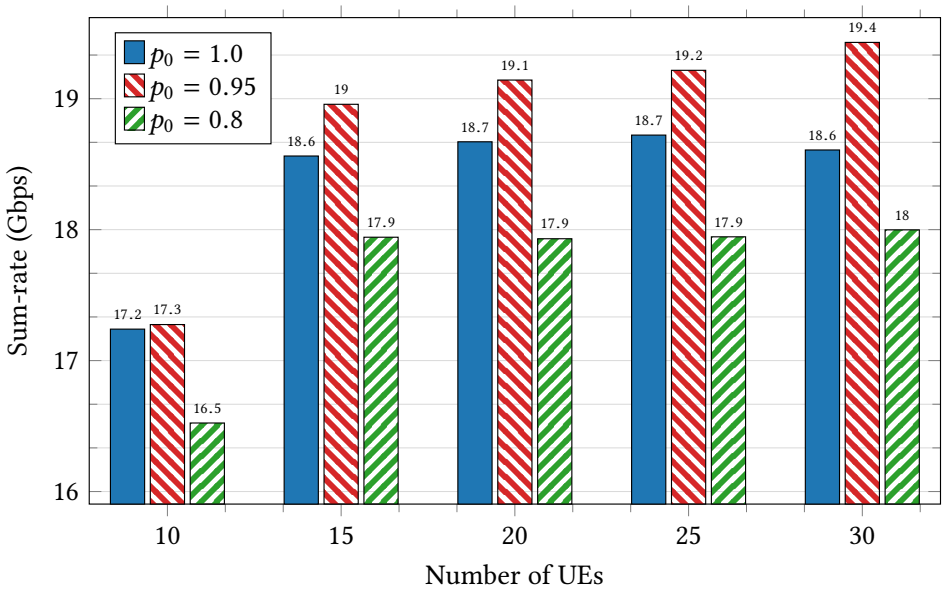


Figure 4.9: Effect of dropout mechanism for different p_0. The policy was trained with the following configuration $K_0 = 15$, $N_i = 3$, $\gamma = 0.9$. Then we evaluate the performance for different number of UEs.

4.5.1.4 Effect of UE dropout mechanism

Here, we evaluate the impact on the system performance of UE random dropout. Figure 4.9 shows the average network performance for different values of dropout rate p_0. For each dropout rate, we train the network for K_0 UEs and then evaluate the performance on a set of $K \in \{10, 20, 25, 30\}$ UEs. We can observe that by playing with the dropout probability p_0, one can improve the network performance. For instance, by taking $p_0 = 0.95$ instead of $p_0 = 1$ (*i.e.*, no dropout), we observe 4% performance improvement when $K = 30$. However, as we decrease p_0 to 0.8, the performance decreases as well to 3% compared to $p_0 = 1$. This is mainly because decreasing p_0 also increases the variance as shown in Figure 4.4, leading to a large discrepancy between episodes. In conclusion, a p_0 close but not equal to 1 is beneficial in this scenario. Therefore, for the rest of the paper, we fix $p_0 = 0.95$.

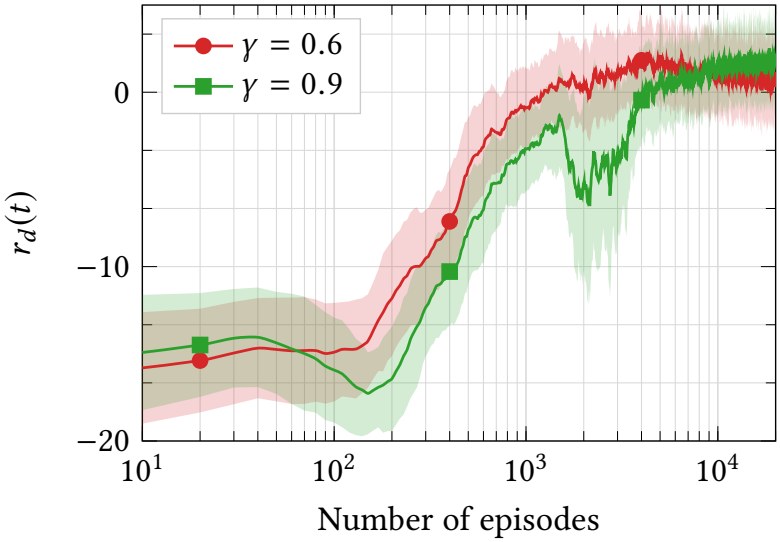


Figure 4.10: Impact of discounting factor γ.

4.5.1.5 Impact of the discounting factor γ

The discounting factor also impacts the learning convergence. Lowering γ accelerates the convergence to the detriment of performance. Increasing γ may improve the performance at the risk of miscoordination. As Figure 4.10 shows, when $\gamma = 0.6$, the convergence is much faster than for $\gamma = 0.9$, which eventually ends up yielding better performance.

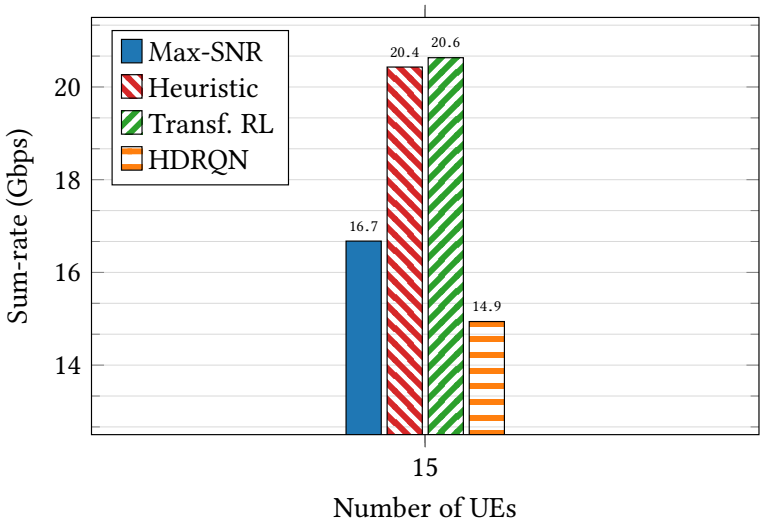


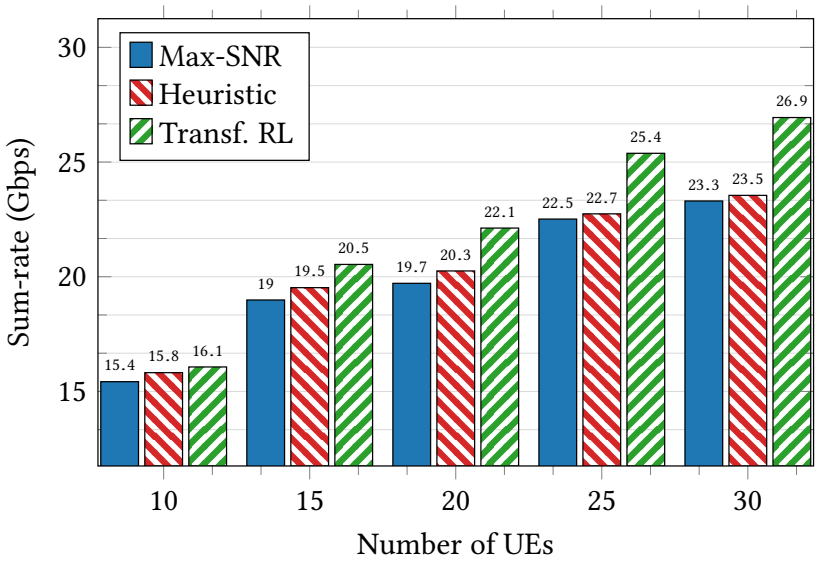
Figure 4.11: Comparison between the proposed transferable user association and the previously proposed solution based on hysteretic deep recurrent Q network (HDRQN).

4.5.2 Performance comparison

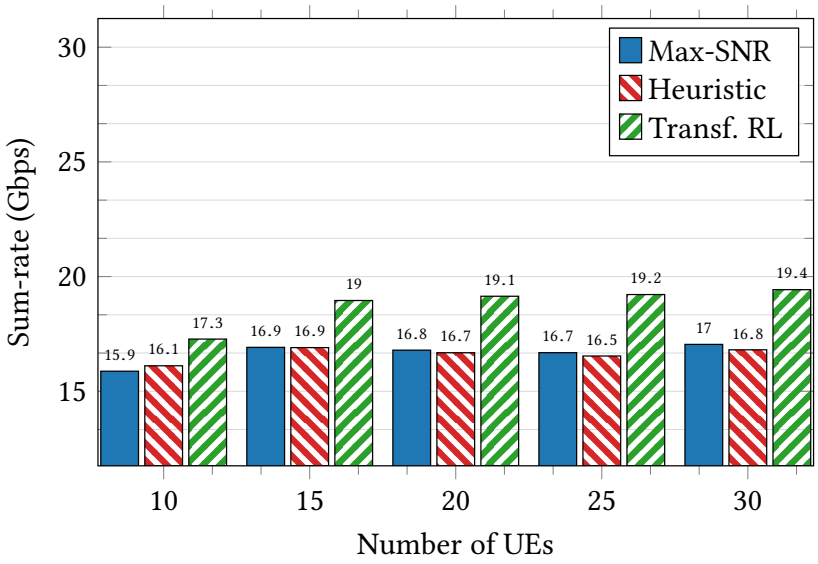
4.5.2.1 Comparison of the proposed solution *w.r.t.* previous works

Here, we aim to prove the advantage of the proposed transferable user association compared to solutions of the literature, including our previous Hysteretic Deep Recurrent Q-Network (HDRQN) algorithm for user association in Chapter 3. Recall that the HDRQN solution is conceived and optimized for

networks in which the UEs' position does not vary. Therefore, applying a policy trained for a given set of fixed UEs to a different network geometry does not work satisfactorily, in general. To illustrate this aspect, we train the HDRQN algorithm in a given deployment of 15 UEs and then evaluate its performance in 500 randomly chosen deployments. The average performance is showed in Figure 4.11 and compared with the new proposed transferable solution and the two baselines solutions. We can observe that the sum-rate performance of the HDRQN falls below the Max-SNR. This exemplifies that the HDRQN algorithm is deployment-specific and its generalization to scenarios with mobility is not straightforward. Indeed, in the HDRQN setting, a new training step is required whenever a new deployment is specified. In contrast, our new proposed solution is adapted to any deployment, even when the number of UEs varies and with zero-shot generalization capability. Moreover, we will show in the following that the performance of our proposed scheme outperforms even more tangibly the other state-of-the-art solutions, specifically when considering dynamic network traffic and UEs' mobility.



(a) $N_i = 15$ (*i.e.*, less collision events).



(b) $N_i = 3$ (*i.e.*, less collision events).

Figure 4.12: Generalization capability of the PNA *w.r.t.* K. Training configuration: ($K_0 = 15$ UEs, $N_i = 3$, $\forall i$). Testing configuration: N_i is kept fixed, K varies.

4.5.3 Policy transferability property

To assess how transferable the proposed algorithm is, we consider training the PNA for a reference number of users, $K_0 = 15$ and for a fixed number of beams per BS, $N_i = 3,\ \forall i$. Then we evaluate the performance of the algorithm for different network deployments with a variable number of UEs $K \in \{10, 20, 25, 30\}$, including changes in the UEs' position.

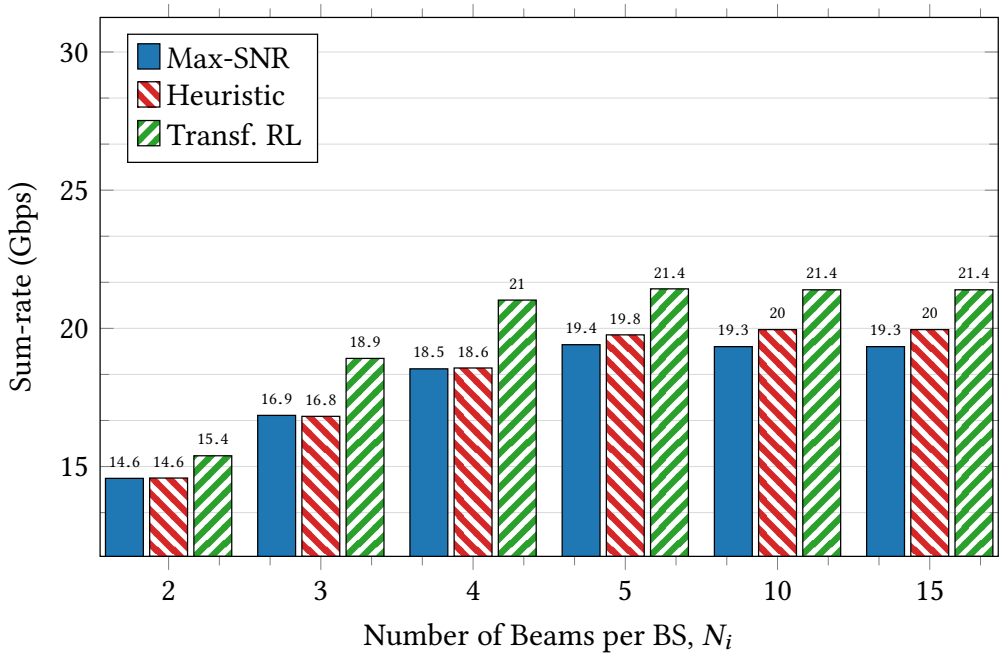


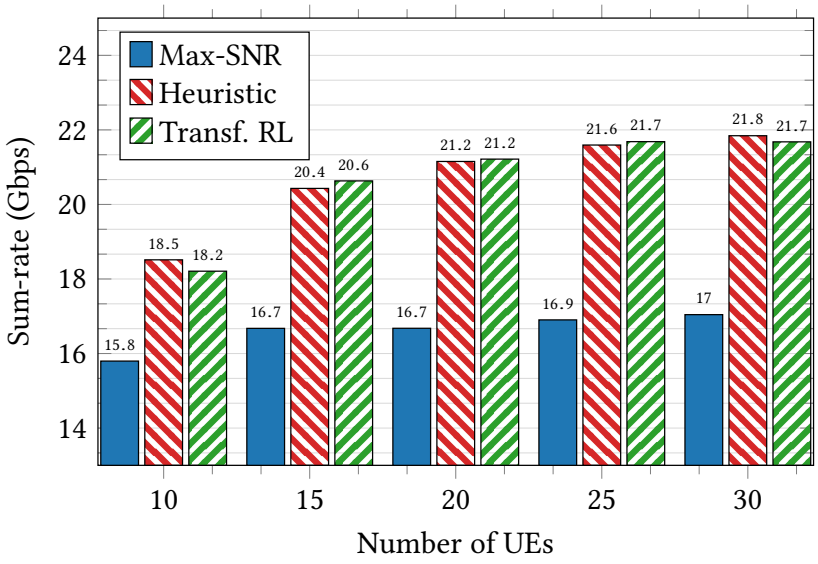
Figure 4.13: Generalization capacity of the PNA *w.r.t.* N_i. Training configuration: ($K_0 = 15$ UEs, $N_i = 3,\ \forall i$). Testing configuration: K is kept fixed and equal to K_0, N_i varies.

4.5.3.1 Zero shot generalization capacity

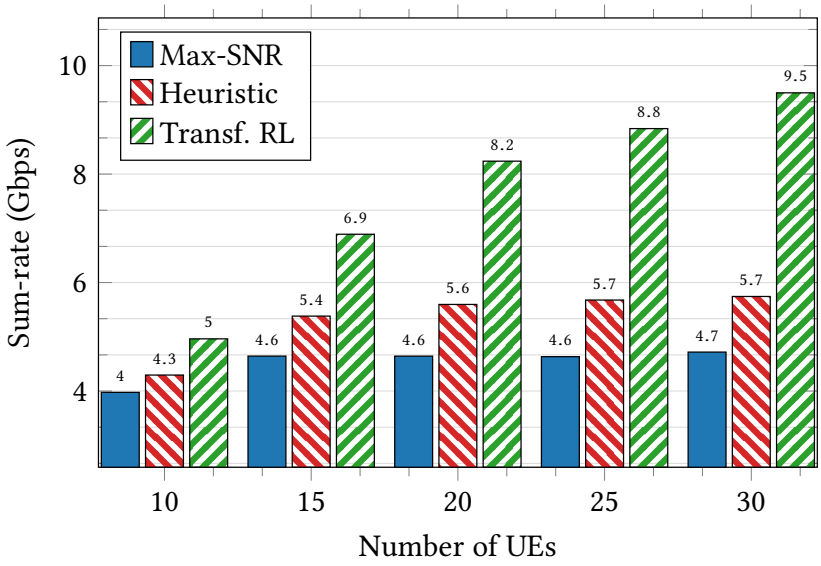
To evaluate the generalization capability of the proposed algorithm, we train the PNA to optimize the network sum-log-rate, *i.e.*, $\alpha = 1$. We remark that the proposed architecture can effectively and efficiently adapt to change in the number of UEs and the number of beams available per BS, without requiring additional training steps. In particular, in Figure 4.12, when the number of UEs doubles *w.r.t.* the reference training point *i.e.*from $K_0 = 15$, to $K = 30$, the proposed transferable PNA exhibits 14.5% and 15.5% increase in network sum-rate compared to max-SNR and the heuristic approach respectively. Moreover, an additional feature of the proposed architecture, is that even when the number of beams available per BS later changes, which impacts the collision events, the algorithm still adapts to maintain the system's performance. Indeed, in Figure 4.13 where we evaluate the performance of the algorithms for different $N_i \in \{2, 3, 4, 5, 10, 15\}$, we can observe that as N_i increases, implying less and less collisions since K is fixed, the algorithm keeps outperforming the two benchmarks. When N_i becomes greater than 5, *i.e.*, $\sum_{i=1}^{3} N_i > K = 15$, there is no improvement in the sum-rate as there are enough beams to serve all UEs.

4.5.3.2 Performance *w.r.t.* network traffic

Now we evaluate the system performance *w.r.t.* network traffic. Here again, the PNA is trained for $K_0 = 15$ to optimize the network sum-rate ($\alpha = 0$). Figure 4.14a shows the case of full-buffer traffic (*i.e.*, $D_j(t) = \infty$) and Figure 4.14b the case of dynamic traffic. We remark that in case of full-buffer traffic,



(a) Full-buffer traffic.



(b) With dynamic traffic.

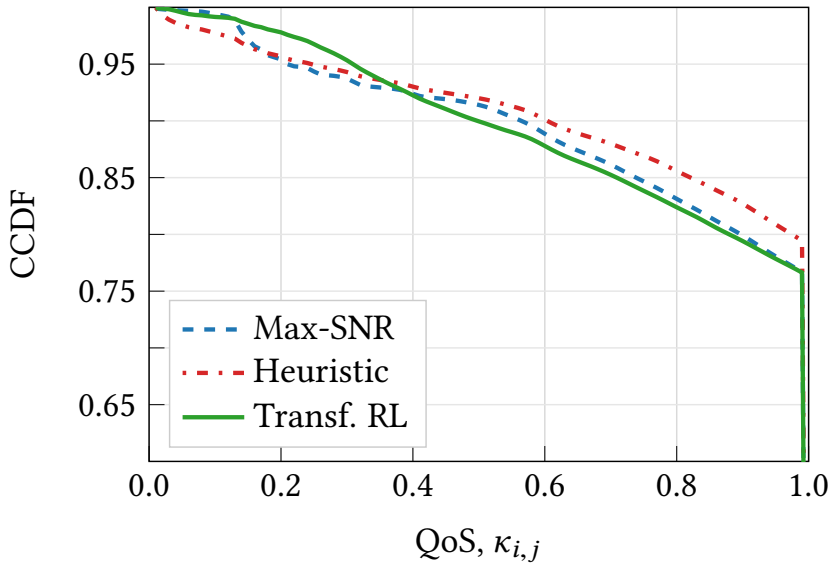
Figure 4.14: Performance of the proposed solution *w.r.t.* network traffic.

the proposed method performs better than the two benchmarks but performs slightly worse[8] than the heuristic algorithm when generalized to $K = 10$ and $K = 30$. However, when we consider the network traffic, the proposed transferable solution clearly outperforms the two benchmarks, yielding 102.1%, 66.66% network sum-rate increase for $K = 30$, *w.r.t.* the max-SNR and heuristic algorithms, respectively.

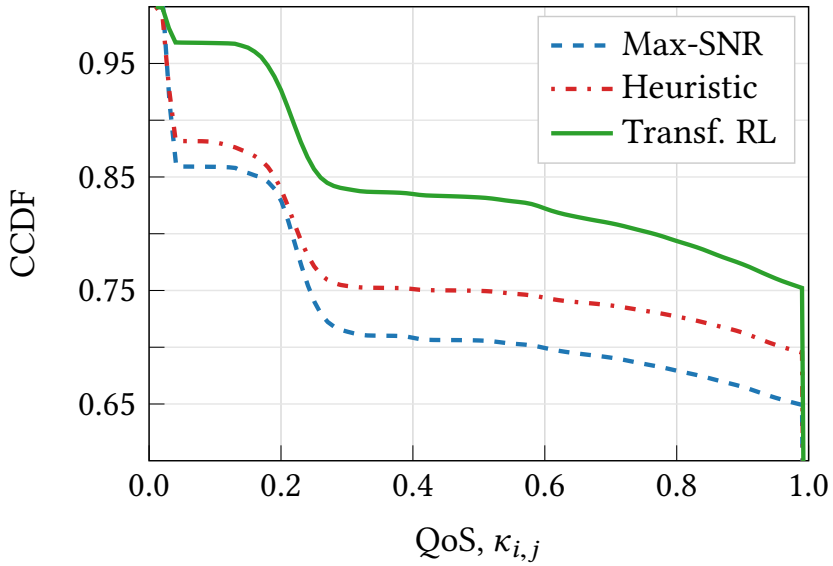
4.5.3.3 QoS satisfaction

In this section, we evaluate the QoS satisfaction of users (denoted $\kappa_{i,j}$ in Eqn. 2.7) for a network of $K = 15$ UEs under two configurations: $N_i = 15$, $\forall i$ and $N_i = 3$, $\forall i$ beams. Figure 4.15 shows the Complementary Cumulative Distribution Function (CCDF) of the QoS satisfaction of UEs. We can remark that when $N_i = 15$, all algorithms exhibit almost the same performance. When $N_i = 3$, *i.e.*, the

[8]Note that when the network dynamic deviates too much from the reference training point, it is possible to retrain the policy, *e.g.*, by utilizing a *curriculum learning* approach [89], where the already trained policy can be used as a starting point.



(a)



(b)

Figure 4.15: UEs' QoS satisfaction when (a) $N_i = 15$, and (b) $N_i = 3$. Here, we consider $\alpha = 0$.

association becomes complex as there are fewer beams than UEs, the performance of the two baselines algorithm falls down in comparison to our proposed solution. For example, 75% of the UEs get fully satisfied with our solution, whereas only 65% is satisfied with the Max-SNR algorithm and 70% with the heuristic approach. However, it is worth noting that UE's satisfaction does not necessarily reflect the global network performance we are interested in. Indeed, two UEs can experience the same QoS satisfaction, whereas they do not have the same contribution to global network performance. For example, for two UEs with a data request of 100 Mbps and 1 Gbps respectively, if there is only one beam, our algorithm will give more importance to the most demanding UE as it contributes most to the global network objective.

4.6 Conclusion and Perspectives

In this chapter, we investigated the problem of transferability of user association policies for 5G and beyond networks. To this end, we proposed a policy network architecture and a learning mechanism that enable users to learn a robust and transferable user association policy. The latter is adapted to withstand the environment dynamics, including fast fading, evolving traffic requirements, and time-varying number and position of UEs. Our proposed solution is based on deep multi-agent reinforcement learning, where agents leverage local and possibly global observations to optimize a network utility function. With our proposed novel architecture, the learned policy has zero-shot generalization capabilities, and can directly be transferred to new incoming UEs, which can start making decisions without requiring additional training steps. Moreover, our solution is flexible as it can be implemented in a centralized, distributed, or hybrid way. Numerical results showed that the proposed solution can achieve large network sum-rate gains especially when we consider network traffic and mobility, indeed, doubling the network sum-rate compared to baseline approaches available in the literature. Eventually, our proposed framework does not only apply to the user association problem but can be exploited to solve other complex radio resource management problems involving decision making.

In the next chapter, we will investigate how the proposed solution can be exploited for uplink communications for dynamic computation offloading enabled by edge computing, which involves optimization of both radio and computing resources.

The technical contributions of this chapter have been validated by the following paper and patent.

[C3] **M. Sana**, N. di Pietro, and E. Calvanese Strinati, *"Transferable and Distributed User Association Policies for 5G and Beyond Networks,"* IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Virtual, Sept. 2021.

[P2] **M. Sana**, N. di Pietro, E. Calvanese Strinati, and B. Miscopein, *"Method for associating user equipment in a cellular network according to a transferable association policy,"* Filed in September 30, 2020, FR2009989.

Chapter 5

Application to Dynamic Computation Offloading

"On ne naît pas tout fait."

"We are not born ready-made."

– Zoseph Ki-Zerbo (1922 – 2006)

Contents

5.1 Introduction

In this chapter, we study the problem of dynamic computation offloading for energy-efficient edge computing. In this problem, the User Equipments (UEs) continuously generate data (possibly with unknown statistics), which require buffering before transmission and processing at a distant Edge Server (ES) through a set of Access Points (APs). In general, this process introduces a queuing delay both from a communication and computation point of view. Accordingly, meeting the quality of service of UEs requires usually imposing average or probabilistic latency constraints on the queuing delay. The resulting problem is challenging, as it requires effective management of limited radio and computation resources in complex and time-varying environments, including channels' dynamic and UEs' mobility. In this work, we focus on the energy-delay trade-off. To this end, UEs, APs, and ES exploit *low-power sleep operations*: they can activate sleep states in which they cannot communicate and/or calculate for a limited period, thus consuming less energy. In this context, our goal is to minimize the long-term system energy consumption under strict end-to-end delay constraints at each UE. By using Lyapunov stochastic optimization tools, we show that this long-term optimization can be reduced to a per slot problem, where solving the latter in a *per slot fashion* guarantees the expected long-term goal. Moreover, we show that the new problem can be decoupled into a CPU scheduling and a user association problem. We efficiently and optimally solve the former using a fast iterative algorithm and hinge on our proposed scheme in Chapter 4 to solve the latter.

5.1.1 Motivations

Wireless communication networks are experiencing an unprecedented revolution, evolving from pure communication systems towards a tight integration of communication, computation, caching, and control [3]. Such a heterogeneous ecosystem requires a flexible network design and orchestration, able to accommodate, on the same network infrastructure, all the different services with their requirements in terms of energy, latency, and reliability. This requires an enhancement of the radio access network, *e.g.*, through the adoption of millimeter-wave (mmWave) communications, densification of APs, and flexible management of the physical layer [90]. In addition, the deployment of computing and storage capabilities at the network edge will enable network function virtualization and fast processing of the myriad of data collected by sensors, cars, mobile devices, etc. For this, Multi-Access Edge Computing (MEC) was conceived to enable energy-efficient, low-latency, highly reliable services by bringing cloud resources close to the users. In this context, *dynamic computation offloading* allows resource-poor devices to transfer application execution to ESs to reduce energy consumption and/or latency. From a network management perspective, this task is complex and requires the joint optimization of radio and computation resources. To address this problem, we introduce L2OFF: Learning to Offload, a framework built on top of the transferable user association policy architecture proposed in Chapter 4 that successfully addresses the problem of computation offloading.

5.1.2 Related work

The problem of dynamic computation offloading has received wide attention from academia and industry [91]. In [92], a scheduling strategy is proposed to counterbalance task completion ratio and throughput, hinging on Lyapunov optimization. [93] aims at minimizing the long-term average delay under a long-term average power consumption constraint. In [94], the long-term average energy consumption of a MEC network is minimized under a delay constraint, using a MEC sleep control. Also, [95] minimizes the energy consumption under a mean service delay constraint, optimizing the number of active base stations and the ESs' computation resource allocation, leveraging sleep modes for APs and ESs. In [96], Lyapunov optimization is used to reduce the energy consumption of a fog network, guaranteeing an average response time. In [97], the authors exploit Lyapunov optimization, Lagrange multipliers, and

sub-gradient techniques are exploited to optimize devices' and APs' energy consumption under latency constraints, with AP sleep states.

Recently, the advances of Machine Learning (ML) and Deep Reinforcement Learning (DRL) in wireless networks have opened up new possibilities for low-complexity and efficient algorithms for MEC [3], especially when model-based optimization is challenged by the difficulty or even impossibility of deriving mathematical models that accurately predict the networks' behavior. In this sense, the authors of [98] propose to couple model-based Lyapunov optimization with model-free DRL and formulate a sum-rate maximization problem under queue stability and long-term device energy constraints. However, their reference scenario considers a single AP, and no CPU scheduling is optimized at the ES. [99] also considers the same approach, intending to minimize the sum of the power consumption of the edge nodes, and a cost charged by a central cloud to help the edge node in processing the tasks under stability constraints. However, they do not consider the energy consumption of end UEs and APs.

5.1.3 Contributions

The contribution of this chapter is as follows:

- *A long-term energy minimization problem*: we consider a scenario in which multiple UEs perform computation offloading and compete for radio and computation resources in a network with many APs deployed with one ES, all exploiting low-power sleep operation modes. In this work, we target to minimize the long-term system's cost measured in terms of money spent on energy consumption. Accordingly, we treat the underlying problem as a long-term system energy minimization under strict delay constraints. We do not assume any knowledge of radio channels and data arrival statistics. Despite this, we come out with an online solution, which in each time slot, optimizes the UE-AP association in a distributed way using Multi Agent Reinforcement Learning (MARL), and the ES's CPU scheduling via a fast iterative algorithm whose solution's complexity scales linearly in the number of UEs. The resulting framework provides near-optimal performance.

- *Lyapunov meets distributed reinforcement learning*: we combine the convenience of a model-based solution that exploits Lyapunov stochastic optimization, with the power of model-free solutions based on MARL, aiming at energy-efficient computation offloading from an overall network perspective.

- *A unified framework for joint radio and computing resource management*: compared to the state-of-the-art works, the originality of our strategy lies in the capability of *simultaneously*: *i*) minimizing the duty cycles of all the network elements under delay constraints; *ii*) effectively managing radio interference; *iii*) being low-complexity; *iv*) combining Lyapunov optimization with DRL; *v*) being distributed and compatible with UE's mobility. The latter point, in sharp contrast with [98], results from the *zero-shot generalization* capability of our solution: it optimizes the learned computation offloading policy for all possible deployments of UEs *using attention neural networks*, and adapts when the number of UEs differs from the initial training point.

The technical content of this chapter is based on the published conference paper [100].

The remainder of this chapter is organized as follows. Section 5.2 introduces the system model and formulates the computation offloading problem as a long-term optimization. Section 5.3 details the proposed solution to efficiently address the formulated problem. We provide numerical results in Section 5.4 and draw conclusions in Section 5.5.

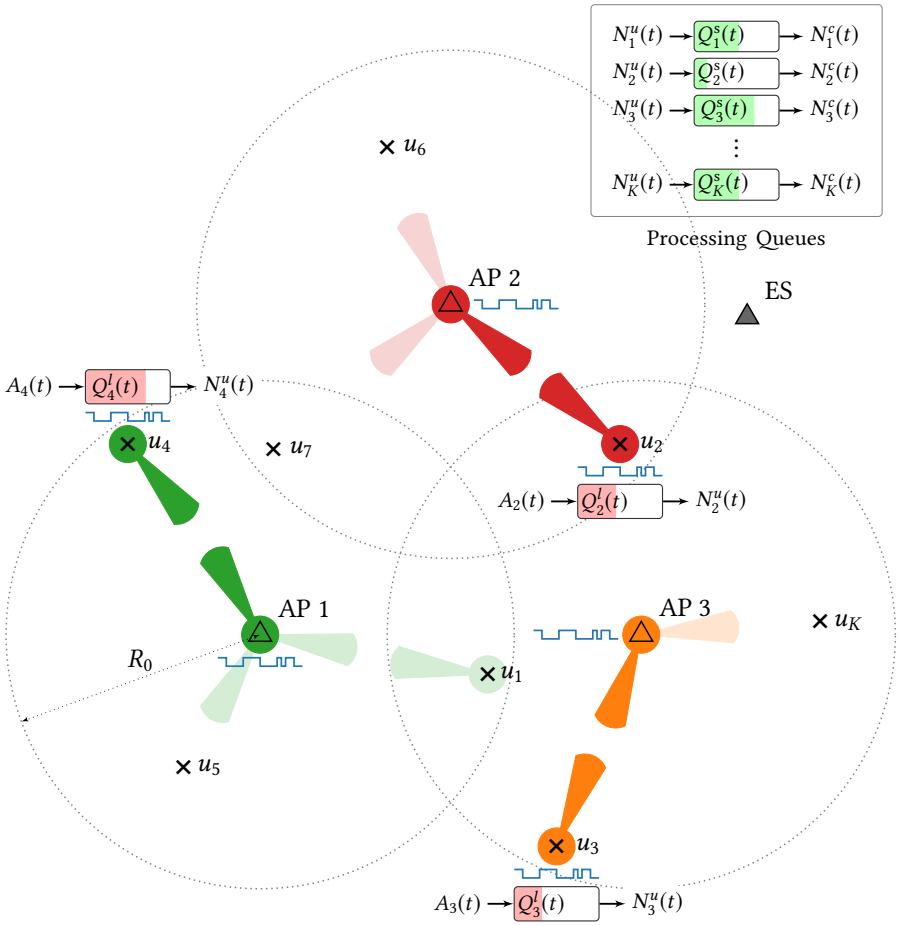


Figure 5.1: Network model with one Es, 3 APs deployed with K UEs.

5.2 Energy-Efficient Edge Computing: System Model

We consider the scenario depicted in Figure 5.1, where K UEs offload computational tasks to an ES, via one out of N possible APs. Let $\mathcal{U}$ and $\mathcal{A}$ be the sets of UEs and APs, respectively. Also, let $\mathcal{A}_j$ be the set of APs UE j can be associated with, which depends on the UE's coverage. In this dynamic system, we divide time into slots of equal duration τ. Specifically, we assume that a fraction $\beta \in (0, 1)$ of each slot is devoted to controlling signaling and $(1 - \beta)$ to data transmission from the UEs to the ES through APs and to data computation at the ES, which can simultaneously occur because they operate on separate data. At each time slot t, the dynamic of the radio channels and data arrivals at the UEs varies with *a priori* unknown statistics. Consequently, the achievable data rate over the radio channels and the computation rate at the ES vary with time. These variations also depend on UEs' mobility and the dynamic of the interference resulting from the communication of multiple UEs that we describe in the sequel.

5.2.1 Radio access and data rate model

In this work, we consider uplink communications for computation offloading. More specifically, we assume Spatial Division Multiple Access (SDMA). The APs serve the UEs over the same time-frequency resources but with different beams. In this scenario, uplink communications are affected by both intra- and inter-cell interference. Indeed, suppose that UE j is served by AP i at time t. Let $p_j^{u,\mathrm{Tx}}(t)$ be the uplink transmit power of UE j, $G_{j,i}^{\mathrm{Ch}}(t)$ the channel gain between UE j and AP i, $G_{j,i}^{\mathrm{Tx}}(t)$ the transmit antenna gain towards the direction of AP i, $G_{j,i}^{\mathrm{Rx}}(t)$ the receive antenna gain, B the allocated bandwidth,

and N_0 the noise power spectral density. Then, the Signal-to-Noise-plus-Interference Ratio (SINR) is given by

$$\mathrm{SINR}_j(t) = \frac{p_j^{u,\mathrm{Tx}}(t) G_{j,i}^{\mathrm{Tx}}(t) G_{j,i}^{\mathrm{Ch}}(t) G_{j,i}^{\mathrm{Rx}}(t)}{\mathcal{I}_{j,i}(t) + N_0 B}, \tag{5.1}$$

where $\mathcal{I}_{j,i}(t) = \sum_{j' \in \mathcal{U} \setminus \{j\}} p_{j'}^{u,\mathrm{Tx}}(t) G_{j',i}^{\mathrm{Tx}}(t) G_{j',i}^{\mathrm{Ch}}(t) G_{j',i}^{\mathrm{Rx}}(t)$ is the overall interference. Then, the achievable rate of UE j at time t is given by the Shannon formula as $R_j(t) = B \log_2(1 + \mathrm{SINR}_j(t))$.

If the j-th UE's offloadable data unit is encoded into S_j bits, the number of data units transmitted in the uplink direction at time t is

$$N_j^u(t) = \left\lfloor \frac{(1-\beta)\tau R_j(t)}{S_j} \right\rfloor. \tag{5.2}$$

Here, $\lfloor \cdot \rfloor$ denotes the Floor operator.

5.2.2 Computation model

We assume that the ES has one core CPU, for which UEs compete for the CPU time in each time slot. In particular, given a CPU core frequency $f_c(t)$ (measured in CPU cycles per second), each UE is allocated a portion $f_j(t)$ of $f_c(t)$ such that $\sum_{j=1}^{K} f_j(t) \leq f_c(t)$. Then, denoting by J_j the number of processed data units per CPU cycle, the number of data units processed over one slot is

$$N_j^c(t) = \lfloor (1-\beta)\tau f_j(t) J_j \rfloor. \tag{5.3}$$

5.2.3 Delay and queuing model

In our setting, computation offloading involves two steps: *i*) an uplink transmission phase of input data from the UEs; *ii*) a computation phase at the ES. New data units are continuously generated from an application at the UE's side and consequently offloaded and processed at the ES. In particular, generated data are queued locally at the UEs, then uploaded to the ES through one AP with time-varying data rate as in Eqn. (5.2). At the ES, received data are queued, waiting to be processed with a time-varying computational rate as in Eqn. (5.3). Thus, we represent the overall system through a simple queuing model involving both queues, synthetically depicted in Figure 5.1. Accordingly, each data unit experiences two different delays: *i*) a communication delay, including buffering at the UE; *ii*) a computation delay, including buffering at the ES. As shown later, we take into account these two delays jointly, as in [101]. UE j's uplink communication queue evolves as

$$Q_j^l(t+1) = \max\left(0, Q_j^l(t) - N_j^u(t)\right) + D_j(t), \tag{5.4}$$

where $D_j(t)$ is the number of newly arrived offloadable data units generated by the application that runs at the UE at time t. It is the realization of a random process whose statistics are unknown *a priori*. The remote computation queue at the ES evolves as

$$Q_j^s(t+1) = \max\left(0, Q_j^s(t) - N_j^c(t)\right) + \min\left(Q_j^l(t), N_j^u(t)\right). \tag{5.5}$$

End-to-end delay constraints. We know from Little's law that given a stationary queueing system, the average overall service delay is proportional to the average queue length [102]. Then, in our system, the overall delay is directly related to the sum of the uplink communication queue and the computation queue $Q_j^{\mathrm{tot}}(t) = Q_j^l(t) + Q_j^s(t)$. In particular, if $\bar{D}_j = \mathbb{E}\left\{D_j(t)/\tau\right\}$ is the average data unit arrival rate, the long-term average end-to-end delay L_j^{avg} experienced by a data unit generated by UE j is given by the ratio between the average of Q_j^{tot} and the average arrival rate. Thus, our first aim is to guarantee a long-term average delay L_j^{avg}, which gives the following constraint:

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\left[Q_j^{\mathrm{tot}}(t)\right] \leq Q_j^{\mathrm{avg}} = L_j^{\mathrm{avg}} \bar{D}_j, \quad \forall j. \tag{5.6}$$

Note that $\bar{D}_j$ is not known *a priori*, but can be estimated online *e.g.*, using moving average window.

5.2.4 Energy consumption model

We exploit low-power operation modes at UEs, APs, and ES: they can activate sleep states in which they cannot communicate and/or calculate for a limited period, thus consuming less energy. However, due to control signaling, UEs, APs, and ES are active for at least $\beta\tau$ seconds in each slot, consuming active power. Hence, we model the energy consumption of each entity as follows:

5.2.4.1 UE Energy Consumption.

Let $x_{j,i}(t) \in \{0,1\}$ be an association variable such that $x_{j,i}(t) = 1$ if and only if UE j offloads its data via AP i at time t, and $x_{j,i}(t) = 0$ otherwise. Also, Let $p_j^{\mathrm{u,off}}$ and $p_j^{\mathrm{u,on}}$ be UE j's sleep and active power, respectively. Then, the total UEs' energy consumption at time t is:

$$\mathrm{E_u}(t) = \sum_{j\in\mathcal{U}} \tau\left[(1-\beta)\left(I_j^{\mathrm{u}}(t)\left(p_j^{\mathrm{u,on}} + p_j^{\mathrm{u,Tx}}(t)\right) + (1 - I_j^{\mathrm{u}}(t))p_j^{\mathrm{u,off}}\right) + \beta p_j^{\mathrm{u,on}}\right], \tag{5.7}$$

where $I_j^{\mathrm{u}}(t) = \max_{i\in\mathcal{A}_j}\{x_{j,i}(t)\}$ indicates if UE j is active or not. Indeed, in a given time slot, a UE j can decide to not associate with any AP, hence, to not transmit. In this case, $I_j^{\mathrm{u}}(t) = 0$, and $p_j^{\mathrm{u,Tx}}(t) = 0$.

5.2.4.2 AP Energy Consumption.

Let $p_i^{\mathrm{a,off}}$ and $p_i^{\mathrm{a,on}}$ be the *i-th* AP's sleep and active power consumption, respectively. The total APs' energy consumption at time t is

$$\mathrm{E_a}(t) = \sum_{i=1}^{N} \tau\left[(1-\beta)\left(I_i^{\mathrm{a}}(t)p_i^{\mathrm{a,on}} + (1 - I_i^{\mathrm{a}}(t))p_i^{\mathrm{a,off}}\right) + \beta p_i^{\mathrm{a,on}}\right], \tag{5.8}$$

where $I_i^{\mathrm{a}}(t) = \max_{j\in\mathcal{U}}\{x_{j,i}(t)\}$ indicates whether AP i is active ($I_i^{\mathrm{a}}(t) = 1$) or not ($I_i^{\mathrm{a}}(t) = 0$).

5.2.4.3 ES Energy Consumption.

To reduce the energy consumption, we adopt both a low-power sleep mode for the ES, when no computation is performed at a given slot t, and a scaling of the CPU frequency $f_c(t)$, when the CPU is active and computing [103]. Namely, the CPU core consumes a power $p_{\mathrm{s}}^{\mathrm{on}}$ in active state, and a power $p_{\mathrm{s}}^{\mathrm{off}} < p_{\mathrm{s}}^{\mathrm{on}}$ in sleep state. When the ES is active, the dynamic power spent for computation is $p_{\mathrm{s}}^{\mathrm{c}}(t) = \kappa f_c^3(t)$, where κ is the effective switched capacitance of the processor [104]. In particular, we assume that $f_c(t)$ can be dynamically selected from a finite set $\mathcal{F} = \{0, \ldots, f_{\max}\}$. Therefore, the ES's energy consumption at time t is

$$\mathrm{E_s}(t) = (1-\beta)\tau\left(I_{\mathrm{s}}(t)\left(p_{\mathrm{s}}^{\mathrm{on}} + p_{\mathrm{s}}^{\mathrm{c}}(t)\right) + (1 - I_{\mathrm{s}}(t))p_{\mathrm{s}}^{\mathrm{off}}\right) + \beta\tau p_{\mathrm{s}}^{\mathrm{on}}, \tag{5.9}$$

where $I_{\mathrm{s}}(t) = \mathbb{1}_{f_c(t)>0}$, with $\mathbb{1}_{f_c(t)>0}$ the indicator function, which equals 1 if $f_c(t) > 0$ and 0 otherwise. Hence, $I_{\mathrm{s}}(t)$ indicates whether the ES is active ($I_{\mathrm{s}}(t) = 1$) or not ($I_{\mathrm{s}}(t) = 0$). From (5.7), (5.8), (5.9), the total system energy consumption at time t is $\mathrm{E_{tot}}(t) = \mathrm{E_s}(t) + \mathrm{E_a}(t) + \mathrm{E_u}(t)$. Next, our objective function is a convex combination of UEs, APs, and ES energy consumption:

$$\mathrm{E_w}(t) = \alpha_1\mathrm{E_u}(t) + \alpha_2\mathrm{E_a}(t) + \alpha_3\mathrm{E_s}(t), \tag{5.10}$$

with $\sum_{i=1}^{3}\alpha_i = 1$. Different α_i lead to different strategies. For example, $\alpha_1 = 1$ models a *user-centric* strategy, where only UEs' energy consumption is optimized. $\alpha_i = 1/3, \forall i$ yields a *holistic* strategy that includes the whole network's energy.

Remark 6. *From an optimization point of view, we can drop the fraction β from equations ($\beta = 0$). This is possible since we do not optimize β as the fraction of time dedicated to signaling and controlling is predefined and fixed. Accordingly, for simplicity, in the following, we consider $\beta = 0$ when deriving equations.*

5.2.5 Proposed long-term energy minimization problem

Following the above definitions, we formulate the following minimization problem on the weighted network energy consumption, subject to (5.6) and instantaneous constraints on the optimization variables:

Long-term energy minimization problem under end-to-end delay constraints.

$$\underset{\{\Psi(t)\}}{\text{minimize}} \quad \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[\mathrm{E_w}(t)\right], \tag{$\mathcal{P}_0$}$$

$$\text{subject to} \quad \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[Q_j^{\mathrm{tot}}(t)\right] \le Q_j^{\mathrm{avg}}, \qquad \forall j; \tag{C1}$$

$$x_{j,i}(t) \in \{0,1\}, \qquad \forall j,i, \tag{C2}$$

$$\sum_{j\in\mathcal{U}} x_{j,i}(t) \le N_i, \qquad \forall i, \tag{C3}$$

$$\sum_{i\in\mathcal{A}_j} x_{j,i}(t) \le 1, \qquad \forall j, \tag{C4}$$

$$f_j(t) \ge 0, \qquad \forall j,t, \tag{C5}$$

$$f_c(t) \in \mathcal{F}, \qquad \forall t, \tag{C6}$$

$$\sum_{j\in\mathcal{U}} f_j(t) \le f_c(t), \qquad \forall t, \tag{C7}$$

where $\Psi(t) = \left[\{x_{j,i}(t)\}_{j,i}, f_c(t), \{f_j(t)\}_j\right]$ and the expectation in Eqns. ($\mathcal{P}_0$) and (C1) is taken with respect to the random input data unit generation and radio channels, whose statistics are unknown. The constraint (C1) is the delay constraint. The constraint (C2) highlights that the UE-AP association variables are binary. The constraints (C3) and (C4) respectively ensure that the number of UEs assigned to each AP cannot exceed a maximum N_i UEs, and that each UE is assigned to at most one AP. Finally, the constraints (C5)-(C7) indicate that the computation frequencies assigned to each user are non negative and their sum cannot exceed the total CPU frequency of the ES, chosen from the finite set $\mathcal{F}$.

Directly solving the problem ($\mathcal{P}_0$) is very challenging due to i) the unavailability of the statistics, ii) non-convexity and NP-hardness of the problem in particular due to binary variables, iii) the long-term nature of the objective function as well as the delay constraint (C1). Therefore, to address this problem, we hinge on Lyapunov stochastic optimization tools [105].

5.3 Lyapunov meets MARL for Energy Efficient Edge Computing

5.3.1 A Lyapunov-aided problem decomposition

To handle the constraint (2.10), following [105], we introduce *virtual queues* $Z_j(t)$, which evolve as

$$Z_j(t+1) = \max(0, Z_j(t) + Q_j^{\mathrm{tot}}(t+1) - Q_j^{\mathrm{avg}}), \quad \forall j \in \mathcal{U}. \tag{5.11}$$

Here, $Z_j(t)$ is a state variable that measures how the system behaves *w.r.t.* the constraint (C1). In particular, it increases if the instantaneous value of $Q_j^{\text{tot}}(t)$ violates the constraint, and decreases otherwise. From [105], we know that the constraint (C1) is guaranteed if the virtual queue $Z_j(t)$, $\forall j$ is *mean rate stable, i.e.,*

$$\lim_{T\to\infty} \frac{\mathbb{E}\{Z_j(T)\}}{T} = 0. \tag{5.12}$$

To ensure this, we introduce the *Lyapunov* function $L(\mathbf{Z}(t))$ and the *drift-plus-penalty* function $\Delta_p(\mathbf{Z}(t))$, which we define as follows:

$$L(\mathbf{Z}(t)) = \frac{1}{2}\sum_{j=1}^{K} Z_j(t)^2, \tag{5.13}$$

$$\Delta_p(\mathbf{Z}(t)) = \mathbb{E}\left[L(\mathbf{Z}(t+1)) - L(\mathbf{Z}(t)) + \Omega \cdot \mathrm{E_w}(t)|\mathbf{Z}(t)\right]. \tag{5.14}$$

Here, $L(\mathbf{Z}(t))$ represents a measure of overall system's congestion, whereas the drift-plus-penalty function $\Delta_p(\mathbf{Z}(t))$ is the conditional expected variation of $L(\mathbf{Z}(t))$ over one slot, plus a penalty factor weighted by Ω, which trades off queue backlogs and the objective function of Eqn. ($\mathcal{P}_0$) [105].

Proposition 1. *If the radio channel states and the input data generation are i.i.d. over time slots, we obtain the* optimal *solution of Eqn. ($\mathcal{P}_0$) by optimally and jointly solving the following two sub-problems for a sufficiently high value of Ω.*
Sub-problem 1: CPU scheduling. *At time t, solve the following optimization problem:*

$$\begin{aligned}\underset{\{f_c(t),\{f_j(t)\}_j\}}{\text{minimize}} \quad & G_1(t) = \Omega\alpha_3 \mathrm{E_s}(t) \\ & + \sum_{j\in\mathcal{U}}\left[-2Q_j^{\mathrm{s}}(t)\tau f_j(t)J_j + \max\left(0, Q_j^{\mathrm{s}}(t) - \tau f_j(t)J_j + 1\right)Z_j(t)\right] \qquad (\mathcal{P}_1)\\ & \text{subject to} \quad \text{(C5)-(C7) } \textit{of } (\mathcal{P}_0).\end{aligned}$$

Sub-problem 2: UE-AP association. *At time t, solve the following optimization problem:*

$$\begin{aligned}\underset{\{x_{j,i}(t)\}_{j,i}}{\text{minimize}} \quad & G_2(t) = \Omega\cdot(\alpha_1\mathrm{E_u}(t) + \alpha_2\mathrm{E_a}(t)) + \sum_{j\in\mathcal{U}}\left[\left(-\frac{3}{2}Q_j^{l}(t) + Q_j^{\mathrm{s}}(t)\right)N_j^{u}(t)\right. \\ & \left. + \max\left(0, Q_j^{l}(t) - N_j^{u}(t)\right)Z_j(t)\right] \qquad (\mathcal{P}_2)\\ & \text{subject to} \quad \text{(C2)-(C4) } \textit{of } (\mathcal{P}_0).\end{aligned}$$

Sketch of proof. From [105], we know that an algorithm, which minimizes the drift-plus-penalty function $\Delta_p(\mathbf{Z}(t))$ in Eqn. (5.14) under the constraints (C2)-(C7), which we later refer to as the *drift-plus-penalty algorithm*, guarantees that the virtual queues $Z_j(t)$'s are mean rate stable and therefore also guarantee that the constraint (C1) is satisfied. However, directly minimizing $\Delta_p(\mathbf{Z}(t))$ is complex due to its non-convexity and non-differentiability. Hence, we hinge on the concept of Γ-approximation of the drift-plus-penalty algorithm.

Definition 3. *For a given constant Γ, a Γ-additive approximation of the drift-plus-penalty algorithm is one that, for a given state $\mathbf{Z}(t)$ at slot t, chooses a (possibly randomized) action $\mathbf{\Psi}(t)$ that yields a conditional expected value of the objective function in Eqn. (5.14) that is within a constant Γ from the infimum over all possible control actions.*

Hence, following this concept of Γ-approximation, our policy proceeds by minimizing a proper

upper bound of the drift-plus-penalty (5.14) to "push" the queues towards lower congestion states, *i.e.* towards system stability. In particular, it can be shown that (5.14) enjoys the following upper-bound:

$$\begin{aligned}\Delta_p(\mathbf{Z}(t)) \leq \zeta + \mathbb{E}\Bigg\{ \sum_{j=1}^{K} \Bigg[& \chi_j(t) - 2Q_j^{\mathrm{s}}(t)\tau f_j(t) J_j \\ & + \Big(\max\Big(0, Q_j^{\mathrm{s}}(t) - N_j^{c}(t)\Big) + \max\Big(0, Q_j^{l}(t) - N_j^{u}(t)\Big)\Big) Z_j(t) \\ & + \left(-\frac{3}{2} Q_j^{l}(t) + Q_j^{\mathrm{s}}(t)\right) N_j^{u}(t) \Bigg] + \Omega \cdot \mathrm{E_w}(t) \Big| \mathbf{Z}(t) \Bigg\}, \end{aligned} \tag{5.15}$$

where $\zeta > 0$ is a constant and $\chi_j(t)$ does not depend on the optimization variables. We defer full derivations of Eqn. (5.15) including expressions of ζ and $\chi_j(t)$ to Appendix C. Next, assuming that the radio channel states and the input data generation are *i.i.d.* over time slots and that $L(\mathbf{Z}(0)) < \infty$, greedily minimizing Eqn. (5.15) under (C2)-(C7) guarantee that the virtual queues are mean rate stable. Moreover from [105, Th. 4.8], we have also that:

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\left[\mathrm{E_w}(t)\right] \leq \mathrm{E_w^{opt}}(t) + \frac{\zeta + \Gamma}{\Omega}, \tag{5.16}$$

where $\mathrm{E_w^{opt}}(t)$ is the infimum time average energy achievable by any policy that meets the required constraints (C2)-(C7). Thus, the long-term solution of Eqn. ($\mathcal{P}_0$) is obtained by optimality minimizing Eqn. (5.15) for sufficiently large value of Ω. Finally, the decomposition into two sub-problems is straightforward because radio and computing optimization variables are decoupled in Eqn. (5.15) and can be treated independently. □

Summary. We summarize the objective of the above mathematical manipulations. First, to ensure the long-term delay constraint (C1), from [105], we need to guarantee the mean rate stability of the virtual queues. For this, it is sufficient to guarantee that the Lyapunov drift-plus-penalty function (5.14) is upper-bounded. Now, assuming that radio channel states and input data generation are *i.i.d.* over slots, we show that minimizing a proper upper bound of Eqn. (5.14) and letting $\Omega \longrightarrow \infty$ under the constraints (C2)-(C7) is equivalent to solving the problem ($\mathcal{P}_0$). Finally, this new problem can be cast into two sub-problems to be solved in a per slot fashion by observing that radio and computing optimization variables are decoupled in the derived bound (5.15).

5.3.2 Proposed fast iterative algorithm for CPU scheduling

Here, our aim is to propose an optimal algorithm to solve the CPU scheduling problem.

Lemma 3 (Maximum frequency constraint). *An optimal frequency scheduler should be such that any time t, for any UE j, it guarantees that* $f_j(t) \leq \min\left(\frac{Q_j^s(t)+1}{\tau J_j}, f_c(t)\right)$.

Proof. From Eqn. (5.5), $\max\left(0, Q_j^s(t) - N_j^c(t)\right)$ is the remaining data in the processing queue after each time slot. Now, we know that $N_j^c(t) = \lfloor \tau f_j(t) J_j \rfloor$ (where we have dropped β following Remark 6) thus, $\tau f_j(t) J_j - 1 \leq N_j^c(t) \leq \tau f_j(t) J_j$. Hence, we have

$$\max\left(0, Q_j^s(t) - N_j^c(t)\right) \leq \max\left(0, Q_j^s(t) - \tau f_j(t) J_j + 1\right).$$

Then, the proof is straightforward and follows by observing that for a given UE j, $Q_j^s(t) - \tau f_j(t) J_j + 1 < 0$ means that the allocated CPU frequency exceeds what is needed to empty the queue $Q_j^s(t)$, which is inefficient. Indeed, at maximum, the allocated frequency is the one that empties the processing queue, *i.e.* $Q_j^s(t) - \tau f_j(t) J_j + 1 \geq 0$. Noting that $f_j(t) \leq f_c(t)$, $\forall j$ completes the proof. □

Then by injecting the maximum frequency constraint in Eqn. ($\mathcal{P}_2$) and replacing $\mathrm{E_s}(t)$ by its expression in Eqn. (5.9), we can write:

$$\begin{aligned} G_1(t) &= \Omega\alpha_3\tau\left(I_s(t)\left(p_s^{on} + p_s^c(t)\right) + (1 - I_s(t))p_s^{off}\right) \\ &\quad + \sum_{j\in\mathcal{U}}\left[-2Q_j^s(t)\tau f_j(t)J_j + \left(Q_j^s(t) - \tau f_j(t)J_j + 1\right)Z_j(t)\right] \\ &= \Omega\alpha_3\tau\left(I_s(t)\left(p_s^{on} + p_s^c(t) - p_s^{off}\right) + p_s^{off}\right) \\ &\quad + \sum_{j\in\mathcal{U}}\left[-\left(2Q_j^s(t) + Z_j(t)\right)\tau f_j(t)J_j + \left(Q_j^s(t) + 1\right)Z_j(t)\right]. \end{aligned} \tag{5.17}$$

Now minimizing $G_1(t)$ under the constraints (C5)-(C7) is equivalent to minimizing a new objective $\tilde{G}_1(t)$ under the same constraints, where

$$\tilde{G}_1(t) = \Omega\alpha_3\tau I_s(t)\left(p_s^{on} - p_s^{off} + \kappa f_c(t)^3\right) - \sum_{j\in\mathcal{U}}\left[\left(2Q_j^s(t) + Z_j(t)\right)\tau f_j(t)J_j\right]. \tag{5.18}$$

Here, $\tilde{G}_1(t)$ is obtained from $G_1(t)$ by dropping the terms, which do not depend on the optimization variables. Solution of the problem ($\mathcal{P}_1$) follows by first observing that if $f_c(t)$ is fixed, $\tilde{G}_1(t)$ is linear *w.r.t.* the optimization variables $\{f_j(t)\}_j$. Hence it can be solved using fast iterative algorithm with a complexity of at most $O(K \times |\mathcal{F}|)$ iterations.

Lemma 4. *Let us define $\tilde{Q}_j(t) = 2Q_j^s(t) + Z_j(t)$. At a given time t, the scheduler first needs to choose the frequency $f_c(t) \in \mathcal{F} = \{0, \ldots, f_{\max}\}$ to be used. If there exists a solution $f_c(t) \in \mathcal{F}\backslash\{0\}$, i.e. such that $f_c(t) > 0$ then necessarily we have:*

$$\frac{\Omega\alpha_3(p_s^{on} - p_s^{off})}{\sum_{j\in\mathcal{U}}\tilde{Q}_j(t)J_j} < f_c(t) \leq \min\left(\sqrt{\frac{\sum_{j\in\mathcal{U}}\tilde{Q}_j(t)J_j}{\kappa\Omega\alpha_3}}, f_{\max}\right). \tag{5.19}$$

Thus we only need to search $f_c(t)$ within this interval.

Proof. First note that,

- if $f_c(t) = 0$, then $I_s(t) = 0 \iff f_j(t) = 0,\ \forall j \in \mathcal{U} \implies \tilde{G}_1(t) = 0$.
- if $f_c(t) > 0 \implies \tilde{G}_1(t) = \Omega\alpha_3\tau\left(p_s^{on} - p_s^{off} + \kappa f_c(t)^3\right) - \sum_{j\in\mathcal{U}} \tilde{Q}_j(t)\tau f_j(t)J_j$

Thus, there exists a solution $f_c(t) > 0$ iff $\Omega\alpha_3\tau\left(p_s^{on} - p_s^{off} + \kappa f_c(t)^3\right) - \sum_{j\in\mathcal{U}} \tilde{Q}_j(t)\tau f_j(t)J_j < 0$.

$$\begin{aligned}\iff \Omega\alpha_3\kappa f_c(t)^3 &< \sum_{j\in\mathcal{U}} \tilde{Q}_j(t)f_j(t)J_j - \Omega\alpha_3\left(p_s^{on} - p_s^{off}\right)\\ &< f_c(t)\sum_{j\in\mathcal{U}} \tilde{Q}_j(t)J_j - \Omega\alpha_3\left(p_s^{on} - p_s^{off}\right) \quad \text{as } f_j(t) \le f_c(t)\end{aligned}$$

Since $f_c(t) > 0$, it implies $f_c(t)\sum_{j\in\mathcal{U}} \tilde{Q}_j(t)J_j - \Omega\alpha_3\left(p_s^{on} - p_s^{off}\right) > 0 \iff f_c(t) > \frac{\Omega\alpha_3(p_s^{on} - p_s^{off})}{\sum_{j\in\mathcal{U}} \tilde{Q}_j(t)J_j}$.
Also, as $p_s^{on} > p_s^{off}$ we have,

$$\Omega\alpha_3\kappa f_c(t)^3 < f_c(t)\sum_{j\in\mathcal{U}} \tilde{Q}_j(t)J_j,\ \text{thus},\ f_c(t) \le \min\left(\sqrt{\frac{\sum_{j\in\mathcal{U}} \tilde{Q}_j(t)J_j}{\kappa\Omega\alpha_3}},\ f_{\max}\right).$$

□

Algorithm 4: ES CPU Scheduling

1 In each time slot t, observe $Q_j^s(t)$, $Q_j^l(t)$, $Z_j(t)$, and compute $\tilde{Q}_j(t) = 2Q_j^s(t) + Z_j(t)$, $\forall j$.
2 Define a vector $\mathcal{F} = [0, \ldots, f_{\max}]$ of ES CPU frequencies available.
3 Define a matrix $F = \{F_{k,j}\}_{k,j}$ of size $|\mathcal{F}| \times K$, and a $|\mathcal{F}|$-sized vector $\mathcal{G}_1 = \{\mathcal{G}_1^k\}_{k=1\ldots|\mathcal{F}|}$.
4 Initialize F and $\mathcal{G}_1$ with zeros *i.e.*, set $F_{k,l} = 0\ \forall k, l$, and $\mathcal{G}_1^k = 0\ \forall k$.
5 **for** $k = 1, \ldots, |\mathcal{F}|$ **do**
6 Let $f_c^k(t) = \mathcal{F}_k$, and $\mathcal{U} = \{k = 1, \ldots, K\}$.
7 **while** $\frac{\Omega\alpha_3(p_s^{on} - p_s^{off})}{\sum_{j\in\mathcal{U}} \tilde{Q}_j(t)J_j} < f_c^k(t) \le \min\left(\sqrt{\frac{\sum_{j\in\mathcal{U}} \tilde{Q}_j(t)J_j}{\kappa\Omega\alpha_3}},\ f_{\max}\right)$ **do**
8 $\tilde{j} = \arg\max_{j\in\mathcal{U}} \left\{\tilde{Q}_j(t)J_j\right\}$.
9 $F_{k,\tilde{j}} = \min\left(\frac{Q_{\tilde{j}}^s(t) + 1}{\tau J_{\tilde{j}}},\ f_c^k(t)\right)$.
10 $\mathcal{U} = \mathcal{U}\backslash\{\tilde{j}\}$.
11 $f_c^k(t) = f_c^k(t) - F_{k,\tilde{j}}$.
12 **if** $\mathcal{U} = \emptyset$ **then**
13 break.
14 **end**
15 **end**
16 Compute the objective function $\mathcal{G}_1^k = \tilde{G}_1(t)$ of Eqn. (5.18) with $f_c(t) = \mathcal{F}_k$ and $f_j(t) = F_{k,j}$, $\forall j$.
17 **end**
18 Find $k^* = \arg\min_k \{\mathcal{G}_1^k\}$, and then set $f_c^*(t) = \mathcal{F}_{k^*},\ f_j^*(t) = F_{k^*,j}\ \forall j$.

The overall procedure to select the optimal CPU frequency $f_c(t)$ and the optimal scheduling frequencies $\{f_j\}_{j\in\mathcal{U}}$ is described in Algorithm 4. In particular, in Algorithm 4, steps 7-15 find the optimal CPU

resource allocation for a given $f_c(t)$ that minimize $\tilde{G}_1(t)$. For this, it iteratively allocates the maximum available CPU to the UE with the highest $\tilde{Q}_j J_j$. This is because of the sign minus in front of $\tilde{Q}_j J_j$ in Eqn. (5.18). This Algorithm 4 requires, in the worst case, at most $K \times |\mathcal{F}|$ iterations.

5.3.3 Proposed MARL framework for UE-AP association

Sub problem ($\mathcal{P}_2$) is more complex as it is non-convex and NP-hard [56]. However, note that it takes a similar form as the user association problem in previous chapters except for the change in the form of the objective function. Therefore, we propose to use our transferable solution presented in 4, where UEs, modeled as autonomous agents, learn to offload tasks over multiple episodes of random deployments to maximize a long-term γ-discounted reward $\sum_{\tau=t+1}^{T_e} \gamma^{\tau-t-1} r(\tau)$, where $r(t) = -G_2(t)$ is the common reward perceived by each UE at time t and T_e is the length of an episode. From a Lyapunov optimization perspective, the long-term goal (minimization of the long-term average energy) is guaranteed when Eqn. ($\mathcal{P}_2$) is solved optimally slot by slot. Here, this is achieved by myopically maximizing the instantaneous reward instead of the long-term reward, *i.e.*, by setting $\gamma = 0$.

Remark 7. *During an episode, $r(t)$ can drop to $-\infty$ due to the presence of queues in the expression of the objective $G_2(t)$, which is not bounded. To solve this problem, note that in a feasible scenario, the queues growing to infinity result from UEs deciding systematically to not offload (which is a wrong policy). Hence, we define two clipping values $Q_j^{\text{clip}} = (1+\alpha_1)Q_j^{\text{avg}}$ and $Z_j^{\text{clip}} = (1+\alpha_2)(Q_j^{\text{avg}})^2$, parameterized by α_1, α_2, which we consider as the maximum tolerable value of physical and virtual queues respectively, above which an episode terminates with a failure. In this way, we improve the learning convergence, as UEs are quickly notified of their failure.*

Thus, as in Chapter 4, let $\mathbf{o}_j^{\text{R}}(t)$ denote the set of "radio observations" of UE j:

$$\mathbf{o}_j^{\text{R}}(t) = \left\{ a_j(t-1), R_{j,a_j(t-1)}, R(t-1), \text{ACK}_j, \left\{\text{RSS}_{j,i}\right\}_{i\in\mathcal{A}_j}, \left\{\vartheta_{j,i}\right\}_{i\in\mathcal{A}_j} \right\}. \tag{5.20}$$

$a_j(t-1) \in \mathcal{A}_j$ denotes UE j's action (*i.e.*, connection request to an AP) at time $t-1$, $R_{j,a_j(t-1)}$ is the perceived rate, $R(t-1)$ the total network sum-rate, and ACK_j the received connection acknowledgment signal. $\left\{\text{RSS}_{j,i}\right\}_{I\in\mathcal{A}_j}$, $\left\{\vartheta_{j,i}\right\}_{i\in\mathcal{A}_j}$ indicate the received signal strength and corresponding angles of arrival (AoA) from UE j to AP n. Similarly, we denote with $\mathbf{o}_j^{\text{C}}(t)$ represents the set of "MEC observations", related to task offloading:

$$\mathbf{o}_j^{\text{C}}(t) = \left\{ (x_j, y_j), f_j(t), Q_j^l(t), Q_j^{\text{s}}(t), Z_j(t) \right\}, \tag{5.21}$$

where $Q_j^l(t)$, $Q_j^{\text{s}}(t)$, $Z_j(t)$ are the queues defined above, (x_j, y_j) are UE j's geographical coordinates, and $f_j(t)$ its allocated CPU frequency at the ES.

Learning to offload (L2OFF) computational tasks. To learn to offload computational tasks, our solution relies on the transferable solution proposed in Chapter 4. In this framework, after observing $\mathbf{o}_k^{\text{R}}(t)$, UE k builds its local state encoding $\mathbf{u}_k$, which represents its "perception" of the radio environment, using an encoding function $f(\cdot)$, *e.g.*, a neural network. Then, based on the aggregated MEC observations of its whole neighborhood, it constructs an encoding vector $\mathbf{v}_k$, which characterizes its perception of the network from a computation viewpoint. UE k then builds its overall context encoding vector $\mathbf{c}_k$ to represent its global understanding of the environment, using an encoding function $h(\cdot)$, *e.g.*, a concatenation operator or a neural network. For each UE, the goal of the MARL framework is to learn an association policy $\pi_{\boldsymbol{\theta}}$ with learnable parameters $\boldsymbol{\theta}$, where $\pi_{\boldsymbol{\theta}}(a_k(t)|\mathbf{o}_k(t)) = p_{a_k(t),k}$ indicates the probability of taking action $a_k(t)$ after observing $\mathbf{o}_k(t) = \{\mathbf{o}_k^{\text{R}}(t), \mathbf{o}_k^{\text{C}}(t)\}$. Note that the probability vector $\mathbf{p}_k(t) = [p_{0,k}, \ldots, p_{N,k}] \in [0,1]^{N+1}$, from which the action $a_k(t)$ of the UE will be sampled, is such that $\sum_{n\in\mathcal{A}} p_{n,k} = 1$ and $p_{n,k} = 0$ for all $n \notin \mathcal{A}_k$.

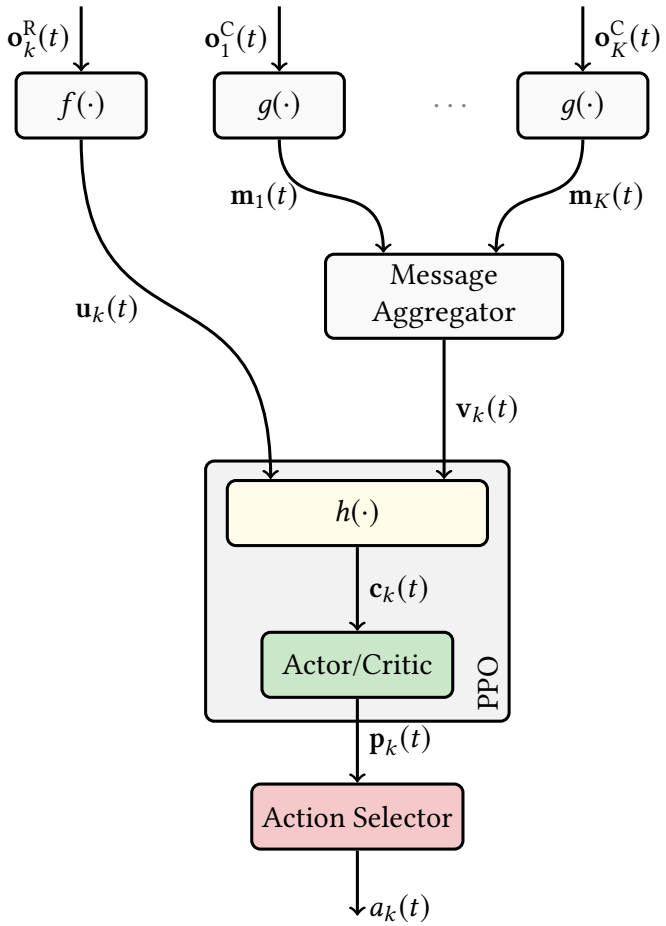


Figure 5.2: Dynamic computation offloading policy network architecture. A UE decides to offload its computation tasks based on its radio observations and after aggregating computation observations from its neighborhood, including its observations. All UEs share the same policy.

Message passing service. Let $\boldsymbol{\theta}_k, \boldsymbol{\theta}_q$, and $\boldsymbol{\theta}_v : \mathbb{R}^6 \times \mathbb{R}^m$ be learnable weights, describing the set of parameters of the encoding function $g(\cdot)$, which we later refer to as the *message generator*. For each UE l, let $\mathbf{k}_l = \boldsymbol{\theta}_k^T \mathbf{o}_l^{\mathrm{C}}(t)$, $\mathbf{q}_l = \boldsymbol{\theta}_q^T \mathbf{o}_l^{\mathrm{C}}(t)$, $\boldsymbol{\nu}_l = \boldsymbol{\theta}_v^T \mathbf{o}_l^{\mathrm{C}}(t)$, and $\mathbf{m}_l = \{\mathbf{q}_l, \boldsymbol{\nu}_l\}$ be the *key*, the *query*, the *value*, and the *message* associated with UE l. Then, each UE k, after aggregating the messages from its neighbors $\mathcal{N}_k$, computes its encoding vector $\mathbf{v}_k = \sum_{l \in \mathcal{N}_k} \alpha_{l,k} \boldsymbol{\nu}_l$, where the score $\alpha_{l,k}$ represents the interaction between UEs l and k (in achieving the underlying optimization goal). This score is calculated using dot-product attention mechanism [84]:

$$\alpha_{l,k} = \mathrm{softmax}\left(\left[\frac{\mathbf{q}_l \mathbf{k}_k^T}{\sqrt{m}}\right]_{l \in \mathcal{N}_k}\right). \tag{5.22}$$

Here, softmax$(\cdot)$ denotes the normalized exponential function. Note that computing $\mathbf{v}_k$ only involves the queries and the values from others UEs in $\mathcal{N}_k$ and not their keys, which are UE-specific and do not need to be transmitted. Such a message-passing service enables the *scalability* and the *transferability* of the learned policy, which is optimized for all possible UE deployments, in sharp contrast with [98], which requires fixed UEs. In other words, in our framework, a change in the number or position of UEs in the network does not require a new policy training and does not impact the architecture of the policy network. Only the number of exchanged messages varies, depending on the variation of a UE's neighborhood. Both, the input variables and the number of neurons of the encoding functions remain fixed. This enables *curriculum learning*, where a policy obtained from *e.g.* 6 UEs can be leveraged as a starting point to train another policy for $K > 6$ UEs. Finally, all the encoding functions, including the message generator, are optimized through end-to-end learning procedure using *proximal policy optimization* (PPO) and an actor-critic framework [87].

Table 5.1: Mobile edge computing parameters

UEs sleep and active power, γ	$p^{\mathrm{u,off}} = 0.346$ W, $p^{\mathrm{u,on}} = 0.9$ W
APs sleep and active power	$p^{\mathrm{a,off}} = 0.278$ W, $p^{\mathrm{a,on}} = 2.2$ W
ES sleep and active power	$p^{\mathrm{s,off}} = 10$ W, $p^{\mathrm{s,on}} = 20$ W
Target Signal-to-noise ratio (SNR)	15 dB
UEs max transmit power	0.1 W
Hysteretic parameter ϵ	$\epsilon_1 = 0.01$, $\epsilon_2 = 0.5$
Number of MLP neurons, m	128
Number of Monte-Carlo simulations, N	200

5.4 Simulation results

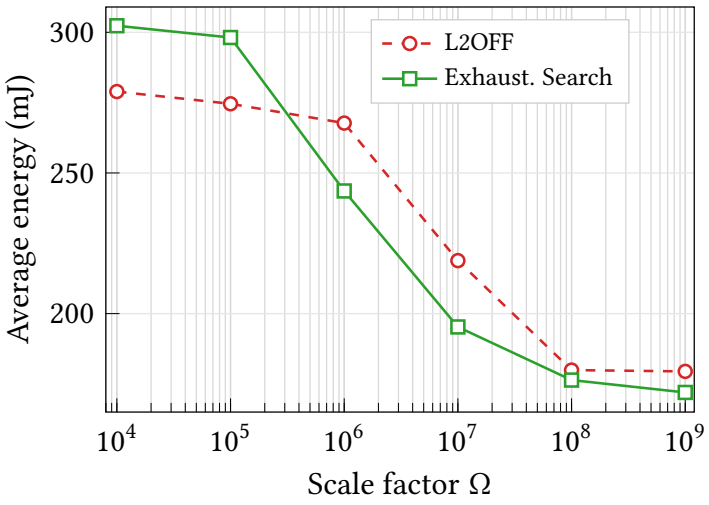
In this section, we assess the effectiveness of the proposed framework in a network of 3 APs operating at 28 GHz mmWave frequencies and for $K \in \{6, 9, 12, 15\}$ UEs. We use $p^{\mathrm{u,off}} = 0.346$ W, $p^{\mathrm{u,on}} = 0.9$ W, $p^{\mathrm{a,off}} = 0.278$ W, $p^{\mathrm{a,on}} = 2.2$ W, $p^{\mathrm{s,off}} = 10$ W, $p^{\mathrm{s,on}} = 20$ W. Each UE transmits with power $p^{\mathrm{u,Tx}}(t) = \min\left(p_j^{\mathrm{tg}}(t), p_{\max}\right)$ over a bandwidth $B = 10$ MHz, where $p_j^{\mathrm{tg}}(t)$ is the power to meet a predefined target SNR of 15 dB and $p_{\max} = 0.1$ W. Each slot lasts 10 ms and we set $\beta = 0.1$, $N_i = 15$, $\kappa = 10^{-27}$, $J_j = 10^{-3}$, $S_j = 1500$ bits $\forall j$ and $\mathcal{F} = \{0, 0.1, \ldots, 1\} \times f_{\max}$, where $f_{\max} = 10^9$ cycle/s. UEs' data generation rate follows a Poisson distribution with mean $D_j = 50 \times S_j$ bits $\forall j$. In our setup, all encoding functions are composed of one multi-layer perceptron (MLP) of $m = 128$ neurons with a rectifier linear unit (ReLu) activation. Both the actor and the critic module comprise $2m$ neurons and we empirically set the learning rate to 10^{-4}, $\alpha_1 = 10$ and $\alpha_2 = 0$. Simulation parameters are summarized in Table 5.1. Additional parameters, including pathloss and antenna diagrams, can be found in Table 3.1 of Chapter 3. To foster the learned policy and enable better generalization, during training, we consider random CPU scheduling[1]. This is possible since the problems ($\mathcal{P}_1$) and ($\mathcal{P}_2$) are completely decoupled, therefore, the policy learned to solve the problem ($\mathcal{P}_2$) must be independent of the ES frequency allocation. We compare our L2OFF solution to two benchmarks:

- Exhaustive search: at each t, we perform an exhaustive search over all possible solutions of ($\mathcal{P}_2$).
- Max-SNR: each UE is associated with a Bernoulli random variable with probability p of being in active state (which models the average duty cycle of UEs). Then, at each t, an active UE gets associated with the AP providing the maximum SNR.

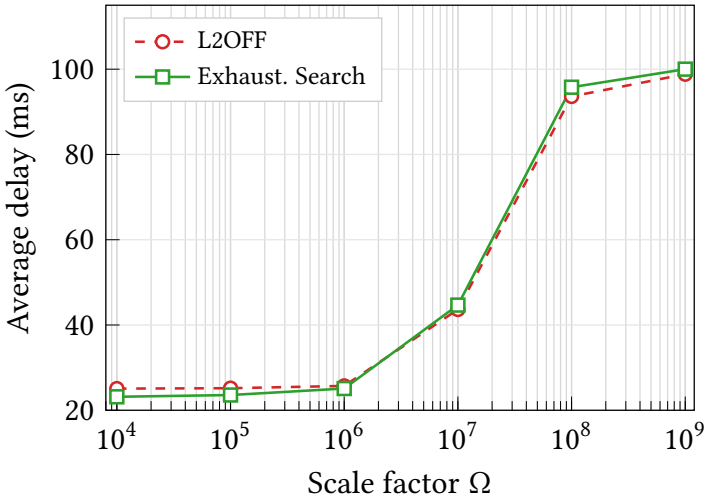
5.4.1 Energy-delay trade-off

Here, we evaluate the performance of our proposed framework for different values of Ω (cf. Eqn. (5.14)), and compare the results to the performance obtained via exhaustive search in Figure 5.3. First, we observe that our method can effectively adapts the duty cycle to minimize energy consumption. Indeed, the results in Figure 5.3 follow our theoretical expectations: when Ω increases, optimally solving the problems ($\mathcal{P}_1$) and ($\mathcal{P}_2$) lowers the duty cycle, consequently leading to a lower energy consumption. Meanwhile, the average delay increases and caps to 100 ms, which is the fixed delay constraint (C1). Interestingly, the proposed scheme exhibits performance close to exhaustive search approach (for $\Omega = 10^9$), reaching up to 96.5% of its performance, for the same delay constraint.

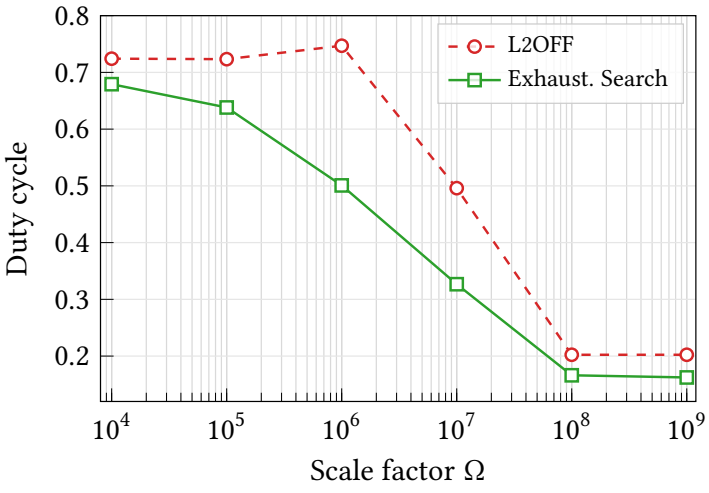
[1]We randomly select $f_c(t) \in \mathcal{F}$ and allocate $\omega_j f_c(t)$ to each UE j such that $\sum_j \omega_j = 1$, where $\{\omega_j\}_j$ follow a symmetric Dirichlet distribution.



(a) Average energy as $\Omega \to \infty$



(b) Average delay as $\Omega \to \infty$



(c) Average duty cycle as $\Omega \to \infty$

Figure 5.3: Energy-delay trade-off *w.r.t.* Ω for $K = 6$ UEs and for a fixed delay constraint of 100 ms.

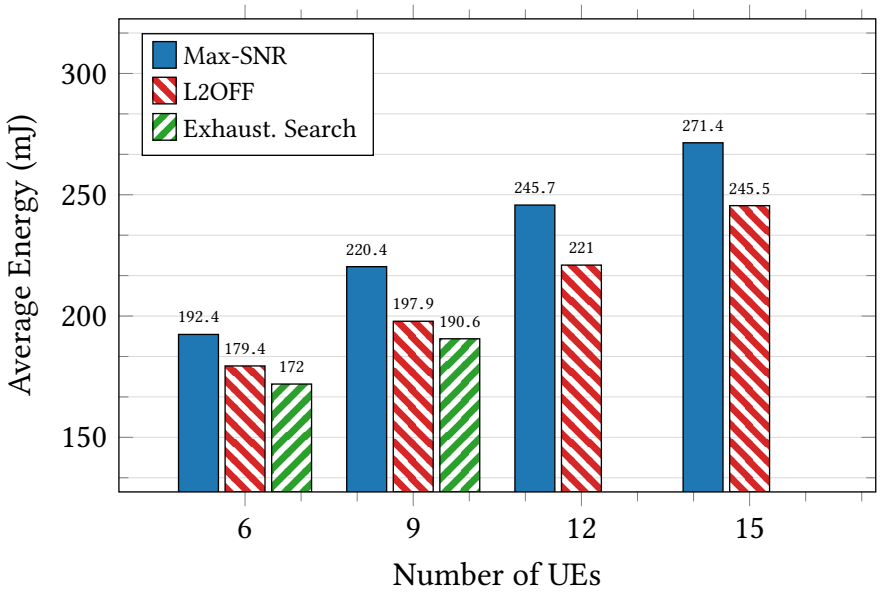


Figure 5.4: Average energy for a fixed average delay of 100 ms. Due to complexity, results for $K \in \{12, 15\}$ UEs cannot be obtained for the exhaustive search.

5.4.2 Performance comparison

To fairly compare the proposed framework with the heuristic based on Max-SNR, we first determine exhaustively the optimal lowest duty cycle that enables the Max-SNR algorithm to guarantee an average delay constraint of 100 ms. Then, the comparison is made for the same delay in Figure 5.4. We can notice how, even by optimally computing the duty cycle for the Max-SNR algorithm, our solution still outperforms, reducing the energy by 10% for 15 UEs compared to Max-SNR solution, as we consider interference, and intelligently orchestrate UEs. Moreover, under the same delay constraint, with our strategy, the network consumes 246 mJ on average for 15 UEs, whereas for the same energy consumption, the Max-SNR can only serve 12 UEs.

5.5 Conclusion and Perspectives

In this chapter, we proposed a novel approach for delay constrained energy-efficient dynamic computation offloading services in dense mmWave networks impaired by interference. We first formulated the computation offloading as a long-term optimization. Then, we applied Lyapunov optimization tools to split the problem into a CPU scheduling problem and a UE-AP association problem. While the first one is easily solvable via an efficient fast iterative algorithm, we solved the second one using multi-agent reinforcement learning with a *distributed and transferable* policy. The proposed solution reaches up to 96.5% of the optimal solution obtained via exhaustive search and can reduce energy consumption up to 10% compared to a heuristic approach based on SNR maximization.

Eventually, if direct information exchange is allowed between users, the performance of our proposed framework can be further improved, which would help to unwind confusing situations. Indeed, consider the example of two users with similar requirements located very close to each other. In this case, each of these users may have the same perception of the radio environment and thus observe the same inputs. As a result, these users may eventually converge to the same behavior, taking the same actions (or resulting in a ping-pong effect) as they share the same knowledge. Thus, in the absence of explicit communication between these users (or priority level), they will tend to connect to the same base station, hence, will experience strong interference from each other. In such a case, a good policy might be to let one of the users communicates or to connect both users to different base stations. However,

our proposed mechanism cannot induce such concurrent behavior because users share the same global knowledge. Therefore, in this scenario, an external arbitration is required. Although rare, this example illustrates the need for inter-agent communications in some situations to reach optimal convergence. However, due to bandwidth constraints, communications between agents must be limited. In addition, only relevant information, sufficient for agents to complete the targeted optimization task must be exchanged. This gives rise to a new paradigm: the *semantic communications*, which we explore in the next chapter as a new fundamental for beyond 5G networks.

The technical contributions of this chapter have been validated by the following paper.

[C4] **M. Sana**, M. Merluzzi, N. di Pietro, and E. Calvanese Strinati, *"Energy Efficient Edge Computing: When Lyapunov Meets Distributed Reinforcement Learning,"* IEEE International Conference on Communications (ICC) Workshops, Virtual, Montreal, Canada, June 2021.

Part III

Exploring new Fundamentals for beyond 5G Networks: The Opportunity of Semantic Communications

Chapter 6

Learning Semantics: An Opportunity for Effective 6G Communications

"There's no sense in being precise when you don't even know what you're talking about."

"Il ne sert à rien d'être précis quand on ne sait même pas de quoi on parle."

– John von Neumann (1903 – 1957)

Contents

6.1 Introduction

Recently, semantic communications are envisioned as a key enabler of future Sixth Generation (6G) networks. Back to Shannon's information theory, the goal of communication has long been to guarantee the correct reception of transmitted messages irrespective of their meaning or the intended goal. However, in general, whenever communication occurs to convey a meaning or to accomplish a goal, what matters is the receiver's understanding of the transmitted message or how the received message help in achieving the targeted goal of the communication (*e.g.*, completing a specified task) and not necessarily the correct reconstruction of transmitted messages. Hence, semantic and goal-oriented communications introduce a new paradigm: transmitting only relevant information sufficient for the receiver to capture the meaning intended or fulfill the targeted goal of communication (*e.g.*, dictated by the application). This can help in saving a lot of communication bandwidth. This chapter provides a global overview of the opportunity offered by semantic and goal-oriented communications for beyond 5G networks. To this end, we present and detail a novel architecture that enables representation learning of semantic symbols for effective semantic communications. We first discuss theoretical aspects and successfully design objective functions, which help learn effective semantic encoders and decoders. Eventually, we show promising preliminary numerical results for the scenario of text transmission, especially when sender and receiver speak different languages.

6.1.1 Motivations

Academia and industry have kicked off research on the future 6G of wireless networks. Speculation about the possible evolution of current 5G technology as well as radical new architectures, approaches, and technologies are being intensely discussed [3, 106, 107]. The expectation is that by 2030 first commercial 6G solutions will be available. This is driven by the current trend, witnessing an unprecedented demand for communication bandwidth to accommodate burgeoning new services like eXtended Reality (XR) or autonomous driving. To meet these challenges, historically in wireless communications, a solution has been to explore higher frequencies to benefit from the available large spectrum resources. Such solutions cyclically face an inevitable bottleneck, represented by the hardware's cost, complexity and energy efficiency of wireless communications. For example, as frequency increases new challenges arise in communication such as blockage, severe pathloss, atmospheric absorption, and power amplifier efficiency [108]. This calls for new paradigms shift for the effective design of 6G communications [2].

In addition, 6G will offer a radical step ahead to Artificial Intelligence (AI) in general and to Machine Learning (ML) in particular. ML and AI are already cornerstones of 5G, allowing to improve operational and service performance. However, 5G has not been designed specifically to support effective interactions between AI agents but rather to collect, exchange, and process data to feed ML applications. In contrast, 6G will be built on the native inclusion of AI as a fundamental component of the connect-compute-control network [109]. This will enable the intertwining of different kinds of intelligence (natural and artificial), requiring a radically new approach in the design of communication systems [2]. In our view, future 6G systems should be engineered to effectively recreate or infer the meaning of what has been communicated rather than to *"just"* optimize opaque data pipes that aim at reproducing exactly exchanged sequences of symbols [2]. Effective communication of *meanings* can be achieved through exchanges of *semantics*. Today fundamental open question to answer is *how to bring the notion of* semantic *from human understanding to machine understanding?* In our view, this requires a radically innovative approach to communications: the semantic and goal-oriented communications [2]. This approach can achieve a significant source data compression gain, which saves a lot of communication bandwidth.

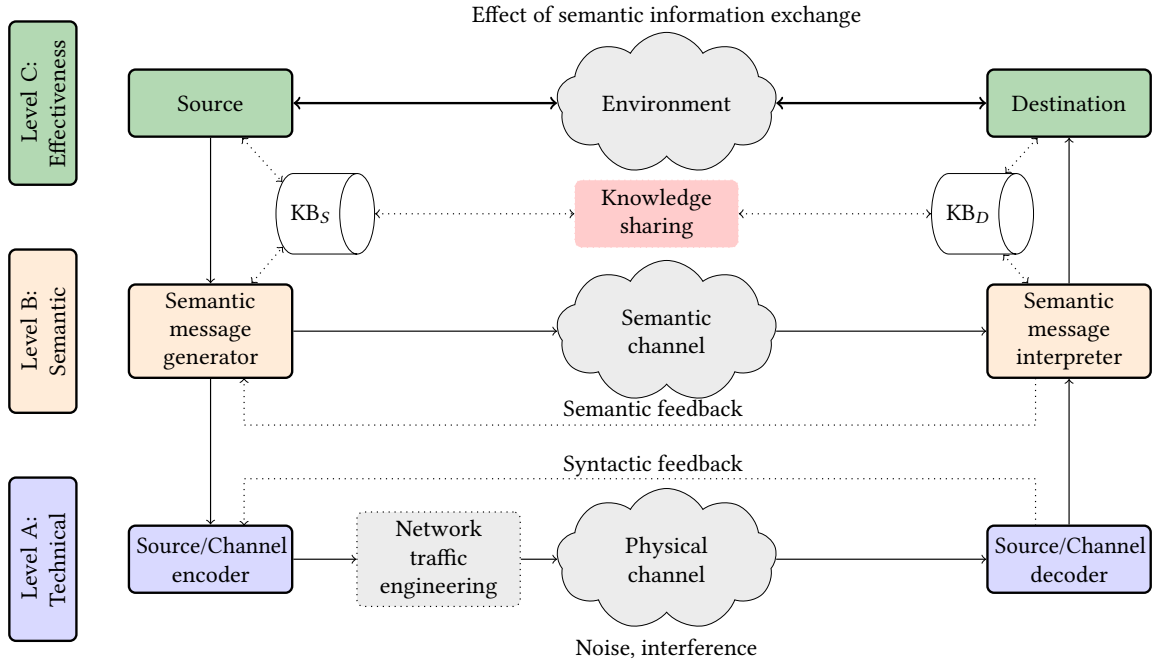


Figure 6.1: Multi-level communication system [2]. Here, KB_S and KB_D denote the knowledge base available at the source and destination, respectively.

6.1.2 Related work

In their seminal work [110, 111], Shannon and Weaver identified three levels of communication (see Figure 6.1):

- *Level A* - the technical problem: how accurately can the symbols of communication be transmitted?
- *Level B* - the semantic problem: how precisely do the transmitted symbols convey the desired meaning?
- *Level C* - the effectiveness problem: how effectively does the received meaning affect conduct in the desired way?

Shannon deliberately focused on the technical problem and the communication systems that we know so far are engineered to optimize the *level A* of communication. Then in 1953, Weaver provided a first attempt for the inclusion of semantics in the communication problem [111]. Bar-Hillel and Carnap provided also outlines of a theory of semantic information, focusing mainly on measuring how informative transmitted message is (informativity measurement) [112]. Recently with the growing interdependence between communication systems and AI, new attempts to include the *Level B* (the semantic problem) in the communication system has started [2, 113, 114, 115]. The ontology for *semantics* is still evolving in the state-of-the-art. In the Internet of Things (IoT) and semantic web contexts, semantic refers to the capability of enriching data and support interoperability mechanism between hardware and software applications belonging to different domains [116]. The semantic is indeed a way to associate documents, collected or processed data (a file, an image, a text, a sensed physical measure, etc.) to information and metadata. This enables to constitute of a knowledge-based decision-making database that can communicate in predefined semantic languages. In application-driven mechanisms, *semantic* refers to an abstraction at higher Open Systems Interconnection (OSI) layers, used for autonomous configuration and reconfiguration of network states leveraging on the principle of information-centric networking [117]. In contrast, as indicated in [2], the end goal of semantic

communication is different. Semantic communications are shaped to effectively compress the exchanged data between communicating parties, improve the communication robustness by incorporating semantic information to the classical *Level A* communication scheme.

This is possible by exploiting the knowledge shared a priori between communicating parties, such as a shared language or logic, shared background and contextual knowledge, and possibly a shared view on the goal of communication. In [113], the authors provide tentative definitions of semantic capacity, semantic noise, and a semantic channel from the perspective of Shannon's statistical measurement of information. Our work focuses on the potential benefits of semantic compression. In [118] the authors refers to semantic as the *semantics of information*, addressing the significance and usefulness of messages by considering the contextual attributes (semantics) of information [119]. In this approach, the Age of the Information (AoI) is key to identify the relevance of the semantic information for the effectiveness of the exchange between communicating parties. Nevertheless, AoI does not necessarily define the meaning of a message in many applications, but rather how a message is still pertinent for an application given its age. Indeed, different definitions of *semantic* carry different measures of semantic information.

Here in this chapter, we focus on semantic communications and particularly on the benefit of semantic compression. We refer to *semantic* as a "meaningful" message (a sequence of well-formed symbols, which are possibly learned from data) that have to be interpreted at the receiver. This requires a reasoning unit (natural or artificial) able to interpret based on a knowledge base: a symbolic knowledge representation of the specific application. Here, we focus on applications for AI and neural networks that exchange, communicate and intertwine. We do not apply this research only to level A, but we jointly design a full end-to-end (E2E) communication-intelligence chain with level A, jointly to Level B. This requires creating an overlay on top of Level A to enable interaction and communication between intelligent machines. In this context, an E2E neural architecture is presented in [114], enabling semantic transmission of sentences. However, their proposed architecture is limited in flexibility: they represent each word in a transmitted sentence with the same and fixed number of semantic symbols irrespective of the conveyed meaning. Authors in [120] apply the same architecture to speech signals transmission. Similarly, the work in [121] presents a deep source-channel coding scheme, which exploits hybrid automatic repeat request (HARQ) to reduce semantic transmission error.

6.1.3 Contributions

The contribution of this chapter is as follows:

- *An E2E semantic communication architecture*: we propose a novel E2E semantic communication architecture incorporating level B to classical level A communications. In this architecture, information from a binary source is encoded with semantic information extracted using neural attention mechanisms [84], to produce sequence of semantic symbols. In contrast to very recent state-of-the-art works [114, 120], which propose an E2E system for semantic text and speech transmission, we formally define a new loss function, which captures the effects of *semantic distortion* to communication. This enables to dynamically trade semantic compression losses with semantic fidelity [113] (*i.e.*, the semantic interpretation correctness).

- *An adaptive mechanism for dynamic semantic symbols representation at the source*: we design a semantic adaptive mechanism, which dynamically optimizes the number of symbols per semantic message based on the trade-off between semantic compression and semantic fidelity that we formally express.

- *A toy example in the scenario of text transmission*: we provide a detailed numerical evaluation that shows the benefits of our proposed adaptive E2E semantic system. Results are provided for the context of Natural Language Processing (NLP), especially when transmitter and receiver speak a different language. In this context, messages are formed and communication parameters are set

to maximize the correct interpretation of semantic messages rather than error-free bit decoding at the receiver.

The technical content of this chapter is based on the conference paper [122].

The remainder of this chapter is organized as follows. Section 6.2 introduces the basic concepts of semantic communications. Section 6.3 details the proposed E2E architecture for semantic representation learning. We provide numerical results in Section 6.4 and draw conclusions in Section 6.5.

6.2 Semantic Communications

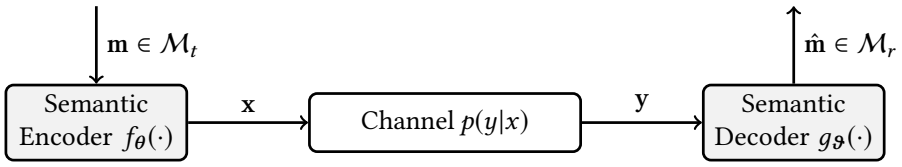


Figure 6.2: Simplified semantic communication system model.

6.2.1 General introduction

A semantic communication system defines a communication framework in which sender and receiver exchange *semantic information* to create a common understanding of exchanged messages.

> **Definition 4.** *We refer to the term* semantic information *as the meaning underlying the data (which can be discrete or continuous) that a sender wants to convey to a receiver.*

Example of data ranges from (random) numbers to texts, audios, images or videos. Formally defining the semantic content (or the meaning) of data is not a trivial task. In [123], the author proposes a definition of semantic content based on data as follows:

> **Definition 5.** *An instance of semantics is defined, if and only if:*
>
> - *the instance consists of at least one datum*
> - *the data are well-formed (i.e., data are organized in a correct way according to the rules (syntax) of a specific system)*
> - *the well-formed data is meaningful (i.e., the data must comply with the meaning of the chosen system, code or language).*

For example, human language uses a structured set of signs, gestures, writings, or words associated with real-world things or abstract thoughts, and rules to convey meanings. Each language has its own structure, which depends on the set of rules used to convey meaningful information. This definition of meaning and language can be extended to artificial languages, such as a computer programming language, after proper identification of language's concepts, rules, and constraints. Here, we focus on applications where AI agents exchange, communicate and intertwine. For this, we adopt *semantic symbols* as a means to represent semantics. Thus, in our scenario, agents exchange semantic symbols depending on the meaning associated with the exchanged data. To do so, agents can also rely on their respective knowledge base (KB), a symbolic knowledge representation of the specific application [124], possibly shared among agents, from which semantic symbols can be inferred or interpreted (see Figure 6.1). A knowledge base can be manually constructed or learned from the ontology, rules, constraints that

govern a specific application. This can be achieved using for *e.g.*, graph-based knowledge representation [124], where relationships, rules between entities, or concepts are organized over a graph.

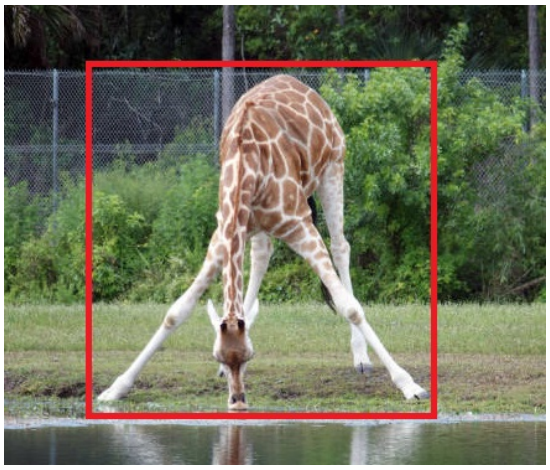


Figure 6.3: A giraffe drinking water.

Example 2 (A giraffe drinking water)**.** *Consider the following example of a picture of a giraffe drinking water (see Figure 6.3). A knowledge base of this picture can be represented as follows: i) this is an image of an animal, ii) the animal is a giraffe, iii) the giraffe is standing on grass, iv) the giraffe is drinking water, v) there is a fence behind the giraffe etc.*

The knowledge base extracted from one specific application can be used to solve related problems [124]. From the above example, to the question "is the giraffe in a zoo?", one may answer "yes" because "there is a fence behind the giraffe". However answering such a question requires reasoning capabilities (here for *e.g.*, finding the relationship between the concept "zoo" in the question and the knowledge base, in particular, the word "fence"). This example also points out a central aspect of the knowledge base, which cannot answer all questions due to a limited and finite set of symbolic structures used to represent knowledge [124]. For example, the knowledge base of the above example cannot answer the question "is the giraffe male or female?". In addition, the knowledge base can be static or dynamic (introduction of new concepts or entities, the evolution of relationships or rules for *e.g.* in a multi-level video game.). Here, for sake of simplicity, we consider a scenario where knowledge bases are static and are inferred from the underlying data available at the source and destination. Hence, we refer to *semantic messages* as a sequence of well-formed symbols generated[1] from the source and destination knowledge bases. Our goal is then to propose a framework, which enables *representation learning* of semantic symbols. Figure 6.2 presents our proposed E2E adaptive semantic communication system. It is composed of a semantic encoder $f_{\boldsymbol{\theta}}(\cdot)$ and a semantic decoder $g_{\boldsymbol{\vartheta}}(\cdot)$, which we describe in the following.

6.2.2 Semantic source and channel coding

The semantic encoder transforms input sequence into semantic symbols to be transmitted through the channel. Let $\mathcal{M}_t$ denotes the source alphabet. Each message m emitted by the source is associated with a symbol $x \in \mathcal{X}$ (possibly a discrete or continuous space) such that $x = f_{\boldsymbol{\theta}}(m)$, where $f_{\boldsymbol{\theta}}(\cdot)$ denotes the semantic encoder with (trainable) parameters $\boldsymbol{\theta}$. This encoder is characterized by the probability distribution $p_{\boldsymbol{\theta}}(x|m)$. Thus, if the source emits a message m with a probability $p_{\mathcal{M}_t}(m)$, the probability

[1]Note that before being able to produce semantic symbols, the source and the destination may first agree on a common mechanism (*e.g.*, a logic). This can be achieved through our proposed E2E learning.

that the encoder emits symbol x is:

$$p_{\boldsymbol{\theta}}(x) = \sum_{\substack{m: x = f_{\boldsymbol{\theta}}(m) \\ m \in \mathcal{M}_t}} p_{\mathcal{M}_t}(m) = \sum_{m \in \mathcal{M}_t} \delta\left(x - f_{\boldsymbol{\theta}}(m)\right) p_{\mathcal{M}_t}(m), \tag{6.1}$$

where $\delta(\cdot)$ is the Dirac distribution. Next, our objective is to define the adequate symbols probability distribution $p_{\boldsymbol{\theta}}(x)$ (or equivalently $p_{\boldsymbol{\theta}}(x|m)$), which ensures semantic "fidelity" of interpreted messages at the receiver. Note that the mapping from m to x is not always bijective [125]. Indeed, it can be one-to-many: a message can be mapped to different symbols, each conveying the same information. In this case, the encoder introduces *semantic redundancy*, *i.e.*, the conditioned entropy $H_{\boldsymbol{\theta}}(X|M) \neq 0$. Conversely, the mapping can be many-to-one, *i.e.*, many messages are mapped to the same symbol: there is a *semantic ambiguity*, and $H_{\boldsymbol{\theta}}(M|X) \neq 0$. As in the rate-distortion Theory [126], such an encoder has a complexity equals to $I_{\boldsymbol{\theta}}(X;M)$, which corresponds to average number of bits needed to represent message m. Hence, as we focus on semantic compression, our first objective is to find $f_{\boldsymbol{\theta}}(\cdot)$, which minimizes this complexity *i.e.*,

$$\arg\min_{\boldsymbol{\theta}} I_{\boldsymbol{\theta}}(X;M) \tag{6.2}$$

Lemma 5. *If there is no redundancy introduced by the semantic encoder, i.e., M determines X as the mapping $f_{\boldsymbol{\theta}} : \mathcal{M}_t \rightarrow \mathcal{X}$ is unique, then,*

$$I_{\boldsymbol{\theta}}(X;M) = H_{\boldsymbol{\theta}}(X), \tag{6.3}$$

in which case, minimizing $I_{\boldsymbol{\theta}}(X;M)$ is equivalent to minimizing $H_{\boldsymbol{\theta}}(X)$.

Proof. First note $I_{\boldsymbol{\theta}}(X;M) = H(X) - H_{\boldsymbol{\theta}}(X|M)$. Thus, proof follows as $H_{\boldsymbol{\theta}}(X|M) = 0$ if there is no redundancy. □

6.2.3 Semantic decoder

The role of the decoder is mainly to infer the meaning intended by the source[2]. In contrast to Shannon's communications paradigm, an exact reconstruction of the transmitted messages is not necessary. Given the receiver alphabet $\mathcal{M}_r$ and the semantic decoder $g_{\boldsymbol{\vartheta}}(\cdot)$ with (trainable) parameters $\boldsymbol{\vartheta}$, the decoded message $\hat{m}$ from a received symbol y is the one that maximizes the estimated posterior probability $q_{\boldsymbol{\vartheta}}(m|y)$ conditioned on the received symbol y at the receiver:

$$\hat{m} = \arg\max_{m' \in \mathcal{M}_r} q_{\boldsymbol{\vartheta}}(m'|y), \tag{6.4}$$

Hence, given the semantic encoder and decoder, a natural measure of the semantic distortion between m and $\hat{m}$ is the expected Kullback-Leibler (KL) divergence between the "true" posterior probability $p_{\boldsymbol{\theta}}(m|y)$ at the encoder and the one captured by the decoder $q_{\boldsymbol{\vartheta}}(m|y)$,

$$\mathbb{E}_y\left\{\mathrm{KL}\left(p_{\boldsymbol{\theta}}(m|y)||q_{\boldsymbol{\vartheta}}(m|y)\right)\right\} = \sum_{m \in \mathcal{M}_r} \int_y p_{\boldsymbol{\theta}}(y) p_{\boldsymbol{\theta}}(m|y) \log\left(\frac{q_{\boldsymbol{\vartheta}}(m|y)}{p_{\boldsymbol{\theta}}(m|y)}\right) \mathrm{d}y. \tag{6.5}$$

Our second objective is then to find $f_{\boldsymbol{\theta}}(\cdot)$ and $g_{\boldsymbol{\vartheta}}(\cdot)$ which minimize the semantic distortion between the intended message m and the decoded message $\hat{m}$, *i.e.*, Eqn. (6.5).

$$\arg\min_{\boldsymbol{\theta}, \boldsymbol{\vartheta}} \mathbb{E}_y\left\{\mathrm{KL}\left(p_{\boldsymbol{\theta}}(m|y)||q_{\boldsymbol{\vartheta}}(m|y)\right)\right\} \tag{6.6}$$

[2]The decoder can also recover an equivalent meaning from Level C perspective, *i.e.*, *w.r.t.* to the targeted goal of the communication.

6.2.4 Semantic channel and noise

To illustrate the notion of semantic channel, let us consider the following example of a conversation between three persons [113].

Example 3 (A conversation between Linda, Pheobe, and Aïda)**.** *Here, Linda is trying to convey meaningful information to Phoebe through Aïda, who serves as a* semantic channel.

- Linda: "Aïda, would Phoebe like to go for hiking in the Bastille's mountain?"

- Aïda: "Pheobe, you want to climb the Bastille?"

- Phoebe: "No, I'm not available today."

In this example, Aïda conveys to Pheobe a message completely different from one transmits by Linda, which may result in an engineering failure from classical level A communication's perspective if we compare e.g., transmit and receive sentence character by character. However, there is no semantic failure as Linda's message to Aïda is semantically equivalent to Aïda's message to Phoebe.

> **Definition 6.** *Two messages are semantically equivalent if they convey the same meaning. In other words, the received message $\hat{m}$ and the transmitted message m are semantically equivalent if $\hat{m}$ is interpreted accurately by the receiver as the meaning intended by the transmitter.*

From the above definition, formally defining the notion of semantic equivalence is not trivial, as it can take different forms depending on the purpose of the communication and the type of data manipulated by the source and the destination. For example, in NLP, two words may be semantically equivalent if they are synonyms. A semantic error may occur during communication as the result of a mismatch between m and $\hat{m}$: the two messages are not semantically equivalent. This error can be introduced by *Level A* channel noise and/or interference, the difference of the level of knowledge available at the source and destination or its incompleteness (at *Level B*) and, by limitation of semantic encoder/decoder not being able to learn the correct semantic representation, *i.e.*, a limitation of the representation space of $f_{\boldsymbol{\theta}}(\cdot)$ and $g_{\boldsymbol{\vartheta}}(\cdot)$. To design an efficient communication system, given the semantic channel with probability density $p(y|x)$, our proposed solution maximizes the mutual information $I_{\boldsymbol{\theta}}(X;Y)$ between the input and output of the channel:

$$\arg\max_{\boldsymbol{\theta}} I_{\boldsymbol{\theta}}(X;Y) \tag{6.7}$$

6.2.5 Proposed semantic representation learning

To optimize our semantic communication system, we adopt an E2E learning mechanism, where our objective is to jointly achieve Eqns. (6.2), (6.6) and (6.7). Overall, we propose to minimize the following objective function $\mathcal{L}^{\alpha,\beta}_{\boldsymbol{\theta},\boldsymbol{\vartheta}}$:

$$\mathcal{L}^{\alpha,\beta}_{\boldsymbol{\theta},\boldsymbol{\vartheta}} \;=\; I_{\boldsymbol{\theta}}(X;M) - (1+\alpha) I_{\boldsymbol{\theta}}(X;Y) + \beta \mathbb{E}_y \left\{ \mathrm{KL}\left(p_{\boldsymbol{\theta}}(m|y) || q_{\boldsymbol{\vartheta}}(m|y) \right) \right\}, \tag{6.8}$$

where $\alpha \geq 0$ and $\beta \geq 0$ are some hyperparameters that trade-off the optimization. To minimize $\mathcal{L}^{\alpha,\beta}_{\boldsymbol{\theta},\boldsymbol{\vartheta}}$, we hinge on the well-known cross-entropy (CE) loss defined as:

$$\mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\vartheta}} \triangleq \mathbb{E}_{m\sim p_{\mathcal{M}}(m), y\sim p_{\boldsymbol{\theta}}(y|m)} \left\{ -\log(q_{\boldsymbol{\vartheta}}(m|y) \right\}. \tag{6.9}$$

Indeed, we have the following Lemmas:

> **Lemma 6.** *Assuming the RX and the TX are sharing the same background i.e., $\mathcal{M}_t = \mathcal{M}_r = \mathcal{M}$, the cross-entropy loss can be decomposed as follows:*
>
> $$\mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\vartheta}} = H_{\boldsymbol{\theta}}(X) - I_{\boldsymbol{\theta}}(X;Y) + \mathbb{E}_y \left\{ \mathrm{KL}\left(p_{\boldsymbol{\theta}}(m|y) || q_{\boldsymbol{\vartheta}}(m|y) \right) \right\}. \tag{6.10}$$

Proof of Lemma 2 Eqn. (6).

$$
\begin{aligned}
\mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\vartheta}} &\overset{\Delta}{=} \mathbb{E}_{m\sim p_{\mathcal{M}}(m),\, y\sim p_{\boldsymbol{\theta}}(y|m)}\left\{-\log(q_{\boldsymbol{\vartheta}}(m|y)\right\} \\
&\overset{(a)}{=} -\sum_{m\in\mathcal{M}} p_{\mathcal{M}}(m)\int_y p_{\boldsymbol{\theta}}(y|m)\log(q_{\boldsymbol{\vartheta}}(m|y))\mathrm{d}y, \\
&\overset{(b)}{=} -\sum_{m\in\mathcal{M}}\int_y p_{\boldsymbol{\theta}}(y)p_{\boldsymbol{\theta}}(m|y)\log(q_{\boldsymbol{\vartheta}}(m|y)\mathrm{d}y, \\
&\overset{(c)}{=} -\sum_{m\in\mathcal{M}}\int_y p_{\boldsymbol{\theta}}(y)p_{\boldsymbol{\theta}}(m|y)\log\left(\frac{q_{\boldsymbol{\vartheta}}(m|y)}{p_{\boldsymbol{\theta}}(m|y)}\right)\mathrm{d}y - \sum_{m\in\mathcal{M}}\int_y p_{\boldsymbol{\theta}}(y)p_{\boldsymbol{\theta}}(m|y)\log(p_{\boldsymbol{\theta}}(m|y))\mathrm{d}y \\
&\overset{(d)}{=} \mathbb{E}_y\left\{\mathrm{KL}\left(p_{\boldsymbol{\theta}}(m|y)||q_{\boldsymbol{\vartheta}}(m|y)\right)\right\} - \sum_{m\in\mathcal{M}}\int_y p_{\mathcal{M}}(m)\int_x p(y|x)\delta(x-f_{\boldsymbol{\theta}}(m))\log(p_{\boldsymbol{\theta}}(m|y))\mathrm{d}x\mathrm{d}y \\
&\overset{(e)}{=} \mathbb{E}_y\left\{\mathrm{KL}\left(p_{\boldsymbol{\theta}}(m|y)||q_{\boldsymbol{\vartheta}}(m|y)\right)\right\} - \int_x\int_y p_{\boldsymbol{\theta}}(x)p(y|x)\log(p_{\boldsymbol{\theta}}(x|y))\mathrm{d}x\mathrm{d}y \\
&= \mathbb{E}_y\left\{\mathrm{KL}\left(p_{\boldsymbol{\theta}}(m|y)||q_{\boldsymbol{\vartheta}}(m|y)\right)\right\} - \int_x\int_y p_{\boldsymbol{\theta}}(x,y)\log(p_{\boldsymbol{\theta}}(x|y))\mathrm{d}x\mathrm{d}y \\
&= \mathbb{E}_y\left\{\mathrm{KL}\left(p_{\boldsymbol{\theta}}(m|y)||q_{\boldsymbol{\vartheta}}(m|y)\right)\right\} - \int_x\int_y p_{\boldsymbol{\theta}}(x,y)\log\left(\frac{p_{\boldsymbol{\theta}}(x,y)p_{\boldsymbol{\theta}}(x)}{p_{\boldsymbol{\theta}}(y)p_{\boldsymbol{\theta}}(x)}\right)\mathrm{d}x\mathrm{d}y \\
&= \mathbb{E}_y\left\{\mathrm{KL}\left(p_{\boldsymbol{\theta}}(m|y)||q_{\boldsymbol{\vartheta}}(m|y)\right)\right\} - I_{\boldsymbol{\theta}}(X;Y) + H_{\boldsymbol{\theta}}(X),
\end{aligned}
$$

(a) comes from the definition of the expectation Eqn. (6.9); (b) is straightforward noting that $p(y|m)p(m) = p(m|y)p(y)$; Note that the first term in (c) is the expectation of the KL-divergence between $p_{\boldsymbol{\theta}}(m|y)$ and $q_{\boldsymbol{\vartheta}}(m|y)$. Then, we use $p_{\boldsymbol{\theta}}(y|m) = \int_x p(y|x)\delta(x - f_{\boldsymbol{\theta}}(m))\mathrm{d}x$ in (d) and apply Eqn. (6.1) in (e), which completes the proof. □

Lemma 7. *If $\alpha \geq 0$ and $0 \leq \beta \leq 1$, then, the objective function (6.8) admits an upper-bound as follows:*

$$
\mathcal{L}^{\alpha,\beta}_{\boldsymbol{\theta},\boldsymbol{\vartheta}} \leq \mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\vartheta}} - \alpha I_{\boldsymbol{\theta}}(X;Y). \tag{6.11}
$$

In particular, equality holds if $\beta = 1$ and if there is no semantic redundancy at the source.

Proof. The proof follows by noting that $\mathcal{L}^{\alpha=0,\beta=1}_{\boldsymbol{\theta},\boldsymbol{\vartheta}} = \mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\vartheta}} - H_{\boldsymbol{\theta}}(X|M)$ and that $H_{\boldsymbol{\theta}}(X|M) \geq 0$. □

Thus, to minimize $\mathcal{L}^{\alpha,\beta}_{\boldsymbol{\theta},\boldsymbol{\vartheta}}$, we can minimize this upper-bound, where $I_{\boldsymbol{\theta}}(X;Y)$ can be estimated using mutual information neural estimator [127].

Remark 8. *Note that in [114], the authors have considered minimizing $\mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\vartheta}} - \alpha I_{\boldsymbol{\theta}}(X;Y)$, where $0 \leq \alpha \leq 1$. However, the paper fails in providing a justification on how the proposed loss optimizes the semantic representation learning. In contrast, our Lemma (7) specifies that the* semantic representation loss *(6.8) admits $\mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\vartheta}} - \alpha I_{\boldsymbol{\theta}}(X;Y)$ as an upper-bounded. Hence, minimizing this upper bound also minimizes the loss function $\mathcal{L}^{\alpha,\beta}_{\boldsymbol{\theta},\boldsymbol{\vartheta}}$.*

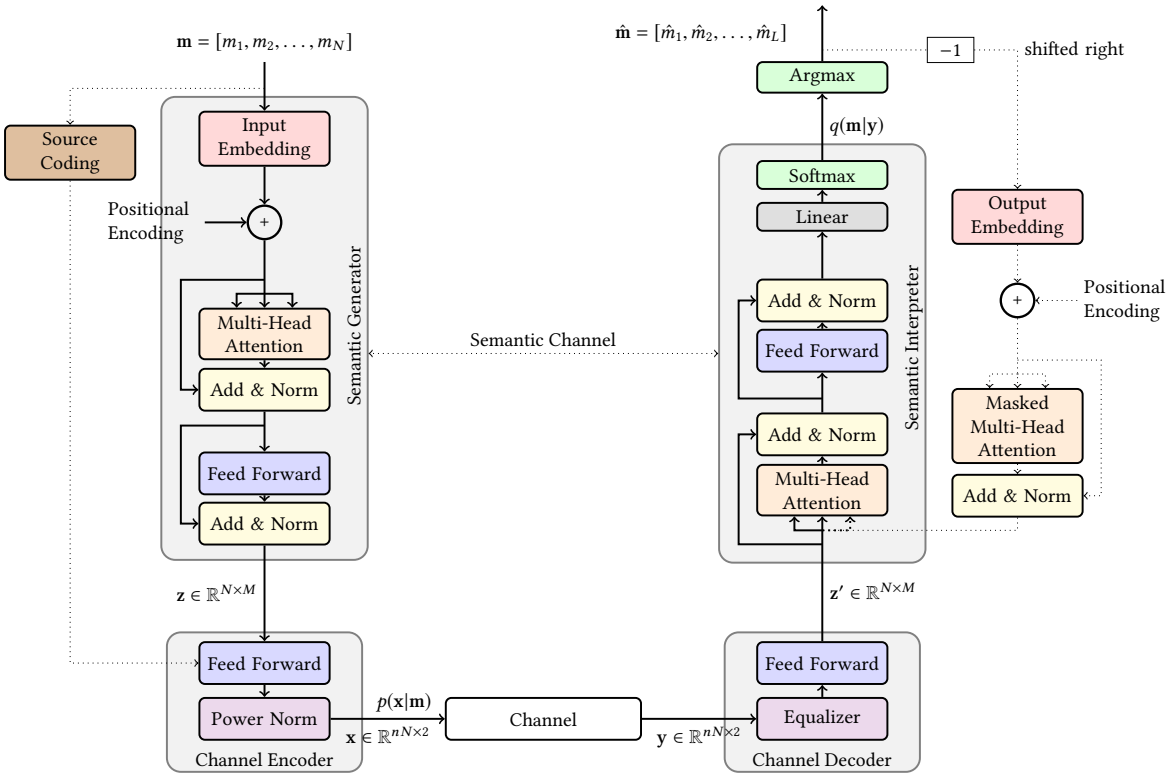


Figure 6.4: Transformer-based semantic communication system architecture

6.3 Transformers Enabled Semantic Communications

Our semantic communication system relies on Transformer architecture [84]. Transformer networks have been introduced as the first transduction model entirely built using self-attention mechanisms able to learn context representation of its input and output. In contrast to solutions based on recurrent and convolution neural networks, Transformer models in general have i) lower computational complexity, ii) more parallelizable computations, and iii) can learn long-range dependencies in input sequence [84].

6.3.1 Background on transformers

The key components of Transformers are *self attention* and *multi-head attention* mechanisms [84].
Self attention mechanism. Given a sequence of size N, let K, Q, and V be the associated *key*, *query*, and *value* matrices respectively, where $K, Q \in \mathbb{R}^{N \times d_k}$, $V \in \mathbb{R}^{N \times d_v}$, $\forall i$, and d_k, d_v are the dimensions of the key, value, respectively. The output A of the attention function can be computed in a matrix form as follows:

$$A = \text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \tag{6.12}$$

where softmax$(\cdot)$ denotes the normalized exponential function.

Multi-head attention mechanism. Consider the following d_m-dimensional key, query, and value matrices $K', Q', V' \in \mathbb{R}^{N \times d_m}$. For each head h, a multi-head attention proceeds by first projecting each row in K', Q', V' into d_k, d_k, d_v dimensional subspace, using linear projectors $\boldsymbol{\theta}_i^K, \boldsymbol{\theta}_i^Q \in \mathbb{R}^{d_m \times d_k}$ and $\boldsymbol{\theta}_i^V \in \mathbb{R}^{d_m \times d_v}$. Here, $\boldsymbol{\theta}_i^K, \boldsymbol{\theta}_i^Q$, and $\boldsymbol{\theta}_i^V$ are learnable weights, describing the set of parameters of the attention head i. Thus, for each head i, the projections gives $K_i = K'\boldsymbol{\theta}_i^K \in \mathbb{R}^{N \times d_k}$, $Q_i = Q'\boldsymbol{\theta}_i^Q \in \mathbb{R}^{N \times d_k}$,

$V_i = V'\boldsymbol{\theta}_i^K \in \mathbb{R}^{N\times d_v}$, and the associated attention value is

$$A_i = \text{Attention}(Q_i, K_i, V_i).$$

Finally, we obtain the output of the multi-head attention mechanism by concatenating the attention's value of all heads and projecting into another linear subspace as follows:

$$\text{MultiHead}(Q, K, V) = \text{concat}(A_1, \ldots, A_h)\boldsymbol{\theta}^O, \tag{6.13}$$

where $\boldsymbol{\theta}^O \in \mathbb{R}^{hd_v\times d_m}$ is another learnable parameter, h is the number of attention heads, and concat$(\cdot)$ is the concatenation operator. Hence, the fundamental idea behind multi-head attention is that each attention head, through its projectors, can extract specific characteristics of inputs sequences. Doing so allows the model to jointly attend to information from different representation subspaces at different positions. This aspect of multi-head attention mechanisms makes them particularly suitable for semantic information extraction.

6.3.2 Architecture description

Figure 6.4 shows our attention-based E2E semantic communication system. Our proposed architecture is composed of a source coder $S(\cdot)$, a semantic generator $G(\cdot)$, a channel encoder $E(\cdot)$, a channel decoder $D(\cdot)$, and a semantic interpreter $I(\cdot)$.

Semantic generator. The key component of the semantic generator is multi-head attention block (see Figure 6.4). It allows features extraction and to find intrinsic relationships between pair of messages (m_i, m_j) in an input sequence $\mathbf{m} = [m_1, m_2, \ldots, m_N]$ generated by the source, where $m_i \in \mathcal{M}_t$. It outputs $\mathbf{z} = G(\mathbf{m}) \in \mathbb{R}^{N\times M}$, in the semantic representation subspace, where each message m_i is mapped into $\mathbb{R}^M$.

Channel encoder. It encodes the message $S(\mathbf{m})$ generated by the source coding block (*e.g.*, using Huffman source coding), with the semantic information provided by the semantic generator $G(\mathbf{m})$: $\mathbf{x} = E(\mathbf{z}, S(\mathbf{m}))$. Here, $E(\cdot)$ is composed of a feed-forward neural network (FNN), followed by a power normalization layer such that $\mathbb{E}[\|\mathbf{x}\|] = 1$ to average the energy of the symbols constellation. Then, each message m_i is encoded in n complex symbols to be transmitted through the wireless channel.

Wireless channel. The channel outputs $\mathbf{y} = h\mathbf{x} + \mathbf{n}$, where h is the fading coefficient matrix, and $\mathbf{n} \sim \mathcal{CN}(0, \sigma_n^2\mathbf{I})$ is an additive Gaussian noise with power σ_n^2 and $\mathbf{I}$ denotes the identity matrix.

Channel decoder. The decoder performs a channel equalization *e.g.*, using Zero Forcing (ZF) method and decodes the received symbols into the semantic representation subspace, $\mathbf{z}' = D(\mathbf{y})$ using a feed-forward neural network.

Semantic interpreter. It plays the inverse role of the generator. it interprets the decoded semantic symbols in the space of possible messages of the receiver alphabet $\mathcal{M}_r$. As the generator, the interpreter is composed of a multi-head attention network. For each decoded message z_i', the output of the interpreter is a probability distribution over all possible messages in $\mathcal{M}_r$: $[q(m|z_i'),\ \forall m \in \mathcal{M}_r]$. Each z_i' is then interpreted as the message $m \in \mathcal{M}_r$ that maximizes $q(m|z_i')$:

$$\hat{m}_i = \underset{m'\in\mathcal{M}_r}{\arg\max}\, q(m'|z_i'),\quad \forall i. \tag{6.14}$$

Remark 9. *Note that the semantic interpreter can also adopt an auto-regressive model, where the previously interpreted message is consumed as an additional input when interpreting the next one. In other words, given a sequence of decoded symbols $\mathbf{z}' = [z_1', \ldots, z_N']$, if the first symbol z_1' is interpreted as $\hat{m}_1$, then the second symbol z_2' is interpreted given $\mathbf{z}'$ and $\hat{m}_1$, then z_3' given $\mathbf{z}'$, $\hat{m}_1$ and*

$\hat{m}_2$, and so on.

$$\hat{m}_i = \arg\max_{m' \in \mathcal{M}_r} q(m'|z'_1, \ldots, z'_i). \tag{6.15}$$

Auto-regressive models are particularly suitable when there is a strong correlation between different messages in the sequence (e.g., text translation). However, it requires interpreting symbols one after the other, thus can result in a large decoding overhead.

6.3.3 Performance measure

To assess the performance of the proposed semantic communication system, we define the following metric and trade-off parameter:

Average transmission rate (bits/s). Let T_s denotes the transmission duration of each symbol. We define the average transmission rate R as the ratio between the amount of transmitted information $I(X;Y)$ and T_s, *i.e.*,

$$R = \frac{I_{\boldsymbol{\theta}}(X;Y)}{T_s} \text{ (bits/s)}. \tag{6.16}$$

Accuracy vs. complexity trade-off. Moreover, we also consider the following metric:

$$\tau = \frac{1}{\mathbb{E}[n]} \times (1 - \psi_{\boldsymbol{\theta},\boldsymbol{\vartheta}}(M, \hat{M})), \tag{6.17}$$

where $\mathbb{E}[n]$ is the average number of symbol per transmitted message. Here, $\psi_{\boldsymbol{\theta},\boldsymbol{\vartheta}}(M, \hat{M})$ measures the semantic error between transmitted message M and interpreted message $\hat{M}$. This error takes different forms depending on the context [118] (*e.g.*, mean square error, cross-entropy or BLEU score in NLP [128]). Thus, τ measures the trade-off between "transmission accuracy" and model complexity in terms of average number of symbols ($\mathbb{E}[n]$) used to represent each message.

6.4 Numerical Results

We provide a detailed evaluation of the performance of our proposed adaptive E2E semantic communication system in the context of natural language processing. Numerical results are reported for text transmission as in [114]. Our reference scenario considers a transmitter communicating with a receiver by sending a block of sentences (sequence of words) through the wireless channel using the previously described semantic communication system. To this end, the transmitter learns to map each word to a sequence of semantic symbols that the receiver has to interpret. Note that such a mapping is learned from the data available at the source. Hence a word can have different symbols representation depending on the sentence it belongs to and the underlying meaning conveyed by both the word and the sentence. In this scenario, once received symbols are interpreted back to words, we measure the transmission accuracy in terms of Bilingual Evaluation Understudy (BLEU) Score, which counts the difference of words (or group of words - n-grams) between the intended sentence and interpreted one [128]. Its value range from zero to one, with one indicating that the interpreted message is the one as the reference. We consider averaging the BLEU score over 1-gram to 4-grams. We use the dataset from Tatoeba Project (translation from English to French data available at http://www.manythings.org/anki/). All FNNs are composed of one multi-layer perceptron with 128 neurons and we use 6 attention heads. Unless otherwise specified, we set $M = 64$, $T_s = 1s$ (normalized), $n = 6$, $\alpha = 0.01$ and $\beta = 1$. Then we train the proposed E2E network for a reference signal-to-noise ratio (SNR) of 7 dB using a batch-size of 256 and then performs tests for different value of SNR.

Impact of the SNR and the source entropy. We first show in Figure 6.5, the impact of SNR and the source entropy on transmission accuracy. We change the source entropy by modifying the distribution

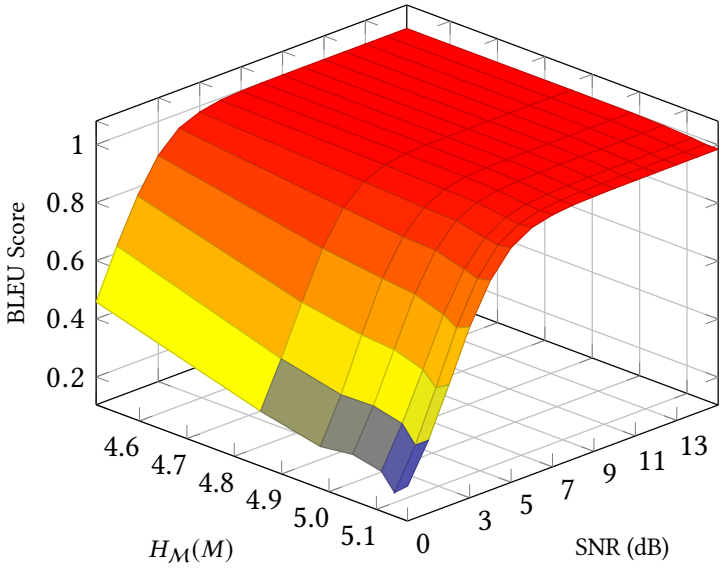


Figure 6.5: Impact of the SNR and $H_{\mathcal{M}}(M)$ on the accuracy. Here we use $n = 6$ symbols/word over AWGN channel.

$p_{\mathcal{M}}(M)$. We observe in Figure 6.5 that the performance slightly decreases when $H_{\mathcal{M}}(M)$ increases since there is more information to convey to the receiver. Also, we observe that the proposed scheme achieves a BLEU score of 1 for SNR $\geq$ 5 dB. In particular, we observe that this threshold varies with the reference SNR for the training, which we set here to 7 dB.

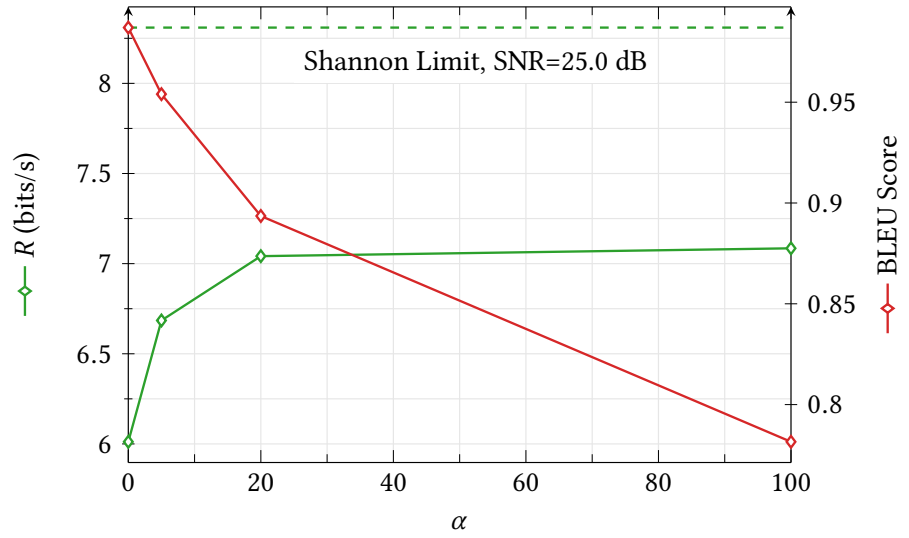


Figure 6.6: Impact of the trade-off parameter α on performances.

Impact of α. Here, we assess the performance of the proposed scheme *w.r.t.* the trade-off parameter α of the objective function (6.8). Figure 6.6 shows the BLEU score and the mutual information of the channel for different α. As α increases we give more importance to $I(X;Y)$ term (6.8), thus increasing the mutual information at the risk of degrading the accuracy.

Impact of the number of symbols per word. Authors in [114] consider a fixed number of symbols per word sent through the channel. However, depending on the length of the words (*e.g.*, the number of characters) and/or the conveyed semantic information, different words may not use the same number of symbols. To show this effect, let $\mathbf{m} = [m_1, m_2, \ldots, m_N]$ be a sequence of words to be transmitted

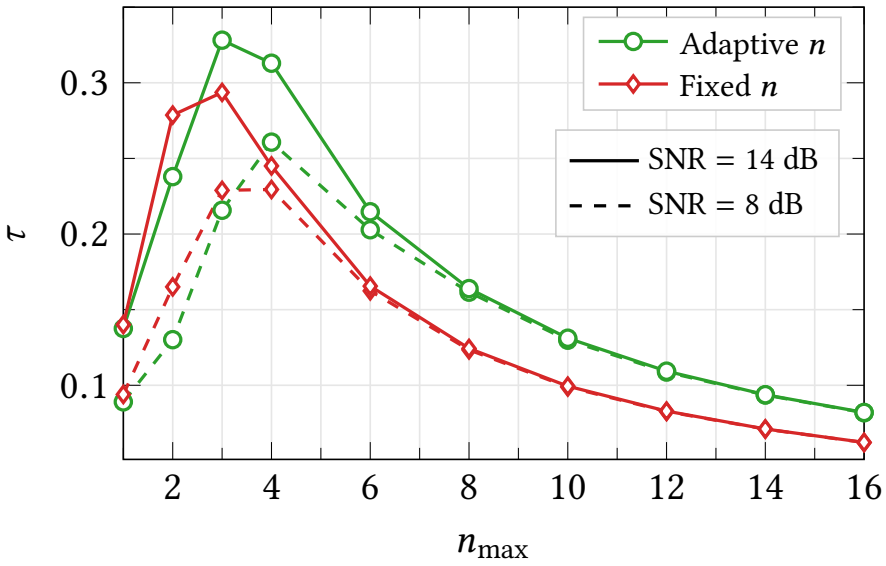


Figure 6.7: Impact of Adaptive vs Fixed number of symbols/word.

and $l(m_i)$ the length of each word m_i on a character basis. Let $L_{\mathbf{m}} = \sum_i l(m_i)$ be the total number of characters in sequence $\mathbf{m}$. we first construct the probability vector $\mathbf{p} = [p_1, \ldots, p_N]$ such that $p_i = \frac{l(m_i)}{L_{\mathbf{m}}}$, $\forall i$. Hence, p_i defines the weight of the word m_i in the sequence in terms of number of characters. Now, let $n_{\max}$ be the maximum number of symbols admissible for each word. Then, we encode each word m_i in n_i (instead of fixed $n = n_{\max}$ as considered in [114]) symbols where,

$$n_i = \min\left(\max\left(n_{\min}, \lfloor n_{\max} N p_i + \frac{1}{2} \rfloor\right), n_{\max}\right). \tag{6.18}$$

Hence, $n_i \in [n_{\min}, n_{\max}]$, $\forall i$. In Figure 6.7, we show the impact of the adaptive vs. fixed encoding on the metric τ, where we arbitrary fix $n_{\min} = 1$ and let $n_{\max} \in [1, 16]$. We first note that in both cases, there is a trade-off between accuracy and complexity, *i.e.*, there is an optimal value (n^*) of $n_{\max}$ depending on the SNR. In particular, for the fixed case ($n = n_{\max}$), and for lower SNR (8dB) we have $n^* = 4$. As the SNR increases to 14dB, only $n^* = 3$ symbols are sufficient to encode each word. In the adaptive case, the number of symbols per word is adapted to the words' length such that on average, $\mathbb{E}[n] \leq n_{\max}$. Therefore, in Figure 6.7, we clearly see that when $n_{\max} \leq 4$, the adaptive method outperforms the fixed one, exhibiting 21.7% increase in τ. This means that for the same accuracy, the adaptive encoding uses a lower number of symbols than the fixed encoding to represent each word. When $n_{\max} \leq 3$, the adaptive method is slightly less efficient: this suggests that there is a minimum number of symbols per word to meet a given accuracy, here *e.g.*, $n_{\min} = 2$.

Impact of languages mismatch. We now show a scenario where the transmitter speaks French and the receiver must understand in English. In this case, the sender and the receiver have different alphabets. This further introduces complexity in symbols interpretation. Indeed, many words in French are written the same way in English leading to semantic ambiguity. The result is 30% decrease in BLEU score performance as show in Figure 6.8. In the same figure, we also show the performance of the classical approach using Huffman/6-bits coding and a 64 QAM modulation. Note that as there is no way to infer English words from decoded symbols in the classical approaches, we rely on Google Translator, although its alphabet is larger than that of our receiver. The proposed semantic communication clearly outperforms the two benchmarks, especially in the low SNR regime.

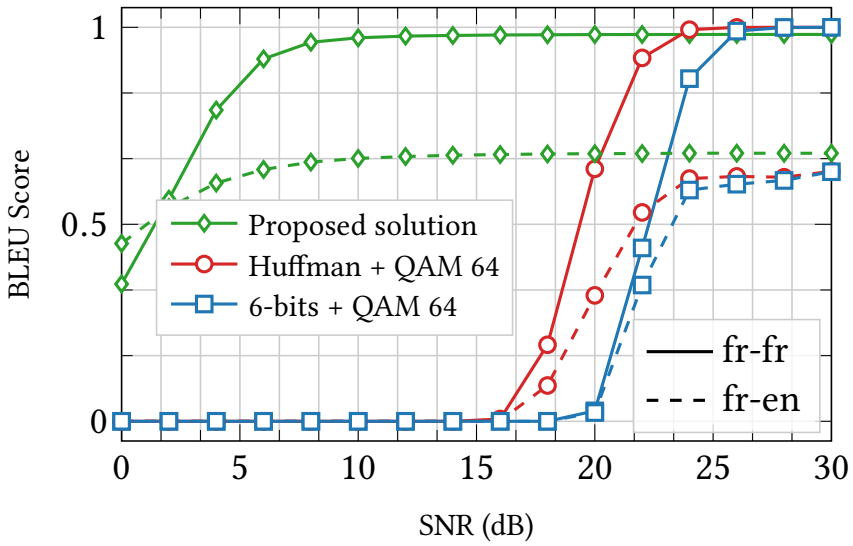


Figure 6.8: 1-gram BLEU Score vs. SNR for French-to-(French/English) translation in the presence of AWGN channel.

6.5 Conclusion and Perspectives

In this work, we focused on showing the potential of semantic compression considering static knowledge bases at the source and destination. To this end, we proposed a novel E2E architecture for an efficient semantic communication system. We started by analyzing theoretical aspects to formulate an objective function for semantic representation learning. Then, we proposed a new metric and trade-off parameter to assess the performance of the proposed system in terms of transmission accuracy and model complexity. Eventually, we proposed a toy example on text transmission, which shows a significant semantic compression gain, especially when sender and receiver speak different languages. In this example, the sender learns to map transmitted sentences into a sequence of well-formed symbols, exploiting the semantic, *i.e.*, the meaning conveyed by these sentences. Then, we proposed a mechanism that adapts the number of symbols per word based on the conveyed semantic, providing up to 21% extra gain compared to state-of-the-art approaches. Importantly, this gain can be significantly extended when applied to multi-modal and data-angry applications such as video-to-text or text-to-video.

Eventually, the work of this last chapter can be extended to fully take advantage of the potential gain of semantic communications for effective and efficient 6G communications. A possible extension first considers dynamic knowledge bases. In this scenario, the knowledge available at the source and destination evolves during the exchanges and according to the shared semantic channels. For example, this is the case when the rules that govern an application change with time, and the source and the destination must dynamically update their reasoning unit accordingly. Another perspective is to study how the obtained semantic compression gain translates into bandwidth saving gain. This would require first defining the semantic capacity, which is still an open research issue. Finally, as in [113], our work focused only on jointly optimizing level A with level B. It is therefore interesting to study how to incorporate level C, *i.e.*, the goal-oriented aspect. One solution is to specify a communication goal: for *e.g.*, the receiver may interpret sentences to execute an action. Then we can optimize the resulting new E2E architecture after a proper definition of the new loss function.

The technical contributions of this chapter have been validated by the following paper.

[C5] **M. Sana** and E. Calvanese Strinati, *"Learning Semantics: An Opportunity for Effective 6G Communications,"* in Proc. IEEE Consumer Communications and Networking Conference (CCNC), Virtual, Las Vegas, January 2022.

Chapter 7

Conclusions and Future Perspectives

"Le sage n'évite pas toujours les erreurs."

"The wise man does not always avoid mistakes."

– Adage Mossi (Burkina Faso)

Contents

7.1 Main Conclusions

In this thesis, we have designed and analyzed novel distributed learning frameworks for radio resource management in 5G and beyond networks. Our proposed approach models user equipments as independent agents, which cooperate or compete for radio or computing resources to optimize network utility functions. To do so, agents learn to make autonomous decisions in a distributed way, based only on their local observations (and global observations if available) using a Multi Agent Reinforcement Learning (MARL) framework. Our proposed method eliminates the need for a cumbersome database or *a priori* modeling, which in practice are infeasible, thus reducing signaling and computational costs. Our proposed solutions jointly incorporate environment's dynamics during learning, including large and small scale fading, intra- and inter-cell interference, users traffic and mobility, as well as radio and computing resources, resulting in near-optimal performance. In addition, by properly designing agent policy network architecture, we ensure flexible, scalable, and transferable solutions. In other words, the learned policies adapt well by design to change in the number of users and their positions and can be transferred to new deployments without requiring substantial additional training. Thus, with the proposed approaches, new users can benefit from the knowledge available in the cell without requiring new learning. Moreover, when a relevant change occurs in the radio environment (*e.g.*, due to fading), our proposed solution is self-reorganized toward the optimal solution.

To come out with all these valuable features, we first proposed a fully distributed and decentralized user association framework in Chapter 3. In this context, we proposed a learning and orchestration mechanism based on *hysteretic deep recurrent Q network*, which allows coordination between users to achieve near-optimal performance without inter-agent communications, thus limiting signaling

overhead. We have validated the proposed solution in millimeter-wave networks with static and dynamic channels, as well as in a mobility context for handover management. However, despite its valuable features, this solution lacks flexibility: it requires a new learning procedure each time a change in the number or position of users occurs *w.r.t.* initial training point. To address this issue, we introduced in Chapter 4, a transferable policy architecture, which allows a user association strategy or policy acquired in a specific scenario (*e.g.*, a network deployment) to be applied to distinct but related scenarios, without having to redefine, recompute, or relearn a new policy. To achieve transferability, our proposed novel architecture conveniently combines *neural attention* mechanisms and multi-agent reinforcement learning and has *zero shot generalization* capacity: a policy learned in a specific deployment can be transferred to another one without requiring substantial additional training procedure. Therefore, as desired, the proposed mechanism adapts well and by design to variations in the number of UEs or changes in the geometry of the network. Such a feature significantly reduces the computational complexity of user association during the network operations and makes the policy suitable for distributed and dynamic scenarios.

Next, based on previous results, we addressed the problem of dynamic computation offloading in Chapter 5. In the considered scenario, multiple users compete for radio and computing resources to offload tasks to an edge server, to reduce energy consumption and/or latency. We first formulated the underlying problem as a long-term minimization problem of system energy consumption under strict end-to-end delay constraints. Then, based on Lyapunov stochastic optimization tools, we decoupled the formulated problem into a *per slot* frequency allocation problem and a radio resource allocation problem, namely a user association problem, which are to be jointly solved. Accordingly, we proposed a fast and efficient iterative algorithm to solve the former problem and we hinged on our transferable user association solution to solve the latter. The resulting framework exhibits near-optimal performance by improving the energy efficiency of the network while significantly reducing complexity.

Finally, our analysis showed that inter-agent communication, although limited, may be necessary for some scenarios to ensure convergence. Thus, in our last study in Chapter 6, we explored the opportunity offered by semantic communications to beyond 5G network management. In this context, what matters in communication between agents is their understanding of the meaning conveyed by exchanged messages and not their correct reconstruction. To this end, in this preliminary work, we focused on semantic compression. In our study, we referred to *semantic* as "meaningful" message (a sequence of well-formed symbols, which are possibly learned from data) that have to be interpreted at the receiver. This required an artificial reasoning unit based on a knowledge base, *i.e.*, a symbolic knowledge representation of the specific application. Therefore, we have proposed and detailed a novel E2E architecture, which allows *representation learning* of semantic symbols for effective semantic communications. We have discussed theoretical aspects and have designed objective functions, which allow learning an efficient semantic encoder and decoder. Our preliminary results have shown significant semantic compression gain, which suggests that semantic communications can bring a significant leap forward to the current 5G networks by enabling efficient and sustainable communications.

7.2 Future Work

Here we present the perspectives of our work, which require further investigation.

7.2.1 UAV assisted wireless networks

An immediate extension of our work concerns application with mobile base stations, namely Unmanned Aerial Vehicles (UAVs) also referred to as drones. Recently UAV applications have gained central interest in the wireless communication community [129]. With their ability to fly, UAVs can be leveraged in a variety of ways to enhance wireless networks. They can be deployed to provide ubiquitous network coverage by assisting the existing wireless communication infrastructure or serve as relays to provide

wireless connectivity between users with no line-of-sight links with surrounding base stations [130]. In this context, an interesting study investigates optimal UAVs deployment and trajectory optimization *w.r.t.* static ground base stations load, UAVs' battery level, and users traffic demand and mobility. Then, a solution to this problem exploits the idea introduced by our proposed user association solution in Chapter 4 to build adaptive and fully transferable Radio Resource Management (RRM) policies. Here, the interaction between multiple UAVs and UEs can be formulated as two distinct MARL problems, where the former optimizes UAVs placement, and the latter optimizes user association.

7.2.2 Explainable policies

Policy explainability is an interesting direction of study [131]. Indeed, the agents in our MARL framework learn to take autonomous decisions, which have an impact on the network performance. It is therefore primordial to understand and to be able to explain the underlying reasoning behind every decision as well as exchanged messages. Doing so will help gain confidence in the learning performance and recover from failure situations, which can occur due to the uncertainties of wireless channels. In the context of user association, an attempt to explain the learned policies in Chapter 3 and 4, may first identify how the different components of UEs' observations impact the output association request. This can be achieved using *e.g.*, classification methods or principal component analysis, which will enable to construct a table mapping the resulting key components to association requests.

7.2.3 Communications for machine learning

Another interesting line of research is to study the impact of wireless communications on machine (or edge) learning applications. One prominent example is federated learning over-the-air, where multiple distributed devices collaboratively perform common learning tasks by exchanging their model parameters rather than raw data, using wireless communication links [132]. In this context, many challenges arise, ranging from learning convergence to optimizing communication and computation resources for communication-efficient learning. Our work in Chapter 5 can be extended, considering that offloaded tasks are now learning tasks (*e.g.*, federated learning) and jointly optimizing learning performance with computation and communication constraints.

7.2.4 Semantic and goal oriented communications

Our preliminary results on semantic communications are promising. In particular, we have shown in Chapter 6 that a significant semantic compression gain can be obtained by transmitting only relevant information, which allow the receiver to correctly extract and understand the intended meaning rather than trying to reproduce the information exactly from one point to another [2]. However, our work did not focus on how a semantic compression gain translates into bandwidth saving. As we mentioned in Chapter 6, this would require first defining the semantic capacity, which is still an open research issue. In addition, as our work focused on applications with the source and destination sharing static knowledge bases, a possible extension considers dynamic knowledge base systems. Such applications require the source and destination to dynamically update their reasoning unit, as their knowledge bases evolve during the exchange of messages and according to the shared semantic channels. Finally, our work focused on jointly optimizing level A with level B. The interesting study to incorporate level C, *i.e.*, the goal-oriented aspect, requires specifying a communication goal. For instance, following our examples of Chapter 6, the receiver may now interpret sentences to execute actions. Then we can optimize the resulting new E2E architecture after a proper definition of the new loss function. We believe that semantic communications together with goal-oriented communications may be one of the cornerstones of sixth-generation (6G) networks [133], thus requiring further research.

Appendices

Appendix A

Résumé étendu de thèse

Les travaux de thèse présentés dans ce manuscrit portent sur les mécanismes d'apprentissage distribués pour la gestion et l'orchestration des réseaux mobiles 5G et au-delà. Plus spécifiquement, nous étudions comment gérer de manière efficiente et efficace les ressources radio et computationnelles des réseaux mobiles en se basant sur des approches d'apprentissage distribuées et sur l'intelligence artificielle. Dans ce qui suit nous résumons brièvement les principaux résultats de nos travaux de recherches. Pour cela nous commençons par situer le contexte d'étude, puis nous décrivons les principaux challenges associés avant de décliner les méthodes proposées pour adresser ces problèmes.

Introduction et contexte d'étude

Les communications sans fil connaissent une demande sans précédent de débit et de bande passante. Non seulement le volume du trafic de données explose, mais les spécificités et la nature des objets communicants se diversifient. Dans le même temps, de nouvelles applications et de nouveaux cas d'utilisation apparaissent, chacun avec des exigences strictes en termes de fiabilité et/ou de latence. Il s'agit par exemple de la réalité augmentée, virtuelle et mixte, de la télémédecine, des véhicules autonomes, des véhicules volants, de l'Internet des objets (IoT), des usines 4.0 et des villes intelligentes. Cela pousse le réseau sans fil à se réinventer constamment pour relever ces défis. L'introduction récente de la cinquième génération (5G) de réseaux mobiles en est un parfait exemple [4]. La technologie 5G représente une avancée considérable dans la conception des réseaux de communication. Elle fournit une infrastructure de communication capable de délivrer simultanément des communications hautement fiables, à faible latence et à débits de données élevés, prenant ainsi en charge une variété de services. Ces services sont généralement répartis en trois grandes catégories :

1. Les communications à haut débit (eMBB) : poussés par la nécessité de fournir un débit de données plus élevé, les services eMBB visent à améliorer la capacité du réseau à prendre en charge des connexions stables avec des débits de données de pointe très élevés (jusqu'à 20 Gbps en liaison descendante [5]) ainsi que des débits de données modérés pour les utilisateurs en bordure de cellule (fournissant globalement un débit de données perçu de 100 Mbps à tout moment et en tout lieu).

2. Les communications massives entre-machines (mMTC): ce service vise à prendre en charge un nombre massif de dispositifs connectés ayant des communications sporadiques (envoi de petits paquets de données) et une faible consommation d'énergie, comme les dispositifs IoT. Parmi les autres cas d'utilisation figurent les réseaux intelligents, l'internet tactile ainsi que les services impliquant des communications de machine à machine.

3. Les communications ultra-fiables à faible latences (URLLC): ce service vise à prendre en charge les applications nécessitant une transmission de paquets courts à faible latence et une fiabilité extrêmement élevée (avec des taux d'erreur des paquets autour de $10^{-5} - 10^{-9}$). Ces applications vont de la télé-chirurgie aux véhicules autonomes en passant par les usines 4.0.

Pour répondre à toutes ces exigences strictes, la 5G adopte principalement des communications dans les bandes millimétriques, l'approche MIMO massif en augmentant le nombre d'antennes par station de base

pour avoir des gains d'antenne important et des communications directives, ainsi qu'un déploiement (ultra) dense des points d'accès mobiles pour booster la capacité du réseau [6]. Cela n'est pas sans complexités additionnelles. En effet, les communications dans les bandes millimétriques souffrent d'une sévère atténuation du canal et sont très sensibles au blocage et aux absorptions atmosphériques. La densification des points d'accès mobile quant à elle, augmente le nombre de stations à gérer, entraînant également des interférences intercellulaires tandis que la gestion simultanée des utilisateurs devient complexe avec le MIMO massif (formation et choix des faisceaux optimaux trop complexes). À cela se rajoutent une complexité liée à la croissance exponentielle du nombre d'utilisateurs connectés, des services hétérogènes et exigeants, des données de trafic variables, des canaux sans fil dynamiques. De ce fait, la gestion des ressources radio devient de plus en plus complexe, nécessitant désormais des solutions avancées, flexibles, évolutives et peu complexes que nous étudions dans cette thèse.

Des solutions distribuées pour la gestion des ressources radio

Dans les communications sans fil, la gestion des ressources radio (RRM) implique toutes les stratégies, procédures et algorithmes utilisés pour gérer efficacement les ressources radio (par exemple la formation de faisceaux, l'allocation de puissance, le choix de la modulation et du schéma de codage de canal, etc.). Traditionnellement, ces algorithmes sont obtenus en résolvant des problèmes d'optimisation basés par exemple sur le trafic de données (instantané), la dynamique des canaux sans fils, les exigences dynamiques de qualité de service des utilisateurs, la charge des stations de base, et cela, sous des contraintes spécifiques de consommation d'énergie des utilisateurs, de latence, de débit, etc. En général, ces problèmes d'optimisation sont des problèmes de programmation en nombres entiers (ou en nombres mixtes), qui sont non convexes et NP-difficile. Par conséquent, les solutions traditionnelles fonctionnent généralement de manière centralisée. En effet, les approches centralisées donnent de meilleurs résultats car les informations provenant de plusieurs nœuds du réseau sont collectées et traitées de manière unifiée. Cependant, elles entraînent une surcharge importante de signalisation et nécessitent un calcul excessif, ce qui n'est pas pratique pour les réseaux 5G en raison du déploiement dense d'utilisateurs et de stations de base. De plus, comme souligné, la gestion des ressources radio fait intervenir de nombreuses variables d'optimisation qui ne sont pas toujours bien définies mathématiquement (en raison de la nature dynamique de l'environnement de propagation, de la mobilité des utilisateurs), ce qui rend difficile la formulation et la résolution de problèmes d'optimisation. Cela motive davantage l'exploration de solutions plus avancées pour la gestion des ressources radio. Cette thèse fait le choix des approches d'apprentissage distribué pour une gestion efficace et efficiente des ressources radio des réseaux mobiles 5G et au-delà. Les solutions distribuées ont l'avantage d'être flexibles, évolutives et robustes face aux perturbations ambiantes. En outre, elles réduisent la surcharge de signalisation et évitent des calculs centralisés laborieux. Cependant, l'apprentissage distribué fait face à plusieurs défis, notamment dans les réseaux 5G denses, en raison d'un environnement sans fil incertain et des ressources radio et de calcul limitées. Motivés par ces défis, nous proposons de nouveaux cadres d'apprentissage distribué basés sur l'apprentissage par renforcement multi-agent, tenant compte de la dynamique de l'environnement (variations des canaux sans fil, interférences intra et intercellulaires, trafic et mobilité des utilisateurs) pour une gestion dynamique des ressources radio. Plus précisément, notre approche modélise les équipements utilisateur comme des agents indépendants, qui collaborent (ou rivalisent) pour accéder à des ressources radio et/ou computationnelles afin d'optimiser des fonctions d'utilité du réseau. Pour cela, les agents s'appuient sur leurs observations locales (et sur d'éventuelles observations globales) pour prendre des décisions autonomes, réduisant ainsi les coûts de signalisation et de calcul.

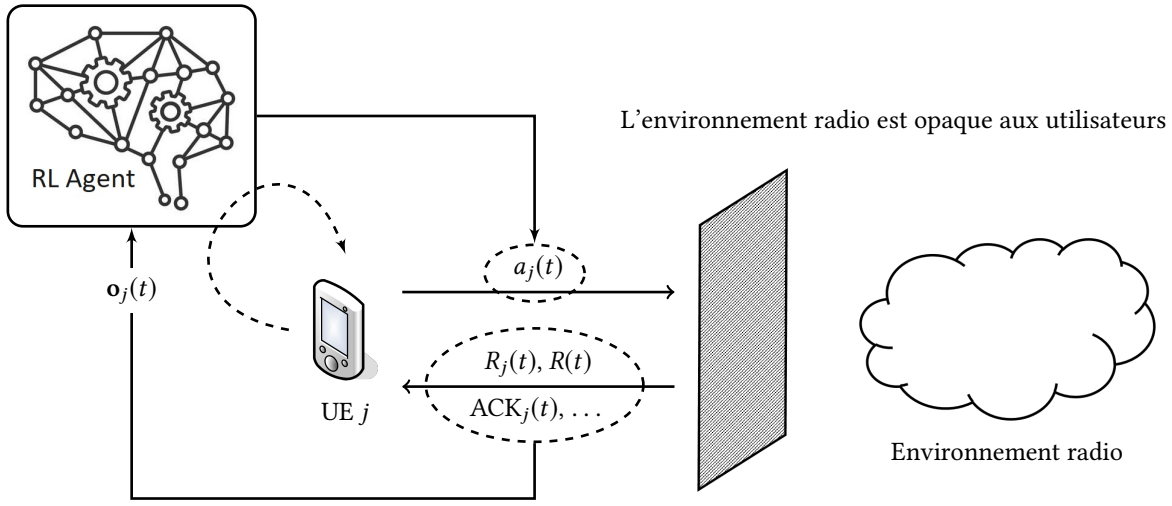


Figure A.1: Illustration du cadre d'apprentissage par renforcement proposé.

Une gestion dynamique basée sur l'apprentissage par renforcement multi-agent

En se basant sur cette approche distribuée, nous proposons dans un premier temps, un cadre d'association d'utilisateurs entièrement distribué et décentralisé pour l'affectation optimale des équipements utilisateurs aux stations de base, ainsi que pour gérer la mobilité. L'association optimale des utilisateurs aux stations de base est une tâche fondamentale, qui est cruciale dans les communications mobiles car elle affecte directement l'efficacité spectrale du réseau ainsi que la qualité de service perçue par les utilisateurs. Cependant elle est difficile à résoudre car c'est un problème combinatoire qui implique généralement des optimisations non convexes et NP-difficile. Pour résoudre ce problème, notre solution associe des mécanismes d'apprentissage par renforcement aux méthodes d'apprentissage machine (et d'apprentissage profonds). En utilisant ce mécanisme, il n'est nullement besoin de bases de données experts labélisés ou de modèles de l'environnement radio, intraitable mathématiquement le plus souvent. Dans notre solution, les agents apprennent leur politique d'association par interaction avec l'environnement radio, de manière à maximiser des fonctions d'utilité du réseau.

Chaque équipement d'utilisateur est modélisé comme un agent indépendant qui prend des décisions autonomes basées sur ces observations locales $\mathbf{o}_j(t)$. Ces observations locales sont choisis avantageusement à l'instant t comme suit :

$$\mathbf{o}_j(t) = \left\{a_j(t-1), R_{a_j(t-1),j}(t-1), R(t-1), \mathrm{ACK}_j(t-1), \mathrm{RSS}_{a_j(t-1),j}(t), D_j(t)\right\}. \tag{A.1}$$

où $a_j(t-1)$ est l'action ayant été effectuée par l'utilisateur j à l'instant précédent, $\mathrm{ACK}_j(t-1)$ est la réponse à la requête d'association renvoyée par la station de base à laquelle elle a été transmise (par exemple $\mathrm{ACK}_j(t-1) = 1$ si l'association était acceptée et $\mathrm{ACK}_j(t-1) = 0$ si elle était refusée). $\mathrm{RSS}_{a_j(t-1),j}(t)$ est à la mesure à l'instant t de la puissance reçue de la station de base à laquelle le terminal mobile s'est associé, $D_j(t)$ est le débit demandé par le terminal mobile à l'instant t, $R_{a_j(t-1),j}(t-1)$ est une estimation de la capacité de canal de la liaison descendante à l'instant précédent $t-1$ (autrement dit $R_{i,j}(t) = B_{i,j}\log_2\left(1 + \mathrm{SINR}_{i,j}(t)\right)$ où $B_{i,j}$ et $\mathrm{SINR}_{i,j}$ sont respectivement la bande passante et le rapport signal à bruit plus interférence lorsque l'utilisateur j est associé à la station de base i). La capacité totale du réseau obtenue à l'instant précédent est alors calculée sur l'ensemble des utilisateurs comme suit :

$$R(t) = \sum_{i\in\mathcal{A}}\sum_{j\in\mathcal{U}} x_{i,j}U_\alpha\left(\min\left(R_{i,j}(t), D_j(t)\right)\right), \tag{A.2}$$

où $\mathcal{A}$ et $\mathcal{U}$ désignent respectivement l'ensemble des stations de base et des utilisateurs dans le réseau. Ici, $U_\alpha(\cdot)$ est une fonction d'utilité permettant d'introduire une équité dans l'association selon le paramètre α [44]. Elle est définie comme suit :

$$U_\alpha(x) = \begin{cases} (1-\alpha)^{-1}x^{1-\alpha}, & \text{si } \alpha \geq 0 \text{ et } \alpha \neq 1, \\ \log(x), & \alpha = 1. \end{cases} \tag{A.3}$$

Ainsi par exemple si $\alpha = 1$, nous avons une équité proportionnelle entre les utilisateur.

L'action d'un équipement utilisateur à l'instant t est définie par la requête d'association de cet utilisateur, soit $a_j(t) = i$, où i désigne l'index de la station de base demandée. Ainsi, après que l'agent associé à l'utilisateur j ait observé le vecteur $\mathbf{o}_j(t)$ et pris l'action $a_j(t)$, celui-ci reçoit une récompense $r(t)$, définie sur la base de la fonction d'utilité du réseau, et commune à tous les utilisateurs. Les équipements utilisateur apprennent ainsi de manière indépendante une stratégie (politique) d'association π_j qui décide pour chaque observation, l'action à effectuer permettant de maximiser la somme de récompenses au cours du temps.

Il est important de noter que dans le cadre proposé, la taille des observations n'évolue pas avec le nombre d'utilisateurs, contrairement à d'autres travaux de la littérature [40]. Aussi, les utilisateurs ne sont pas conscients de leur présence mutuelle dans le réseau et les observations de chaque utilisateur informent partiellement de l'état du réseau. Ce faisant, malgré la bonne association d'un utilisateur donné, la fonction d'utilité résultante peut décroître du fait du mauvais comportement des autres utilisateurs générant ainsi de fortes interférences. Cela peut pousser cet utilisateur à changer sa stratégie bien qu'étant bonne. Ce problème de non-stationnarité de l'environnement dû à l'interaction de multiple agents est fondamental dans l'apprentissage par renforcement multi-agent [63].

L'approche que nous proposons résous ce problème en introduisant le principe d'hystérésis dans l'apprentissage, permettant de traiter différemment les récompenses positives et négatives perçues par les utilisateurs au cours du temps [64]. Plus précisément, un utilisateur, décidant d'être optimiste, accorde moins d'importance à la faible récompense reçue après son action, faisant l'hypothèse que cela est probablement dû au mauvais comportement des autres utilisateurs. Il ignore donc de ce fait cette récompense, en maintenant sa stratégie apprise. Nous montrons qu'en choisissant bien le degré d'optimisme de chaque utilisateur, nous améliorons considérablement les performances d'apprentissage. Outre cela, la solution que nous proposons intègre la dynamique de l'environnement (interférence des canaux, évanouissement rapide et trafic réseau) pendant la phase d'apprentissage, de sorte que l'association des utilisateurs se réorganise d'elle-même vers l'association optimale lorsqu'un changement pertinent se produit dans l'environnement. Par conséquent, nous réduisons davantage les coûts de signalisation ainsi que la complexité de calcul. Ceci est en contraste avec les solutions actuelles de l'état de l'art, qui ne prennent pas en compte la nature dynamique des réseaux sans fil, nécessitant ainsi de recalculer périodiquement ou à chaque fois qu'un changement notable se produit dans l'environnement pour corriger les dérives possibles de l'association optimale. Nous validons cette approche à la fois dans un réseau hétérogène statique et dynamique, comprenant des stations de base millimétriques et sub-6 GHz. Nous montrons notamment que notre approche permet d'atteindre jusqu'à 98.7% de la performance optimale obtenue par une recherche exhaustive, augmentant les gains de performance de 40% comparés à des solutions de l'état de l'art. Dans le cas de la mobilité, notre solution permet de réduire de 70% la fréquence de transfert d'un utilisateur d'une station de base à une autre, très coûteux aussi bien énergétiquement que matériellement.

La nouveauté de cette contribution est ensuite validée dans les articles de conférence, de journal et de brevet suivants:

[C1] **M. Sana**, A. De Domenico, and E. Calvanese Strinati, *"Multi-Agent Deep Reinforcement Learning based User Association for Dense mmWave Networks,"* In Proc. IEEE Global Communications Conference (GLOBECOM), HI, USA, pages 1–6., Dec 2019.

[C2] **M. Sana**, A. De Domenico, E. Calvanese Strinati, and A. Clemente, *"Multi-Agent Deep Reinforcement Learning for Distributed Handover Management In Dense MmWave Networks,"* In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Madrid, Spain, pages 8976–8980., May 2020.

[J1] **M. Sana**, A. De Domenico, W. Yu, Y. Lostanlen, and E. Calvanese Strinati, *"Multi-Agent Reinforcement Learning for Adaptive User Association in Dynamic mmWave Networks,"* IEEE Transactions on Wireless Communications, 19 (10):6520–6534, 2020.

[P1] **M. Sana**, A. De Domenico, *"Method for associating user equipment in a cellular network via multi-agent reinforcement learning,"* Issued in May 20, 2021, US17099922.

Des solutions d'association d'utilisateurs transférables

L'une des principales limites des algorithmes de gestion des ressources radio est qu'ils sont souvent fondés sur des hypothèses assez rigides, telles que des ensembles pré-dimensionnés et fixes de stations de base et d'utilisateurs statiques, des conditions de canal favorables, l'absence d'interférence intra ou intercellulaire. Pourtant, dans les réseaux dynamiques à ondes millimétriques, en particulier dans les réseaux denses, le nombre d'utilisateurs, leur position les uns par rapport aux autres et par rapport aux stations de base, ainsi que les exigences de performance des services auxquels ils accèdent sont susceptibles de changer au fil du temps avec une grande dynamique. Même dans des scénarios relativement stables du point de vue du canal radio et du trafic de données, l'arrivée dans le réseau ou le départ du réseau d'un ou plusieurs utilisateurs impacte les performances globales du réseau. Cela nécessite donc une adaptation constante de l'association d'utilisateurs pour garantir dynamiquement la meilleure qualité de service possible. Pour résoudre ces problèmes, nous proposons dans le chapitre 4 une politique d'association d'utilisateurs évolutive et facile à gérer. Plus précisément, contrairement à la solution précédente formulée pour des scénarios prédéfinis, cette nouvelle solution se concentre sur l'aspect central de la *transférabilité*. Elle permet d'appliquer la stratégie ou la politique d'une association d'utilisateurs acquise dans un scénario spécifique (par exemple, un déploiement donné de réseau) à un autre scénario distinct mais connexe, sans qu'il soit nécessaire de revoir la conception, de recalculer ou de réapprendre une nouvelle politique. De plus, la solution que nous proposons, a une erreur de généralisation quasi-nulle ("zero-shot" learning) : elle s'adapte bien par conception aux variations du nombre d'utilisateurs et de leurs positions sans nécessiter de procédure d'entrainement supplémentaire. Cela réduit considérablement la complexité de calcul de l'association des utilisateurs pendant la phase opérationnelle du réseau et rend la politique adaptée aux scénarios distribués et dynamiques. Nos résultats de simulation montrent que la solution proposée est capable de s'adapter efficacement même lorsque le nombre d'utilisateurs double par rapport à la référence d'apprentissage, avec des gains de performance pouvant atteindre 100% comparés à des solutions de l'état de l'art. Nous validons ensuite la nouveauté de cette proposition dans les contributions suivantes:

[C3] **M. Sana**, N. di Pietro, and E. Calvanese Strinati, *"Transferable and Distributed User Association Policies for 5G and Beyond Networks,"* IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Virtual, Sept. 2021.

[P2] **M. Sana**, N. di Pietro, E. Calvanese Strinati, and B. Miscopein, *"Method for associating user equipment in a cellular network according to a transferable association policy,"* Filed in September 30, 2020, FR2009989.

De l'association d'utilisateur au déchargement à faible consommation d'énergie de tâches de calcul sur des périphéries distantes

Jusqu'à présent, nous avons étudié les mécanismes d'association des utilisateurs pour améliorer l'efficacité spectrale du réseau. Nous proposons maintenant de consolider tous ces acquis pour résoudre le problème du déchargement de calcul efficace en énergie sur des serveurs distants. En effet, avec le déploiement de capacités de calcul et de stockage à la périphérie du réseau, l'informatique de périphérie (également connue sous le nom d'informatique périphérique mobile multi-accès (MEC)) a été conçue pour permettre des services à faible latence, hautement fiables et économes en énergie, en rapprochant les ressources et les capacités de calcul du "cloud" au plus près des utilisateurs finaux. Dans ce contexte, le déchargement dynamique de tâches de calcul permet aux dispositifs pauvres en ressources de calcul, de transférer l'exécution des applications à des serveurs distants afin de réduire la consommation d'énergie et la latence. Dans le scénario envisagé, plusieurs utilisateurs se disputent simultanément des ressources radio et de calcul en périphérie limitées pour obtenir le traitement des tâches déchargées sous des contraintes de délai. Pour cela, nous exploitons les modes de veille à faible consommation à tous les nœuds du réseau. Autrement dit, les utilisateurs, tout comme les stations de base et les serveurs distant peuvent décider de se mettre en "mode veille" pour réduire la consommation d'énergie. Il s'agit ensuite de déterminer quand éteindre ou rallumer un noeud. Du point de vue de la gestion du réseau, cette tâche est complexe et nécessite une optimisation conjointe des ressources radio et de calcul. Dans le chapitre 5, nous formulons le problème sous-jacent comme un problème *d'optimisation dynamique à long terme* visant à réduire la consommation d'énergie à long terme sous des contraintes strictes de délai. Ensuite, en se basant sur les outils d'optimisation stochastique de Lyapunov, nous montrons que ce problème peut être découplé en un problème d'ordonnancement de fréquence CPU et un problème d'allocation des ressources radio, à savoir un problème d'association d'utilisateurs. Nous proposons donc un algorithme itératif rapide, particulièrement efficace pour résoudre le premier problème et nous nous appuyons sur le cadre d'association d'utilisateurs proposé précédemment pour résoudre le second. Dans l'ensemble, l'originalité de la solution résultante réside dans sa capacité à *simultanément* : *i*) minimiser les rapports cycliques de mise en veille de tous les éléments du réseau sous contraintes de délai; *ii*) gérer efficacement les interférences radio; *iii*) être peu complexe; *iv*) combiner les méthodes d'optimisation stochastique de Lyapunov avec l'apprentissage par renforcement multi-agent (MARL); *v*) être distribué et compatible avec la mobilité des UE. Nos résultats de simulation montrent alors que la méthode proposée atteint 96.5% des performances optimales obtenues par recherche exhaustive onéreuse et permet de réduire la consommation d'énergie de 10% comparée à un algorithme heuristique proposée. La nouveauté de cette contribution est validée par l'article de conférence suivant:

[C4] **M. Sana**, M. Merluzzi, N. di Pietro, and E. Calvanese Strinati, *"Energy Efficient Edge Computing: When Lyapunov Meets Distributed Reinforcement Learning,"* IEEE International Conference on Communications (ICC) Workshops, Virtual, Montreal, Canada, June 2021.

Vers des communications sémantiques pour des réseaux au-delà de la 5G encore plus performants

Nous avons montré dans les travaux précédents que la communication entre agents, bien que limitée, peut être nécessaire dans certains scénarios pour garantir la convergence. Si l'on revient à la théorie de l'information de Shannon, l'objectif de la communication a longtemps été d'assurer la réception correcte des messages transmis, indépendamment de leur signification. Cependant, pour que la communication soit efficace, ce qui importe est que les agents comprennent le sens véhiculé par les messages échangés et non leur reconstruction correcte. Ce paradigme fait référence aux *communications sémantiques* : transmettre uniquement les informations pertinentes suffisantes pour que les agents saisissent le sens

voulu (l'objectif visé) permet d'économiser beaucoup de bande passante de communication. Dans cette dernière contribution, nous proposons d'explorer l'opportunité des communications sémantiques comme nouveau fondamental pour les réseaux au-delà de la 5G. Pour cela, dans le chapitre 6, nous proposons et détaillons une nouvelle architecture qui permet l'apprentissage de la représentation des symboles sémantiques pour des communications efficaces entre agents. Nous discutons des aspects théoriques et concevons avec succès des fonctions objectives qui permettent d'apprendre des codeurs et des décodeurs sémantiques efficaces. Nous proposons également un mécanisme adaptatif, qui optimise dynamiquement le nombre de symboles de chaque message transmis. Enfin, nous validons notre approche dans un scénario de transmission de texte, où un expéditeur - un agent IA - transmet des phrases dans une langue que le récepteur doit décoder et comprendre dans une autre langue. Nos résultats numériques préliminaires sont prometteurs et montrent le potentiel des communications sémantiques pour les futurs réseaux 6G. Les résultats de cette contribution ont été acceptés pour publication dans la conférence suivante:

[C5] **M. Sana** and E. Calvanese Strinati, *"Learning Semantics: An Opportunity for Effective 6G Communications,"* in Proc. IEEE Consumer Communications and Networking Conference (CCNC), Virtual, Las Vegas, January 2022.

Conclusion

Dans cette thèse, nous avons conçu et analysé de nouveaux cadres d'apprentissage distribué pour la gestion des ressources radio dans les réseaux mobiles 5G et au-delà. L'approche que nous proposons modélise les équipements des utilisateurs comme des agents indépendants, qui coopèrent ou rivalisent pour des ressources radio ou de calcul afin d'optimiser les fonctions d'utilité du réseau. Pour ce faire, ils apprennent à prendre des décisions autonomes de manière distribuée, en se basant uniquement sur leurs observations locales (et les observations globales si elles sont disponibles) en utilisant un cadre d'apprentissage par renforcement multi-agent. Cette méthode élimine le besoin d'une base de données onéreuse à constituer ou d'une modélisation a priori de l'environnement radio, qui en pratique sont infaisables, réduisant ainsi les coûts de signalisation et de calcul. Les solutions que nous proposons intègrent conjointement la dynamique de l'environnement pendant l'apprentissage, y compris les évanouissements à grande et petite échelle des canaux, les interférences intra et intercellulaires, le trafic et la mobilité des utilisateurs, ainsi que les ressources radio et computationnelles, ce qui permet d'obtenir des performances quasi-optimales. De plus, en concevant correctement l'architecture neuronale de la politique des agents, nous garantissons des solutions flexibles, évolutives et transférables. En d'autres termes, les politiques apprises s'adaptent bien par conception aux changements du nombre d'utilisateurs et de leurs positions et peuvent être transférées à de nouveaux déploiements sans nécessiter de procédures d'entrainement substantielles. Ainsi, avec les approches proposées, les nouveaux utilisateurs peuvent bénéficier des connaissances disponibles dans la cellule sans nécessiter un nouvel apprentissage. De plus, lorsqu'un changement pertinent se produit dans l'environnement radio (par exemple, en raison de l'évanouissement des canaux sans fil), notre solution proposée s'auto-réorganise vers la solution optimale. Enfin, dans notre dernière étude, nous avons exploré l'opportunité des communications sémantiques comme nouveau fondamental pour les communications au-delà des réseaux 5G. Dans ce contexte, ce qui importe dans la communication entre agents est leur compréhension du sens véhiculé par les messages échangés et non leur reconstruction correcte. Par conséquent, nous avons proposé et détaillé une nouvelle architecture, qui permet l'apprentissage de la représentation des symboles sémantiques pour des communications sémantiques efficaces. Nos résultats préliminaires se sont avérés prometteurs et suggèrent que les communications sémantiques apporteront un bond en avant significatif aux réseaux 5G actuels.

Appendix B

Training transferable policies

We provide in this Appendix, the sequence diagram and the algorithm used to derive transferable user association policies of Chapter 4.

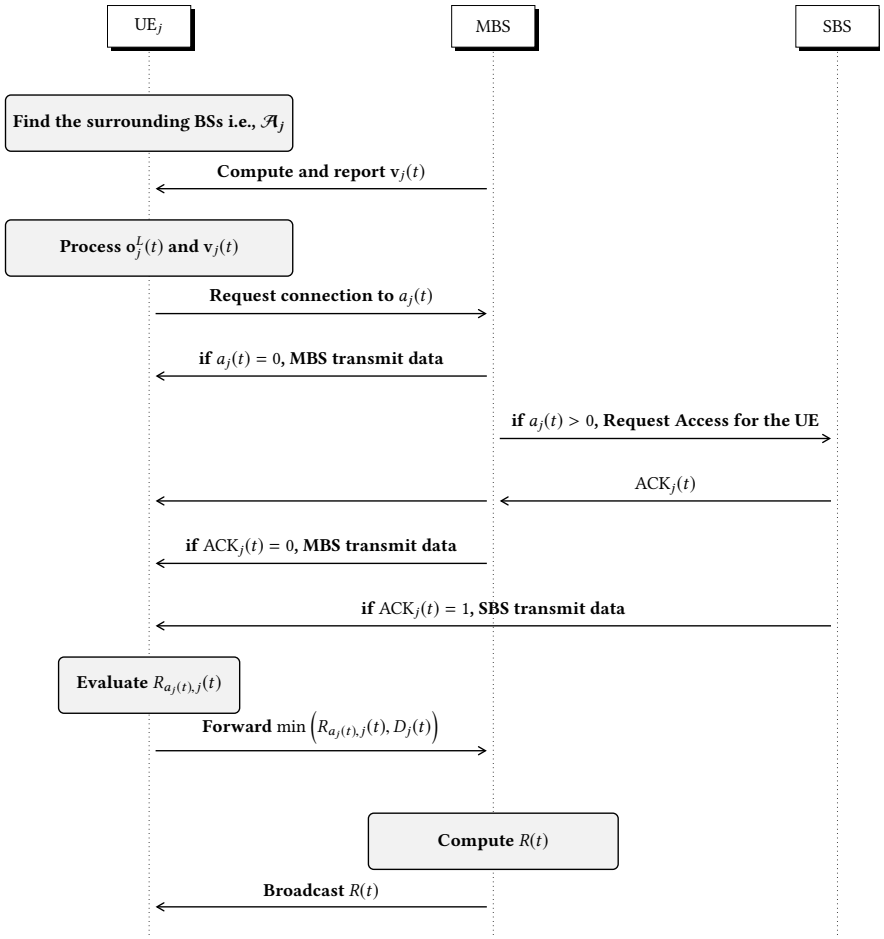


Figure B.1: Message sequence chart for a distributed implementation of the proposed mechanism.

Fig. B.1 shows the flow diagram for a possible implementation of the proposed framework in a distributed fashion. Here, we assume a central controller collocated with the MBS. Each UE j first identifies the set of BSs $\mathcal{A}_j$ it could connect to, which also represents its action space, *i.e.* , an action $a_j(t) \in \mathcal{A}_j$ denotes the index of the BS to which the UE j requests connection at time t. Accordingly, at each time step, UE j observes its local observations $\mathbf{o}_j^L(t)$ and computes the local encoding vector $\mathbf{u}_j(t)$. Then given the available global information, the central controller computes for each UE j, the global encoding vector $\mathbf{v}_j(t)$. Based on $\mathbf{u}_j(t)$ and $\mathbf{v}_j(t)$, UE j selects an action $a_j(t)$ and informs the MBS of the association request. If $a_j(t) = 0$, the MBS grants the connection request and sets up communication. Otherwise, the MBS forwards the connection request to the corresponding SBS. Depending on the overall received requests, the SBS sends an acknowledgement signal ($\text{ACK}_j(t)$) to the MBS. If $\text{ACK}_j(t) = 1$, the SBS grants a connection to the UE; otherwise, the MBS establishes the default data link with the

UE j. Next, each UE j evaluates the perceived data rate, *i.e.* , $R_j(t) = B_{a_j(t),j}\log_2(1 + \text{SINR}_{a_j(t),j})$ and forwards this value to the MBS. Then, the MBS computes the total network utility $R(t)$ and sends it each UE, which use this information to evaluate the goodness of the action selection strategy, and to define future actions accordingly. In Algorithm 5 we summarizes the main steps used for training the proposed transferable user association policies using Proximal Policy Optimization (PPO).

Algorithm 5: Transferable User Association: Training Procedure

Initialize *actor* and *critic* network. `// Note: UEs share the same policy network.`
Initialize global memory $\mathcal{M}$.
for N *episodes* **do**
  Randomly deploy K UEs.
  Apply dropout mechanism with probability p_0.
  Free global memory $\mathcal{M}$.
  while $t < T_e$ **do**
    for $j \in \mathcal{U}$ **do**
      Observe state $\mathbf{o}_j(t) = \{\mathbf{o}_j^L(t), \mathbf{o}_j^G(t)\}$.
      Use the *actor* and compute the *association probability vector* $\mathbf{p}_j(t) = [p_{0,j}, \ldots, p_{N_s,j}]$.
      Sample action $a_j(t)$ in $\mathcal{A}_j$ from distribution $\mathbf{p}_j(t)$.
      if $a_j(t) == 0$ **then**
        MBS grants access.
        $\text{ACK}_j(t) \leftarrow 1$. `// the UE is requesting a connection to the MBS.`
      end
    end
    for $i \in \mathcal{A}\backslash\{0\}$ **do**
      if $\sum_j \mathbb{1}_{a_j(t)=i} > N_i$ **then**
        Admit only the best N_i UEs *w.r.t.* their $\mathbf{p}_j(t)$ and set $\text{ACK}_j \leftarrow 1$ for these UEs.
        Redirect the others UEs towards the MBS and set $\text{ACK}_j \leftarrow 0$ for these UEs.
      else
        $ACK \leftarrow 1$ for all SBS' UEs.
      end
    end
    for $j \in \mathcal{U}$ **do**
      Measure $R_{a_j(t),j}$.
    end
    $R(t) \leftarrow \sum_{i\in\mathcal{A}} \sum_{j\in\mathcal{U}} x_{i,j} U_\alpha \left(\min\left(R_{i,j}(t), D_j(t)\right)\right)$. `// compute network utility.`
    for $j \in \mathcal{U}$ **do**
      $r_j(t) = R(t)$. `// UEs are equally rewarded.`
      Observe the new state $\mathbf{o}_j(t+1) = \{\mathbf{o}_j^L(t+1), \mathbf{o}_j^G(t+1)\}$.
      Store experience $e_j(t) = \left\{\mathbf{o}_j(t), a_j(t), r_j(t), \mathbf{o}_j(t+1)\right\}$ into global memory $\mathcal{M}$.
    end
    $t = t + 1$.
  end
  Sample a batch of experiences from $\mathcal{M}$.
  Update *actor* network to minimize the *PPO loss*: $\mathbb{E}_\pi\left[\min\left(\zeta(\boldsymbol{\theta})\hat{A}, \textbf{clip}\left(\zeta(\boldsymbol{\theta}), 1-\epsilon_1, 1+\epsilon_2\right)\hat{A}\right)\right]$
  Update *critic* network to minimize the *TD error*: $\delta(t) = r(t) + \gamma V(\mathbf{o}_j(t+1)) - V(\mathbf{o}_j(t))$
end
Note that the "gray parts" can be computed in parallel.

Appendix C

Upper bound of the Lyapunov drift-plus-penalty function

In this Appendix, we provide full derivations of the Lyapunov drift-plus-penalty's upperbound used for Proposition 1 in Chapter 5. For this let us recall the definition of the virtual queues, which evolve as:

$$Z_j(t+1) = \max(0, Z_j(t) + Q_j^{\text{tot}}(t+1) - Q_j^{\text{avg}}), \quad \forall j. \tag{C.1}$$

Here, $Q_j^{\text{tot}}(t) = Q_j^l(t) + Q_j^{\text{s}}(t)$ is the sum of the uplink communication queue $Q_j^l(t)$ and the computation queue $Q_j^{\text{s}}(t$, which evolve as follow:

$$Q_j^l(t+1) = \max\left(0, Q_j^l(t) - N_j^u(t)\right) + D_j(t), \tag{C.2}$$

$$Q_j^{\text{s}}(t+1) = \max\left(0, Q_j^{\text{s}}(t) - N_j^c(t)\right) + \min\left(Q_j^l(t), N_j^u(t)\right), \tag{C.3}$$

where $D_j(t)$ is the number of newly arrived offloadable data units generated by the application that runs at the UE at time t, $N_j^u(t)$ and $N_j^c(t)$ are the number of data units offloaded and processed over one slot respectively.

Lyapunov function

Our initial objective is to ensure the mean rate stability of the virtual queues $Z_j(t)\ \forall j$. For this, we introduce the Lyapunov function $L(\mathbf{Z}(t))$ as:

$$L(\mathbf{Z}(t)) = \frac{1}{2}\sum_{j=1}^{K} Z_j(t)^2. \tag{C.4}$$

Note that the lower is $L(\mathbf{Z}(t))$, the lower the virtual queues. Also, we introduce the associated Lyapunov drift-plus-penalty function, which is defined as follows:

$$\Delta_p(\mathbf{Z}(t)) = \mathbb{E}\left[L(\mathbf{Z}(t+1)) - L(\mathbf{Z}(t)) + \Omega \cdot \mathrm{E}_{\mathrm{w}}(t) | \mathbf{Z}(t)\right]. \tag{C.5}$$

Here, $\Delta_p(\mathbf{Z}(t))$ is the conditional expected change of the Lyapunov function over one slot plus a penalty factor that weights the objective function of ($\mathcal{P}_0$) using parameter Ω. Now, if $\Delta_p(\mathbf{Z}(t))$ is bounded $\forall t$, all virtual queues are mean rate stable [105]. Thus, our objective is to define an upperbound of $\Delta_p(\mathbf{Z}(t))$.

Upperbound derivation

To derive an upper bound of $\Delta_p(\mathbf{Z}(t))$, first note that from [105, p. 59], given a generic queue $X(t)$ evolving as

$$X(t+1) = \max(0,\ X(t) + y(t+1) - \bar{y}), \tag{C.6}$$

we have,

$$\frac{X(t+1)^2 - X(t)^2}{2} \leq \frac{(y(t+1) - \bar{y})^2}{2} + X(t)y(t+1) - X(t)\bar{y}.$$

To simplify notations, let $\Delta X(t)^2 = X(t+1)^2 - X(t)^2$. By applying (C.7) to the virtual queue $Z_j(t)$ defined in (C.1) and noting that $(x+y)^2 \leq 2x^2 + 2y^2$, we can write

$$\begin{aligned}\frac{\Delta Z_j(t)^2}{2} &\leq \frac{\left(Q_j^{\text{tot}}(t+1) - Q_j^{\text{avg}}\right)^2}{2} + Z_j(t)\left(Q_j^{\text{tot}}(t+1) - Q_j^{\text{avg}}\right), \\ &= \frac{1}{2}\left(Q_j^{\text{tot}}(t+1)\right)^2 - Q_j^{\text{tot}}(t+1)Q_j^{\text{avg}} + \frac{1}{2}\left(Q_j^{\text{avg}}\right)^2 + Z_j(t)\left(Q_j^{\text{tot}}(t+1) - Q_j^{\text{avg}}\right), \\ &\leq \left(Q_j^{l}(t+1)\right)^2 + \left(Q_j^{\text{s}}(t+1)\right)^2 + \frac{1}{2}\left(Q_j^{\text{avg}}\right)^2 + Z_j(t)\left(Q_j^{\text{tot}}(t+1) - Q_j^{\text{avg}}\right).\end{aligned}$$

Now, note that for $A, b \geq 0$ we have from [105]:

$$(\max(0, Q-b) + A)^2 \leq Q^2 + A^2 + b^2 + 2Q(A-b). \tag{C.7}$$

Now, recalling the evolution of the physical queues (C.2), we can write

$$\begin{aligned}\frac{\Delta Z_j(t)^2}{2} \leq{}& \left(Q_j^{l}(t)\right)^2 + \left(D_j(t)\right)^2 + 2Q_j^{l}(t)\left(D_j(t) - N_j^{u}(t)\right) + \left(N_j^{u}(t)\right)^2 \\ &+ \left(\min\left(Q_j^{l}(t), N_j^{u}(t)\right)\right)^2 + \left(Q_j^{\text{s}}(t)\right)^2 + 2Q_j^{\text{s}}(t)\left(\min\left(Q_j^{l}(t), N_j^{u}(t)\right) - N_j^{c}(t)\right) \\ &+ \left(N_j^{c}(t)\right)^2 + \frac{1}{2}\left(Q_j^{\text{avg}}\right)^2 + Z_j(t)\left(Q_j^{\text{tot}}(t+1) - Q_j^{\text{avg}}\right).\end{aligned} \tag{C.8}$$

Next, applying the following inequalities

- $D_j(t) \leq D_j^{\max}$,
- The frequency allocated to each UE j is such that $f_j(t) \leq f_{\max}$,
- $N_j^{c}(t) \leq N_j^{c,\max} = \lfloor \tau f_{\max} J_j \rfloor$, where J_j is the number of processed data units per CPU cycle,
- $N_j^{u}(t) \leq N_j^{u,\max} = \left\lfloor \frac{\tau R_j^{\max}}{S_j} \right\rfloor$, where $R_j^{\max}$ is the maximum rate of UE j.

we have,

$$\begin{aligned}\frac{\Delta Z_j(t)^2}{2} \leq{}& \left(Q_j^{l}(t)\right)^2 + (D_j^{\max})^2 + 2Q_j^{l}(t)D_j(t) + (N_j^{u}(t))^2 - 2Q_j^{l}(t)N_j^{u}(t) \\ &+ \left(\frac{1}{2}(N_j^{u}(t) + Q_j^{l}(t))\right)^2 + (N_j^{c}(t))^2 + 2Q_j^{\text{s}}(t)\left(\frac{1}{2}(N_j^{u}(t) + Q_j^{l}(t))\right) \\ &+ (Q_j^{\text{s}}(t))^2 - 2Q_j^{\text{s}}(t)N_j^{c}(t) + \frac{1}{2}\left(Q_j^{\text{avg}}\right)^2 + Z_j(t)\left(Q_j^{\text{tot}}(t+1) - Q_j^{\text{avg}}\right).\end{aligned} \tag{C.9}$$

Now, recalling the definition of the total queue and utilizing the fact that

$$\tau f_j(t) J_j - 1 \leq N_j^{c}(t) = \lfloor \tau f_j(t) J_j \rfloor \leq \tau f_j(t) J_j,$$

and after rearranging terms, we have

$$\begin{aligned}\frac{\Delta Z_j(t)^2}{2} \leq{}& \left(D_j(t)\right)^2 + \frac{1}{2}\left(Q_j^{\text{avg}}\right)^2 + \frac{5}{4}\left(N_j^{u,\max}\right)^2 + \left(N_j^{c,\max}\right)^2 + \frac{5}{4}\left(Q_j^{l}(t)\right)^2 \\ &+ Q_j^{\text{s}}(t)Q_j^{l}(t) + 2Q_j^{l}(t)D_j(t) + \left(Q_j^{\text{s}}(t)\right)^2 - Z_j(t)Q_j^{\text{avg}} + \min\left(Q_j^{l}(t), N_j^{u}(t)\right)Z_j(t) \\ &+ Z_j(t)D_j(t) - 2Q_j^{\text{s}}(t)\left(\tau f_j(t) J_j - 1\right) + \max\left(0, Q_j^{\text{s}}(t) - (\tau f_j(t) J_j - 1)\right)Z_j(t) \\ &- \frac{3}{2}Q_j^{l}(t)N_j^{u}(t) + Q_j^{\text{s}}(t)N_j^{u}(t) + \max\left(0, Q_j^{l}(t) - N_j^{u}(t)\right)Z_j(t).\end{aligned} \tag{C.10}$$

Then, by summing over all UEs and taking the expectation, we have,

$$\begin{aligned}\Delta_p(\mathbf{\Theta}(t)) \leq \zeta + \chi(t) + \mathbb{E}\Big\{ & \Omega \cdot \mathrm{E}_{\mathrm{tot}}(t) \\ & + \sum_{k=1}^{K}\Big[-2Q_j^{\mathrm{s}}(t)\tau f_j(t)J_j + \max\Big(0, Q_j^{\mathrm{s}}(t) - \tau f_j(t)J_j + 1\Big) Z_j(t)\Big] \\ & + \sum_{k=1}^{K}\Big[\Big(-\frac{3}{2}Q_j^{l}(t) + Q_j^{\mathrm{s}}(t)\Big) N_j^{u}(t) + \max\Big(0, Q_j^{l}(t) - N_j^{u}(t)\Big) Z_j(t)\Big]\Big|\mathbf{\Theta}(t)\Big\}, \end{aligned} \tag{C.11}$$

where,

$$\begin{aligned}\zeta &= \mathbb{E}\Big\{ \sum_{k=1}^{K}\Big(D_j^{\max}\Big)^2 + \frac{1}{2}\Big(Q_j^{\mathrm{avg}}\Big)^2 + \frac{5}{4}\Big(N_j^{u,\max}\Big)^2 + \Big(N_j^{c,\max}\Big)^2\Big\} \\ &= \frac{1}{2}\sum_{k=1}^{K}\Big[2D_j^{\max 2} + (Q_j^{\mathrm{avg}})^2 + \frac{5}{2}\Big(N_j^{u,\max}\Big)^2 + 2\Big(N_j^{c,\max}\Big)^2\Big] \end{aligned} \tag{C.12}$$

$$\begin{aligned}\chi(t) = \mathbb{E}\Big\{ & \sum_{k=1}^{K}\frac{5}{4}\Big(Q_j^{l}(t)\Big)^2 + Q_j^{\mathrm{s}}(t)^2 + Q_j^{\mathrm{s}}(t)Q_j^{l}(t) + 2Q_j^{\mathrm{s}}(t) + 2Q_j^{l}(t)D_j(t) \\ & - Z_j(t)Q_j^{\mathrm{avg}} + \min\Big(Q_j^{l}(t), N_j^{u}\Big) Z_j(t) + Z_j(t)D_j(t)\Big\} \\ = & \sum_{k=1}^{K}\Big[\frac{5}{4}(Q_j^{l}(t))^2 + (Q_j^{\mathrm{s}}(t))^2 + 2Q_j^{\mathrm{s}}(t) + Q_j^{\mathrm{s}}(t)Q_j^{l}(t) \\ & + 2Q_j^{l}(t)D_j(t) + \min\Big(Q_j^{l}(t), N_j^{u}\Big) Z_j(t) + Z_j(t)D_j(t) - Z_j(t)Q_j^{\mathrm{avg}}\Big] \end{aligned} \tag{C.13}$$

Here, ζ is constant and independent of time t and $\chi(t)$ is constant at time t and does not depend on the optimization variables.

Bibliography

[1] R. J. Mailloux, *Phased Array Antenna Handbook*, 3rd ed. Norwood, MA, USA: Artech House, Inc., 2017. (Cited on pages viii and 31.)

[2] E. C. Strinati and S. Barbarossa, "6G networks: Beyond Shannon towards Semantic and Goal-Oriented Communications," *Computer Networks*, vol. 190, p. 107930, 2021. (Cited on pages ix, 89, 90 and 105.)

[3] E. Calvanese Strinati, S. Barbarossa, J. L. Gonzalez-Jimenez, D. Ktenas, N. Cassiau, L. Maret, and C. Dehos, "6G: The Next Frontier: From Holographic Messaging to Artificial Intelligence Using Subterahertz and Visible Light Communication," *IEEE Vehicular Technology Magazine*, vol. 14, no. 3, pp. 42–50, Sep. 2019. (Cited on pages 1, 4, 20, 71, 72 and 89.)

[4] 3GPP TR 21.915, "Release description (Release 15)," Oct 2019. (Cited on pages 1 and 107.)

[5] 3GPP TR 38.913, "5G; Study on Scenarios and Requirements for Next Generation Access Technologies (Release 15)," Sept 2018. (Cited on pages 1, 14 and 107.)

[6] A. De Domenico, R. Gerzaguet, N. Cassiau, A. Clemente, R. D'Errico, C. Dehos, J. L. Gonzalez, D. Ktenas, L. Manat, V. Savin, and A. Siligaris, "Making 5G Millimeter-Wave Communications a Reality [Industry Perspectives]," *IEEE Wireless Communications*, vol. 24, no. 4, pp. 4–9, Aug 2017. (Cited on pages 2, 13 and 108.)

[7] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, "Millimeter Wave Mobile Communications for 5G Cellular: It Will Work!" *IEEE Access*, vol. 1, pp. 335–349, 2013. (Cited on page 2.)

[8] E. C. Strinati, G. C. Alexandropoulos, V. Sciancalepore, M. Di Renzo, H. Wymeersch, D.-T. Phan-huy, M. Crozzoli, R. D'Errico, E. De Carvalho, P. Popovski *et al.*, "Wireless Environment as a Service enabled by Reconfigurable Intelligent Surfaces: The RISE-6G Perspective," *arXiv preprint arXiv:2104.06265*, 2021. (Cited on page 3.)

[9] E. Basar, M. Di Renzo, J. De Rosny, M. Debbah, M.-S. Alouini, and R. Zhang, "Wireless Communications Through Reconfigurable Intelligent Surfaces," *IEEE Access*, vol. 7, pp. 116 753–116 773, 2019. (Cited on page 3.)

[10] ETSI GS MEC 003 V2.1.1, "Multi-access Edge Computing (MEC); Framework and Reference Architecture," Jan 2019. (Cited on page 4.)

[11] J. Park, S. Samarakoon, M. Bennis, and M. Debbah, "Wireless Network Intelligence at the Edge," *Proceedings of the IEEE*, vol. 107, no. 11, p. 2204–2239, Nov 2019. (Cited on page 4.)

[12] E. Peltonen, M. Bennis, M. Capobianco, M. Debbah, A. Ding, F. Gil-Castiñeira, M. Jurmu, T. Karvonen, M. Kelanti, A. Kliks *et al.*, "6G White Paper on Edge Intelligence," *arXiv preprint arXiv:2004.14850*, 2020. (Cited on pages 4 and 20.)

[13] Y. Deng, "Deep Learning on Mobile Devices: a Review," in *Mobile Multimedia/Image Processing, Security, and Applications*, vol. 10993. International Society for Optics and Photonics, 2019, p. 109930A. (Cited on pages 4 and 20.)

[14] J. Lee, N. Chirkov, E. Ignasheva, Y. Pisarchyk, M. Shieh, F. Riccardi, R. Sarokin, A. Kulik, and M. Grundmann, "On-device Neural Net Inference with Mobile GPUs," *arXiv preprint arXiv:1907.01989*, 2019. (Cited on pages 4 and 20.)

[15] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," *IEEE Communications Surveys Tutorials*, vol. 21, no. 4, pp. 3039–3071, 2019. (Cited on page 4.)

[16] A. Zappone, M. Di Renzo, M. Debbah, T. T. Lam, and X. Qian, "Model-aided wireless artificial intelligence: Embedding expert knowledge in deep neural networks for wireless system optimization," *IEEE Vehicular Technology Magazine*, vol. 14, no. 3, pp. 60–69, 2019. (Cited on page 4.)

[17] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. The MIT Press, 2018. (Cited on pages 4, 24, 26 and 56.)

[18] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level Control through Deep Reinforcement Learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015. (Cited on pages 4, 24, 29 and 33.)

[19] S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar, and A. Y. Zomaya, "Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7457–7469, 2020. (Cited on page 5.)

[20] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A Joint Learning and Communications Framework for Federated Learning Over Wireless Networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 269–283, 2021. (Cited on page 5.)

[21] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečný, S. Mazzocchi, B. McMahan, T. Van Overveldt, D. Petrou, D. Ramage, and J. Roselander, "Towards Federated Learning at Scale: System Design," in *Proceedings of Machine Learning and Systems*, A. Talwalkar, V. Smith, and M. Zaharia, Eds., vol. 1, 2019, pp. 374–388. (Cited on page 5.)

[22] M. Chen, D. Gündüz, K. Huang, W. Saad, M. Bennis, A. V. Feljan, and H. V. Poor, "Distributed Learning in Wireless Networks: Recent Progress and Future Challenges," *IEEE Journal on Selected Areas in Communications*, pp. 1–1, 2021. (Cited on page 5.)

[23] M. Mohammadi Amiri and D. Gündüz, "Machine Learning at the Wireless Edge: Distributed Stochastic Gradient Descent Over-the-Air," *IEEE Transactions on Signal Processing*, vol. 68, pp. 2155–2169, 2020. (Cited on page 5.)

[24] L. Busoniu, R. Babuŝka, and B. D. Schutter, "A Comprehensive Survey of Multiagent Reinforcement Learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 2, pp. 156–172, March 2008. (Cited on pages 5 and 23.)

[25] D. Lopez-Perez, I. Guvenc, G. de la Roche, M. Kountouris, T. Q. Quek, and J. Zhang, "Enhanced Intercell Interference Coordination Challenges in Heterogeneous Networks," *IEEE Wireless Communications*, vol. 18, no. 3, pp. 22–30, 2011. (Cited on page 13.)

[26] A. De Domenico, V. Savin, and D. Ktenas, "A Backhaul-Aware Cell Selection Algorithm for Heterogeneous Cellular Networks," in *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Sep. 2013, pp. 1688–1693. (Cited on page 13.)

[27] N. Sapountzis, T. Spyropoulos, N. Nikaein, and U. Salim, "User Association in HetNets: Impact of Traffic Differentiation and Backhaul Limitations," *IEEE/ACM Transactions on Networking*, vol. 25, no. 6, pp. 3396–3410, Dec 2017. (Cited on page 13.)

[28] J. Hoydis, S. ten Brink, and M. Debbah, "Massive MIMO in the UL/DL of Cellular Networks: How Many Antennas Do We Need?" *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 2, pp. 160–171, 2013. (Cited on page 13.)

[29] Q. Ye, O. Y. Bursalioglu, H. C. Papadopoulos, C. Caramanis, and J. G. Andrews, "User Association and Interference Management in Massive MIMO HetNets," *IEEE Transactions on Communications*, vol. 64, no. 5, pp. 2049–2065, 2016. (Cited on page 13.)

[30] Y. Xu and S. Mao, "User Association in Massive MIMO HetNets," *IEEE Systems Journal*, vol. 11, no. 1, pp. 7–19, 2017. (Cited on page 13.)

[31] T. Bai, R. Vaze, and R. W. Heath, "Analysis of Blockage Effects on Urban Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 9, pp. 5070–5083, Sep. 2014. (Cited on pages 13, 14 and 16.)

[32] O. Semiari, W. Saad, M. Bennis, and B. Maham, "Mobility Management for Heterogeneous Networks: Leveraging Millimeter Wave for Seamless Handover," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Dec 2017, pp. 1–6. (Cited on page 13.)

[33] F. Baccelli and B. Błaszczyszyn, *Stochastic Geometry and Wireless Networks*. Now Publishers Inc, 2010, vol. 1. (Cited on page 14.)

[34] G. Athanasiou, P. C. Weeraddana, C. Fischione, and L. Tassiulas, "Optimizing Client Association for Load Balancing and Fairness in Millimeter-Wave Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 23, no. 3, pp. 836–850, June 2015. (Cited on pages 14 and 23.)

[35] K. Shen and W. Yu, "Fractional Programming for Communication Systems—Part I: Power Control and Beamforming," *IEEE Transactions on Signal Processing*, vol. 66, no. 10, pp. 2616–2630, 2018. (Cited on page 14.)

[36] A. Alizadeh and M. Vu, "Load Balancing User Association in Millimeter Wave MIMO Networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 6, pp. 2932–2945, 2019. (Cited on pages 14 and 23.)

[37] Y. Liu, X. Fang, M. Xiao, and S. Mumtaz, "Decentralized Beam Pair Selection in Multi-Beam Millimeter-Wave Networks," *IEEE Transactions on Communications*, vol. 66, no. 6, pp. 2722–2737, June 2018. (Cited on pages 14, 23 and 55.)

[38] G. Ghatak, A. De Domenico, and M. Coupechoux, "Coverage Analysis and Load Balancing in HetNets With Millimeter Wave Multi-RAT Small Cells," *IEEE Transactions on Wireless Communications*, vol. 17, no. 5, pp. 3154–3169, May 2018. (Cited on page 14.)

[39] P. Zhou, X. Fang, X. Wang, Y. Long, R. He, and X. Han, "Deep Learning-Based Beam Management and Interference Coordination in Dense mmWave Networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 1, pp. 592–603, Jan 2019. (Cited on pages 14, 23, 31, 48 and 60.)

[40] N. Zhao, Y.-C. Liang, D. Niyato, Y. Pei, M. Wu, and Y. Jiang, "Deep Reinforcement Learning for User Association and Resource Allocation in Heterogeneous Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 11, pp. 5141–5152, 2019. (Cited on pages 14, 23, 27, 48, 54, 58 and 110.)

[41] R. Liu, M. Lee, G. Yu, and G. Y. Li, "User association for millimeter-wave networks: A machine learning approach," *IEEE Transactions on Communications*, vol. 68, no. 7, pp. 4162–4174, 2020. (Cited on pages 14 and 23.)

[42] Y. Li, J. G. Andrews, F. Baccelli, T. D. Novlan, and C. J. Zhang, "Design and Analysis of Initial Access in Millimeter Wave Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 10, pp. 6409–6425, Oct 2017. (Cited on page 16.)

[43] T. Bai and R. W. Heath, "Coverage and Rate Analysis for Millimeter-Wave Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 14, no. 2, pp. 1100–1114, 2015. (Cited on page 16.)

[44] R. Srikant and L. Ying, *Communication Networks: An Optimization, Control and Stochastic Networks Perspective*. USA: Cambridge University Press, 2014. (Cited on pages 18, 19 and 110.)

[45] D. Liu, L. Wang, Y. Chen, M. Elkashlan, K. Wong, R. Schober, and L. Hanzo, "User Association in 5G Networks: A Survey and an Outlook," *IEEE Communications Surveys Tutorials*, vol. 18, no. 2, pp. 1018–1044, 2016. (Cited on pages 19, 20 and 23.)

[46] 3GPP TR 36.300 V16.5.0, "Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2 (Release 16)," March 2021. (Cited on page 19.)

[47] D. D. Nguyen, H. X. Nguyen, and L. B. White, "Evaluating Performance of RAT Selection Algorithms for 5G Hetnets," *IEEE Access*, vol. 6, pp. 61 212–61 222, 2018. (Cited on page 20.)

[48] G. Caso, Ö. Alay, G. C. Ferrante, L. D. Nardis, M.-G. D. Benedetto, and A. Brunstrom, "User-Centric Radio Access Technology Selection: A Survey of Game Theory Models and Multi-Agent Learning Algorithms," *IEEE Access*, vol. 9, pp. 84 417–84 464, 2021. (Cited on pages 20 and 23.)

[49] M. S. Mollel, A. I. Abubakar, M. Ozturk, S. F. Kaijage, M. Kisangiri, S. Hussain, M. A. Imran, and Q. H. Abbasi, "A Survey of Machine Learning Applications to Handover Management in 5G and Beyond," *IEEE Access*, vol. 9, pp. 45 770–45 802, 2021. (Cited on page 20.)

[50] W. Wu, "100 prisoners and a lightbulb," *Technical report, OCF, UC Berkeley*, 2002. (Cited on page 22.)

[51] J. N. Foerster, Y. M. Assael, N. de Freitas, and S. Whiteson, "Learning to Communicate to Solve Riddles with Deep Distributed Recurrent Q-Networks," in *IJCAI Workshop on Deep Reinforcement Learning: Frontiers and Challenges*, July 2016. (Cited on page 23.)

[52] Q. Ye, B. Rong, Y. Chen, M. Al-Shalash, C. Caramanis, and J. G. Andrews, "User Association for Load Balancing in Heterogeneous Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2706–2716, 2013. (Cited on page 23.)

[53] A. Alizadeh and M. Vu, "Distributed User Association in B5G Networks Using Early Acceptance Matching Game," *IEEE Transactions on Wireless Communications*, vol. 20, no. 4, pp. 2428–2441, 2021. (Cited on page 23.)

[54] A. Alizadeh and M. Vu, "Multi-Armed Bandit Load Balancing User Association in 5G Cellular HetNets," in *IEEE Global Communications Conference (GLOBECOM)*, 2020, pp. 1–6. (Cited on page 23.)

[55] Q. Mao, F. Hu, and Q. Hao, "Deep Learning for Intelligent Wireless Networks: A Comprehensive Survey," *IEEE Communications Surveys Tutorials*, vol. 20, no. 4, pp. 2595–2621, Fourthquarter 2018. (Cited on page 23.)

[56] M. Sana, A. De Domenico, W. Yu, Y. Lostanlen, and E. Calvanese Strinati, "Multi-Agent Reinforcement Learning for Adaptive User Association in Dynamic mmWave Networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 10, pp. 6520–6534, 2020. (Cited on pages 24, 48 and 81.)

[57] M. Sana, A. De Domenico, and E. Calvanese Strinati, "Multi-Agent Deep Reinforcement Learning based User Association for Dense mmWave Networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*. HI, USA, Dec 2019, pp. 1–6. (Cited on page 24.)

[58] M. Sana, A. De Domenico, E. Calvanese Strinati, and A. Clemente, "Multi-Agent Deep Reinforcement Learning For Distributed Handover Management In Dense MmWave Networks," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Madrid, Spain, 2020, pp. 8976–8980. (Cited on page 24.)

[59] M. Sana and A. De Domenico, "Method for Associating User Equipment in a Cellular Network via Multi-Agent Reinforcement Learning," May 20 2021, US Patent App. 17/099,922. (Cited on page 24.)

[60] Z. Wang, T. Schaul, M. Hessel, H. Hasselt, M. Lanctot, and N. Freitas, "Dueling Network Architectures for Deep Reinforcement Learning," in *Proc. International Conference on Machine Learning (PMLR)*, vol. 48, Jun 2016, pp. 1995–2003. (Cited on page 25.)

[61] L. Matignon, G. J. Laurent, and N. Le Fort-Piat, "Independent Reinforcement Learners in Cooperative Markov Games: a Survey Regarding Coordination Problems," *The Knowledge Engineering Review*, vol. 27, no. 1, pp. 1–31, 2012. (Cited on pages 25, 26 and 30.)

[62] O. Naparstek and K. Cohen, "Deep Multi-User Reinforcement Learning for Distributed Dynamic Spectrum Access," *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 310–323, Jan 2019. (Cited on page 27.)

[63] S. Omidshafiei, J. Pazis, C. Amato, J. P. How, and J. Vian, "Deep Decentralized Multi-task Multi-Agent Reinforcement Learning under Partial Observability," in *Proc. International Conference on Machine Learning (ICML)*, vol. 70. PMLR, 06–11 Aug 2017, pp. 2681–2690. (Cited on pages 27, 28, 48, 56 and 110.)

[64] L. Matignon, G. J. Laurent, and N. Le Fort-Piat, "Hysteretic Q-Learning: An Algorithm for Decentralized Reinforcement Learning in Cooperative Multi-agent Teams," in *Proc. International Conference on Intelligent Robots and Systems (IEEE/RSJ)*, 2007, pp. 64–69. (Cited on pages 28, 29, 48, 57 and 110.)

[65] M. Hausknecht and P. Stone, "Deep Recurrent Q-Learning for Partially Observable MDPs," in *AAAI Fall Symposium on Sequential Decision Making for Intelligent Agents (AAAI-SDMIA15)*, November 2015. (Cited on page 28.)

[66] 3GPP TR 36.872, "Small Cell Enhancements for E-UTRA and E-UTRAN - Physical layer aspects (Release 12)," Dec 2013. (Cited on page 32.)

[67] T. S. Rappaport, Y. Xing, G. R. MacCartney, A. F. Molisch, E. Mellios, and J. Zhang, "Overview of Millimeter Wave Communications for Fifth-Generation (5G) Wireless Networks With a Focus on Propagation Models," *IEEE Transactions on Antennas and Propagation*, vol. 65, no. 12, pp. 6213–6230, 2017. (Cited on page 32.)

[68] 3GPP TR 36.814, "Evolved Universal Terrestrial Radio Access (E-UTRA) - Further advancements for E-UTRA physical layer aspects (Release 9)," Mars 2017. (Cited on page 32.)

[69] R. Chevillon, G. Andrieux, R. Négrier, and J. Diouris, "Spectral and Energy Efficiency Analysis of mmWave Communications With Channel Inversion in Outband D2D Network," *IEEE Access*, vol. 6, pp. 72 104–72 116, 2018. (Cited on page 37.)

[70] 3GPP TR 36.839 V11.1.0, "Evolved Universal Terrestrial Radio Access (E-UTRA); Mobility enhancements in heterogeneous networks (Release 11)," Jan 2013. (Cited on pages 39 and 42.)

[71] 3GPP TS 22.261 V17.0.1, "Service Requirements for Next Generation New Services and Markets (Release 17)," Oct 2019. (Cited on page 39.)

[72] L. Yan, H. Ding, L. Zhang, J. Liu, X. Fang, Y. Fang, M. Xiao, and X. Huang, "Machine Learning Based Handovers for Sub-6 GHz and mmWave Integrated Vehicular Networks," *IEEE Transactions on Wireless Communications*, pp. 1–1, 2019. (Cited on page 39.)

[73] Y. Koda, K. Yamamoto, T. Nishio, and M. Morikura, "Reinforcement learning based Predictive Handover for Pedestrian-aware mmWave Networks," in *Proc. IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Apr 2018, pp. 692–697. (Cited on page 39.)

[74] Z. Wang, L. Li, Y. Xu, H. Tian, and S. Cui, "Handover Optimization via Asynchronous Multi-User Deep Reinforcement Learning," in *Proc. IEEE International Conference on Communications (ICC)*, May 2018, pp. 1–6. (Cited on page 39.)

[75] H. Park, Y. Lee, T. Kim, B. Kim, and J. Lee, "Handover Mechanism in NR for Ultra-Reliable Low-Latency Communications," *IEEE Network*, vol. 32, no. 2, pp. 41–47, Mar 2018. (Cited on page 39.)

[76] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010. (Cited on page 47.)

[77] C. T. Nguyen, N. Van Huynh, N. H. Chu, Y. M. Saputra, D. T. Hoang, D. N. Nguyen, Q.-V. Pham, D. Niyato, E. Dutkiewicz, and W.-J. Hwang, "Transfer Learning for Future Wireless Networks: A Comprehensive Survey," *arXiv preprint arXiv:2102.07572*, 2021. (Cited on page 47.)

[78] Q. Zhao and D. Grace, "Agent Transfer Learning for Cognitive Resource Management on Multi-hop Backhaul Networks," in *Future Network Mobile Summit*, 2013, pp. 1–10. (Cited on page 47.)

[79] Q. Zhao and D. Grace, "Transfer learning for QoS aware topology management in energy efficient 5G cognitive radio networks," in *International Conference on 5G for Ubiquitous Connectivity*, 2014, pp. 152–157. (Cited on page 47.)

[80] Q. Zhao, D. Grace, and T. Clarke, "Transfer Learning and Cooperation Management: Balancing the Quality of Service and Information Exchange overhead in Cognitive Radio Networks," *Transactions on Emerging Telecommunications Technologies*, vol. 26, no. 2, pp. 290–301, 2015. (Cited on page 47.)

[81] Y. Wu, F. Hu, S. Kumar, J. D. Matyjas, Q. Sun, and Y. Zhu, "Apprenticeship Learning Based Spectrum Decision in Multi-Channel Wireless Mesh Networks with Multi-Beam Antennas," *IEEE Transactions on Mobile Computing*, vol. 16, no. 2, pp. 314–325, 2017. (Cited on page 47.)

[82] M. Sana, N. di Pietro, and E. C. Strinati, "Transferable and Distributed User Association Policies for 5G and Beyond Networks," in *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2021, pp. 966–971. (Cited on page 48.)

[83] A. A. Rusu, S. G. Colmenarejo, Ç. Gülçehre, G. Desjardins, J. Kirkpatrick, R. Pascanu, V. Mnih, K. Kavukcuoglu, and R. Hadsell, "Policy Distillation," in *Proc. International Conference on Learning Representations (ICLR)*, May 2016. (Cited on pages 49 and 50.)

[84] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008. (Cited on pages 55, 56, 82, 91 and 97.)

[85] L. Buşoniu, R. Babuška, and B. De Schutter, "Multi-agent Reinforcement Learning: An Overview," in *Innovations in multi-agent systems and applications-1*. Springer, 2010, pp. 183–221. (Cited on page 56.)

[86] J. Schulman, P. Moritz, S. Levine, M. I. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," *CoRR*, vol. abs/1506.02438, 2016. (Cited on page 56.)

[87] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms." *CoRR*, vol. abs/1707.06347, 2017. (Cited on pages 57, 59, 60 and 82.)

[88] E. Wijmans, A. Kadian, A. S. Morcos, S. Lee, I. Essa, D. Parikh, M. Savva, and D. Batra, "DD-PPO: Learning Near-Perfect PointGoal Navigators from 2.5 Billion Frames," in *ICLR*, 2020. (Cited on page 57.)

[89] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum Learning," in *Proc. International Conference on Machine Learning*, 2009, pp. 41–48. (Cited on page 67.)

[90] S. Ahmadi, *5G NR: Architecture, Technology, Implementation, and Operation of 3GPP New Radio Standards*. Elsevier Science, 2019. (Cited on page 71.)

[91] Q.-V. Pham, F. Fang, V. N. Ha, M. J. Piran, M. Le, L. B. Le, W.-J. Hwang, and Z. Ding, "A Survey of Multi-Access Edge Computing in 5G and Beyond: Fundamentals, Technology Integration, and State-of-the-Art," *IEEE Access*, vol. 8, pp. 116 974–117 017, 2020. (Cited on page 71.)

[92] L. Li, Q. Guan, L. Jin, and M. Guo, "Resource Allocation and Task Offloading for Heterogeneous Real-Time Tasks With Uncertain Duration Time in a Fog Queueing System," *IEEE Access*, vol. 7, pp. 9912–9925, 2019. (Cited on page 71.)

[93] L. Chen, S. Zhou, and J. Xu, "Energy Efficient Mobile Edge Computing in Dense Cellular Networks," in *Proc. IEEE International Conference on Communications (ICC)*, 2017, pp. 1–6. (Cited on page 71.)

[94] S. Wang, X. Zhang, Z. Yan, and W. Wenbo, "Cooperative Edge Computing with Sleep Control under Non-uniform Traffic in Mobile Edge Networks," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4295–4306, 2018. (Cited on page 71.)

[95] P. Chang and G. Miao, "Resource Provision for Energy-Efficient Mobile Edge Computing Systems," in *Proc. Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–6. (Cited on page 71.)

[96] Y. Nan, W. Li, W. Bao, F. C. Delicato, P. F. Pires, Y. Dou, and A. Y. Zomaya, "Adaptive Energy-Aware Computation Offloading for Cloud of Things Systems," *IEEE Access*, vol. 5, pp. 23 947–23 957, 2017. (Cited on page 71.)

[97] B. Yu, L. Pu, Q. Xie, J. Xu, and J. Zhang, "U-MEC: Energy-Efficient Mobile Edge Computing for IoT Applications in Ultra Dense Networks," in *Wireless Algorithms, Systems, and Applications*, 2018, pp. 622–634. (Cited on page 71.)

[98] S. Bi, L. Huang, H. Wang, and Y.-J. A. Zhang, "Lyapunov-guided Deep Reinforcement Learning for Stable Online Computation Offloading in Mobile-Edge Computing Networks," *IEEE Transactions on Wireless Communications*, pp. 1–1, 2021. (Cited on pages 72 and 82.)

[99] S. Bae, S. Han, and Y. Sung, "A Reinforcement Learning Formulation of the Lyapunov Optimization: Application to Edge Computing Systems with Queue Stability," *arXiv preprint arXiv:2012.07279*, 2020. (Cited on page 72.)

[100] M. Sana, M. Merluzzi, N. d. Pietro, and E. Calvanese Strinati, "Energy Efficient Edge Computing: When Lyapunov Meets Distributed Reinforcement Learning," in *Proc. IEEE International Conference on Communications Workshops (ICC Workshops)*, 2021, pp. 1–6. (Cited on page 72.)

[101] M. Merluzzi, P. Di Lorenzo, S. Barbarossa, and V. Frascolla, "Dynamic Computation Offloading in Multi-Access Edge Computing via Ultra-Reliable and Low-Latency Communications," *IEEE Transactions on Signal and Information Processing over Networks*, pp. 1–1, 2020. (Cited on page 74.)

[102] J. D. C. Little, "A Proof for the Queuing Formula: $L = \lambda W$," *Oper. Res.*, vol. 9, no. 3, p. 383–387, Jun. 1961. (Cited on page 74.)

[103] E. Le Sueur and G. Heiser, "Dynamic Voltage and Frequency Scaling: The Laws of Diminishing Returns," in *Proc. HotPower*, 2010, pp. 1–8. (Cited on page 75.)

[104] T. D. Burd and R. W. Brodersen, "Processor design for portable systems," *J. VLSI Signal Process. Syst.*, vol. 13, no. 2-3, pp. 203–221, 8 1996. (Cited on page 75.)

[105] M. J. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan & Claypool Publishers, 2010. (Cited on pages 76, 77, 78, 116 and 117.)

[106] M. Z. Chowdhury, M. Shahjalal, S. Ahmed, and Y. M. Jang, "6G Wireless Communication Systems: Applications, Requirements, Technologies, Challenges, and Research Directions," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 957–975, 2020. (Cited on page 89.)

[107] P. Popovski, F. Chiariotti, K. Huang, A. E. Kalør, M. Kountouris, N. Pappas, and B. Soret, "A Perspective on Time towards Wireless 6G," *arXiv preprint arXiv:2106.04314*, 2021. (Cited on page 89.)

[108] D. Belot, J. L. González Jiménez, E. Mercier, and J.-B. Doré, "Spectrum Above 90 GHz for Wireless Connectivity: Opportunities and Challenges for 6G," *Microwave Journal*, vol. 63, no. 9, 2020. (Cited on page 89.)

[109] K. B. Letaief, W. Chen, Y. Shi, J. Zhang, and Y.-J. A. Zhang, "The Roadmap to 6G: AI Empowered Wireless Networks," *IEEE Communications Magazine*, vol. 57, no. 8, pp. 84–90, 2019. (Cited on page 89.)

[110] C. E. Shannon, "A Mathematical Theory of Communication," *The Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948. (Cited on page 90.)

[111] W. Weaver, "Recent Contributions to the Mathematical Theory of Communication," *ETC: a review of general semantics*, pp. 261–281, 1953. (Cited on page 90.)

[112] Y. Bar-Hillel and R. Carnap, "An Outline of a Theory of Semantic Information," *Language and information: Selected essays on their theory and application. London*, 1964. (Cited on page 90.)

[113] J. Bao, P. Basu, M. Dean, C. Partridge, A. Swami, W. Leland, and J. A. Hendler, "Towards a Theory of Semantic Communication," in *IEEE Network Science Workshop*, 2011, pp. 110–117. (Cited on pages 90, 91, 95 and 102.)

[114] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, "Deep Learning Enabled Semantic Communication Systems," *IEEE Transactions on Signal Processing*, pp. 1–1, 2021. (Cited on pages 90, 91, 96, 99, 100 and 101.)

[115] G. Shi, D. Gao, X. Song, J. Chai, M. Yang, X. Xie, L. Li, and X. Li, "A new communication paradigm: from bit accuracy to semantic fidelity," *CoRR*, vol. abs/2101.12649, 2021. (Cited on page 90.)

[116] T. Berners-Lee, J. Hendler, and O. Lassila., "The Semantic Web," *Scientific American*, pp. 34–43, 2001. (Cited on page 90.)

[117] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of Information-centric Networking," *IEEE Communications Magazine*, vol. 50, no. 7, pp. 26–36, 2012. (Cited on page 90.)

[118] M. Kountouris and N. Pappas, "Semantics-Empowered Communication for Networked Intelligent Systems," *CoRR*, vol. abs/2007.11579, 2020. (Cited on pages 91 and 99.)

[119] A. Kosta, N. Pappas, and V. Angelakis, "Age of Information: A New Concept, Metric, and Tool," *Foundations and Trends in Networking*, vol. 12, no. 3, pp. 162–259, 2017. (Cited on page 91.)

[120] Z. Weng, Z. Qin, and G. Y. Li, "Semantic Communications for Speech Signals," *arXiv preprint arXiv:2012.05369*, 2020. (Cited on page 91.)

[121] P. Jiang, C.-K. Wen, S. Jin, and G. Y. Li, "Deep Source-Channel Coding for Sentence Semantic Transmission with HARQ," *arXiv preprint arXiv:2106.03009*, 2021. (Cited on page 91.)

[122] M. Sana and E. C. Strinati, "Learning Semantics: An Opportunity for Effective 6G Communications," *arXiv preprint arXiv:2110.08049*, 2022. (Cited on page 92.)

[123] L. Floridi, *Philosophical Conceptions of Information.* Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 13–53. (Cited on page 92.)

[124] M. Chein and M.-L. Mugnier, *Graph-Based Knowledge Representation: Computational Foundations of Conceptual Graphs*, ser. Advanced Information and Knowledge Processing. Springer, 2008. (Cited on pages 92 and 93.)

[125] P. Basu, J. Bao, M. Dean, and J. Hendler, "Preserving Quality of Information by using Semantic Relationships," *Pervasive and Mobile Computing*, vol. 11, pp. 188–202, 2014. (Cited on page 94.)

[126] C. E. Shannon, "Coding Theorems for a Discrete Source with a Fidelity Criterion," *IRE Nat. Conv. Rec*, vol. 4(142-163), p. 1, 1959. (Cited on page 94.)

[127] M. I. Belghazi, A. Baratin, S. Rajeshwar, S. Ozair, Y. Bengio, A. Courville, and D. Hjelm, "Mutual Information Neural Estimation," in *Proc. International Conference on Machine Learning*, vol. 80. PMLR, 10–15 Jul 2018, pp. 531–540. (Cited on page 96.)

[128] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A Method for Automatic Evaluation of Machine Translation," in *Proc. Association for Computational Linguistics*, ser. ACL '02, 2002, p. 311–318. (Cited on page 99.)

[129] A. Fotouhi, H. Qiang, M. Ding, M. Hassan, L. G. Giordano, A. Garcia-Rodriguez, and J. Yuan, "Survey on UAV Cellular Communications: Practical Aspects, Standardization Advancements, Regulation, and Security Challenges," *IEEE Communications Surveys Tutorials*, vol. 21, no. 4, pp. 3417–3442, 2019. (Cited on page 104.)

[130] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless Communications with Unmanned Aerial Vehicles: Opportunities and Challenges," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 36–42, 2016. (Cited on page 105.)

[131] E. Puiutta and E. M. S. P. Veith, "Explainable Reinforcement Learning: A Survey," in *Machine Learning and Knowledge Extraction*, A. Holzinger, P. Kieseberg, A. M. Tjoa, and E. Weippl, Eds. Cham: Springer International Publishing, 2020, pp. 77–95. (Cited on page 105.)

[132] M. Chen, H. V. Poor, W. Saad, and S. Cui, "Convergence Time Optimization for Federated Learning Over Wireless Networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 4, pp. 2457–2471, 2021. (Cited on page 105.)

[133] M. A. Uusitalo, M. Ericson, B. Richerzhagen, E. U. Soykan, P. Rugeland, G. Fettweis, D. Sabella, G. Wikström, M. Boldi, M.-H. Hamon *et al.*, "Hexa-X The European 6G flagship project," in *2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. IEEE, 2021, pp. 580–585. (Cited on page 105.)