Protocol Emergence with Multi-Agent Reinforcement Learning

Mateus Pontes Mota

N°d'ordre NNT : 2024ISAL0011

THESE de DOCTORAT DE L'INSA LYON,
membre de l'Université de Lyon

Ecole Doctorale N° 160
Électronique, Électrotechnique et Automatique
(ED EEA)

Spécialité/ discipline de doctorat :
Traitement du Signal et de l'Image

Soutenue publiquement/à huis clos le 23/01/2024, par :
Mateus Pontes Mota

Protocol Emergence with Multi-Agent Reinforcement Learning

Devant le jury composé de:

Rapporteurs:		
Belmega, Veronica	Professeure	Université Gustave Eiffel, France
Stefanovic, Cedomir	Professeur	Université de Aalborg, Danemark
Examinateurs:		
Navarro, Monica	Chargée de Recherche, HDR	CTTC, Espagne
Mary, Philippe	Professeur	Université de Rennes, France
Directeur de thèse:		
Gorce, Jean-Marie	Professeur	Université de Lyon, France
Co-encadrant de thèse:		
Valcarce, Alvaro	Chargé de Recherche	Nokia Bell Labs, France

INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Département de la Formation par la Recherche et des Études Doctorales (**FEDORA**)

Bâtiment INSA direction, 1er étage
37, av. J. Capelle
69621 Villeurbanne Cédex
fedora@insa-lyon.fr

Référence : TH1067_PONTES MOTA Mateus

L'INSA Lyon a mis en place une procédure de contrôle systématique via un outil de détection de similitudes (logiciel Compilatio). Après le dépôt du manuscrit de thèse, celui-ci est analysé par l'outil. Pour tout taux de similarité supérieur à 10%, le manuscrit est vérifié par l'équipe de FEDORA. Il s'agit notamment d'exclure les auto-citations, à condition qu'elles soient correctement référencées avec citation expresse dans le manuscrit.

Par ce document, il est attesté que ce manuscrit, dans la forme communiquée par la personne doctorante à l'INSA Lyon, satisfait aux exigences de l'Etablissement concernant le taux maximal de similitude admissible.

Département FEDORA – INSA Lyon - Ecoles Doctorales

SIGLE	ECOLE DOCTORALE	NOM ET COORDONNEES DU RESPONSABLE
ED 206 CHIMIE	**CHIMIE DE LYON** https://www.edchimie-lyon.fr Sec. : Renée EL MELHEM Bât. Blaise PASCAL, 3e étage secretariat@edchimie-lyon.fr	**M. Stéphane DANIELE** C2P2-CPE LYON-UMR 5265 Bâtiment F308, BP 2077 43 Boulevard du 11 novembre 1918 69616 Villeurbanne directeur@edchimie-lyon.fr
ED 341 E2M2	**ÉVOLUTION, ÉCOSYSTÈME, MICROBIOLOGIE, MODÉLISATION** http://e2m2.universite-lyon.fr Sec. : Bénédicte LANZA Bât. Atrium, UCB Lyon 1 Tél : 04.72.44.83.62 secretariat.e2m2@univ-lyon1.fr	**Mme Sandrine CHARLES** Université Claude Bernard Lyon 1 UFR Biosciences Bâtiment Mendel 43, boulevard du 11 Novembre 1918 69622 Villeurbanne CEDEX e2m2.codir@listes.univ-lyon1.fr
ED 205 EDISS	**INTERDISCIPLINAIRE SCIENCES-SANTÉ** http://ediss.universite-lyon.fr Sec. : Bénédicte LANZA Bât. Atrium, UCB Lyon 1 Tél : 04.72.44.83.62 secretariat.ediss@univ-lyon1.fr	**Mme Sylvie RICARD-BLUM** Laboratoire ICBMS - UMR 5246 CNRS - Université Lyon 1 Bâtiment Raulin - 2ème étage Nord 43 Boulevard du 11 novembre 1918 69622 Villeurbanne Cedex Tél : +33(0)4 72 44 82 32 sylvie.ricard-blum@univ-lyon1.fr
ED 34 EDML	**MATÉRIAUX DE LYON** http://ed34.universite-lyon.fr Sec. : Yann DE ORDENANA Tél : 04.72.18.62.44 yann.de-ordenana@ec-lyon.fr	**M. Stéphane BENAYOUN** Ecole Centrale de Lyon Laboratoire LTDS 36 avenue Guy de Collongue 69134 Ecully CEDEX Tél : 04.72.18.64.37 stephane.benayoun@ec-lyon.fr
ED 160 EEA	**ÉLECTRONIQUE, ÉLECTROTECHNIQUE, AUTOMATIQUE** https://edeea.universite-lyon.fr Sec. : Philomène TRECOURT Bâtiment Direction INSA Lyon Tél : 04.72.43.71.70 secretariat.edeea@insa-lyon.fr	**M. Philippe DELACHARTRE** INSA LYON Laboratoire CREATIS Bâtiment Blaise Pascal, 7 avenue Jean Capelle 69621 Villeurbanne CEDEX Tél : 04.72.43.88.63 philippe.delachartre@insa-lyon.fr
ED 512 INFOMATHS	**INFORMATIQUE ET MATHÉMATIQUES** http://edinfomaths.universite-lyon.fr Sec. : Renée EL MELHEM Bât. Blaise PASCAL, 3e étage Tél : 04.72.43.80.46 infomaths@univ-lyon1.fr	**M. Hamamache KHEDDOUCI** Université Claude Bernard Lyon 1 Bât. Nautibus 43, Boulevard du 11 novembre 1918 69 622 Villeurbanne Cedex France Tél : 04.72.44.83.69 direction.infomaths@listes.univ-lyon1.fr
ED 162 MEGA	**MÉCANIQUE, ÉNERGÉTIQUE, GÉNIE CIVIL, ACOUSTIQUE** http://edmega.universite-lyon.fr Sec. : Philomène TRECOURT Tél : 04.72.43.71.70 Bâtiment Direction INSA Lyon mega@insa-lyon.fr	**M. Etienne PARIZET** INSA Lyon Laboratoire LVA Bâtiment St. Exupéry 25 bis av. Jean Capelle 69621 Villeurbanne CEDEX etienne.parizet@insa-lyon.fr
ED 483 ScSo	**ScSo**[1] https://edsciencessociales.universite-lyon.fr Sec. : Mélina FAVETON Tél : 04.78.69.77.79 melina.faveton@univ-lyon2.fr	**M. Bruno MILLY** (INSA : J.Y. TOUSSAINT) Univ. Lyon 2 Campus Berges du Rhône 18, quai Claude Bernard 69365 LYON CEDEX 07 Bureau BEL 319 bruno.milly@univ-lyon2.fr

[1] ScSo : Histoire, Géographie, Aménagement, Urbanisme, Archéologie, Science politique, Sociologie, Anthropologie

Protocol Emergence with Multi-Agent Reinforcement Learning

Mateus Pontes Mota

Academic Advisor: Jean-Marie Gorce

Industry Advisor: Alvaro Valcarce

Doctoral School of Electronics, Electrical Engineering, Automation

National Institute of Applied Sciences of Lyon

November 2023

Abstract

In this work, we propose an automated protocol design technique called protocol emergence. In protocol emergence, the network nodes exchange control messages in order to coordinate to deliver data across the network, but without any prior agreement on the meaning of those messages. This can be seen as a joint signaling and network optimization technique. With protocol emergence, it is possible to reduce the signaling overhead, appealing to massive machine-type communications (mMTC) scenarios, and design application-tailored protocols, which can be useful for somewhat stable scenarios such as indoor factory.

Firstly, the fundamentals of protocol emergence are presented by introducing a framework describing methods of protocol performance evaluation, characterization, cross-node coordination and interpretation. This framework is studied in a slotted multiple-access problem.

In the second part of this work, we evaluate the performance of protocol emergence in a scenario involving contiguous resource allocation. This scenario is used to evaluate the learning capabilities and limitations of protocol emergence, illustrating the robustness to some parameters and the challenges involved with user equipment (UE) scalability, for example.

In the third part of this work, we evaluate protocol emergence under signaling constraints, in a non-contiguous allocation scenario with intermittent signaling. In this study, we focus on producing a method in which the control bit-rate used by the protocol can be controlled. The results highlight the effect of the signaling cost on reducing the control bit-rate and the effect it has on coordination and performance.

In conclusion, this thesis frames the protocol emergence problem, making progress on how to autonomously learn an effective control signaling scheme, while also addressing some of the challenges in protocol design and opening new and exciting avenues for future research. Such progress include a framework for producing and studying emergent protocols, a deep analysis of the learning capabilities and limitations of such methods and how to control the amount of signaling used by the emerged protocols.

Keywords— Multi-Agent Reinforcement Learning, Protocol Emergence, Wireless Communications, Scheduling

Acknowledgements

It has been a long journey, and I would like to offer my sincere thanks to all the people that helped me. First and foremost, I would like to thank my supervisors, Dr. Jean-Marie Gorce and Dr. Alvaro Valcarce, for their guidance and support throughout my PhD. I would also like to thank Dr. Mehdi Bennis and his group for hosting my in the Oulu University and for the colaborations that followed. I extend my thanks to Nokia Bell Labs and all my colleagues over this time, Vinicius, Eloise, Mathieu, Bryan, Ibtissam, Nicolas, Pavan, Alix, Abanoub and others. I would like to thank all the members of the H2020 Windmill project for the memorable experiences we shared. I would also like to thank the European Union for funding my research. I am grateful to all my friends, from Brazil and also from across Europe, thanks! Last but not least, I would like to thank my family and Sarah for their support and love.

Table of Contents

List of Figures

List of Tables

List of Abbreviations and Acronyms

1G first generation

2G second generation

3G third generation

4G fourth generation

5G fifth generation

6G sixth generation

AC adaptive coding

ACK acknowledgement

AMC adaptive modulation and coding

API application programming interface

ARQ automatic repeat request

BS base station

BWP bandwidth part

CBLER control block error rate

CDM code division multiplexing

CDMA code division multiple access

CI confidence interval

CQI channel quality indicator

CSI channel state information

CTDE centralized training and decentralized execution

CUPS control and user plane separation

D2RL deep dense architecture for reinforcement learning

DCI downlink control information

DCM downlink control message

DDPG deep deterministic policy gradient

Dec-POMDP decentralized partially observable Markov decision process

DIAL differentiable inter-agent learning

DL downlink

DQN deep Q-network

DRQN deep recurrent Q-network

EDGE Enhanced Data Rates for GSM Evolution

EE energy efficiency

FDM frequency division multiplexing

FDMA frequency-division multiple access

GPRS general packet radio service

GSM global system for mobile communications

HD high definition

HSPA high-speed packet access

IC instantaneous coordination

IIoT industrial internet-of-things

IR immediate reply

JFI Jain's fairness index

JSRL jump-start reinforcement learning

KPI key performance indicator

L2C learning to communicate

LTE long term evolution

MAC medium access control

MADDPG multi-agent deep deterministic policy gradient

MARL multi-agent reinforcement learning

MCS modulation and coding scheme

MDP Markov decision process

MIMO multiple-input multiple-output

ML machine learning

MLP multilayer perceptron

mMTC massive machine-type communications

MU multi-user

NACK negative acknowledgement

NR new radio

OFDM orthogonal frequency-division multiplexing

OFDMA orthogonal frequency-division multiple access

PDU protocol data unit

PF proportional-fair

PHR power headroom report

PHY physical layer

PMI precoding matrix indicator

POMDP partially observable Markov decision process

PRB physical resource block

QoS quality of service

RB resource block

RBG resource block group

RE resource element

ReLU rectified linear unit

RI rank indicator

RIAL reinforced inter-agent learning

RIV resource indicator value

RL reinforcement learning

RRC radio resource control

SC-FDMA single carrier frequency-division multiple access

SCS subcarrier spacing

SDM space division multiplexing

SDMA space-division multiple access

SDU service data unit

SG scheduling grant

SNR signal-to-noise ratio

SR scheduling request

SRS sounding reference signal

TB transport block

TBLER transport block error rate

TBS transport block size

TDM time division multiplexing

TDMA time division multiple access

TPC transmit power control

TPMI transmit precoding matrix indicator

TTI transmission time interval

UCM uplink control message

UE user equipment

UL uplink

UL-SCH uplink shared channel

UMTS universal mobile telecommunications system

uRLLC ultra-reliable and low-latency communications

USC uplink slotted shared channel

V2X vehicle-to-everything

VRB virtual resource block

XAI explainable artificial intelligence

XR extended reality

List of Symbols

P resource block group size

μ numerology

$N_{\text{symb}}^{\text{sh}}$ number of symbols in a slot

$N_{\text{slot}}^{\text{subframe}}$ number of slots in a subframe

$N_{\text{slot}}^{\text{frame}}$ number of slots in a frame

N_{RBG} number of resource block groups

$N_{\text{BWP}}^{\text{size}}$ number of resource block in the bandwidth part

$N_{\text{BWP}}^{\text{start}}$ starting resource block for the bandwidth part

RB_{start} starting resource block

L_{RBs} length of transmission in resource blocks

s environment state

a action

r reward signal

o observation

h action-observation history

π policy

$\mathcal{R}$ return

γ discount factor of MDP

x agent state

ω policy network

θ actor network parameters

φ critic network parameters

ι soft-update parameter

U number of users

B buffer size

p_{a} probability of arriving a new packet

L_{TB} transport block size

T maximum number of TTIs

λ average number of packets to transmit

V_{DCM} downlink control message vocabulary size

V_{UCM} uplink control message vocabulary size

Υ bitlength

t time step

T_{TTI} transmission time interval duration

Γ collision-rate

N_{c} number of collisions

Δ delay

ξ received time

τ generated time

Λ average delay

Ψ average reliability

I mutual information

n uplink control message

m downlink control message

χ concatenation of action, observation and message

κ reward shaping factor

H entropy

ζ Gumbel-softmax temperature factor

N_{train} number of training episodes

N_{eval} number of evaluation episodes

N_{test} number of test episodes

N_{rep} number of test episodes

M number of resource blocks

L_{SDU} size of SDU in bits

λ_u average number of bits to transmit per UE

ρ path loss

σ noise standard deviation

η spectral efficiency

v_r code rate

ϱ_m modulation order

n_{RB} number of allocated resource blocks

$N_{\mathrm{sc}}^{\mathrm{RB}}$ number of subcarriers in a resource block

G goodput

c channel state relative to resource block

g total amount of data received

$\tilde{n}_{\mathrm{RB}}$ number of requested resource blocks

F geometric mean of goodputs

R_{cp} control channel throughput

Chapter 1

Introduction

Contents

1.1 Background

Wireless communications became an integral component of the modern world, as society got more interconnected over time. The progress of wireless networks is motivated by the need to enhance not only the current performance in existing applications, but also in response to emerging use-cases and their associated requirements [1]. The evolutionary tree of cellular networks starts in the 1980s with the analog signals for voice call used in first generation (1G). In the 1990s, the digital system of second generation (2G) global system for mobile communications (GSM) integrated services such as text messages and digital voice. The third generation (3G) arrived in the early 2000s

and provided services such as video calls, locating services and mobile television, with its first standard, universal mobile telecommunications system (UMTS), being released in 2001. The advent of fourth generation (4G) around the 2010s allowed high data rates applications, such as video streaming and high definition (HD) television, while fifth generation (5G) diversified mobile services beyond humans, to connect things [2], [3].

For the purpose of accommodating multiple services and requirements, the systems need to operate in orderly manner, while also providing a degree of flexibility. A *communication protocol* is a set of rules and conventions dictating how data is transmitted and received in a network. For the intricate protocols used in cellular networks, it is important that the nodes exchange system information and coordinate, thus ensuring that the system operates efficiently and reliably. This communication of the nodes is done by means of *control signaling*, which is an integral part of protocols. The communication protocols define the structure of control signaling messages, their exchange process, and the responses triggered by particular control signals.

As the communication systems evolve and employ more techniques, the control signaling and protocols need to be updated in order to accommodate these new techniques. For example, the evolved 2G in general packet radio service (GPRS) introduced adaptive coding (AC), which later evolved into adaptive modulation and coding (AMC) used in 2G Enhanced Data Rates for GSM Evolution (EDGE) [4], and, initially it was done at the receiver side with the modulation and coding scheme (MCS) decision being informed to the transmitter [5]. This procedure changed in 3G high-speed packet access (HSPA), with the decision being taken by the base station (BS) even in downlink, with the user equipment (UE) indicating its channel conditions by a channel quality indicator (CQI) carried via uplink control signaling, while the chosen MCS is informed

in a downlink control message [6]. Although multiple-input multiple-output (MIMO) was present in previous releases, it became a key technology in 4G long term evolution (LTE), leading to modifications in the control signaling, as the number of transmission layers and antenna-ports needed to be coordinated [7]. In 5G new radio (NR), the support for MIMO increased, requiring a higher degree and flexibility, leading to an increase in the signaling necessary to support it.

This system advancement poses a significant challenge in terms of protocol design, as the systems need more information exchanged to perform well while also having flexibility to accommodate different services and requirements. To illustrate this, the first release of LTE had 7 downlink control information (DCI) formats [8], while the current release of NR, 17.6.0, has a total of 19 DCI formats [9]. In release 14 of LTE, the control and user plane separation (CUPS) was introduced [10], which allowed the control plane and user plane to be decoupled. This feature allows more flexibility to the control plane, which is responsible for the exchange of the control-relevant information, and, in terms of protocol design, it enabled the independent scaling of control and user planes.

Protocol Design

Although the control signaling and communication protocols remain an integral part of cellular systems, traditional protocol design approaches are often based on static and predetermined rules. With the new services provided by 5G and the future challenges for sixth generation (6G), general-purpose protocols may struggle to the dynamic demands of future applications, while application-tailored protocols can provide superior performance. Considering this, automated design of protocols is a

promising subject, with learning based methods being of particular interest. The current thesis explores this idea by leveraging machine learning (ML), more specifically multi-agent reinforcement learning (MARL), to be able to emerge wireless communication protocols. It explores the intersection of protocol design and MARL, with the goal of enabling autonomous, adaptive, and intelligent protocol emergence.

MARL and Communication Emergence

MARL enables protocol emergence through the use of learning to communicate (L2C) techniques, also called emergent communication [11]. In reinforcement learning (RL), an agent has to solve a task while optimizing a reward function related to the task, while MARL extends this paradigm to a scenario with multiple agents. L2C is most often applied in fully cooperative setting, where all agents receive the same reward, and communication is used as a means to help them coordinate and achieve their goals. Although previous research in L2C existed [12], [13], research in this field gathered more attention after 2016 due to the seminal works Foerster *et al.* [14] and Sukhbaatar *et al.* [15].

The L2C techniques are of particular interest to protocol emergence since they enable the jointly learning of communication, or language, and of a cooperative task. As both are learned together, the emerged communication is optimized to the specific task. In wireless communications, we view the control signalling as the language of the network, allowing the nodes to solve their task. As such, protocol emergence can be seen as learning a network language together with the network optimization problem, in the current thesis case, scheduling. Due to this, L2C techniques and MARL are the framework of choice for protocol emergence.

Protocol Emergence and Multiple-Access

In this thesis, protocol emergence is applied to uplink multiple-access problems [16], as such, this work studies medium access control (MAC) protocol emergence. Multiple-access techniques determine how different devices can access the communication channel, transmit their data and avoid interference with one another, enabling multiple devices to share the same wireless medium simultaneously. Since the wireless spectrum is limited, an efficient and organized usage of resources is critical in wireless networks. This problem is interesting to protocol emergence due to the degree of coordination needed, as the network has to coordinate the multiple transmitting devices through control signaling, orchestrating the resource sharing.

Research Significance

The overarching idea in the present thesis is to let the meaning of signaling messages emerge in a reinforced manner at the network nodes while they perform a communication task. By *meaning* of messages we mean a mapping between the control messages received at time $t < t'$ and the actions (control signaling actions, as well as channel-access actions) taken at time $t \geq t'$. As the signaling is jointly learned with the task, a task-specific protocol can be produced this way. We expect protocol learning techniques to contribute increasingly to the native intelligence of the air interface, whose main advantages are:

- Automation: Reproducibility and speed can be achieved with automated protocol generation.
- Customization: The ability to design application-tailored protocols.

- Optimality: Automated search of the signaling space, enabling the selection of protocols with the lowest overhead.

1.2 State-of-the-Art

The related research presented in this section is divided into topics, as this thesis touches on a number of different subjects.

Control Signaling

In [17], a scheme to reduce signaling in a scheduling problem is proposed by exploring the correlation between the channel state information (CSI) reports and the scheduling decisions. In proposed scheme, the UEs make tentative scheduling themselves and the BS, instead of transmitting scheduling assignments, it transmits agreement maps, which need less resources to be transmitted, therefore reducing the amount of signaling needed for scheduling. The scheduling assignment correlation between UEs is exploited in [18] in order to reduce the signaling traffic. It exploits the fact that a single UE is scheduled per resource and that all UEs can listen to the information sent by the BS, in downlink, thus enabling the high signal-to-noise ratio (SNR) UEs to infer their resources from the information gathered from the other UEs assignments. An example of a method for designing a protocol reducing the control signaling can be found in [19], in which a MAC protocol for collaborative sensing is designed by finding closed form solutions for the probability of detection and then choosing the best parameters for minimizing signaling overhead.

Emergent Communication

In [14], two solutions for learning discrete communication protocols to carry out a cooperative task are proposed. The first, reinforced inter-agent learning (RIAL) is based on deep recurrent Q-networks (DRQNs) combined with independent learners with the action depending on the observation and communication message. Differentiable inter-agent learning (DIAL) adds the relaxation to allow the messages to become differentiable, thus, during execution the messages are discrete, while during training they are relaxed to a continuous valued. In [15], a method allowing the learning of purely continuous communication protocols is proposed, with results indicating that agents learn to encode the meaningful information in a sparse communication protocol. In both works, the communication messages are broadcasted to all agents.

Attention mechanisms were applied to L2C to allow agents to determine with whom, when and what to communicate. For example, in [20], a selective communication approach is utilized to allow agents to decide when communication is needed, form collaborative groups to communicate with and how to select the information needed. In [21], a multi-round communication architecture is proposed, where agents have to carry out a cooperative task, learn what messages to send and whom to address them to. An architecture to control when to communicate is proposed for mixed cooperative-competitive setting in [22], where a gating mechanism allows agents to block their communications.

Although task performance may increase when adding communication capabilities to the agents, an evaluation of the communication abilities of the agents is needed. In [23], the authors study metrics that measure the quality of the learned communication and provided recommendations on their use. In this context, [24] uses similar

metrics not to evaluate communication, but as means to introduce inductive biases for positive signaling and listening, as such, such metrics serve as an optimization objective.

Goal-oriented Communications

In [25], [26], L2C is used to emerge a coding scheme by joint learning of communication and cooperation to solve a task with the help of a noisy communication channel. The proposition of both works is to emerge a coding scheme that is tailored to the application. This idea is expanded further in [27], allowing the learning of a physical layer (PHY) in an effective communication scenario, by learning both the channel coding and a modulation. In [28], a MARL solution with a message module, a communication module and an action module is proposed for a task-oriented communication scenario in which the channel allocation is also considered. It uses an attention mechanism to enable an agent to pay more attention to messages more relevant to itself, enabling a more efficient use of communication.

Multiple-Access and RL

RL has been applied to random access schemes in a number of scenarios, such as machine-type communication [29], [30], satellite networks [31] and ad-hoc networks [32]. A listen-before-talk protocol based on MARL is proposed for distributed channel access in [33].

MAC Protocol Design

An example of a classical approach for MAC protocol design is the meta-MAC method proposed in [34], which combines any group of MAC protocols into a unique upper layer protocol. Other examples include the self-adjusting approach employed in [35] and the adaptive distributed method in [36].

RL has also been used for MAC protocol design. For instance, [37] proposes a meta-protocol to decide which MAC protocol to use. [38] proposes another algorithm to decide which protocol features to use. Unlike this previous research, *protocol learning* and *protocol emergence* are concerned with learning a channel access policy and its signaling.

Protocol Learning and Emergence

In [39], the idea of using MARL to learn a predefined control signaling and a new channel access policy was first proposed. This idea was extended in [40], where state abstraction is used to improve the generalization and scalability of the produced protocols. However, in both papers the agents only learn to use an already known MAC signaling, rather than developing a new one.

Emerging a control signaling while learning a channel-access policy is studied in [41] and its scalability in [42]. In [43], a semantic protocol is produced by reducing the redundancy in the control messages of emerged protocols through merging.

Contributions

In light of this related research, the present thesis contributes to protocol emergence, by providing its fundamentals and the steps to study emerged protocols. It also studies the learning capabilities, illustrating the main scalability challenges and the adaptability advantages.

In terms of problem formulation, this thesis provides two new formulations, which model the contiguous and the non-contiguous resource allocation as MARL problems. In terms of signalling, both persistent signalling and intermittent signaling are studied.

1.3 Objectives and Thesis Structure

Objectives

The main objectives of this work are:

1. Provide the fundamentals for protocol emergence to enable future advancements in this area for next-generation communication systems.
2. Introduce MAC protocol emergence, highlighting the different challenges and advantages introduced by it.
3. Develop a framework for MAC protocol emergence that enables not only producing new protocols, but also the comparison of different protocols and the interpretation of the signaling messages.
4. Study the performance of MAC protocols in a challenging resource allocation problem, evaluating its learning capabilities under different system conditions.
5. Provide a formulation for controlling the amount of signaling exchanges, allow-

ing fine-grained control of the signaling throughput.

6. Indicate directions for future research in the field, both on the study of protocol emergence and other applications.

Thesis Structure and contributions

Chapter 2 presents an overview of the main wireless communication and RL concepts explored in this thesis. More specifically, it gives a short description of multiple-access schemes, highlighting the scheduling procedure employed in 5G NR and the signaling involved in it, while also providing an overview of some fundamental concepts of RL and MARL. Chapter 3 introduces a framework for MAC protocol emergence, which focus not only on producing protocols, but also on coordination evaluation, protocol characterization and signaling interpretation. The performance of the MAC protocol emergence method is studied deeply in chapter 4 with a frequency resource allocation problem examined to compare the learning capabilities of the proposed method. In chapter 5, MAC protocol emergence is studied under signaling constraints in a non-contiguous resource allocation problem, by leveraging an intermittent signaling formulation. At last, chapter 6 summarizes the main conclusions of this work and give directions for future research.

1.4 Tools and Methods

Regarding the tools used during this PhD research, we highlight the following software libraries:

- Pytorch [1] [44]: A library for deep learning. It was used in the RL algorithms and deep learning techniques.

[1] https://pytorch.org/

- Gym [2] [45]: An application programming interface (API) for RL environments, that is mainly used to model the partially observable Markov decision processs (POMDPs) in this thesis.
- Sacred [3] [46]: A software library to manage experiments. It provides tools to configuration, logging, organization and reproduce experiments. It was used in this thesis to help keep track of the thousands of experiments done throughout this research period, which were saved in a database.
- Incense [4]: A toolbox to query the sacred database and select experiments. It was used to access the experiments and select the results.
- Joblib [5]: A set of tools providing pipelining in Python. It was used to provide parallelization.

In terms of hardware, the computational workflow involved simulations on remote multi-core servers. Although GPUs were available, the neural networks used were shallow enough (4 layers maximum) that no benefit was observed from using them and the multiple CPUs were used instead.

1.5 Scientific Contributions

Although this thesis contains only currently unpublished and original work, this PhD research produced two related conference publications. It also led to a collaboration with the University of Oulu that produced a conference paper with an extending journal paper in writing, and a submitted survey on communication and control co-design.

[2] https://gymnasium.farama.org/
[3] https://github.com/IDSIA/sacred
[4] https://github.com/JarnoRFB/incense
[5] https://joblib.readthedocs.io

The bibliographic information of the published papers is:

- **M. P. Mota**, A. Valcarce, J.-M. Gorce, and J. Hoydis, "The emergence of wireless MAC protocols with multi-agent reinforcement learning," in *2021 IEEE Globecom Workshops (GC Wkshps)*, IEEE, 2021, pp. 1–6
- **M. P. Mota** *et al.*, "Scalable joint learning of wireless multiple-access policies and their signaling," in *2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)*, IEEE, 2022, pp. 1–5
- S. Mostafa, **M. P. Mota**, A. Valcarce, and Mehdi Bennis, "Emergent communication protocol learning for task offloading in industrial internet of things," in *IEEE Global Communications Conference (GLOBECOM)*, 2023, pp. 1–6

In the first paper, protocol emergence is introduced in a multiple-access task where the UEs have to deliver a fixed number of packets as fast as possible to the BS. The results illustrate the benefits of protocol emergence while highlighting the importance of communication and a multi-agent technique, as both non-communicating solution and a independent learners solution are unable to perform well in this task. The second paper, extends the previous one by modifying the task to a Bernoulli traffic model, where the task is to deliver as many packets as possible under a fixed horizon. The results analyze the scalability issues in terms of traffic and number of UEs.

The third conference paper applies protocol emergence to a joint task offloading decision and scheduling of computation tasks in an industrial internet-of-things (IIoT) scenario. The simulation results indicate the effectiveness of the learned protocols in maintaining highly efficient task offloading and maximizing the number of successfully computed tasks within the deadline constraint compared to traditional approaches.

It is worth mentioning that this thesis was developed under the context Marie Sklodowska-Curie actions (MSCA-ITN-ETN 813999 WINDMILL) in which 2 technical reports have been delivered.

Chapter 2

Theoretical Framework

Contents

This chapter provides some basic concepts that serve as a background to the subsequent chapters. Section 2.1 gives an overview of the relevant wireless communication concepts, namely multiple-access schemes and scheduling. In section 2.2, some fundamental concepts of reinforcement learning (RL) and multi-agent reinforcement learning (MARL) are briefly surveyed.

2.1 Wireless Multiple Access and Scheduling

Multiple access schemes allows a set of user equipments (UEs) to share a transmission medium, in the case of this thesis, a wireless communication channel. Such techniques enable the sharing of resources, such as frequency and time, by using multiplexing division schemes. In cellular networks, the medium access control (MAC) protocol is responsible for controlling the access to the physical medium. MAC protocols can be categorized by the topology used [47]:

- Centralized: A central node coordinates channel access, by allocating resources to the UEs sharing the medium. Due to the high degree of coordination of this topology, collisions can be totally avoided.
- Distributed: In this approach, the nodes take decisions by themselves and coordination is distributed. In the case of multi-hop networks, the nodes are not only in charge of transmitting their own traffic, but also of forwarding data from other nodes [48].

Another common approach to categorize multiple access protocols is with respect to the channel access method: contention-based, contention-free or hybrid [49].

In contention-based protocols, also known as random access methods, the transmitting nodes attempt to access the same channel to transmit their data, as such, they are competing for the ownership of the medium. The receiving nodes will detect the possible collisions or receptions and inform the transmitting node through either an acknowledgement (ACK) or a negative acknowledgement (NACK). A collision happens if two or more nodes simultaneously transmit data. In case of collision or channel error, retransmission is attempted accordingly to the protocol used. As there is no central

decision, contention-based protocols have either minimal or non-existent coordination [50].

In contention-free protocols, collision are avoided by ensuring that each node access their allocated resources exclusively. As such, coordination between nodes is necessary to ensure that each one knows the which resources to use. Usually, as for example in fifth generation (5G), the resource allocation is decided by a central entity and this information is reported to the nodes through control signaling.

The hybrid protocols mix the contention-based and contention-free approaches, combining their positive points and mitigating their negatives. For example, a hybrid approach can use contention-based access in low traffic scenarios and switch to contention-free mode when traffic increases. This would profit of the reduced delay of contention-based protocols in low traffic and of the reduced collision and higher rate of contention-free in high traffic.

It is also possible to classify MAC protocols with respect to the techniques used, such as by the multiplexing division used or by the channel access mechanism used.

Multiplexing Divisions

Multiplexing techniques enable the division of the medium for transmission of different signals to/from multiple nodes. Protocols that leverage multiplexing to provide multiple-access, separating sub-channels for each node, are usually categorized as **channelization protocols** and are mostly centralized [51]. The division of the medium depends on the multiplexing technique used. Examples of divisions are [52]:

- Sub-band for frequency division: Used in frequency division multiplexing (FDM). FDM divides the bandwidth into different sub-bands which are allocated to dif-

ferent signals. The different sub-bands are separated with guard bands to protect against interference.

- Time-slot for time division: Used in time division multiplexing (TDM), where the time is divided into slots, which are used to transmit a single signal. Time synchronization is needed, and guard times are used to avoid collisions.
- Subcarrier for orthogonal frequency division: Used in orthogonal frequency-division multiplexing (OFDM). In OFDM, the frequency band is divided into small parts called subcarriers. The subcarriers are orthogonal to each other and there are no guard bands, which provides better spectral efficiency than FDM. However, guard intervals are introduced in the time-domain to protect against inter-symbol interference. The frequency spacing between adjacent subcarriers is called subcarrier spacing (SCS).
- Code for code division: Used in code division multiplexing (CDM). In CDM, multiple signals are transmitted simultaneously over the same band by leveraging the spread spectrum technique [53] and a special coding scheme. Different signals are encoded using with either orthogonal codes or pseudo-random spreading sequences.
- Beam for spatial division: In space division multiplexing (SDM), different data streams are transmitted by different antenna-ports or different beams. In theory, if the transmitter has N_t antennas and the receiver N_r antennas, the number of parallel streams that can be transmitted is $\min(N_t, N_r)$.

Access Mechanisms

A common classification of MAC protocols is to divide them by the access mechanism used [52], [54]. Examples of access mechanisms include:

- Free access: The communication medium can be accessed anytime. ALOHA is an example of protocol using this technique.
- Slotted access: Time is divided into slots and transmission occur only at the beginning of slots. It reduces collision between an ongoing transmission and a new transmission attempt.
- Probabilistic access: Each node has a probability of accessing the medium.
- Back-off: A transmitting node waits an amount of time before accessing the medium. It is used together with other access methods, for example, a random backoff period is used in the event of a collision in ALOHA.
- Carrier sensing: The nodes sense the medium to identify if it is idle or busy and can only transmit if the medium is idle. Similarly to slotted access, it reduces the collision between ongoing transmissions and new attempts, but collisions due to simultaneous attempts can still occur.
- Polling: A central node sends a pool packet, which indicates the node that was selected to transmit, triggering its data transmission. The pool packet is broadcasted to all nodes, but only the selected one transmits. It is usually implemented together with reservation requests, or request-to-send, in order to avoid pooling nodes without data to transmit.
- Reservation: A reservation period is separated for the nodes to indicate if they

have data to transmit, reserving their transmissions. The nodes then transmit in the reserved times according to some priority. Reservation methods are usually implemented with slotted-time division.

- Messaging: A message carrying explicit information is sent from a node to others.
- Handshake: A more sophisticated messaging scheme in which the receiver must reply to the transmitter according to a predefined procedure.
- Cross-layer design: The design of the MAC takes into consideration information or functions from other layers, such as the physical layer (PHY).

2.1.1 Multiple Access Schemes

In this section, some relevant multiple-access schemes are detailed, with a selected few being illustrated in fig. 2.1.

2.1.1.1 ALOHA

In pure ALOHA, whenever a node has data it transmits immediately. As the transmitters share a single channel in this scheme, there is a potential for collisions. The transmitters rely on ACKs from the receiver, and if an ACK is not received after a time-out, a retransmission is issued after a back-off time. In pure ALOHA, the back-off time is random and a common method used is the binary exponential back-off.

The ALOHA protocol includes other variants, such as:

- Slotted ALOHA: Introduces discrete time-slots, leading to fewer collisions, as transmissions can only start at the beginning of a time-slot.
- Reservation ALOHA [55]: Further builds on top of slotted ALOHA while adding reservation period. The reservation period follows a slotted-ALOHA approach,

where stations compete to acquire reservations.

2.1.1.2 Protocols based on channelization schemes

Channelization schemes divide the channel into sub-channels according to the multiplexing division used. In their most simple form, which is fixed allocation, the protocol divides the resources between devices and each multiplexing division is allocated to one device. Taking the time division multiple access (TDMA) as an example, the frame would be divided into N time-slots and each UE would receive one slot to transmit while remaining silent during the non-allocated slots, with this process repeating every frame. However, dynamic allocation is also possible, with the central node dynamically allocating resources based on the system conditions, therefore, allowing the number of resources allocated to be different for each UE and to vary frame by frame [56].

Frequency-division multiple access (FDMA) Each frequency sub-band is assigned to a single UEs. Frequency-division multiple access (FDMA) has low implementation complexity, and it improves the distribution of resources in frequency selective channels. It was the main scheme used in first generation (1G) [57, p. 9], in which the device retained the sub-band for the whole duration of the call [58].

Time division multiple access (TDMA) Each time-slot is allocated to a single UE to receive or transmit data, and the devices share the frequency bandwidth. TDMA was used in second generation (2G), for example in global system for mobile communications (GSM) [59].

Code division multiple access (CDMA) Each UE is assigned a unique coding sequence used to encode the data, which has low cross-correlation with the other UEs. Each UE occupies the whole bandwidth, like TDMA, however, it can also transmit con-

tinuously like FDMA [60]. It has a higher receiver complexity compared with TDMA and FDMA. It was the main scheme used third generation (3G) systems [61].

Orthogonal frequency-division multiple access (OFDMA) Each UE is assigned to a subset of time-frequency resources, with the resource grid being 2D regions over time and frequency. Over time, the division is done in terms of OFDM symbols and the frequency is divided into subcarriers [62]. It was the scheme of choice for both fourth generation (4G) and 5G systems [3].

Space-division multiple access (SDMA) Precoding matrices are used to optimize the radiation patterns of the receiver and transmitter antennas [63], allowing UEs within the same cell to use the same radio channel depending on their spatial disposition [64]. Space-division multiple access (SDMA) was a prominent technology employed in 4G and 5G to exploit spatial diversity and increase the capacity [65].

Channelization schemes can be used together, and for some schemes, such as SDMA, this is the most common approach. For example, FDMA was used together with TDMA in 2G and with code division multiple access (CDMA) in 3G even if it was not the main scheme [66, p. 54].

2.1.2 5G-NR Scheduling

In the 5G-NR, multiple-access is based mainly on orthogonal frequency-division multiple access (OFDMA), with multi-user (MU)-multiple-input multiple-output (MIMO) being enabled through SDMA. For uplink transmission, single carrier frequency-division multiple access (SC-FDMA) can also be used [67]. The scheduling procedure involves the assignment of the resources by a central scheduler. The information needed by the scheduler is sent through the uplink control channel and the scheduling decision is contained in the downlink control information (DCI) and transmitted through the

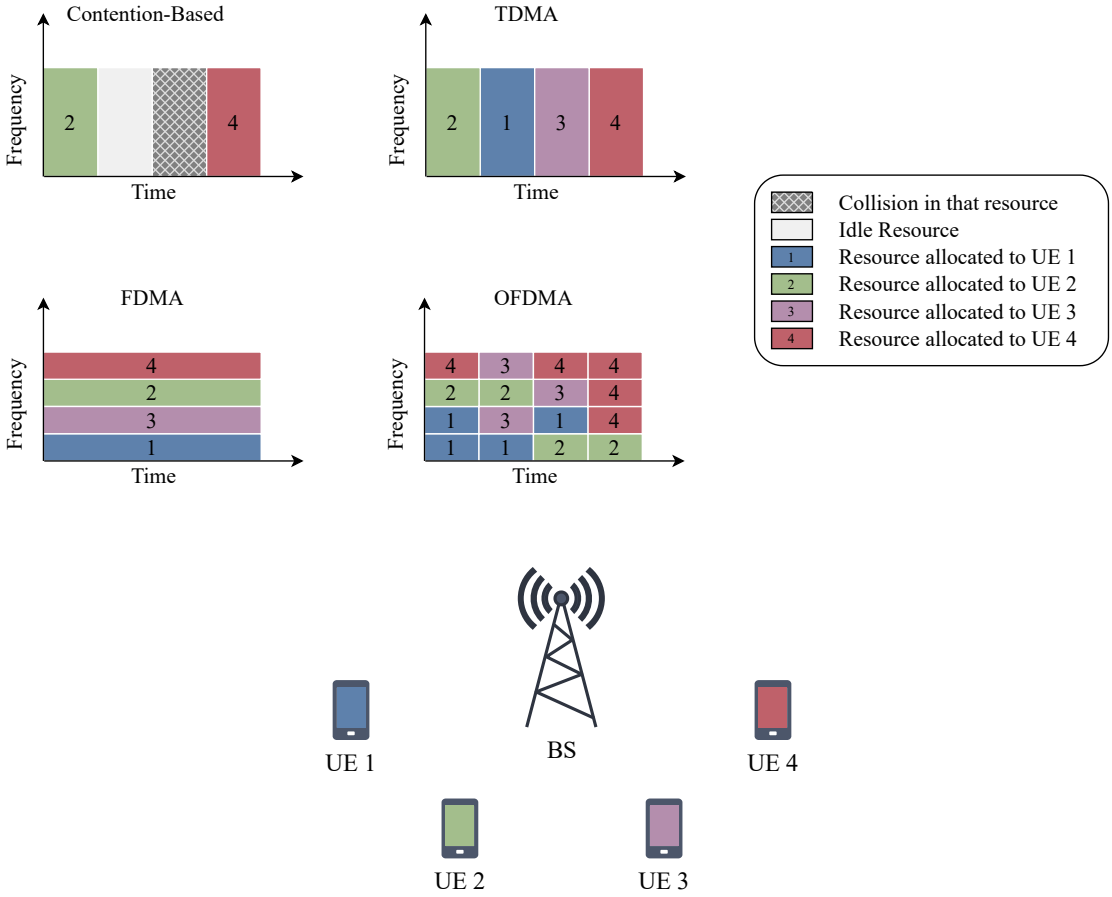


Figure 2.1: Illustration of multiple-access schemes.

downlink control channel.

In *TS 38.300* [68], the basic scheduler operation is described. The MAC in the base station (BS) includes dynamic resource scheduling in order to allocate the PHY resources. The scheduler operation takes into account the following information:

- UE buffer status.
- UE quality of service (QoS) requirements.
- Radio conditions.
- Power headroom reports.

Although the multiple-access problems studied in this thesis can be seen as similar to random-access, there are some differences involved. In the context of 5G-NR, the random-access procedure is triggered in the following cases [68]:

1. Initial access.
2. Connection Re-establishment procedure.
3. Handover.
4. downlink (DL) or uplink (UL) data arrival when UL is non-syncronised.
5. Transition from inactive.
6. To establish time alignment.
7. Request for other system information.
8. Beam failure recovery.

As such, random-access is used to provide system connection, reconnection, recovery or some information. The present work addresses uplink data transmission, as such, it is mainly interested in the uplink shared channel (UL-SCH).

2.1.2.1 Resource Grid

The scheduling procedure allocates resources both in time domain and frequency domain. As such, an overview of the new radio (NR) resource grid is needed. In terms of time-domain, the NR has a frame structure divided into frames, subframes, slots and OFDM symbols, with variable number of symbols in a slot depending on the numerology:

- Frame: A radio frame in NR has a fixed duration of 10 ms and consists of 10 subframes.

- Subframe: A subframe has a fixed duration of 1 ms and consists of a variable number of slots, depending on the numerology used.
- Slot: A slot with normal cyclic prefix contains 14 OFDM symbols and a slot with extended prefix contains 12 symbols.

The NR supports a total of five numerologies $\mu \in \{0, 1, 2, 3, 4\}$ and the number of slots in a subframe, $N^{\text{sh}}_{\text{symb}}$, depends on the numerology according to:

$$N^{\text{sh}}_{\text{symb}} = 2^{\mu}\,. \tag{2.1}$$

The slot format indicates how the symbols within the slot are used, for example, slot formats 0 and 1 use all symbols for downlink and uplink, respectively. There are a total of 56 slot formats predefined in *TS 38.213* [69], with extra 199 reserved formats and one format indicating that the UE determines the slot format based on other configurations.

In the frequency domain, the smallest unit is the OFDM subcarrier, however, the definition of resource elements (REs), resource blocks (RBs), resource block groups (RBGs) is important to understand the resource allocation procedure in NR.

- Resource element (RE): It is the smallest physical resource, and it is made up of one subcarrier in the frequency domain and one OFDM symbol in time domain.
- Resource block (RB): Defined as 12 consecutive subcarriers in frequency domain. Resource block group (RBG): A group of consecutive virtual resource blocks (VRBs). The number of RBs in a RBG depends on the bandwidth part size and the configuration used and may be not constant for all RBGs.

Differently from long term evolution (LTE), where the RB is defined as 12 subcarriers in frequency domain and one slot in time domain, the NR defines a RB only for the

Table 2.1: Effect of numerology

μ	SCS	Bandwidth per RB	N^{sh}_{symb}	$N^{subframe}_{slot}$	N^{frame}_{slot}	Slot duration
0	15kHz	180kHz	14	1	10	1ms
1	30kHz	360kHz	14	2	20	500µs
2	60kHz	720kHz	14	4	40	250µs
3	120kHz	1440kHz	14	8	80	125µs
4	240kHz	2880kHz	14	16	160	625µs

frequency domain. The allocation is done in terms of VRBs and not in terms of physical resource blocks (PRBs), and the mapping of VRBs-to-PRBs is either interleaved or non-interleaved. This means that a continuous set of VRBs maps to a continuous set of PRBs in the non-interleaved case, but it can also map to a non-continuous set of PRBs in the interleaved case.

Another important concept to define is the bandwidth part (BWP), which is a specific subset of the overall carrier bandwidth allocated to a UEs operation. This provides power savings, as the UEs have to monitor a narrower band, and also flexibility in the spectrum use.

Table 2.1 shows how the numerology, μ affect different parameter of the system, both in the frequency domain due to the change in the subcarrier spacing (SCS), and in the time domain due to the change in symbol duration.

2.1.2.2 Resource Allocation

The NR supports three types of resource allocation in the frequency domain, which are detailed in *TS 38.214* [70]:

- Type 0: Non-contiguous allocation using a bitmap.
- Type 1: Contiguous allocation defining the start RB and number of RBs.

Table 2.2: RBG size P

BWP size	Configuration 1	Configuration 2
$1 - 36$	2	4
$37 - 72$	4	8
$73 - 144$	8	16
$145 - 275$	16	16

- Dynamic switch: The allocation is determined at transmission time through the DCI.

In the DCI, the frequency domain resource assignment field specifies the resources allocated to an UE, depending on the allocation type used. Under dynamic switch, this field includes an extra bit used to indicate the allocation type, 0 or 1, for that scheduled transmission, but the allocation follows either type 0 or 1.

Type 0 allocation

When under type 0 allocation, the frequency domain resource assignment field in the DCI follows a bitmap format indicating which RBGs are allocated to that UE. The RBG size depends on the configuration used and on the BWP size. The RBG configuration is a higher layer parameter with two possible values, and it is informed through a radio resource control (RRC) message. The BWP size represents the number of PRBs there are in that BWP. Table 2.2 shows how the RBG size P is calculated.

The total number of RBGs N_{RBG} in a given BWP of size $N_{\text{BWP}}^{\text{size}}$ PRBs starting at PRB $N_{\text{BWP}}^{\text{start}}$ can be calculated as:

$$N_{\text{RBG}} = \left\lceil \frac{N_{\text{BWP}}^{\text{size}} + \left(N_{\text{BWP}}^{\text{start}} \bmod P\right)}{P} \right\rceil. \tag{2.2}$$

The size of the first RBG is:

$$\mathrm{RBG}_0^{\mathrm{size}} = P - N_{\mathrm{BWP}}^{\mathrm{start}} \bmod P \,. \tag{2.3}$$

The last RBG size is calculated as:

$$\mathrm{RBG}_{\mathrm{last}}^{\mathrm{size}} = \begin{cases} \left(N_{\mathrm{BWP}}^{\mathrm{size}} + N_{\mathrm{BWP}}^{\mathrm{start}}\right) \bmod P \ \text{, if } \left(N_{\mathrm{BWP}}^{\mathrm{start}} + N_{\mathrm{BWP}}^{\mathrm{size}}\right) \bmod P > 0 \\ P \ \text{, otherwise.} \end{cases} \tag{2.4}$$

And the size of all other RBGs is P.

With respect to the signaling, the bitmap has size N_{RBG} with one bit per RBG, and the RBGs are indexed in order of frequency. If the bit corresponding to a RBG has a value of 1, this RBG is allocated to the UE.

Type 1 allocation

In the case of allocation following type 1, the frequency domain resource assignment field in the DCI indicates a set of contiguously allocated VRBs. The mapping of VRB to PRB is always non-interleaved under this allocation type. The assignment field contains a resource indicator value (RIV) identifying the starting RB $\mathrm{RB}_{\mathrm{start}}$ and the length in terms of contiguously allocated RBs L_{RBs}. The RIV is calculated as follows:

$$\mathrm{RIV} = \begin{cases} N_{\mathrm{BWP}}^{\mathrm{size}} \left(L_{\mathrm{RBs}} - 1\right) + \mathrm{RB}_{\mathrm{start}} \text{ , if } \left(L_{\mathrm{RBs}} - 1\right) \lfloor N_{\mathrm{BWP}}^{\mathrm{size}}/2 \rfloor \\ N_{\mathrm{BWP}}^{\mathrm{size}} \left(N_{\mathrm{BWP}}^{\mathrm{size}} - L_{\mathrm{RBs}} + 1\right) \left(N_{\mathrm{BWP}}^{\mathrm{size}} - \mathrm{RB}_{\mathrm{start}} - 1\right) \text{ , otherwise} \end{cases} \tag{2.5}$$

with $1 \leq L_{\text{RBs}} \leq \left(N_{\text{BWP}}^{\text{size}} - \text{RB}_{\text{start}}\right)$. The calculation above is a way of compressing the starting RB and the length into a single value.

Release 16 introduced type 1 allocation with RBGs instead of RBs, reducing the number of bits used in the resource assignment field. This allocation happens when the DCI follows format *0_2*, and it is mostly used for ultra-reliable and low-latency communications (uRLLC) use-cases. The calculation for the RIV is similar, just changing from RB to RBG.

2.1.2.3 Scheduling Objectives

A scheduler is the logical entity responsible for the resource allocation. In order to design and compare schedulers and protocols, it is important to highlight some of the performance metrics that are taken into consideration [71]:

- Delay: The average time needed for a packet to be successfully transmitted since it has been received by the MAC layer.
- Goodput: The amount of useful data successfully transmitted by the nodes.
- Fairness: Evaluates if the radio resources are being similarly or fairly shared between the UEs.
- Energy consumption: As most wireless devices have limited battery, power consumption has to be taken into consideration when evaluating MAC protocols.

Other possible performance metrics include reliability, stability and QoS [72].

Although it would be optimal for a protocol to minimize delay, maximize goodput, provide fairness guarantees and maximize energy efficiency, there can be trade-offs involved [73]. Therefore, a scheduler design has to take into consideration the use-case in order to optimize for the most suited performance metric. For instance, delay is a

key performance metric for uRLLC use-cases, while not being as important for massive machine-type communications (mMTC) use-cases, which must prioritize energy efficiency [74].

2.1.2.4 Scheduler Examples

In this section, we illustrate some schedulers used in dynamic allocation with channel state information. Such schedulers, also known as opportunistic schedulers, can leverage the channel information to provide channel-dependent scheduling [75]. A more comprehensive list of scheduling algorithms can be found in [76].

Maximum signal-to-noise ratio (SNR)

In this scheduling solution, the objective is to greedily maximize the rate. It allocates the resources to the UE with the best radio conditions. However, this scheduler is not fair, as in a system with two UEs, one close to the BS and another at the edge, the closest UE will most likely receive more resources than the one at the edge.

Proportional fairness scheduler

The proportional-fair (PF) scheduling algorithm aims to improve fairness while maintaining a good cellwide rate [77]. For each scheduling slot, the algorithm uses the achievable data rate and the average throughput to select the UE or UEs. The achievable data rate of a UE u at a given time-slot t, $R_u(t)$, is the rate that the channel of UE u currently supports. In a scheduling slot t and with average throughput of UE u represented as $G_u(t)$, the algorithm selects UE $u^\star$ such that:

$$u^\star = \underset{u}{\operatorname{argmax}} \frac{R_u(t)}{G_u(t)} \tag{2.6}$$

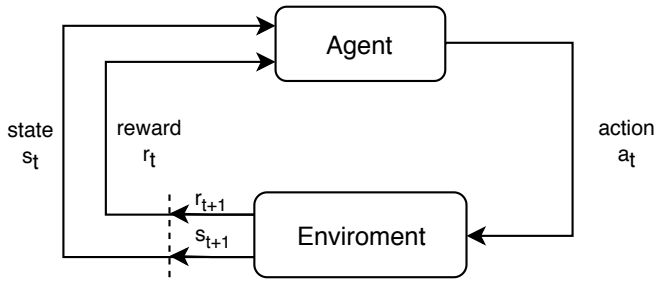


Figure 2.2: Basic diagram of a RL scheme

The average throughput is updated according to:

$$G_u(t+1) = \begin{cases} \left(1 - \frac{1}{t_c}\right) G_u(t) + \frac{1}{t_c} R_u(t), u = u^\star \\ \left(1 - \frac{1}{t_c}\right) G_u(t), u \neq u^\star, \end{cases} \tag{2.7}$$

where the parameter t_c defines the averaging time window and $u^\star$ is the selected UE.

The above algorithm is defined under single-user scheduling, but a multi-user PF scheduler is proposed in [78], by selecting a set of UEs, $I^\star$, instead of a single UE, $u^\star$, in (2.6) and (2.7). The PF scheduler is known to maximize the sum of the logarithmic average rates [79].

2.2 Reinforcement Learning

RL refers to the problem encountered by an agent that must learn behavior by engaging in trial-and-error interactions with a dynamic environment [80]. However, RL can also refer to the machine learning (ML) technique used in such problems [81]. As such, RL refers to both the problem, the class of methods to solve this problem, while also referring to the field that studies the problem and its solutions [82].

2.2.1 RL: The Problem

The main elements in a basic model of a RL problem are: the environment's *state*, the agent's *action* and the *reward* generated from this action. Figure 2.2 shows a simple block diagram of the RL problem depicting the interaction between an agent, which is the learner and decision maker, and its environment through the execution of actions. The environment is affected by the agent's actions and transitions to a new state. The reward received by the agent is a scalar signal associated with the transition. The behavior of the agent should aim to take actions that maximize the accumulated reward. In light of this, we highlight some basic components of a RL problem and solution:

State space $\mathcal{S}$

Set of all possible states of the environment. The environment's state at time step t is denoted s_t, with $s_t \in \mathcal{S}$.

Action Space $\mathcal{A}$

Set of all actions that can be taken by the agent. The action taken at time step t is denoted a_t, with $a_t \in \mathcal{A}$.

Transition Function $\mathbb{P}(s_{t+1}|s_t, a_t) : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$

The transition function $\mathbb{P}(s_{t+1}|s_t, a_t)$ describes the dynamics of the system. It expresses the probability of transitioning to state s_{t+1} after taking action a_t in state s_t.

Reward Function $R(s_t, a_t) : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$

The stochastic reward function R describes the immediate payoff from taking an action a_t in a state s_t. The reward received by the agent at time step t, which is sampled from $R(s_t, a_t)$, is denoted r_t.

At each time step t, the agent observes the state of the environment $s_t \in \mathcal{S}$, and based on that chooses an action $a_t \in \mathcal{A}$. As consequence of its action, the environment transitions to a new state, s_{t+1}, and the agent receives a reward r_{t+1}. A RL problem is commonly framed as a Markov decision process (MDP), because the probability of the next state depends only on the last state and action, ensuring the Markov property:

$$\mathbb{P}(s_{t+1}|s_t, a_t, \ldots, s_0, a_0) = \mathbb{P}(s_{t+1}|s_t, a_t) \tag{2.8}$$

Full Observability and Partial Observability

In fully observable MDPs the agent observes the state of the environment or a function of it that maintains the Markov property. However, in partially observable Markov decision processs (POMDPs) [83] the state of the environment is not directly observed by the agent. Instead, the agent receives an observation $o \in \mathcal{O}$ according to an observation function $O(s, a) : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{O}$ [84], [85].

A POMDP is a MDP in which the true state is clouded and the agent receives an ambiguous observation that does not ensure the Markov property. As such, the agent needs to keep track of the action-observation history h in order to make decisions, the action-observation history up to time-step t is denoted $h_{0:t} = \{o_0, a_0, \ldots, o_{t-1}, a_{t-1}, o_t\}$. An agent may not need the full action-observation history, $h_{0:t}$, but just few k samples, $h_{(t-k):t}$, for instance, the proposed architecture in the deep Q-network (DQN) paper [86] uses the past 4 frames.

Although the action-observation history can be used to solve a POMDP, it can also be used with the environment dynamics to generate a belief state [87]. However, due to needing the environment dynamics [88], belief states can only be used in model-based

RL. The action-observation history is commonly used in POMDPs, but other versions of the history are commonly used in other RL problems, such as meta-RL [89], [90] that also includes the reward and termination. The interested reader can look at [88] for more information on leveraging the history for other RL problems.

2.2.2 RL: Agent Modeling

The goal of the agent is to optimize its policy, π, in order to maximize the expected discounted return $\mathcal{R}$. The expected return given the policy of the agent induces the definition of value-functions, for which the action-value function, Q, is of particular interest for this thesis. As such, the policy and the discounted return are defined as:

Policy $\pi(a_t|s_t) : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$

The policy $\pi(a_t|s_t)$ is the function mapping states to the probability distributions of actions. In case of deterministic policies, the policy is a function of states to actions, $\pi(s_t) : \mathcal{S} \rightarrow \mathcal{A}$.

Discounted Return $\mathcal{R}_t$

The discounted return is given by $\mathcal{R}_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} = r_{t+1} + \gamma \mathcal{R}_{t+1}$ [80].

Q-function $Q^{\pi}(s_t, a_t) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t, a_t, \pi\right]$

Called action-value function, is the overall expected reward for taking an action a_t in a state s_t and then following a policy π. It can also be simply denoted as $Q(s_t, a_t)$.

The parameter γ is called *discount factor*, or discount rate, with $0 \leq \gamma < 1$. The discount factor is used to control the importance given to future rewards in comparison with immediate rewards. The infinity sum $\sum_{t=0}^{\infty} \gamma^t r_{t+1}$ has a finite value if $\gamma \leq 1$, as long as the sequence $\{r_k\}$ is bounded [91]. The process is called undiscounted if $\gamma = 1$.

2.2.3 Multi-Agent Reinforcement Learning

In this thesis, we will be considering multi-agent scenarios. More specifically, we consider fully cooperative scenarios with communication. This can be formalized with a decentralized partially observable Markov decision processs (Dec-POMDPs) [92] augmented with communication.

A Dec-POMDP for n agents is defined by the global state space $\mathcal{S}$, an action space $\mathcal{A}_1, \ldots, \mathcal{A}_n$, and an observation space $\mathcal{O}_1, \ldots, \mathcal{O}_n$ for each agent. In a Dec-POMDP, the agent observation does not fully describe the environment state. In a cooperative setting, all agents share the same reward. Furthermore, the action space of each agent is subdivided into one environment action space and a communication action space. The communication action represents the message sent by an agent, and it does not affect the environment directly, but it may be passed to other agents. This setting is illustrated in fig. 2.3.

MARL introduces some new challenges, such as partial observability and non-stationarity [93]. Communication addresses the partial observability, by providing agents with the ability to communicate their information to other agents. Across this thesis, the agent's internal state, x, comprises not only the current observation, but also previous observations, actions and received messages, similarly to the action-observation history. The non-stationarity comes from the perception of one agent that the transition model of the environment changes whenever another agent changes its policy, because the environment transition depends on the actions of all agents.

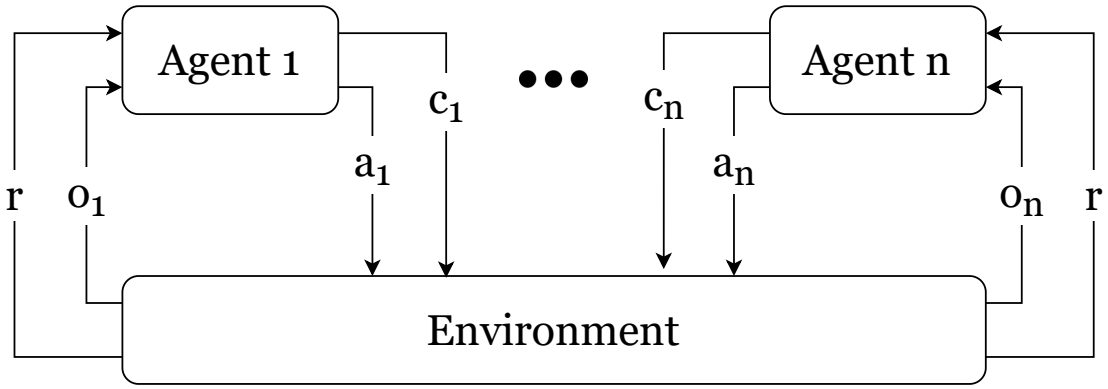


Figure 2.3: Illustration of cooperative MARL with communication.

2.2.4 Multi-agent deep deterministic policy gradient

In this work, we adopt the multi-agent deep deterministic policy gradient (MADDPG) algorithm [94], an extension of the deep deterministic policy gradient (DDPG) algorithm [95] to multi-agent problems with centralized training and decentralized execution (CTDE). It addresses the non-stationarity problem by using a centralized critic.

Each agent has an actor network that depends only on its own agent's state in order to learn a decentralized policy ω_i with parameters θ_i. During the training, each agent has a centralized critic that receives the agent states and actions of all agents in order to learn a joint action value function $Q_i(\boldsymbol{x}, \boldsymbol{a})$ with parameters φ_i, where $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ is a vector containing all the agents' states and $\boldsymbol{a} = (a_1, a_2, \ldots, a_n)$ contains the actions taken by all the agents.

The critic network parameters φ are updated by minimizing the loss given by the temporal-difference error

$$L^i := \mathbb{E}_{\boldsymbol{x},\boldsymbol{a},r,\boldsymbol{x}'\sim\mathcal{D}}\left[y^i - Q_i(\boldsymbol{x}, \boldsymbol{a}; \varphi_i)\right] \tag{2.9}$$

where $\mathcal{D}$ denotes the experience replay buffer in which the transition tuples $(\boldsymbol{x}, \boldsymbol{a}, r, \boldsymbol{x}')$ are stored. The temporal-difference target y^i is given by

$$y^i \coloneqq r + \gamma Q_i'(x', a_1', \ldots, a_n'; \varphi_i')\big|_{a_k'=\mu_k'(x_k)} \tag{2.10}$$

where Q' and ω' represent the target critic network and the value of the target actor network, with parameters φ' and θ', respectively, and γ is the discount factor. The actor network parameters θ are updated using the sampled policy gradient

$$\nabla_{\theta_i} J = \mathbb{E}_{x,a\sim\mathcal{D}} \left[\nabla_{a_i} Q_i(\boldsymbol{x}, \boldsymbol{a}) \nabla_{\theta_i} \omega_i(x_i) \mid a_i = \omega_i(x_i) \right]. \tag{2.11}$$

The target networks parameters are updated as

$$\varphi_i' \leftarrow \iota\varphi_i + (1-\iota)\varphi_i' \tag{2.12}$$

$$\theta_i' \leftarrow \iota\theta_i + (1-\iota)\theta_i' \tag{2.13}$$

where $\iota \in [0, 1]$ is the soft-update parameter. Smaller values of ι lead to slow target network changes and are generally preferred [95].

2.2.4.1 Other Techniques

The basic MADDPG algorithm is incremented with two techniques in this thesis, namely the Gumbel-Softmax [96], [97] and deep dense architecture for reinforcement learning (D2RL) [98].

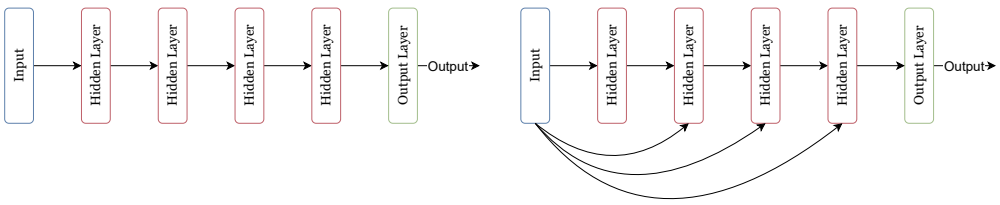
Gumbel-Softmax. The Gumbel-Softmax is a continuous distribution that allows sampling from a categorical distribution in the forward pass of a neural network. This allows the DDPG to be used with discrete action spaces, as in its original form it

can only be used with continuous actions due to the gradient of $Q(s, a)$ w.r.t the actions. The MADDPG is originally implemented for discrete action spaces and uses the Gumbel-Softmax distribution in [94].

D2RL. The D2RL technique modifies the multilayer perceptron (MLP) architectures commonly used in deep RL with dense connections (input concatenations). This improves the feature extraction and allows increasing the number of hidden layers without losing performance as it usually happens in simple feed-forward architectures used in RL. It is done by concatenating the input of the network for every hidden layer of the network except the output layer. As such, the input of a hidden layer is the output of the previous layer concatenated with the input of the network, in this case the state for the actor or state-action for the critic. The difference in architecture is illustrated in fig. 2.4.

JSRL. Another improvement used is the jump-start reinforcement learning (JSRL) [99], which leverages a prior policy to improve the learning performance. JSRL is a meta-algorithm that uses a pre-existing policy to bootstrap an RL algorithm. It uses two policies:

1. Exploration-policy π^e: An RL policy that is trained from the experience gathered



(a) A standard feed-forward architecture.

(b) The D2RL architecture making use of dense connections.

Figure 2.4: Difference between the standard feed-forward and the D2RL architectures.

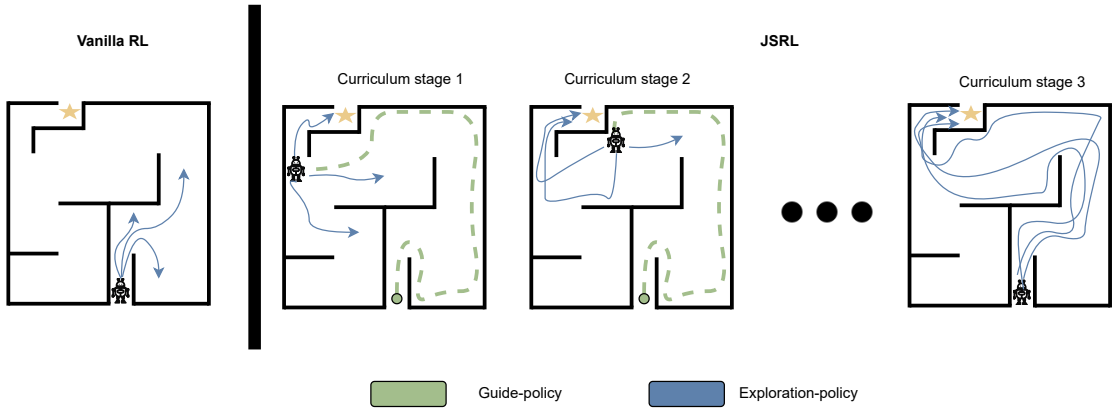


Figure 2.5: Difference between the vanilla reinforcement learning and JSRL. The goal of the agent is to reach the star. In JSRL, the guide-policy takes the agent closer to the goal and the exploration-policy in training only has to learn a simpler version of the original task.

from the environment.

2. Guide-policy π^g : The pre-existing policy that is not updated during training. The guide-policy only requirement are that is better than random exploration and that it can select actions based on environment observations.

Initially, the agent follows the guide-policy to reach states closer to the goal. Then, it switches to the exploration-policy, which continues to act to reach the goal. As the exploration-policy gains proficiency, the guide-policy becomes less active, until the exploration-policy takes full control. This process effectively creates a learning curriculum, allowing the exploration-policy to focus on mastering gradually more challenging tasks.

In a task with horizon H, we roll π^g for a number of h steps and then let π^e take over the extra $H-h$ steps, until the goal or the horizon H is reached. The data collected from both policies is used to train π^e and then, the combined policy is evaluated. If the performance evaluation achieves a threshold β, the number of guide-policy steps is reduced, this decreasing h. In summary, as π^e improves, it should take over the task

earlier until only π^e is used.

Chapter 3

Framework for MAC Protocol Emergence

Contents

In this chapter, we focus on the study of protocol emergence within a multiple-access problem. The main goals of this chapter are to lay the foundations for protocol emergence, investigate its benefits, discuss how to effectively study protocols and suggest future research directions. As such, the main contributions of this chapter are:

1. Comparison of different methods to learn a protocol.
2. **Vocabulary size study**: We evaluate the capabilities of protocol emergence to reduce the amount of signaling used.
3. **Protocol characterization**: We propose new key performance indicators (KPIs) and metrics to compare the emerged protocols and to interpret them using information theory.
4. **Protocol interpretation**: We study the usage of the control messages by the emerged protocol.

3.1 System Model

Consider a single cell with a base station (BS) serving U user equipments (UEs) in a uplink slotted shared channel (USC), where each UE needs to deliver data to the BS. For the frequency domain, the available bandwidth is contained in a single resource. Each UE has a transmission buffer of capacity B service data units (SDUs) initially empty. The SDU arrival is modeled as a Poisson process with probability of arrival p_{a}. So, at each time step, a new SDU of size L_{TB} is added to the buffer with probability p_{a}, until a maximum number T steps is achieved. The average number of SDUs arriving at each

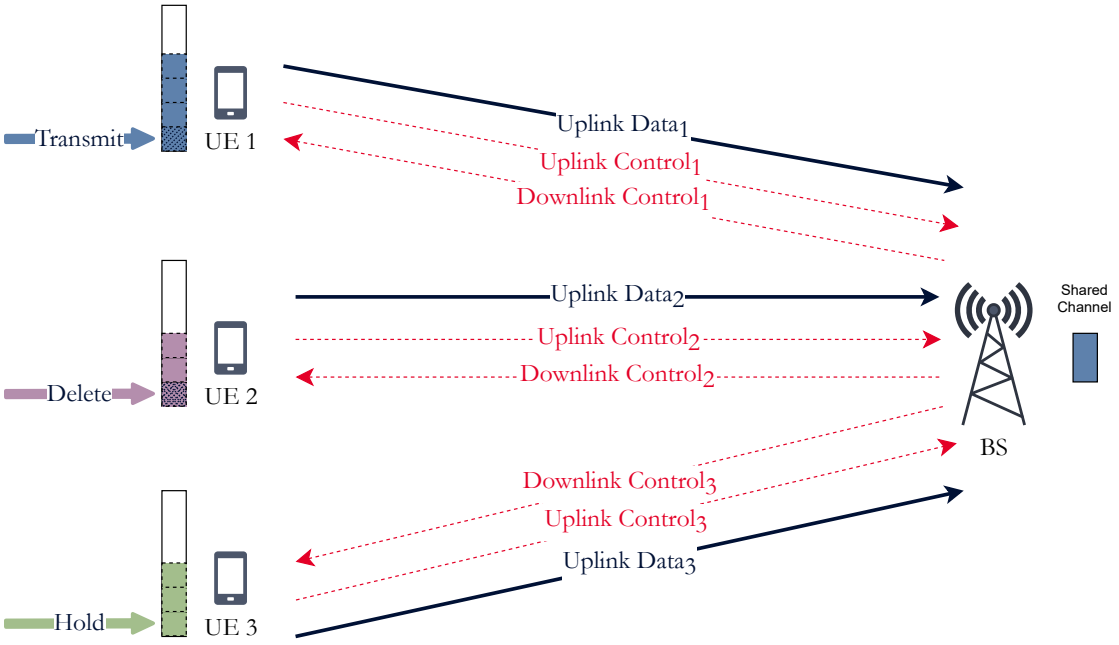


Figure 3.1: System model scheme for the multiple-access problem. The buffer and decisions of each UE are highlighted besides it. The wireless channel is shown besides the BS, indicating the state of the wireless channel, in this case, it indicates that it received data from UE 1.

UE's buffer in any given episode of duration T is then:

$$\lambda = p_a T \,. \tag{3.1}$$

The BS and UEs constitute the nodes of the system who act as independent agents. The network nodes can exchange information using messages through the control channels. In the remainder of this chapter, we refer to the UE medium access control (MAC) agents and the BS MAC agent as UE and BS , respectively.

The channel for the uplink data transmission is modeled as a packet erasure channel, where a transport block (TB) is incorrectly received with a probability referred to as the transport block error rate (TBLER). The UEs use the same frequency resources

on the uplink shared channel (UL-SCH), where collisions may occur. The downlink control messages (DCMs) and uplink control messages (UCMs) are transmitted over the downlink (DL) and uplink (UL) control channels respectively, which are assumed to be dedicated and error free, so without any contention or collision.

We assume that the sets of possible DL and UL control messages have cardinality V_{DCM} and V_{UCM}, respectively. For a DL (resp. UL) control vocabulary of size V_{DCM}, the bitlength Υ_{DL} is equal to $\lceil(\log_2 V_{\text{DCM}})\rceil$, where $\lceil\cdot\rceil$ represents the ceiling function.

At each time step t of duration T_{TTI}, the BS can send one control message to each UE and each UE can send one control message to the BS. Furthermore, the UEs can also a send protocol data units (PDUs) through the UL-SCH or delete a SDU from its buffer at each time step. Figure 3.1 illustrates the system model, highlighting the decisions taken by each UE and its effect on the system.

3.1.1 Network KPIs

In order to evaluate and compare different protocols, let us introduce the following KPIs. The cellwide goodput is the main KPI for the task above, but it is not sufficient. We propose also to evaluate additional metrics linked to the reliability, the latency or the energy efficiency.

The cellwide goodput G (in $\text{Mbit}\,\text{s}^{-1}$) is the number of information bits received by the BS per unit of time.

$$G = \frac{N_{\text{RX}}\, L_{\text{TB}}}{T\, T_{\text{TTI}}} \tag{3.2}$$

where N_{RX} represents the number of SDUs received by the BS serviced by the MAC layer and the SDUs received by the BS several times are only counted once. The collision-rate Γ is the number of steps in which a collision happened divided by the

total number of time steps:

$$\Gamma = \frac{N_c}{T}. \tag{3.3}$$

where N_c represents the total number of time steps in which at least two SDUs collided. Energy efficiency (EE) is defined as the number of received SDUs divided by the total number of transmissions attempted by the UEs N_{TX}:

$$EE = \frac{N_{RX}}{N_{TX}}. \tag{3.4}$$

It is important to highlight that N_{TX} includes retransmissions, even in the case of successful receptions. As such, $N_{TX} \neq N_{RX} + N_{lost}$.

The amount of signaling overhead in bits per transmission time interval (TTI) Υ is defined as the sum of the number of symbols used in the DL and UL control vocabularies:

$$\Upsilon = \Upsilon_{DL} + \Upsilon_{UL}. \tag{3.5}$$

Since we are evaluating a MAC protocol, a useful definition of delay would be the average time it takes for an SDU to be received once it is generated. The delay, Δ, is defined as the time difference between the TTI in which an SDU s is received, ξ_s and the TTI that this SDU was generated, τ_s.

$$\Delta_s = \xi_s - \tau_s. \tag{3.6}$$

As such, the average delay Λ (in ms) is defined as:

$$\Lambda = \frac{\sum \Delta_s}{N_{RX}}. \tag{3.7}$$

Lastly, it is important to evaluate the reliability of a protocol. The reliability Ψ can be defined as the number of unique SDUs received, N_{RX}, divided by the sum of the number of SDUs lost, N_{lost}, and N_{RX}:

$$\Psi = \frac{N_{\text{RX}}}{N_{\text{RX}} + N_{\text{lost}}}. \tag{3.8}$$

3.1.2 Coordination Metrics

Network KPIs provide a way to evaluate and compare channel-access policies. However, these KPIs are not the best way to evaluate cross-node coordination through signaling because a protocol could, for example, perform better based on a superior channel-access policy only. To illustrate this, imagine two contention-based protocols relying on acknowledgements (ACKs), but with two different transmission probabilities. As such, their goodputs will be different, however, their use of the information and the meaning of it is the same.

Hence, metrics that evaluate the use of the communication and the degree of coordination of the system are needed. To evaluate the level of coordination between agents, we make use of emergent communication metrics as proposed in [23], [100]. The metrics used in this chapter are derived from the mutual information. Given two random variables X, Y defined on discrete sets $\mathcal{X}$ and $\mathcal{Y}$, the mutual information is defined as

$$I(X, Y) = \sum_{x \in X} \sum_{y \in Y} \mathbb{P}_{XY}(x, y) \log_2 \left(\frac{\mathbb{P}_{XY}(x, y)}{\mathbb{P}_X(x)\ \mathbb{P}_Y(y)} \right) \tag{3.9}$$

The two metrics used in this chapter are:

1. **Instantaneous coordination (IC)**: Introduced in [100]. Quantifies the relation-

ship between an agent's environment action and the message it received from other agents. This is defined as the mutual information between the received message and the environment action.

2. **Immediate reply (IR)**: We introduce this metric to quantify the dependence between a signaling message received by a node, and subsequent signaling message emitted by the node. This is defined as the mutual information between the received message and the communication action.

Formally:

$$\text{IC} = I(M, A) \tag{3.10}$$

and

$$\text{IR} = I(M, C), \tag{3.11}$$

where, M, A and C are discrete random variables representing the received message, environment action and communication action.

The IC depends on environment actions and is used for the UEs, whereas the IR metric is used for the BS, since the BS works as an orchestrator for the UL transmission task. These metrics can be generalized to model relationships across varying time offsets (i.e., between the received message and the selected action).

Both metrics measure the coordination degree of a protocol. In terms of the system model studied in this chapter, the IC measure the influence of the BS in the decisions of the UEs while the IR measuring the degree in which the BS communication instruction is affected by the received UL message. In applications were a tighter coordination is desired, protocols with higher coordination metrics should be preferred. In [100], a method to motivate higher IC is proposed and it would be interesting to evaluate how

this approach can be useful when both IC and IR are considering in a multiple-access problem.

3.2 Emerging a MAC Protocol with MARL

3.2.1 MARL Formulation

We formulate the problem defined above as a multi-agent reinforcement learning (MARL) cooperative task, where the MAC layers of the network nodes (UEs and BS) are reinforcement learning (RL) agents that need to learn how to communicate with each other to solve an uplink transmission task. In addition, the UE agents need to learn when to send data through the UL-SCH and when to delete an SDU, in other words, to learn how to correctly manage the buffer. To decide how to act, an agent needs to consider the messages received from the other agents. Furthermore, the UEs also take into account their buffer status when taking actions, while the BS takes into account the state of the UL-SCH, i.e. idle, busy or collision-free reception.

We model this problem as a decentralized partially observable Markov decision process (Dec-POMDP) [92], augmented with communication. A Dec-POMDP for n agents is defined by the global state space $\mathcal{S}$, an action space $\mathcal{A}_1, \ldots, \mathcal{A}_n$, and an observation space $\mathcal{O}_1, \ldots, \mathcal{O}_n$ for each agent. In Dec-POMDP, the agent observation does not fully describe the environment state.

All agents share the same reward and the action space of each agent is subdivided into one environment action space and a communication action space. The communication action represents the message sent by an agent and it does not affect the environment directly, but it may be passed to other agents. In this chapter, the agent internal state x_i may comprise not only the agent's current observation, but also pre-

vious observations, actions and received messages. We assume the episode ends when a maximum number of steps T is reached.

We use the following notations:

- o_t^{u}: Observation received by the u^{th} UE at time step t.
- o_t^{b}: Observation received by the BS at time step t.
- n_t^{u}: The UCM sent from the u^{th} UE at time step t.
- m_t^{u}: The DCM sent to the u^{th} UE at time step t.
- a_t^{u}: Environment action of the u^{th} UE at time step t.
- x_t^{u}: Agent state of the u^{th} UE at time step t.
- x_t^{b}: Agent state of the BS at time step t.

Observations

The observation $o_t^{\mathrm{u}} \in \{0, \ldots, B\}$ is an integer representing the number of SDUs in the buffer of the UE u at that time t. Similarly, the observation o_t^{b} received by the BS is a discrete variable with $U + 2$ possible states:

$$o_t^{\mathrm{b}} = \begin{cases} 0, \text{ if the UL-SCH is idle} \\ \mathrm{u}, \text{ if the UL-SCH is detected busy with a single PDU from UE u correctly decoded} \\ U + 1, \text{ non-decodable energy in the UL-SCH,} \end{cases} \tag{3.12}$$

where $\mathrm{u} \in \{1, \ldots, U\}$.

Actions

The environment action $a_t^{\mathrm{u}} \in \mathcal{A}_e = \{0, 1, 2\}$ is interpreted as follows:

$$a_t^{\mathrm{u}} = \begin{cases} 0\text{: do nothing} \\ 1\text{: transmit the oldest SDU in the buffer} \\ 2\text{: delete the oldest SDU in the buffer.} \end{cases} \tag{3.13}$$

Communication actions

We highlight that the DCM and UCM messages, m and n, are communication actions that the agents select while also being information available to the other agent's state as received message. The BS communication action is the vector comprising the DCMs sent to all UEs $\boldsymbol{m}_t \in \mathcal{D}^U$, with $\mathcal{D} = \{0, \ldots, V_{\mathrm{DCM}} - 1\}$ and $\mathcal{D}^n$ denotes the n-ary cartesian power of set $\mathcal{D}$. The communication action of a UE u is a single UCM $n_t^u \in \mathcal{U} = \{0, \ldots, V_{\mathrm{UCM}} - 1\}$ and the vector of all sent UCMs is $\boldsymbol{n}_t$.

Input states

The agent state at time step t is a tuple comprising the most recent k observations, actions and received messages, and it represents the input of the policy to determine the next action.

- UE u: $x_t^{\mathrm{u}} = (\chi_t^{\mathrm{u}} \ldots, \chi_{t-k}^{\mathrm{u}})$, where $\chi_t^{\mathrm{u}} = (o_t^{\mathrm{u}}, a_t^{\mathrm{u}}, n_t^{\mathrm{u}}, m_t^{\mathrm{u}})$. Highlighting that o^{u} is the local observation of that UE, a^{u} is the environment action taken, n^{u} represents the communication action taken and m^{u} is the received DCM.

- BS: $x_t^{\mathrm{b}} = (\chi_t^{\mathrm{b}} \ldots, \chi_{t-k}^{\mathrm{b}})$, where $\chi_t^{\mathrm{b}} = (o_t^{\mathrm{b}}, \boldsymbol{n}_t, \boldsymbol{m}_t)$, with $\boldsymbol{n}$ and $\boldsymbol{m}$ containing the messages to and from all UEs.

This formulation is illustrated in fig. 3.2 for the BS and UE policies, highlighting the observations and actions.

Given the above definitions, the size of the input and output layers for the actors are:

- UE: Input size is $k(V_{\mathrm{DCM}} + V_{\mathrm{UCM}} + 4)$, as it has 3 possible actions and the observation is a single scalar. Output size is $3 + V_{\mathrm{UCM}}$, corresponding to the environment actions and the UCM.
- BS: Input size is $k(2 + U + UV_{\mathrm{UCM}} + UV_{\mathrm{DCM}})$, as the observation is a one-hot vector of size $2 + U$. Output size is UV_{DCM}, corresponding to communication actions to all UEs.

The critic input size is the sum of all nodes actors inputs and outputs.

Reward

The reward function adopted tries to maximize the goodput while also trying to maximize the reliability. The reward is the same for all agents and it is calculated as the sum of contributions of each UE:

$$r_t = \sum r_t^u \tag{3.14}$$

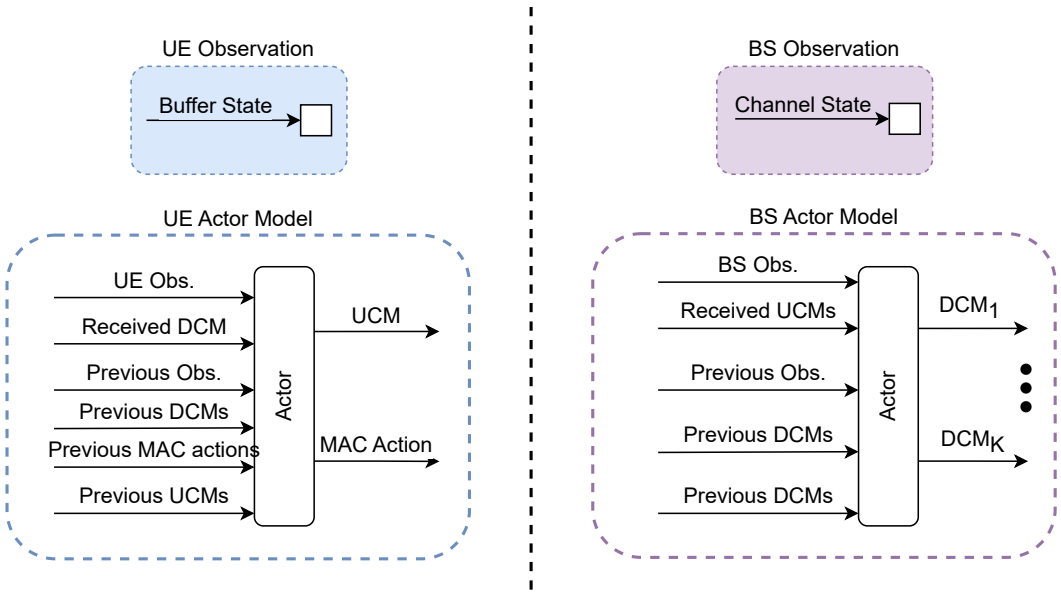


Figure 3.2: Scheme of the inputs and outputs of the policy network of the UE and BS in the multiple-access problem. The information contained in the observation is highlighted, while also indicating the other input information. The different actions are shown as outputs of the policies.

The reward term associated with each UE at each time step is:

$$r_t^u = \begin{cases} +\kappa, \text{if a new SDU from UE } u \text{ is received by the BS} \\ -\kappa, \text{if UE } u \text{ deletes a SDU not received by the BS} \\ 0, \text{ else,} \end{cases} \tag{3.15}$$

where κ is a positive integer. This choice of reward is possible by leveraging the centralized training and decentralized execution (CTDE). During the centralized training, a centralized reward system can be used to observe the buffers of the BS and UEs in order to assign the reward.

Coordination Metrics

With the above definitions, we can revisit the IC and IR equations, highlighting that the IC is defined for the UEs and the IR for the BS. The IC is the mutual information between the received DCM m^u_{t-1} and the action a^u_t, since the communication action of a node is the received message of another node at the next time-step, it measures the influence of the communication of an agent in the action taken by another. Hence, omitting the indexes:

$$\text{IC} = \sum_{m \in \mathcal{D}} \sum_{a \in \mathcal{A}_a} \mathbb{P}(m, a) \, \log_2 \left(\frac{\mathbb{P}(m, a)}{\mathbb{P}(m) \, \mathbb{P}(a)} \right) \tag{3.16}$$

Similarly, the IR is the mutual information of the vector of received messages $\boldsymbol{n}_{t-1}$ and the communication action $\boldsymbol{m}_t$:

$$\text{IR} = \sum_{\boldsymbol{n} \in \mathcal{U}^U} \sum_{\boldsymbol{m} \in \mathcal{D}^U} \mathbb{P}(\boldsymbol{n}, \boldsymbol{m}) \, \log_2 \left(\frac{\mathbb{P}(\boldsymbol{n}, \boldsymbol{m})}{\mathbb{P}(\boldsymbol{n}) \, \mathbb{P}(\boldsymbol{m})} \right) \tag{3.17}$$

where U is the number of UEs.

To calculate the maximum values of the IC and IR we use the fact that the mutual information of finite countable discrete random variables X and Y is bounded as:

$$I(X, Y) \leq \min \left(H(X), \, H(Y) \right) \tag{3.18}$$

where $H(X)$ is the entropy of the discrete random variable X. For a discrete random variable X with n_x categories:

$$\max \left(H(X) \right) = \log_2 n_x \tag{3.19}$$

Hence, for the IC:

$$\mathrm{IC} \leq \min\left(\log_2 V_{\mathrm{DCM}}, \log_2 3\right) \tag{3.20}$$

since the cardinality of $\mathcal{A}_e$ is 3. Similarly for the IR:

$$\mathrm{IR} \leq U \cdot \min\left(\log_2 V_{\mathrm{UCM}}, \log_2 V_{\mathrm{DCM}}\right) \tag{3.21}$$

3.2.2 Training Algorithm

The RL solution used in this chapter is based on the multi-agent deep deterministic policy gradient (MADDPG) algorithm [94]. This algorithm is well suited to Dec-POMDP when strong coordination is needed, due to its centralized critic architecture. Each agent has its own actor network that depends only on this agent's state in order to learn a decentralized policy ω_i parametrized by θ_i. Each agent also has a centralized critic network that receives the agent states and actions of all agents in order to learn a joint action value function $Q_i(x, a)$ parametrized by φ_i, where $x = (x_1, x_2, \ldots, x_n)$ contains all the agents' states and $a = (a_1, a_2, \ldots, a_n)$ contains the actions taken by all of the agents. The critic networks are only used during the centralized training. For more details of this algorithm, the interested reader can find it on the original paper [94].

Similarly to the original work [94], we use the Gumbel-softmax [96] trick to soft-approximate the discrete actions to continuous ones. The Gumbel-softmax reparameterization also works to balance exploration and exploitation. The exploration-exploitation trade-off is controlled by the temperature factor ζ.

We use the MADDPG as the reference training algorithm and propose some improvements to it. First, we make use of parameter sharing [14] for similar network nodes, in this case the UEs, meaning that these nodes have the same actor and critic

networks parameters.

Because of parameter sharing and also that UE index is not included in the agent's state, any policy that leverages the agent's identity is not capable of effectively solving the task due to the parameter sharing, because it would lead to collisions. This approach is employed as our goal is to learn a generic channel access policy, rather than a customized one. Although UE customized access policies might be interesting in some scenarios, such as indoor factories, for mobile phones it is preferred to have the access policy as general as possible. An example of a policy that leverages the agent's identity would be a round-robin policy.

Secondly, we modify the neural networks architectures by using deep dense architecture for reinforcement learning (D2RL) [98]. The main idea of D2RL is to use dense connections to pass the input of the network before every hidden layer, improving feature extraction and allowing deeper networks.

The actor and critic networks have the same architecture; a fully connected multilayer perceptron (MLP) with two hidden layers when D2RL is disabled and four hidden layers when using it. Each hidden layer has 64 neurons and their activation function is the rectified linear unit (ReLU).

Another improvement used is the jump-start reinforcement learning (JSRL) [99], which leverages an expert policy to improve the learning performance. The idea is that, with episodes duration of T, during the first t_{jsrl} steps, the expert policy is used and the learning agent policy is only used on the last $T - t_{\text{jsrl}}$ steps. When the performance of this composite policy surpasses a threshold, t_{jsrl} is decreased until only the learning agent policy is used. We use JSRL on top of a MADDPG algorithm improved with D2RL.

3.3 Results

3.3.1 Baseline Solutions

We compare the proposed solution with a contention-free (i.e. BS-controlled, scheduled) and a contention-based (i.e. grant-free) baseline.

In the contention-free protocol, the UE sends a scheduling request (SR) if its transmission buffer is not empty and it only transmits if it has received a scheduling grant (SG). Similarly, it only deletes a TB from the transmission buffer after the reception of an ACK. At each time step, the BS receives zero or more SRs. It then chooses one of the requesters at random to transmit in the next time-step, sending a SG to the selected UE. However, if the UE had made a successful data transmission simultaneously with an SR, the BS will send an ACK to this UE and its SR is ignored.

In the contention-based protocol, each UE transmits with probability p_t if its transmission buffer is not empty. Similarly to the contention-free baseline, the UE only deletes a TB after the reception of an ACK. At each time step, the BS sends an ACK to a UE if it receives a TB from the UE. For each experiment, the transmission probability chosen is the one that performs better in terms of goodput.

3.3.2 Simulation Procedure and Parameters

The transmission buffer of each user starts empty and the SDU arrival probability p_a is fixed and remains constant for all UEs. The system is trained for a fixed number of episodes N_train. During training, we evaluate the policy on a fixed set of N_eval evaluation episodes with disabled exploration and disabled learning to assess the current performance of the communication protocol. We then select the historical best protocol, which is the best performing one on the evaluation episodes during the whole

Table 3.1: Simulation Parameters

Parameter	Symbol	Value
Number of UEs	U	$[2, 4]$
Size of transmission buffer	B	20
Avg. number of SDUs per UE	λ	8
SDU arrival probability	p_{a}	$[0.16, 0.33]$
Transport block error rate	TBLER	$\left[10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\right]$
Transport block size (bit)	L_{TB}	1024
DCM vocabulary size	V_{DCM}	3
UCM vocabulary size	V_{UCM}	2
Duration of episode (TTIs)	T	24
TTI duration (ms)	T_{TTI}	0.1
Reward function parameter	κ	3
Number of training episodes	N_{train}	100k
Number of evaluation episodes	N_{eval}	500
Number of test episodes	N_{test}	5000
Number of randomized repetitions	N_{rep}	8

training procedure and its performance is further assessed in N_{test} episodes with exploration and learning disabled, this is the final testing phase. This whole procedure represents a single training repetition.

We evaluate a total of N_{rep} repetitions, each with a different random seed. After training finishes, we have successfully trained a population of N_{rep} protocols, and we can select the best performing protocol on the training episodes, which is the historical best across all repetitions. This selection step can be seen as a *survival-of-the-fittest* approach because only one protocol of the population of N_{rep} is chosen going forward. A summary of the main simulation parameters is provided in table 3.1, while the parameters of the MARL training algorithms is listed in table 3.2.

Table 3.2: Training Algorithm Parameters

Parameter	Symbol	Value
Memory length	k	3
Replay buffer size		10^5
Batch size		1024
Number of neurons per hidden layer		64
Interval between updating policies		96
Number hidden layers		$[2, 4]$
Activation function of hidden neurons		ReLU
Optimizer algorithm		Adam
Learning rate		10^{-3}
Discount factor		0.9
Policy regularizing factor		10^{-3}
Gumbel-softmax temperature factor	ζ	1
Target networks soft-update factor		10^{-3}

3.3.3 Emerging or Learning?

As discussed in section 3.1.2, a new protocol could achieve higher performance due to, e.g. a better channel-access policy. In this case, the merit for the gains would not be on the learned signaling. Hence, to evaluate if holistic protocol emergence can outperform protocol learning, we compare them next in a channel-access learning setup. Let us first introduce some definitions:

- **Protocol emergence**: Both control-plane vocabulary, policies and channel-access policies of UEs and the BS are learned.
- **Protocol learning**: The control-plane and channel-access policies of the UEs are learned, while the BS follows the grant-based baseline and an a-priori fixed control-plane vocabulary.
- **Channel-access learning**: Only the medium access policy of the UEs is learned while their control-plane and the BS follow the grant-based baseline and an a-

priori fixed control-plane vocabulary.

The MARL learning algorithm we used is MADDPG with D2RL.

In fig. 3.3, these three different ways of producing protocols are compared together with the contention-free baseline. Figure 3.3a shows that convergence is faster with channel-access learning and protocol learning, and that protocol emergence produces better performing protocols. Figure 3.3b shows that even though the channel-access and protocol learning solutions have better performance than the baseline, they have a ceiling when training under different TBLERs scenarios, whereas protocol emergence is capable of continously improve and produce better protocols for scenarios with reduced TBLER.

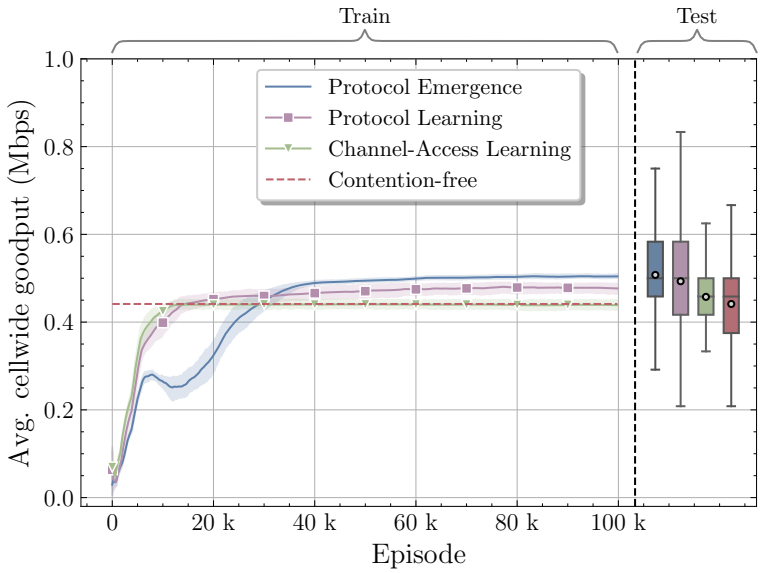
The performance of channel-access learning is limited by the control-plane signaling used, as it has a similar performance than the baseline used for its control-plane. However, protocol learning allows a higher degree of freedom when compared with the channel-access learning, which leads to a increase in performance. Protocol emergence further improves on this allowing the design of better performing protocols.

In general terms, the control-plane of the channel-access learning has to follow the rules set by the pre-defined protocol. Protocol learning allows breaking some of the control-plane rules, whilst protocol emergence allows new control-plane rules to emerge.

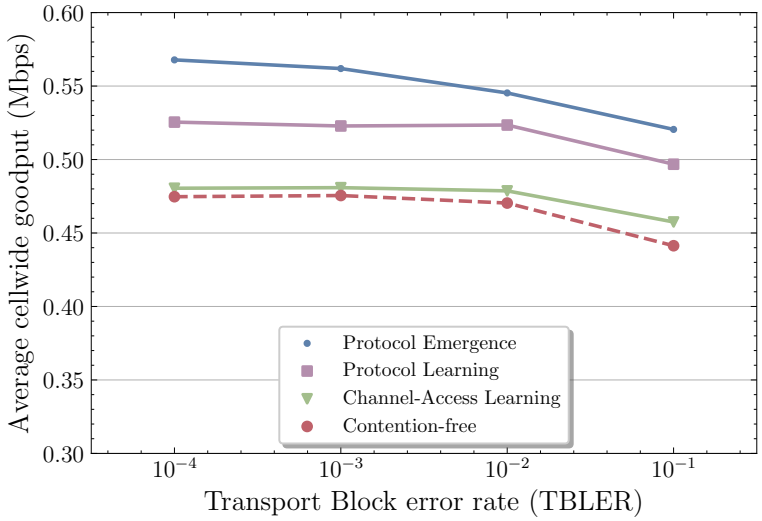
3.3.4 Producing a Protocol

MARL algorithm design

In fig. 3.4 we compare the performance of three different techniques for emerging a protocol and of the baselines on two scenarios, an easy one with 2 UEs and a harder one



(a) Learning curves comparison for TBLER = 10^{-1}.



(b) Comparison of the final performance of the best protocol produced over different TBLERs.

Figure 3.3: Comparison of different ways to produce a protocol. For the learning curve in (a), the solid lines indicate average performance in the evaluation episodes, the shaded region depict the 95% confidence interval (CI) and the boxplots compare the performance of the best historical protocol on the test episodes. The training is done for different TBLERs and the final performance of the best protocol produced for each TBLER is compared in (b). The mean number of SDUs in the cell per episode fixed is $U \cdot \lambda = 16$; UL and DL vocabulary sizes are $V_{\text{UCM}} = 2$ and $V_{\text{DCM}} = 3$.

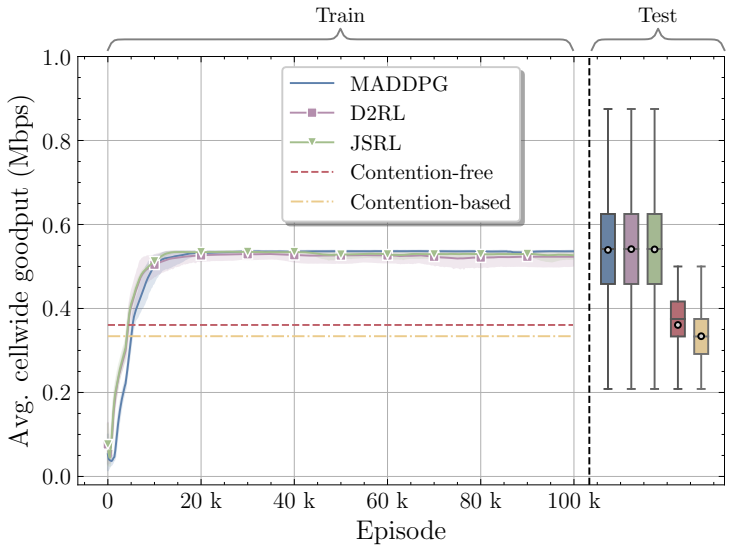
with 4 UEs. On the easy task, the emerged protocols all have similar learning curves and produce similar performing protocols at the end. However, on the harder task the benefits of adding the D2RL become apparent and, although the end performance is the same, using JSRL on top of the already improved algorithm hastens learning even further. For the remainder of this chapter, we will focus on the harder task.

Leveraging an expert

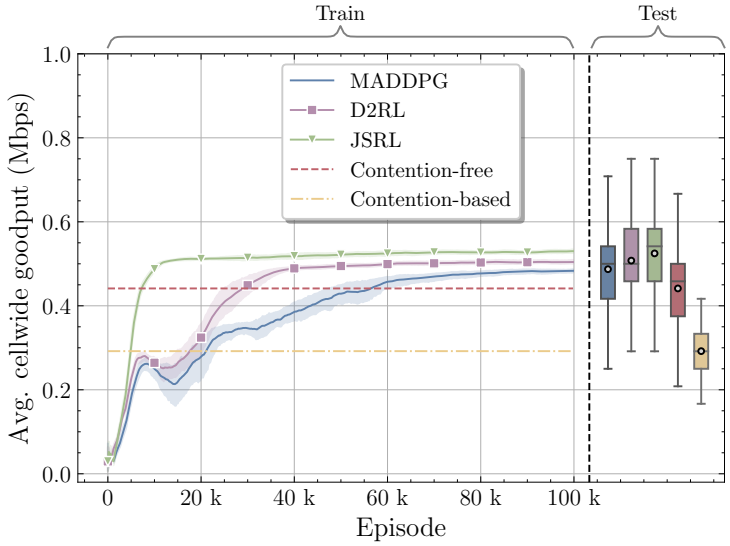
The results in fig. 3.4 indicate that using a expert policy to accelerate learning, JSRL, is a good practice as it hastens learning. This idea can not only be used as a way of leveraging a general-purpose protocol to help emerge an application-tailored one, but also for improving an already well defined application-tailored protocol. However, this approach limits the vocabulary sizes to the same as the expert, hindering the search of protocols across the signaling spaces. JSRL is one technique that can be used to accelerate learning with prior knowledge, others include learning from demonstrations [101], imitation learning [102] and supervised self-play [103]. JSRL is used in this chapter because it is compatible with any RL algorithm, it uses the expert policy directly instead of needing a dataset and it showed faster convergence compared with other imitation learning techniques.

How much signaling is needed?

One of the benefits of emerging a protocol from scratch with MARL is that it can produce protocols with different amounts of signaling by varying the UL and DL vocabulary sizes. This allows a search across the protocol space in order to control the trade-off between signaling and performance or to search for the minimum amount of



(a) Avg. number of SDUs per UE: $\lambda = 8$. Arrival rate: $p_a = 0.33$. Number of UEs: $U = 2$



(b) Avg. number of SDUs per UE: $\lambda = 4$; Arrival rate: $p_a = 0.16$. Number of UEs: $U = 4$

Figure 3.4: Comparison of learning curves during the training procedure in terms of goodput. Three training solutions for emerged protocols are compared with the baselines: MADDPG, MADDPG with D2RL and MADDPG with D2RL and JSRL. The solid lines show the average performance in the evaluation episodes and the shaded areas represent the 95% CI. The boxplots compare the performance of the best historical protocol on the test episodes. Avg. number of SDUs in the cell per episode fixed: $U \cdot \lambda = 16$; TBLER = 10^{-1}; UL and DL vocabulary sizes are $V_{\text{UCM}} = 2$ and $V_{\text{DCM}} = 3$.

signaling needed for a specific task. Figure 3.5 illustrates such study, which indicates that the only constraint in terms of vocabulary size to effectively solve this particular task is $V_{\text{DCM}} \geq 2$. More complex tasks may need larger vocabularies, making automatic protocol emergence appealing, since the minimum amount of signaling needed is unknown beforehand.

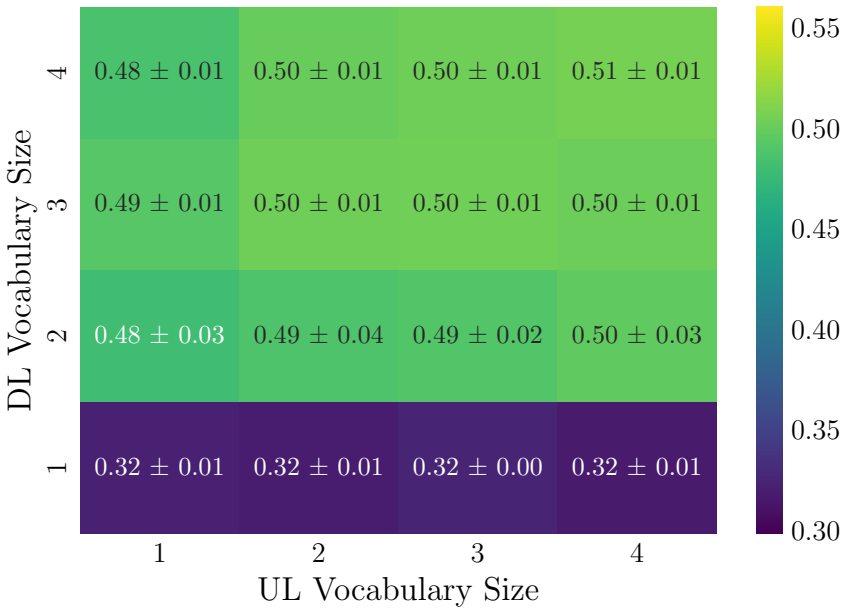
3.3.5 Coordination Analysis

Studying the Coordination

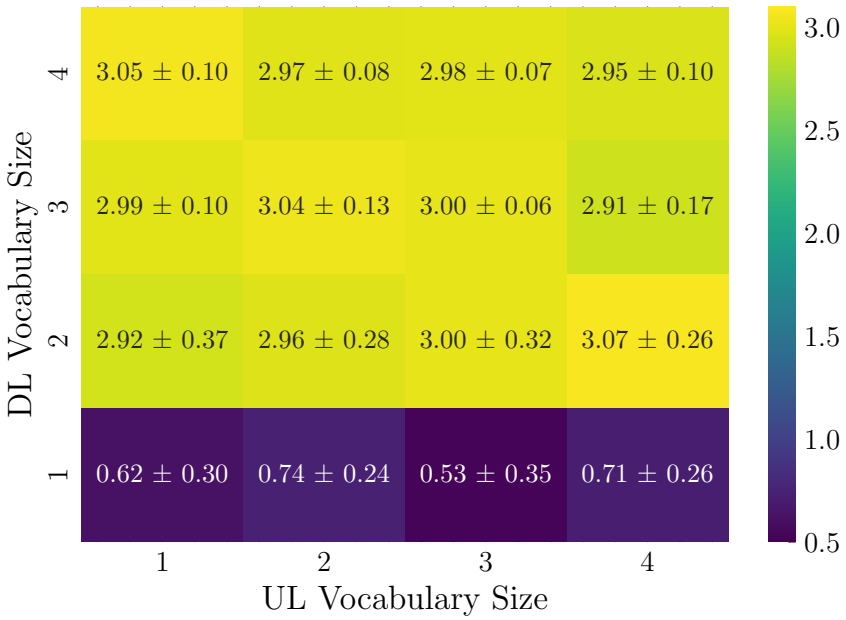
To effectively produce a protocol with good performance, the network nodes have to coordinate with one another contructively. Hence, one important step of protocol emergence is the analysis of the coordination and how it correlates with the other network KPIs. In this chapter, we are mainly interested in the IR and IC metrics, which are calculated empirically for the whole set of test episodes. The IR is the mutual information between the messages received and the communication action, so, it is calculated for the BS. The IC is the mutual information between the received message and the environment action taken and it is calculated for the UEs.

Effect of coordination

Figure 3.6 illustrates the correlation between IC and IR and the effect of coordination on network performance. A key observation is that tighter signaling-based coordination across radio nodes leads to improved performance for the emerged protocols, as illustrated in the increase in goodput in fig. 3.6a. This is even more prominent in fig. 3.6b, where the effect signaling coordination on collision-rate is illustrated, as coordination is necessary to avoid collisions between the UEs. It is also important to highlight that random initializations of the protocol models and learning algorithms

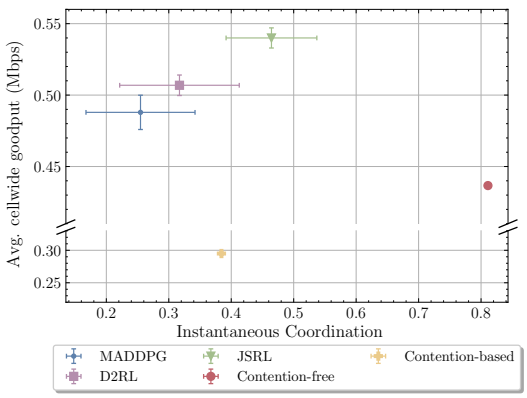


(a) Heatmap of the average goodput

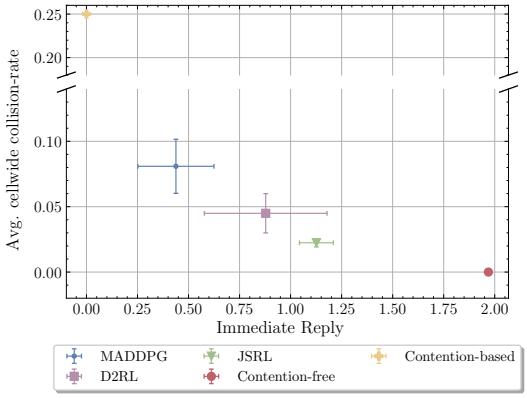


(b) Heatmap of the average delay

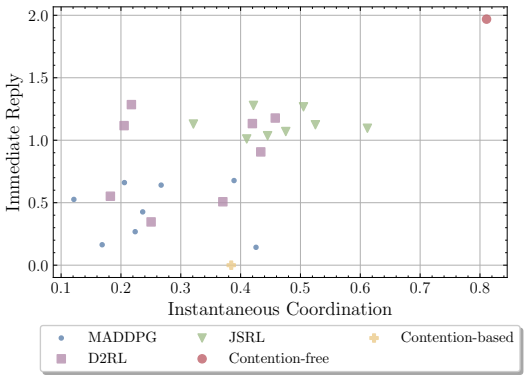
Figure 3.5: Study of signaling vocabulary size and its impact on performance. The values represent the average of all repetitions on the final test episodes. The annotations indicate the mean value and 95% CI. The training algorithm used is the MADDPG with D2RL. Avg. number of SDUs per UE: $\lambda = 4$; Arrival rate: $p_a = 0.16$; Number of UEs: $U = 4$; TBLER = 10^{-1}.



(a) IC and goodput



(b) IR and collision-rate



(c) IC and IR

Figure 3.6: Study of the correlation between the coordination metrics with themselves and with some network KPIs. For the emerged protocols the UL and DL vocabulary sizes are $V_{\text{UCM}} = 2$ and $V_{\text{DCM}} = 3$. Avg. number of SDUs per UE: $\lambda = 4$; Arrival rate: $p_{\text{a}} = 0.16$; Number of UEs: $U = 4$; TBLER = 10^{-1}. The error bars show the 95% CI.

will produce a large and heterogeneous collection of protocols for the same hyperparameters. This variance is illustrated by the error bars in fig. 3.6 and is the main reason for the survival-of-the-fittest approach to protocol selection discussed in section 3.3.2.

We highlight that it is possible to motivate the agents to achieve better coordination by adding a term related to the coordination to the reward of an agent, as in [100]. For example, the IR could be added to the reward for the BS and the IC for the UEs. This idea may be important for some scenarios where a higher degree of coordination is absolutely necessary and maybe difficult to achieve.

3.3.6 Profiling the Protocols

In order to compare protocols, we characterize them in terms of network KPIs and coordination metrics. This way, it is possible to draw a profile of each protocol. For example, in table 3.3 we compare the best protocol of each RL solution with the baselines.

At the top of table 3.3, we compare the different ways of designing protocols and on the bottom, different solutions for protocol emergence. We add the protocol with least signaling that obeys the constrain from the vocabulary size study illustrated in fig. 3.5a with $D = 2$ and $U = 1$ and labeled D2RL-MS. A variation of the MADDPG with D2RL is also compared, labeled D2RL-EE, where the reward function is modified with a penalization for collisions. In this case, the penalization is equal to the number of UEs that collided. The scenario in consideration is the same as in fig. 3.4b.

Comparing protocols

When comparing multiple protocols like this, one can select the best-suited protocol in terms of multiple constraints. For example, the JSRL is the most well-rounded,

Name	**Goodput** (Mbit s^{-1})	**Collision-rate**	**Delay** Λ (ms)	**EE**	**Reliability** Ψ	**Signaling Overhead** Υ	**IR**	**IC**
Protocol Emergence	**0.52**	5%	2.80	0.76	99.98%	3.00	1.01	0.40
Protocol Learning	0.50	2%	3.18	0.84	**100**%	3.00	1.85	0.70
Channel-Access Learning	0.46	4%	3.45	0.62	99.85%	3.00	1.93	0.51
Contention-Free	0.44	**0.0**	4.47	**0.90**	**100**%	3.00	1.95	0.80
Contention-Based	0.29	25%	5.08	0.34	**100**%	**1.00**	0.0	0.37
MADDPG	0.49	9%	3.14	0.68	1.00	3.00	0.64	0.41
JSRL	**0.53**	3%	**2.72**	0.83	99.98%	3.00	0.87	0.45
D2RL	0.52	5%	2.80	0.76	99.98%	3.00	1.01	0.40
D2RL-EE	0.50	**0.0**	3.13	**0.90**	99.99%	3.00	1.49	0.49
D2RL-MS	0.50	5%	3.07	0.77	99.99%	**1.00**	0.0	0.36

Table 3.3: Protocol profiles. For the solutions at the top, the MADDPG with D2RL is used as the learning architecture.

whereas if the interest is on the best goodput with low signaling, then the emerged protocol with minimum signaling, D2RL-MS, would be the best protocol.

Effect of reward

One of the advantages of designing a protocol with RL is the potential to influence its characteristics through the objective function, in this case, the reward function. This is illustrated in table 3.3 by the difference between D2RL-EE and the other emerged protocols in terms of the trade-off between collision-rate and delay. By changing the reward function, the emerged protocol lowers the collision-rate, improving the EE while increasing the delay.

3.3.7 Interpreting the Protocols

On emerged protocols, the control messages initially have no assigned meaning. However, as the training goes on, the agents agree upon how should they communicate and behave, thus defining the meaning of the messages and how they are used by their policies. Thus, studying the meaning of these messages is a necessary step for interpreting and explaining the emerged protocol. For simplicity, we will focus on the study of the downlink messages and how they influence the UE behavior.

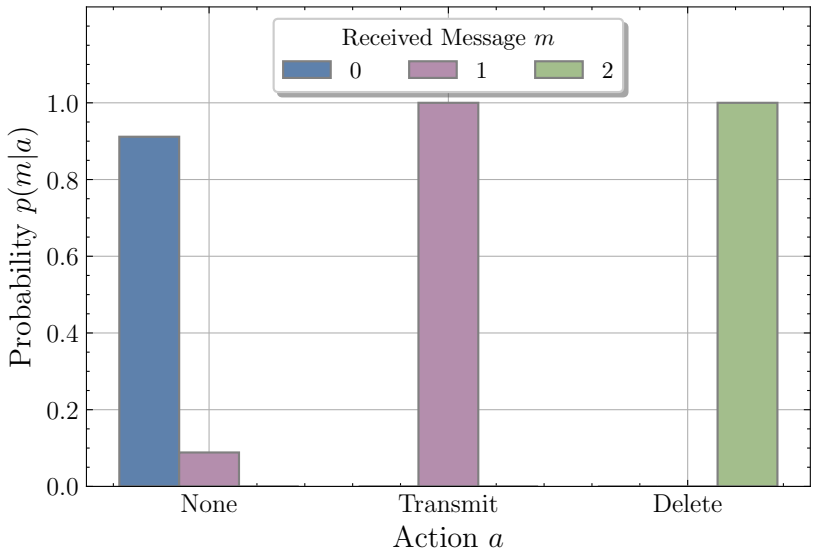
We use the conditional probability of a control message m being the received one given that action a was taken, $\mathbb{P}(m|a)$, to assess how the control messages are used by a protocol. The probabilities are calculated empirically through the transitions across N_{test} episodes.

Deriving the meaning

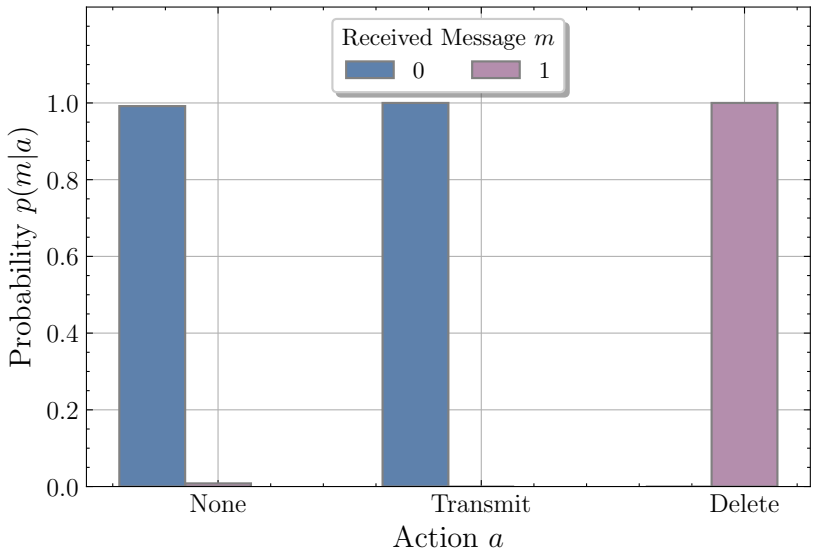
This approach is illustrated in fig. 3.7 for the baseline approaches, contention-free and contention-based and fig. 3.8 for emerged protocols. As an example on how to derive the meaning from the probabilities, in fig. 3.7a, the delete action is only taken when message $m = 2$ is received. This suggests that message $m = 2$ behaves as an ACK. A similar approach can be used to indicate that $m = 1$ is the SG for the contention-free baseline and that $m = 1$ is the ACK for the contention-based solution in fig. 3.7b.

Meaning on emerged protocols

When we analyze the meaning of the messages for the JSRL protocol, we notice that it is very similar to the contention-free baseline, which is used as the expert, the ACK is related to $m = 2$ and the SG to $m = 1$. This behavior is the same for all the repetitions with JSRL. However, we can see that there are some cases that $m = 2$ leads to a transmission, so its meaning is not only of ACK. This meaning fluidity is even more noticeable on a protocol emerged from scratch, as in fig. 3.8b where $m = 0$ is the only message interpreted as ACK, but it also can lead to a transmit action on some cases. This meaning fluidity leads to *contextual meaning*, as the meaning of a message is not firmly defined and depends on the context.

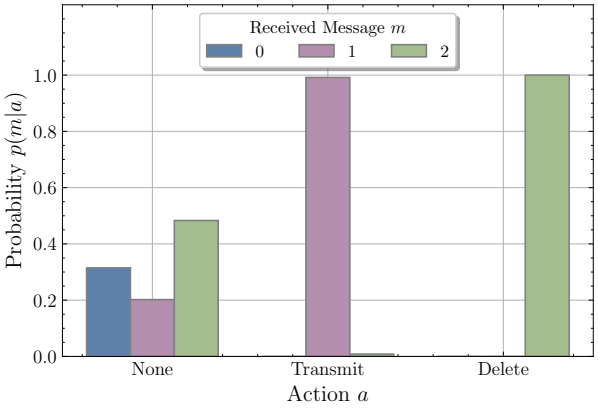


(a) Contention-free baseline

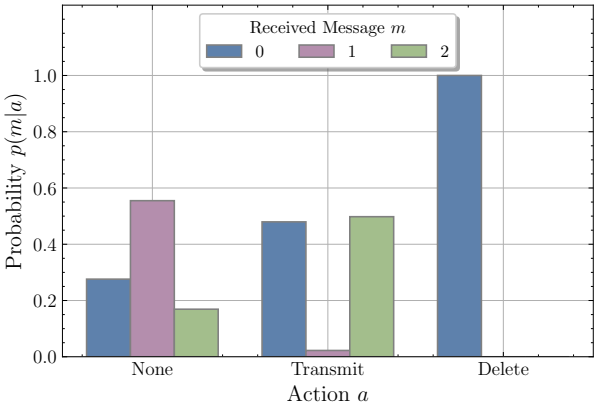


(b) Contention-based baseline

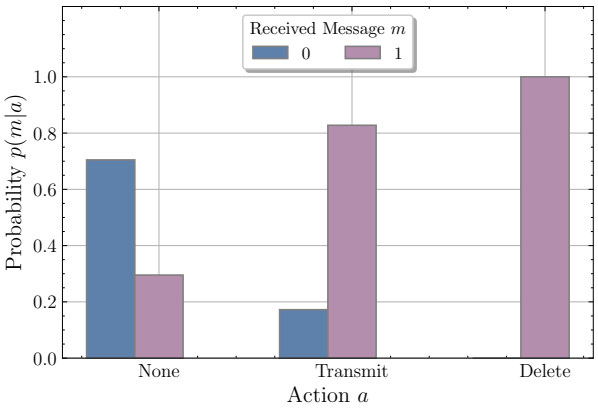
Figure 3.7: Study of the meaning of the downlink messages for the reference protocols. Avg. number of SDUs per UE: $\lambda = 4$; Arrival rate: $p_a = 0.16$; Number of UEs: $U = 4$; TBLER = 10^{-1}.



(a) JSRL: DL vocabulary size $V_{\text{DCM}} = 3$



(b) D2RL: DL vocabulary size $V_{\text{DCM}} = 3$



(c) D2RL-MS: DL vocabulary size $V_{\text{DCM}} = 2$

Figure 3.8: Study of the meaning of the downlink messages for the emerged protocols. The UL vocabulary size is $V_{\text{UCM}} = 2$. Avg. number of SDUs per UE: $\lambda = 4$; Arrival rate: $p_{\text{a}} = 0.16$; Number of UEs: $U = 4$; TBLER = 10^{-1}.

Contextual meaning

The ability to derive a contextual meaning is an important feature of emerged protocols and it explains the signaling constraint of $V_{\mathrm{DCM}} \geq 2$, for the vocabulary size study as illustrated in fig. 3.8c. The emerged protocol, D2RL-MS, uses this feature in order to have the same control message leading to a delete action in some cases, similarly to an ACK, and to a transmit action on others, as a SG. Because of this, the BS can orchestrate the UEs and effectively control the system even with less signaling, while the UE can use other information such as its previously transmitted UCM and its buffer status to determine which action to take.

3.4 Chapter Summary

In this chapter, we have proposed a novel framework for the study of emergent protocols, which includes the protocol production, coordination analysis, protocol profiling and interpretation. We have shown how, using protocol emergence techniques, radio nodes can learn the signaling vocabulary and policy, as well as the channel-access policy, which is equivalent to physical layer (PHY) control by the agents.

We have shown that the average emerged protocol outperforms two very different baselines (contention-free and contention-based), and highlighted the strengths of these methods for searching the vast space of signaling vocabularies to achieve solutions with low signaling overhead. The second step of our framework is concerned with the relationship between coordination and network performance. Our results indicate that a better learning algorithm achieves better performance not necessarily only through a superior channel-access policy, but also thanks to improved coordination through signaling.

The third step of our framework lets us compare different protocols in terms of network KPIs and coordination metrics. This creates a protocol profile that is useful to evaluate the trade-off when selecting one protocol over another in a specific scenario. The last step discusses interpretability, presenting a simple way of understanding how the control messages are used through conditional probability.

Chapter 4

MAC Protocol Emergence for Contiguous Resource Allocation

Contents

In this chapter, we focus on studying the performance of protocol emergence in a more challenging problem, evaluating its learning capabilities, scalability, and limitations. Differently from chapter 3, the scenario of this study involves multiple frequency

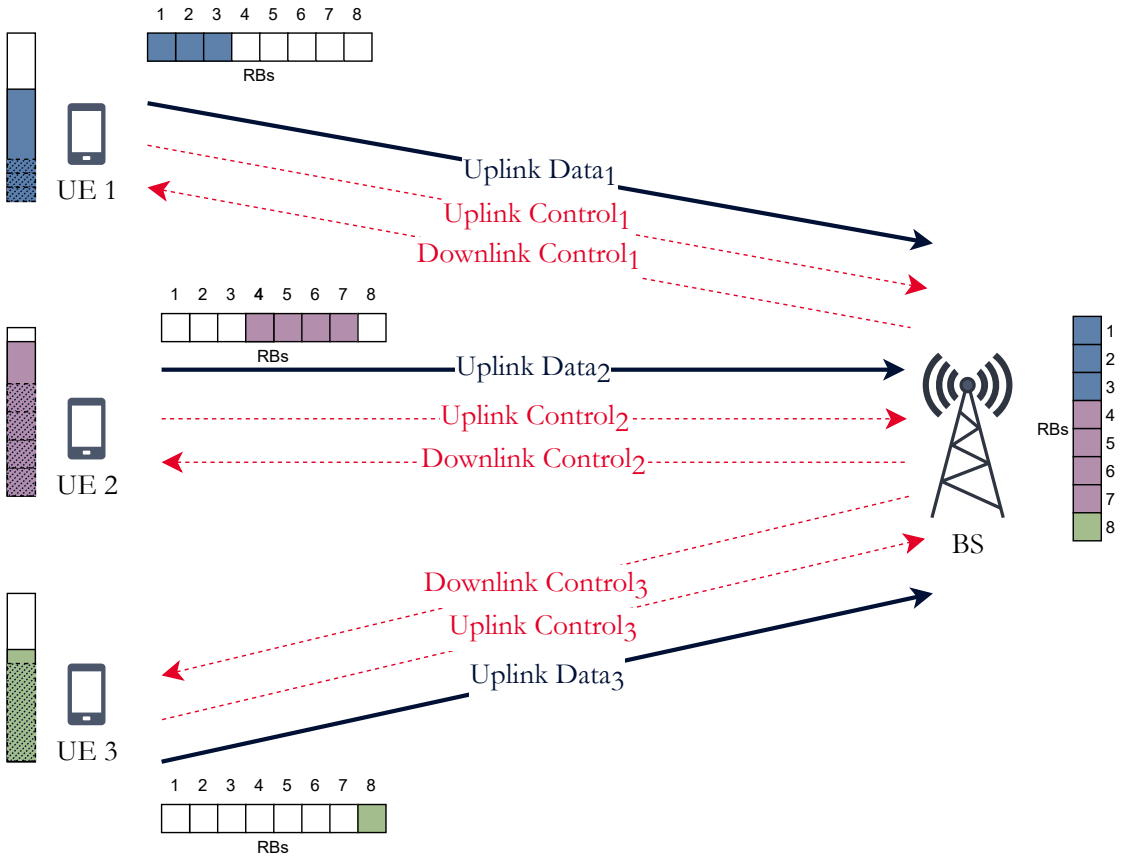


Figure 4.1: System model scheme with contiguous resource allocation. The buffer and decisions of each user equipment (UE) are highlighted besides it. The wireless channel is shown besides the base station (BS), highlighting the frequency resource axis on that time slot, indicating the resource blocks (RBs) used by each UE.

resources in a scheduling problem with contiguous frequency-domain resource allocation.

4.1 System Model

Consider a single cell with a BS serving U UEs in a uplink slotted shared channel (USC), where each UE needs to deliver data to the BS. For the frequency domain, the available bandwidth is divided into M RBs. Time domain is assumed slotted with a fixed time-step duration of T_{TTI}. We assume an episode of duration T T_{TTI}s.

The BS and UEs constitute the nodes of the system who act as independent agents. The network nodes can exchange information, using messages through the control channels. In the remainder of this paper, we refer to the UE medium access control (MAC) agents and the BS MAC agent as UE and BS, respectively.

Each UE has a transmission buffer of capacity B in bits initially empty. Data arrives in service data units (SDUs) of fixed size L_{SDU}. The SDU arrival is modeled as a Bernoulli process with probability of arrival p_{a}. So, at each time step, a new SDU of size L_{SDU} is added to the buffer with probability p_{a}, until a maximum number T steps is achieved. The average number of information bits arriving at each UE's buffer in any given episode of duration T is then:

$$\lambda_u = p_{\text{a}} T L_{\text{SDU}} \tag{4.1}$$

We assume that a automatic repeat request (ARQ) process is used by the network nodes to handle the buffer management, as such, upon successful reception of a transport block (TB) by the BS, the UE remove those bits from its buffer.

The system is channel-aware, with each UE having a signal-to-noise ratio (SNR) given by:

$$\text{SNR}_u = \frac{|\tilde{h}_u \sqrt{\rho_u}|^2}{\sigma^2} p_t \,, \tag{4.2}$$

where $\tilde{h}_u \sqrt{\rho_u}$ is the effective channel of UE u, with $\tilde{h}$ denoting the complex gain and ρ denoting the path loss. The path loss and gains are assumed constant for the episode duration, due to its short duration. A time-correlated flat channel with Rayleigh fading based on the Jakes model is assumed for the complex gain calculations [104].

Given a SNR_u, a modulation and coding scheme (MCS) is chosen by a look-up table

link adaptation method. The method chooses the highest MCS with an average error rate lower than the target transport block error rate (TBLER) for that SNR. The spectral efficiency, η, is the maximum number of information bits in a resource element (RE), given by the product of the code rate ν_r and the modulation order ϱ_m:

$$\eta = \varrho_m \nu_r \,. \tag{4.3}$$

With a selected MCS and given the number of RBs for transmission, n_{RB}, it is possible to calculate the maximum number of information bits that can be transmitted, which is the maximum transport block size (TBS), as:

$$\mathrm{TBS}_{\max} = n_{\mathrm{RB}} N_{\mathrm{sc}}^{\mathrm{RB}} N_{\mathrm{symb}}^{\mathrm{sh}} \varrho_m \nu_r \,, \tag{4.4}$$

where $N_{\mathrm{sc}}^{\mathrm{RB}}$ is the number of subcarriers in a RB and $N_{\mathrm{symb}}^{\mathrm{sh}}$ is the number of symbols in a slot. The slot follows format 1, that is, all orthogonal frequency-division multiplexing (OFDM) symbols within a slot are used for uplink shared channel (UL-SCH) data transmission. Transmission errors affect the whole TB.

The channel for the uplink data transmission is modeled as a packet erasure channel, where a TB is incorrectly received with a probability referred to as the TBLER. As contiguous resource allocation is assumed, data transmission occurs in adjacent RBs for a given UE. Collisions may happen if more than one UE transmit over the same resource, in this case the whole TBs are not received correctly. The downlink control messages (DCMs) and uplink control messages (UCMs) are transmitted over the downlink (DL) and uplink (UL) control channels respectively, which are assumed to be dedicated and error free, so without any contention or collision. In this scenario, the

DCMs are usually used to indicate the resource allocation of choice and the UCMs are usually used to indicate the amount of traffic or priority of a UE.

We assume that the sets of possible DL and UL control messages have cardinality V_{DCM} and V_{UCM}, respectively. For a DL (resp. UL) control vocabulary of size V_{DCM}, the bitlength Υ_{DL} is equal to $\lceil(\log_2 V_{\text{DCM}})\rceil$, where $\lceil\cdot\rceil$ represents the ceiling function.

At each time step t of duration T_{TTI}, the BS can send one control message to each UE and each UE can send one control message to the BS while being able to send protocol data units (PDUs) through the UL-SCH. The PDU contains a TB and the UE has to select the RBs for transmission, which is done by deciding the initial RB and also the number of RBs. It is important to differentiate the SDU and PDU, the SDU is the packet that arrives on the MAC from the upper layers, while the MAC PDU is represents the TB sent to the physical layer (PHY). As such, a MAC PDU can contain multiple SDUs or even a part of a SDU. Figure 4.1 illustrates the system model, highlighting the decisions taken by each UE and its effect on the system.

4.1.1 Network KPIs

Differently from chapter 3 where performance was mainly compared in terms of bit-rate, the goal for this task is to produce good protocols in terms of fairness. As such, defining the goodput G_u (in Mbit s^{-1}) of an UE u as the number of information bits received by the BS from UE u per unit of time:

$$G_u = \frac{N_{\text{RX}}}{T\, T_{\text{TTI}}} \tag{4.5}$$

The main performance metrics of interest are the average goodput, its geometric mean and the Jain's fairness index (JFI) [105], defined as:

$$\text{JFI} = \frac{\left(\sum_u G_u\right)^2}{U \sum_u (G_u)^2}\,. \tag{4.6}$$

The collision-rate Γ is the total number of resources in which a collision happened divided by the total number of resources in the grid:

$$\Gamma = \frac{N_\text{c}}{TM}. \tag{4.7}$$

where N_c represents the total number of resources in which at least two SDUs collided.

The other metrics used in chapter 3 such as delay, reliability, and coordination metrics will not be evaluated here as the goal of this chapter is not a deep study and comparison of protocols, but of the capabilities and limitations of the proposed method of joint learning signaling and scheduling. Signaling overhead will be mainly evaluated from the DL and UL vocabulary sizes.

4.2 MARL Problem Formulation

4.2.1 MARL Formulation

The problem defined above is formulated, similarly to the chapter 3, as a multi-agent reinforcement learning (MARL) cooperative task. The UE agents need to learn when to send data and in which RBs. Differently from the previous chapter, the UEs do not need to learn the buffer management, as this handled by an ARQ process. Both the UEs and BS also learn how to communicate with each other through the control

channel, sending the relevant information in order to cooperate and avoid collisions, which is used as input by the network nodes to decide how to act. The BS also takes into account the spectral efficiencies of the MCS of each UE, the normalized amount of data received from each UE and the state of each RB, i.e. idle, busy or collision-free reception. Aside from the control messages, the UEs takes into consideration their buffer status in order to inform its decision.

This problem is modeled as a decentralized partially observable Markov decision process (Dec-POMDP) [92], augmented with communication. The action space of each agent is subdivided into one environment action space and a communication action space. The communication action represents the message sent by an agent, and it does not affect the environment directly, but it may be passed to other agents. We also assume that this is a cooperative problem, as such, all agents share the same reward. In this thesis, the agent internal state x_i may comprise not only the agent's current observation, but also previous observations, actions and received messages. We assume the episode ends when a maximum number of steps T is reached.

We use the following notations:

- o_t: Observation received by the agent at time step t.
- n_t^{u}: The UCM sent from the u^{th} UE at time step t.
- m_t^{u}: The DCM sent to the u^{th} UE at time step t.
- a_t^{u}: Environment action of the u^{th} UE at time step t.
- x_t: Agent state at time step t.

Observations

The UE observation o_t^{u} is an integer representing the integer division of the number of bits in the buffer of the UE u and its current spectral efficiency at that time t. This value represents the number RBs needed to send all the data.

The observation o_t^{b} received by the BS is a combination of three vectors, the channel information $\boldsymbol{c}$, the received data information $\boldsymbol{g}$ and the spectral efficiency information $\boldsymbol{\eta}$:

$$o_t^{\mathrm{b}} = [\boldsymbol{c}\ \boldsymbol{g}\ \boldsymbol{\eta}]\ . \tag{4.8}$$

The vector $\boldsymbol{c}$ contains the channel information relative to each RB, c_i, which is a discrete variable with $U + 2$ possible states:

$$c_i = \begin{cases} 0, \text{ if the RB } i \text{ is idle} \\ u, \text{ if successfully received data from UE } u \text{ on RB } i \\ U + 1, \text{ non-decodable energy in the RB } i, \end{cases} \quad , \forall i \in \{1, \ldots, M\} \tag{4.9}$$

where $u \in \{1, \ldots, U\}$. The received data information is the normalized amount of bits received from each UE

$$g_u = \frac{N_{\mathrm{RX}}^u}{\max\limits_i N_{\mathrm{RX}}^i} \tag{4.10}$$

and $\boldsymbol{\eta}$ contains the current spectral efficiency of the MCS of each UE, η_u.

Actions

Since the UE decision involves selecting the initial RB and the number of RBs for transmission, we model the action as vector containing the main action, a_{m} and two

action parameters, a_{idx} and a_{off}, similarly to the action design in [106]. The main action $a_{\text{m}} \in \mathcal{A}_e = \{0, 1\}$ is interpreted as follows:

$$a_{\text{m}} = \begin{cases} 0\text{: do nothing} \\ 1\text{: transmit data} \end{cases} \tag{4.11}$$

The first action parameter $a_{\text{idx}} \in \{1, \cdots, M\}$ indicates the initial RB for the contiguous allocation and the second action parameter $a_{\text{off}} \in \{1, \cdots, M\}$, represents the number of RBs allocated for transmission, also called the offset.

When compared with the formulation of chapter 3, the above formulation introduces the two new action parameters, while removing one main action, the delete option. In terms of parameters for the actor network, this action formulation has a total size of $2 + 2M$.

Communication actions

The DCM and UCM messages, m and n, are communication actions that the agents select while also being information available to the other agent's state as received message. The BS communication action is the vector comprising the downlink messages sent to all UEs $\boldsymbol{m}_t \in \mathcal{D}^U$, with $\mathcal{D} = \{0, \ldots, V_{\text{DCM}} - 1\}$ and $\mathcal{D}^n$ denotes the n-ary Cartesian power of set $\mathcal{D}$. The communication action of a UE u is a single UCM $n_t^u \in \mathcal{U} = \{0, \ldots, V_{\text{UCM}} - 1\}$ and the vector of all sent UCMs is $\boldsymbol{n}_t$.

Input states

The agent state at time step t is a tuple comprising the most recent k observations, actions and received messages.

- UE u: $x_t^{\mathrm{u}} = \left[\xi_t^{\mathrm{u}} \ldots, \xi_{t-k}^{\mathrm{u}}\right]$, where $\xi_t^{\mathrm{u}} = \left[o_t^{\mathrm{u}}, a_t^{\mathrm{u}}, n_t^{\mathrm{u}}, m_t^{\mathrm{u}}\right]$.
- BS: $x_t^{\mathrm{b}} = (\xi_t^{\mathrm{b}} \ldots, \xi_{t-k}^{\mathrm{b}})$, where $\xi_t^{\mathrm{b}} = (o_t^{\mathrm{b}}, \boldsymbol{n}_t, \boldsymbol{m}_t)$, with $\boldsymbol{n}$ and $\boldsymbol{m}$ containing the messages to and from all UEs.

This formulation is illustrated in fig. 4.2 for the BS and UE policies, highlighting the observations and actions.

Given the above definitions, the size of the input and output layers for the actors are:

- UE: Input size is $k(1 + V_{\mathrm{DCM}} + V_{\mathrm{UCM}} + 2 + 2M)$, as the action has size $2 + 2M$ and the observation is a single scalar. Output size is $2 + 2M + V_{\mathrm{UCM}}$, corresponding to the environment actions and the UCM.
- BS: Input size is $k((2+U)M + 2U + UV_{\mathrm{UCM}} + UV_{\mathrm{DCM}})$, as the observation has size $(2 + U)M + 2U$. Output size is UV_{DCM}, corresponding to communication actions to all UEs.

The critic input size is the sum of all nodes actors inputs and outputs.

Reward

The reward function adopted is the same for all agents and defined as the geometric mean of the goodputs of all UEs:

$$r_t = \sqrt[U]{\prod_{u=1}^{U} G_u} \tag{4.12}$$

The case in which $G_u = 0$ is handled by assuming that at least one bit was received from each UE, that is, if for an UE $N_{\mathrm{RX}} = 0$, then the reward is calculated using $N'_{\mathrm{RX}} =$

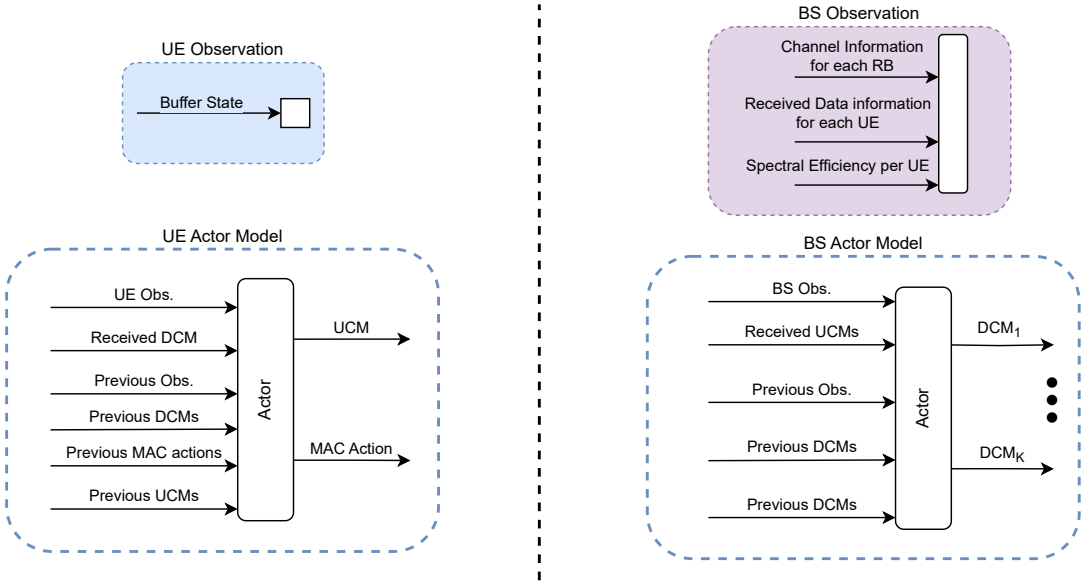


Figure 4.2: Scheme of the inputs and outputs of the policy network of the UE and BS in the contiguous allocation problem. The information contained in the observation is highlighted, while also indicating the other input information. The different actions are shown as outputs of the policies.

1 instead. This correction is only done for the reward calculation, not affecting the network performance calculations, which are done using N^u_{RX}.

4.2.2 Training Algorithm

Similarly to chapter 3, the reinforcement learning (RL) solution used in this chapter is based on the multi-agent deep deterministic policy gradient (MADDPG) algorithm [94]. Each agent has its own actor network that depends only on this agent's state in order to learn a decentralized policy ω_i parametrized by θ_i. Each agent also has a centralized critic network that receives the agent states and actions of all agents in order to learn a joint action value function $Q_i(x, a)$ parametrized by φ_i, where $x = (x_1, x_2, \ldots, x_n)$ contains all the agents' states and $a = (a_1, a_2, \ldots, a_n)$ contains the actions taken by all the agents.

Similarly to the original work [94], we use the Gumbel-softmax [96] trick to soft-approximate the discrete actions to continuous ones. The Gumbel-softmax reparameterization also works to balance exploration and exploitation. The exploration-exploitation trade-off is controlled by the temperature factor ζ.

Even though we use the MADDPG as the base training algorithm, we add some improvements to it. First, we make use of parameter sharing [14] for similar network nodes, in this case the UEs, meaning that the UEs have the same actor and critic networks parameters.

Secondly, we modify the neural networks architectures by using D2RL [98]. The main idea of D2RL is to use dense connections to pass the input of the network before every hidden layer, improving feature extraction and allowing deeper networks.

The actor and critic networks have the same architecture; a fully connected multilayer perceptron (MLP) with four hidden layers. Each hidden layer has 64 neurons and their activation function is the rectified linear unit (ReLU).

4.3 Results and discussion

4.3.1 Baseline Solutions

We compare the proposed solution with a request-grant proportional-fair (PF) protocol. In this protocol, the UE sends a scheduling request (SR) if its transmission buffer is not empty. The UCM sent indicates how many resources are needed by that UE, $\tilde{n}_{\mathrm{RB}}$, varying from 0 to M. At each time step, the BS receives zero or more SRs. The resource allocation is done by selecting the UE u with the largest

$$\frac{\eta_u}{G_u} \tag{4.13}$$

Table 4.1: Simulation Parameters

Parameter	Symbol	Value
Number of UEs	U	$[2, 4, 6, 8]$
Number of RBs	M	$[2, 4, 6, 8, 10, 12]$
Size of transmission buffer	B	∞
SDU arrival probability	p_{a}	$[0.15, 0.3, 0.6,]$
Transport block error rate	TBLER	10^{-1}
DCM vocabulary size	V_{DCM}	$[1, 2, 4, 8, 16, 32, 37, 48]$
UCM vocabulary size	V_{UCM}	$[1, 2, 4, 8, 9, 12, 16]$
Duration of episode (TTIs)	T	$[24, 36, 48, 96, 192]$
TTI duration (ms)	T_{TTI}	0.5
Numerology		1
Number of training episodes	N_{train}	100k
Number of evaluation episodes	N_{eval}	500
Number of test episodes	N_{test}	1000
Number of randomized repetitions	N_{rep}	8

and allocating either the number of requested resources $\tilde{n}_{\mathrm{RB}}$ or the remaining unallocated RBs and repeating this selection until either all RBs or all the requesters are allocated. The ratio in Eq. (4.13) is the achievable rate for one RE divided by the goodput. The allocation is informed to the UE through a DCM that contains a bit-field for scheduling grant (SG) and a field for the RB allocation, that maps to the initial RB and the total number of RBs.

4.3.2 Simulation Procedure and Parameters

For the performance evaluation, the transmission buffer of each UE starts empty and the traffic model is homogeneous across UEs, with SDU arrival probability p_{a} and the size of a SDU L_{SDU} both are constant for all UEs. The buffer size of each UE is assumed to have infinite capacity. The system is trained for a fixed number of episodes N_{train}. During training, we evaluate the policy on a fixed set of N_{eval} evaluation episodes with disabled exploration and disabled learning to assess the current

Table 4.2: Training Algorithm Parameters

Parameter	**Symbol**	**Value**
Memory length	k	3
Replay buffer size		10^5
Batch size		1024
Number of neurons per hidden layer		64
Interval between updating policies		96
Number hidden layers		4
Activation function of hidden neurons		ReLU
Interval between updating policies		96
Optimizer algorithm		Adam
Learning rate		10^{-3}
Discount factor		0.9
Policy regularizing factor		10^{-3}
Gumbel-softmax temperature factor	ζ	1
Target networks soft-update factor		10^{-3}

performance of the communication protocol. We then select the historical best protocol, which is the best performing one on the evaluation episodes during the whole training procedure and its performance is further assessed in N_{test} episodes with exploration and learning disabled, this is the final testing phase. This whole procedure represents a single training repetition.

We evaluate a total of N_{rep} repetitions, each with a different random seed. After training finishes, we have successfully trained a population of N_{rep} protocols, and we can select the best performing protocol on the training episodes, which is the historical best across all repetitions. This selection step can be seen as a *survival-of-the-fittest* approach because only one protocol of the population of N_{rep} is chosen going forward. A summary of the main simulation parameters is provided in table 4.1, while the parameters of the MARL training algorithms is listed in table 4.2.

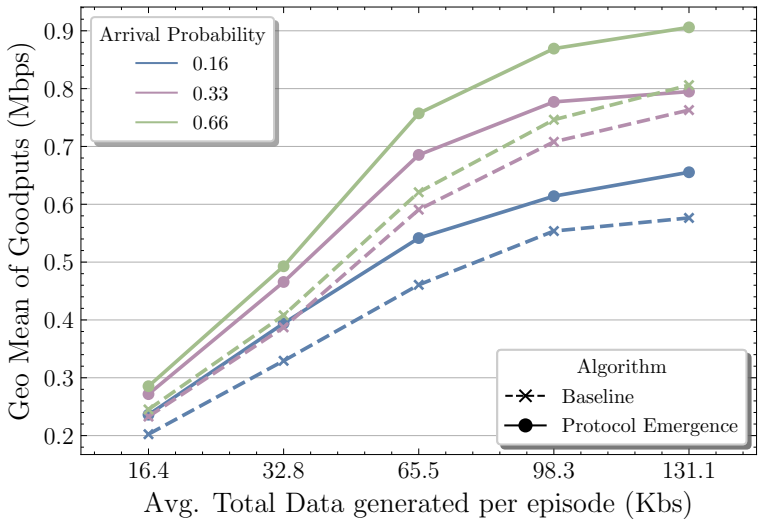


Figure 4.3: Comparison of final performance for different traffic. Parameters: Number of UEs $U = 4$; number of resources $M = 8$; episode length $T = 24$; UL and DL vocabulary sizes are $V_{\text{UCM}} = 9$ and $V_{\text{DCM}} = 37$.

4.3.3 Learning Capabilities

In this first set of results, we evaluate the capabilities of the proposed solution to learn protocols for different scenarios, by evaluating its performance across different traffic models, number of RBs, number of UEs and objective.

Traffic Scalability

Figure 4.3 shows the performance of the emerged protocols produced by training under different traffic models, with the x-axis representing the average total data arriving in an episode, $\lambda_{\text{total}} = \lambda_u U$. Due to how λ_u is calculated eq. (4.1), λ_{total} depends on the arrival probability and the size of each arrival. In low traffic densities, the performance gain from protocol emergence is small. However, the improvement over the compared baseline increases both with arrival probability, and with total arrival in bits.

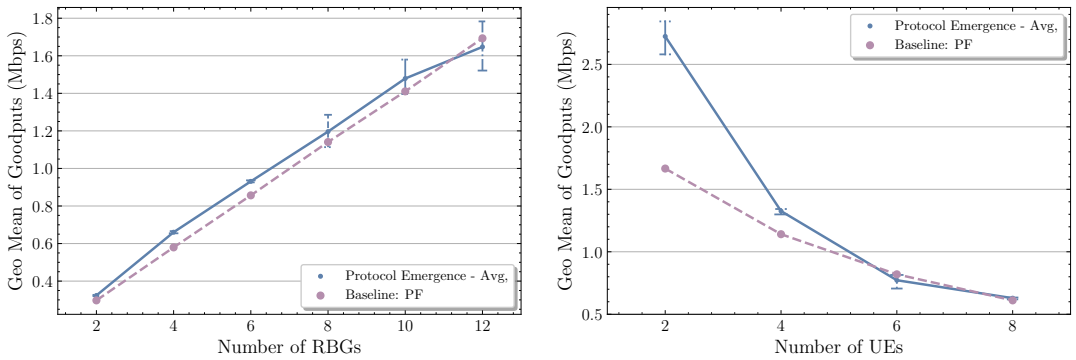
Scalability with RBs

Figure 4.4a illustrate the final performance of emerged protocols when varying the number of RBs while maintaining other values fixed. In the case of traffic, as increasing the number of RBs allows more data to be transmitted, the traffic density is fixed. In this case, the overall system traffic is assumed to have a density of 100%, that is, the arrival bit-rate on average is equal to the amount of bits needed to fill the resource grid. We observe the overall final performance difference between the proposed technique and the PF baseline remains constant when increasing the number of RBs.

Scalability with UEs

In fig. 4.4b, scalability with UEs is evaluated by comparing the performance of protocol emergence with the baseline. Since the overall traffic density in the system is constant, when increasing the number of UEs the arrival probability is constant and the size of arriving SDUs decrease. Increasing the number of UEs reduces the performance gain from using protocol emergence, as the grant-based PF protocol performs well for a high number of UEs.

It is important to highlight that scalability, both in terms of number of agents and size of the state and action spaces, is currently a challenge in MARL [107]. This is mainly due to the increased challenge in coordination and of the exploration. Moreover, this an important issue due to the architecture and formulation used in this work, as increasing the number of UEs directly affects the input and output size of the BS neural networks.



(a) Scalability study in terms of the number of RBs, assuming fixed number of UEs $U = 6$. Traffic is assumed to have a density of 100% with arrival probability $p_a = 0.33$.

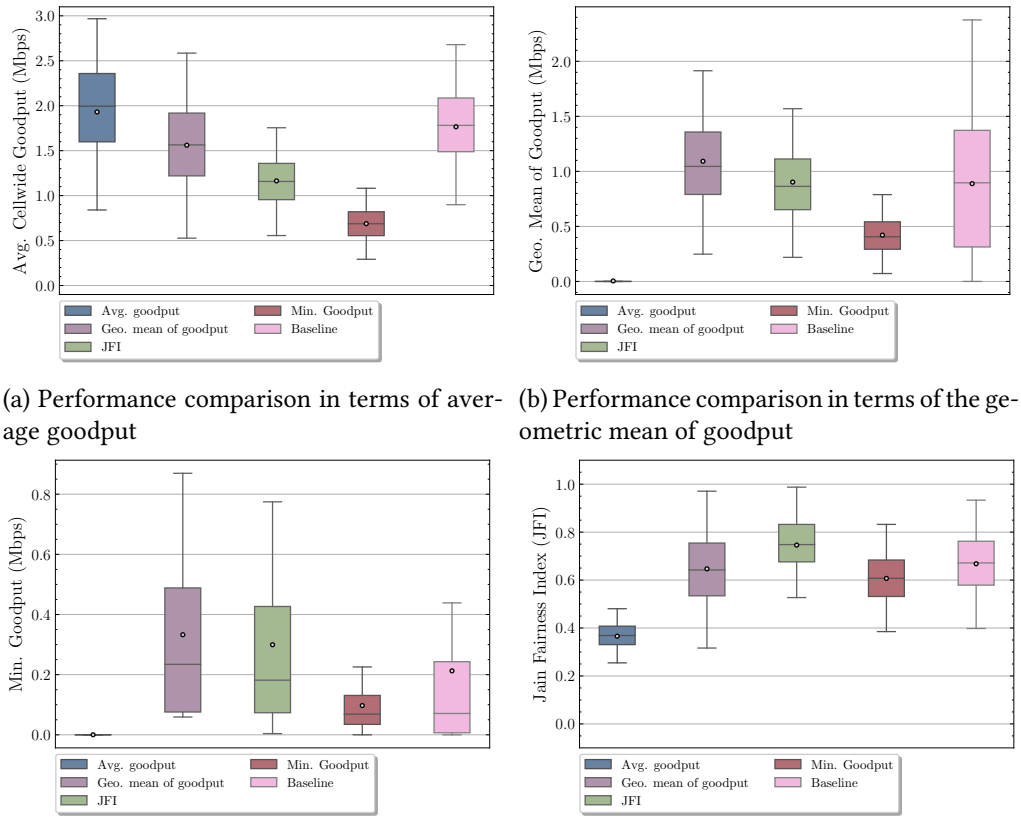
(b) Scalability study in terms of the number of UEs. Number of RBs is constant, $M = 8$ and traffic is assumed to have a density of 100% with arrival probability $p_a = 0.33$.

Figure 4.4: Scalability on terms of UEs and RBs.

Learning capabilities for different objectives

One of the advantages of protocol emergence is the ability to change the objective that the protocol should optimize for. In order to analyze the performance of protocols emerged with different objectives, the reward function is modified in order to produce four sets of protocols emerged for different objectives, geometric mean of goodput, average goodput, minimum goodput and JFI. The best protocol of each set with respect to its objective is selected, and they are compared on a set of episodes, together of the PF baseline. The JFI objective is not purely the JFI, but it also includes the average goodput, otherwise it would produce protocols with very low rates, but similar rates for the UEs. Figure 4.5 highlights the performance of these protocols.

From fig. 4.5, we observe that the PF baseline is a well-rounded protocol with high fairness, high overall goodput. However, protocol emergence allows the design of protocols that outperform it on terms of cellwide goodput, minimum service and fairness, both in terms of JFI and geometric mean of goodput. The maxmin of goodput seems a



(a) Performance comparison in terms of average goodput

(b) Performance comparison in terms of the geometric mean of goodput

(c) Performance comparison in terms of minimum goodput

(d) Performance comparison in terms of JFI

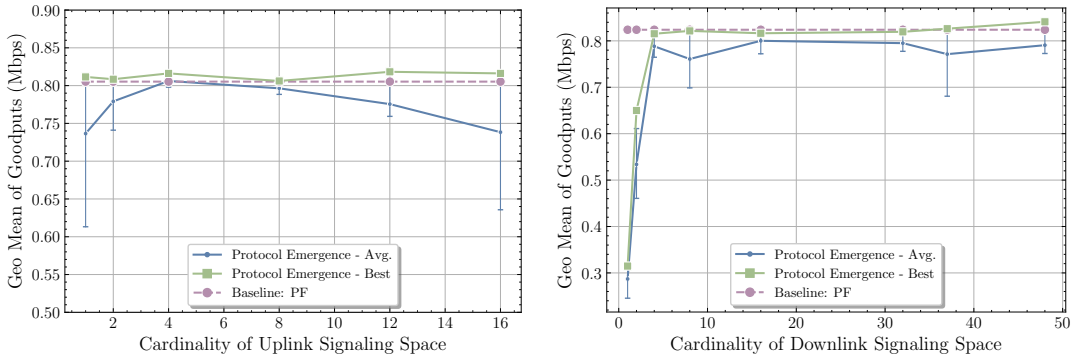
Figure 4.5: Comparison of the performance of protocols emerged with different objectives. Number of RBs is constant, $M = 8$ and traffic is assumed to have a density of 100% with arrival probability $p_{\text{a}} = 0.33$ and arrival size $L_{\text{SDU}} = 18\text{kbit}$. Number of UEs is $U = 6$ and episode duration $T = 96$.

difficult objective to maximize as the solution training with this reward function does not perform very well even for its objective, only outperforming the direct goodput objective in terms of minimum service. The protocol trained to maximize the average goodput performs really for its objective, but is a very unfair protocol, as expected. The fairness-based objectives produce the best protocols in terms of its selected metric, while performing better than the baseline even on terms of minimum service as shown in fig. 4.5c.

4.3.4 Effect of vocabulary sizes

For studying the effect of the UL and DL vocabulary sizes, a different scenario is analyzed, in which the channel of each UE varies over time according to a Jakes fading model. Also, the arrival rates of each UE is different, and these values change from episode to episode. The arrival rates are random, with their average being constant. A time-variant channel is important for the DL signaling, as the resource allocation should adapt to channel changes. The heterogeneous traffic is important for the UL signaling, so that the BS needs traffic information from the UE instead of learning the traffic pattern.

The results of such experiments are highlighted in fig. 4.6. In fig. 4.6a, we observe that it is possible to produce good performing protocols with any value of UL vocabulary size, but in general, learning is better for values between four and ten. Increasing the vocabulary size beyond that point makes learning more difficult, as it makes the



(a) UL vocabulary study with values in the set $V_{\text{UCM}} \in [1, 2, 4, 8, 12, 16]$. Baseline vocabulary is equal to 9, $M + 1$.

(b) DL vocabulary study with values in the set $V_{\text{UCM}} \in [1, 2, 4, 8, 16, 32, 37, 48]$. Baseline vocabulary is equal to 37, $\frac{(M+1)M}{2} + 1$.

Figure 4.6: UL and DL vocabulary studies. Number of RBs is constant, $M = 8$ and traffic is assumed to have a density of 100% with arrival size $L_{\text{SDU}} = 9$kbit. Number of UEs is $U = 6$ and episode duration $T = 96$.

signaling space too large without it being useful.

The main learning constraint is the DL vocabulary, as shown in fig. 4.6b. For low values, even the best protocol in the population is unable to perform well, only producing well performing protocols with $V_{\text{DCM}} \geq 4$. Differently from the UL case, increasing the DL vocabulary size does not hamper learning.

4.4 Chapter Summary

In this chapter, we have evaluated the capabilities of the proposed protocol emergence technique in a problem more complex than that of chapter 3, namely a contiguous resource allocation problem. We provided a RL formulation for tackling this problem, with a policy method selecting the starting RB and the total number of RBs allocated, while also learning the signaling necessary for coordination between BS and UEs.

The proposed solution is compared with a grant-based PF scheduler. We highlight that the PF scheduler maximizes the sum of the logarithmic average user rates [79], which is equivalent to maximizing the geometric mean of the average user goodput. Due to this, frequency-domain resource allocation for the PF scheduler is optimal for geometric mean maximization and any performance gain of the proposed solution must come from an improved signaling and better use of time-domain resources.

Overall, protocol emergence is robust to the scaling of the system. The main conclusion of the learning capabilities analysis are:

1. Traffic: The proposed technique is robust to different traffic, learning well in both low, medium and high traffic scenarios. However, the benefits of protocol emergence are more clear on medium-high load scenarios, as shown in fig. 4.3.

2. RBs: Again, the proposed technique is robust to scalability of RBs. However, differently from the traffic analysis, there is no scenario that is shown to perform better with the difference in performance being similar as shown in fig. 4.4a.

3. UEs: The challenge in terms of scalability is in terms of UEs, as highlighted in fig. 4.4b. Scaling with traffic and RBs slightly increases the required training time, however the training time increases significantly with the number of UEs, needing a larger number of training episodes to converge. Besides this, the benefit in terms of performance diminishes, as the grant-based baseline is optimized for a large number of UEs.

4. Objective function: Protocol emergence has shown the capabilities of producing well performing protocols for different objectives, as highlighted in fig. 4.5. This is one of its biggest strengths, as it is possible to search in the protocol space for protocols optimizing even joint objectives, for example, a protocol maximizing goodput constrained by a minimum level of service.

The vocabulary size study was conducted with experiments designed in a way that coordination between the nodes is necessary. This analysis showed that, although it is possible to produce well performing individual protocols with low signaling, providing a reasonable level of granularity to the UL and DL signaling helps produce a better set instead of a few good ones.

Chapter 5

Protocol Emergence under Signaling Constraints

Contents

In this chapter, protocol emergence is used to minimize the volume of control-plane traffic in a non-contiguous resource allocation scenario. As such, instead of persistent signaling as in chapter 3 and chapter 4, this chapter introduces an intermittent signaling scenario where the nodes decide if they will send a control message or not.

As a consequence of the intermittent signaling, it is possible to reduce the amount of data that goes through the control channels.

5.1 System Model

Consider a single cell with a base station (BS) serving U user equipments (UEs) in a uplink slotted shared channel (USC), where each UE needs to deliver data to the BS. For the frequency domain, the available bandwidth is divided into M resource block groups (RBGs). A RBG is a set of consecutive resource blocks (RBs). Time domain is assumed slotted with a fixed time-step duration of T_{TTI}. We assume an episode of duration T T_{TTI}s.

The BS and UEs constitute the nodes of the system who act as independent agents. The network nodes may exchange information, using messages through the control channels. The decision of sending a control message is taken by the network nodes. For example, in a given time slot, the BS may decide to send a downlink control message (DCM) to UE 1, but not to UE 2, while only UE 2 sends a uplink control message (UCM).

Each UE has a transmission buffer of capacity B in bits initially empty. Data arrives in service data units (SDUs) of fixed size L_{SDU}. The SDU arrival is modeled as a Bernoulli process with probability of arrival p_{a}. So, at each time step, a new SDU of size L_{SDU} is added to the buffer with probability p_{a}, until a maximum number T steps is achieved. The average number of information bits arriving at each UE's buffer in any given episode of duration T is then:

$$\lambda_u = p_{\text{a}} T L_{\text{SDU}} \tag{5.1}$$

We assume that a automatic repeat request (ARQ) process is used by the network nodes

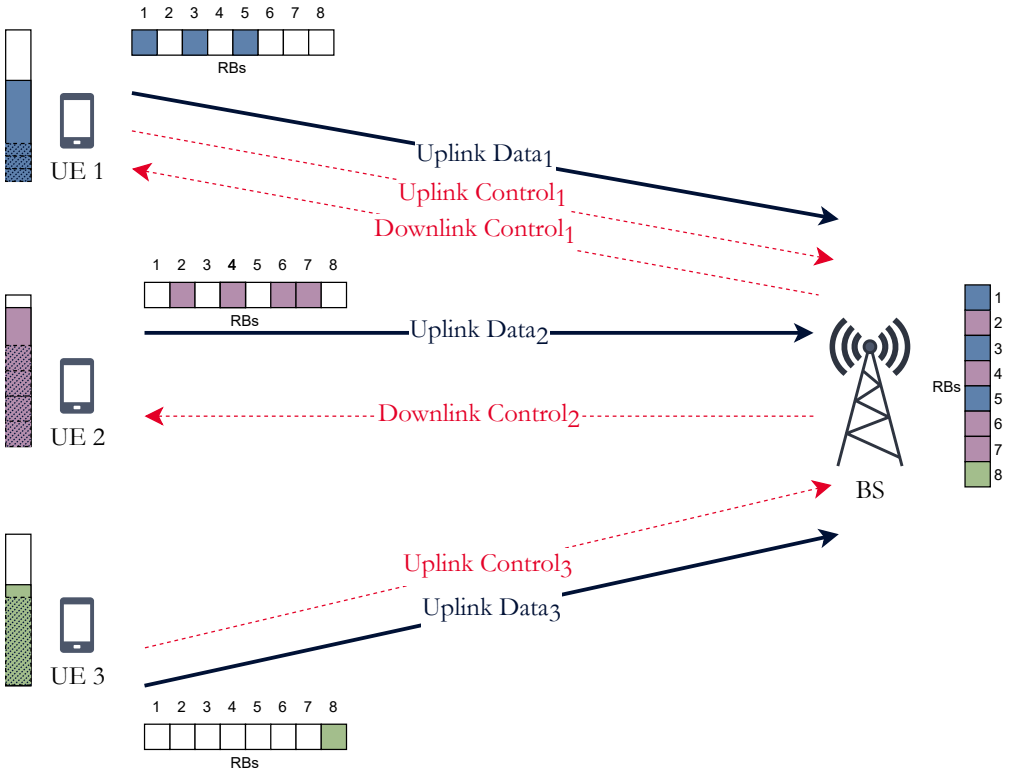


Figure 5.1: System model scheme with non-contiguous resources allocation. The buffer and decisions of each UE are highlighted besides it. The wireless channel is shown besides the BS, highlighting the frequency resource axis on that time slot, indicating the RBGs used by each UE. Sending a control message is also a decision taken by the UEs and BS, this is highlighted in how for UEs 2 and 3 the uplink (UL) and downlink (DL) control messages are not sent, respectively.

to handle the buffer management, as such, upon successful reception of a transport block (TB) by the BS, the UE remove those bits from its buffer.

The system is channel-aware, with each UE having a signal-to-noise ratio (SNR) given by:

$$\mathrm{SNR}_u = \frac{|\tilde{h}_u \sqrt{\rho_u}|^2}{\sigma^2} p_t \,, \tag{5.2}$$

where $\tilde{h}_u \sqrt{\rho_u}$ is the effective channel of UE u, with $\tilde{h}$ denoting the complex gain and ρ denoting the path loss. The path loss and gains are assumed constant for the episode

duration, due to its short duration. An uncorrelated flat channel with Rayleigh fading is assumed for the complex gain calculations.

Given a SNR_u, a modulation and coding scheme (MCS) is chosen by a look-up table link adaptation method. The method chooses the highest MCS with an average error rate lower than the target transport block error rate (TBLER) for that SNR. The spectral efficiency, η, is the maximum number of information bits in a resource element (RE), given by the product of the code rate ν_r and the modulation order ϱ_m:

$$\eta = \varrho_m \nu_r \,. \tag{5.3}$$

With a selected MCS and given the number of RBs for transmission, n_{RB}, which depends on the number of RBGs, it is possible to calculate the maximum number of information bits that can be transmitted, which is the maximum transport block size (TBS), as:

$$\mathrm{TBS}_{\max} = n_{\mathrm{RB}} N_{\mathrm{sc}}^{\mathrm{RB}} N_{\mathrm{symb}}^{\mathrm{sh}} \varrho_m \nu_r \,, \tag{5.4}$$

where $N_{\mathrm{sc}}^{\mathrm{RB}}$ is the number of subcarriers in a RB and $N_{\mathrm{symb}}^{\mathrm{sh}}$ is the number of symbols in a slot. The number of RBs in a RBG depends on the bandwidth part size. Transmission errors affect the whole TB.

The channel for the uplink data transmission is modeled as a packet erasure channel, where a TB is incorrectly received with a probability referred to as the TBLER. As non-contiguous resource allocation is assumed, in which the UEs decide which RBGs will carry its transmission. Collisions may happen if more than one UE transmit over the same resource, in this case the TBs are not received correctly. The DCMs and UCMs are transmitted over the DL and UL control channels respectively, which is not

error-free, having an error probability given by control block error rate (CBLER). In this scenario, the DCMs are usually used to indicate the resource allocation of choice and the UCMs are usually used to indicate the amount of traffic or priority of a UE.

We assume that the sets of possible DL and UL control messages have cardinality V_{DCM} and V_{UCM}, respectively. For a DL (resp. UL) control vocabulary of size V_{DCM}, the bitlength Υ_{DL} is equal to $\lceil(\log_2 V_{\text{DCM}})\rceil$, where $\lceil\cdot\rceil$ represents the ceiling function. One of the messages in a vocabulary is used to indicate if a control message was not received, although not exclusively, as this same message is also available to be sent directly as a control message too.

At each time step t of duration T_{TTI}, the BS may send one control message to each UE and each UE may send one control message to the BS while being able to send protocol data units (PDUs) through the uplink shared channel (UL-SCH). The PDU contains a TB and the UE has to select the RBGs for transmission, which is done by directly deciding if it will use a RBG or not. Figure 5.1 illustrates the system model, highlighting the decisions taken by each UE and its effect on the system. The BS and UEs have an extra action that decides if the control message will be transmitted or not, per time-step.

As the nodes do not need to send a control message at every step, this formulation introduces intermittent signalling. In chapters 3 and 4, the nodes had to send a control message at every step, which means that the control signalling was persistent in both scenarios.

5.1.1 Performance Metrics

The performance metrics evaluated in this chapter are related to the received bit-rate, or goodput. The goodput of an UE u, G_u (in Mbit s^{-1}), is defined as the number of

information bits received by the BS from UE u per unit of time:

$$G_u = \frac{N_{\mathrm{RX}}}{T\, T_{\mathrm{TTI}}} \tag{5.5}$$

We are mainly interested in the geometric mean of the goodput, as it provides a notion of fairness:

$$F = \sqrt[U]{\prod_{u=1}^{U} G_u}\ . \tag{5.6}$$

Another metric of interest is the collision-rate Γ defined as the total number of resources in which a collision happened divided by the total number of resources in the grid:

$$\Gamma = \frac{N_{\mathrm{c}}}{TM}. \tag{5.7}$$

where N_{c} represents the total number of resources in which at least two SDUs collided.

Due to the intermittent signaling, the control channel throughput, R_{cp}, is the main performance metric regarding the signaling overhead. It is defined as the total number of bits transmitted in the control channel by all nodes per unit of time. It is calculated (in kbit s^{-1}) as:

$$R_{\mathrm{cp}} = \frac{N_{\mathrm{DCM}}\Upsilon_{\mathrm{DL}} + N_{\mathrm{UCM}}\Upsilon_{\mathrm{UL}}}{T\, T_{\mathrm{TTI}}}\ , \tag{5.8}$$

where N_{DCM} and N_{UCM} represent the total number of transmitted DL and UL control messages.

5.2 MARL Problem Formulation

5.2.1 MARL Formulation

The problem defined above is formulated, similarly to chapters 3 and 4, as a multi-agent reinforcement learning (MARL) cooperative task. The UE agents need to learn when to send data and in which RBGs. The medium access control (MAC) agents also learn how and when to communicate with each other through the control channel, sending the relevant information in order to cooperate and avoid collisions, which is used as input by the network nodes to decide how to act. The how to communicate is decided as a communication action indicating which control message to send. The when to communicate is decided by an extra action that indicates if the control message will be transmitted or not, this action is called communication control action. The BS also takes into account the spectral efficiencies of the MCS of each UE, the normalized amount of data received from each UE and the state of each RBG, i.e idle, busy or collision-free reception. Aside from the control messages, the UEs takes into consideration their buffer status in order to inform its decision.

Similarly to the previous chapters, this problem is modeled as a decentralized partially observable Markov decision process (Dec-POMDP) [92], augmented with communication. The action space of each agent is subdivided into one environment action space and a communication action space. The communication action represents the message sent by an agent, and it does not affect the environment directly, but it may be passed to other agents. Unlike chapters 3 and 4, the transmission of the control action depends on the extra communication control action. We also assume that this is a cooperative problem, as such, all agents share the same reward. In this chapter, the

agent internal state x_i may comprise not only the agent's current observation, but also previous observations, actions and received messages. We assume the episode ends when a maximum number of steps T is reached.

We use the following notations:

- o_t: Observation received by the agent at time step t.
- n_t^{u}: The UCM sent from the u^{th} UE at time step t.
- m_t^{u}: The DCM sent to the u^{th} UE at time step t.
- a_t^{u}: Environment action of the u^{th} UE at time step t.
- a_t^{b}: Environment action of the BS at time step t.
- x_t: Agent state at time step t.

Besides the reward and the action modeling, the problem formulation is similar to chapter 4.

Observations

The UE observation o_t^{u} is an integer representing integer division of the number of bits in the buffer of the UE u and its current spectral efficiency at that time t.

Similarly to chapter 4, the observation o_t^{b} received by the BS is a combination of three vectors, the channel information $\boldsymbol{c}$, the received data information $\boldsymbol{g}$ and the spectral efficiency information $\boldsymbol{\eta}$:

$$o_t^{\mathrm{b}} = [\boldsymbol{c}\ \boldsymbol{g}\ \boldsymbol{\eta}]\ . \tag{5.9}$$

The vector $\boldsymbol{c}$ contains the channel information relative to each RBG, c_i, which is a

discrete variable with $U + 2$ possible states:

$$c_i = \begin{cases} 0, \text{ if the RBG } i \text{ is idle} \\ u, \text{ if successfully received data from UE } u \text{ on RBG } i \\ U + 1, \text{ non-decodable energy in the RBG } i, \end{cases} \quad , \forall i \in \{1, \ldots, M\} \tag{5.10}$$

where $u \in \{1, \ldots, U\}$. The received data information is the normalized amount of bits received from each UE

$$g_u = \frac{N_{\text{RX}}^u}{\max\limits_i N_{\text{RX}}^i} \tag{5.11}$$

and $\boldsymbol{\eta}$ contains the current spectral efficiency of the MCS of each UE, η_u.

Actions

The environment action of the UE comprises the data transmission action and the UCM control action. The environment action of the BS contains just the DCMs control actions.

For the UE action regarding the transmission of data, the UE decision involves the selection the RBGs used for data transmission. We model the action as vector containing the main action, a_{m} and an action parameter, a_{bmp}, similarly to the action design in [106]. The main action $a_{\text{m}} \in \mathcal{A}_e = \{0, 1\}$ is interpreted as follows:

$$a_{\text{m}} = \begin{cases} 0\text{: do nothing} \\ 1\text{: transmit data.} \end{cases} \tag{5.12}$$

The action parameter is a bitmap indicating if a RBG is used for transmission or not.

If bit m of the bitmap has value 1, the UE will use RBG m for transmission.

Both the UEs and BS have an action, a_{c}, determining the transmission of the control message, called DCM or UCM control actions. For the UE, this action is interpreted as follows:

$$a_{\mathrm{c}} = \begin{cases} 0\text{: do not transmit the UCM} \\ 1\text{: transmit the UCM.} \end{cases} \tag{5.13}$$

For the BS, this action is a vector controlling the transmission of each of the U DCMs. The interpretation of each element of this action vector is similar to the UE case.

Communication actions

Coinciding with the chapter 4, the DCM and UCM messages, m and n, are communication actions that the agents select while also being information available to the other agent's state as received message. The BS communication action is the vector comprising the downlink messages sent to all UEs $\boldsymbol{m}_t \in \mathcal{D}^U$, with $\mathcal{D} = \{0, \ldots, V_{\mathrm{DCM}} - 1\}$ and $\mathcal{D}^n$ denotes the n-ary Cartesian power of set $\mathcal{D}$. The communication action of a UE u is a single UCM $n_t^u \in \mathcal{U} = \{0, \ldots, V_{\mathrm{UCM}} - 1\}$ and the vector of all sent UCMs is $\boldsymbol{n}_t$.

Input states

The agent state at time step t is a tuple comprising the most recent k observations, actions and received messages.

- UE u: $x_t^{\mathrm{u}} = \left[\xi_t^{\mathrm{u}} \ldots, \xi_{t-k}^{\mathrm{u}}\right]$, where $\xi_t^{\mathrm{u}} = \left[o_t^{\mathrm{u}}, a_t^{\mathrm{u}}, n_t^{\mathrm{u}}, m_t^{\mathrm{u}}\right]$.
- BS: $x_t^{\mathrm{b}} = (\xi_t^{\mathrm{b}} \ldots, \xi_{t-k}^{\mathrm{b}})$, where $\xi_t^{\mathrm{b}} = (o_t^{\mathrm{b}}, \boldsymbol{n}_t, \boldsymbol{m}_t)$, with $\boldsymbol{n}$ and $\boldsymbol{m}$ containing the

messages to and from all UEs.

This formulation is illustrated in fig. 5.2 for the BS and UE policies, highlighting the observations and actions.

Given the above definitions, the size of the input and output layers for the actors are:

- UE: Input size is $k(1 + V_{\text{DCM}} + V_{\text{UCM}} + 2 + M + 2)$, as the action has size $2 + M + 2$, corresponding to the main action, the action parameter and the communication control action. The observation is a single scalar. Output size is $2+2M+2+V_{\text{UCM}}$, corresponding to the environment actions and the UCM.
- BS: Input size is $k((2+U)M+2U+UV_{\text{UCM}}+UV_{\text{DCM}}+2U)$, as the observation has size $(2+U)M+2U$. Output size is $UV_{\text{DCM}}+2U$, corresponding to communication actions to all UEs and the communication control actions.

The critic input size is the sum of all nodes actors inputs and outputs.

Reward

The reward is divided into two components, the performance objective, r_g, and the signaling penalization, r_s.

$$r_t = r_g - \kappa r_s \,, \tag{5.14}$$

where κ denotes the signaling cost. The performance objective is the geometric mean of the goodput:

$$r_g = \sqrt[U]{\prod_{u=1}^{U} G_u} \,. \tag{5.15}$$

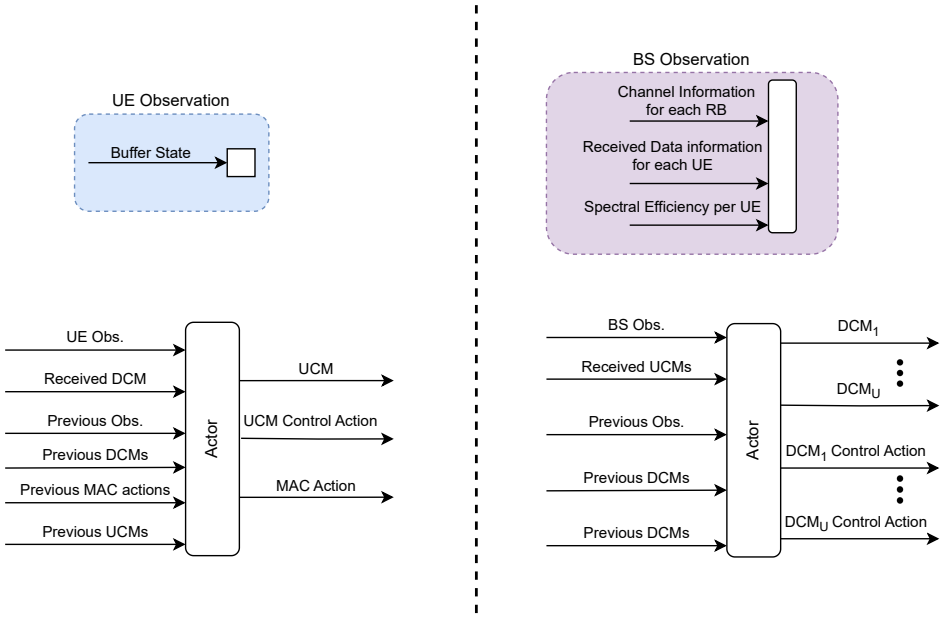


Figure 5.2: Scheme of the inputs and outputs of the policy network of the UE and BS in the non-contiguous allocation problem. The information contained in the observation is highlighted, while also indicating the other input information. The different actions are shown as outputs of the policies.

Similarly to chapter 4, the case in which $G_u = 0$ is handled by assuming that at least one bit was received from each UE, and this correction is used only for the reward calculation, not for the performance evaluations, as such, the calculation of r_g is different from F.

The cost is fixed per message, and it does not depend on whether it is a DCM or UCM, as such, each message costs the same, even if they have different bitlengths. Also, differently from the performance objective, the signaling penalization is instantaneous, only counting the penalization regarding the current time step. The signaling

penalization is calculated as:

$$r_s = \frac{N^t_{\text{UCM}} + N^t_{\text{DCM}}}{2U}, \tag{5.16}$$

where N^t_{UCM} and N^t_{DCM} represent the number of uplink and downlink control messages transmitted at time-step t. As $r_s \in [0, 1]$, r_g is also normalized to be under the same interval, with κ controlling the effect of the signaling cost.

5.2.2 Training Algorithm

Similarly to chapters 3 and 4, the reinforcement learning (RL) solution used in this chapter is based on the multi-agent deep deterministic policy gradient (MADDPG) algorithm [94], improved with deep dense architecture for reinforcement learning (D2RL), as explained in section 2.2.3. Each agent has its own actor network that depends only on this agent's state in order to learn a decentralized policy ω_i parametrized by θ_i. Each agent also has a centralized critic network that receives the agent states and actions of all agents in order to learn a joint action value function $Q_i(x, a)$ parametrized by φ_i, where $x = (x_1, x_2, \ldots, x_n)$ contains all the agents' states and $a = (a_1, a_2, \ldots, a_n)$ contains the actions taken by all agents.

Similarly to the original work [94], we use the Gumbel-softmax [96] trick to soft-approximate the discrete actions to continuous ones. Additionally, we add some techniques to the MADDPG that improve its learning capabilities. First, we make use of parameter sharing [14] for similar network nodes, in this case the UEs, meaning that the UEs have the same actor and critic networks parameters. We also make use of D2RL [98] as the architecture of choice, using dense connections to pass the input of the network before every hidden layer, improving feature extraction and allowing

Table 5.1: Simulation Parameters

Parameter	Symbol	Value
Number of UEs	U	4
Number of RBGs	M	4
Number of RBs per RBG		1
Size of transmission buffer	B	∞
SDU arrival probability	p_{a}	0.3
Transport block error rate	TBLER	10^{-1}
Control block error rate	CBLER	$\left[0, 10^{-1}\right]$
DCM vocabulary size	V_{DCM}	37
UCM vocabulary size	V_{UCM}	9
Duration of episode (TTIs)	T	24
TTI duration (ms)	T_{TTI}	0.5
Numerology		1
Number of training episodes	N_{train}	100k
Number of evaluation episodes	N_{eval}	500
Number of test episodes	N_{test}	1000
Number of randomized repetitions	N_{rep}	8

deeper networks. The actor and critic networks have the same architecture; a multi-layer perceptron (MLP) with four hidden layers with dense connections. Each hidden layer has 64 neurons and their activation function is the rectified linear unit (ReLU).

5.3 Results and discussion

5.3.1 Baseline Solutions

We compare the proposed solution with a request-grant proportional-fair (PF) protocol, in which the UE sends a scheduling request (SR) if its transmission buffer is not empty. The UCM sent indicates how many RBGs are needed by that UE, $\tilde{n}_{\mathrm{RB}}$, varying from 0 to M. At each time step, the BS receives zero or more SRs. The resource

allocation is done by selecting the UE u with the largest

$$\frac{\eta_u}{G_u} \tag{5.17}$$

and allocating either the number of requested resources $\tilde{n}_{\mathrm{RB}}$ or the remaining unallocated RBGs and repeating this selection until either all RBGs or all the requesters are allocated. The ratio in Eq. (5.17) is the achievable rate for one RE divided by the goodput. The allocation is informed to the UE through a DCM that contains a bit-field for scheduling grant (SG) and a bitmap for the RBG allocation.

5.3.2 Simulation Procedure and Parameters

For the performance evaluation, the transmission buffer of each UE starts empty, while the traffic model is homogeneous across UEs, with the SDU arrival probability p_{a} and the size of a SDU L_{SDU} both being constant for all UEs. For simplicity and ease of comparison with chapter 4, we assume that each RBG contains one RB. The buffer size of each UE is assumed to have infinite capacity. The system is trained for a fixed number of episodes N_{train}. During training, we evaluate the policy on a fixed set of N_{eval} evaluation episodes with disabled exploration and disabled learning to assess the current performance of the communication protocol. We then select the historical best protocol, which is the best performing one on the evaluation episodes during the whole training procedure and its performance is further assessed in N_{test} episodes with exploration and learning disabled, this is the final testing phase. This whole procedure represents a single training repetition.

We evaluate a total of N_{rep} repetitions, each with a different random seed. After training finishes, we have successfully trained a population of N_{rep} protocols, and we

Table 5.2: Training Algorithm Parameters

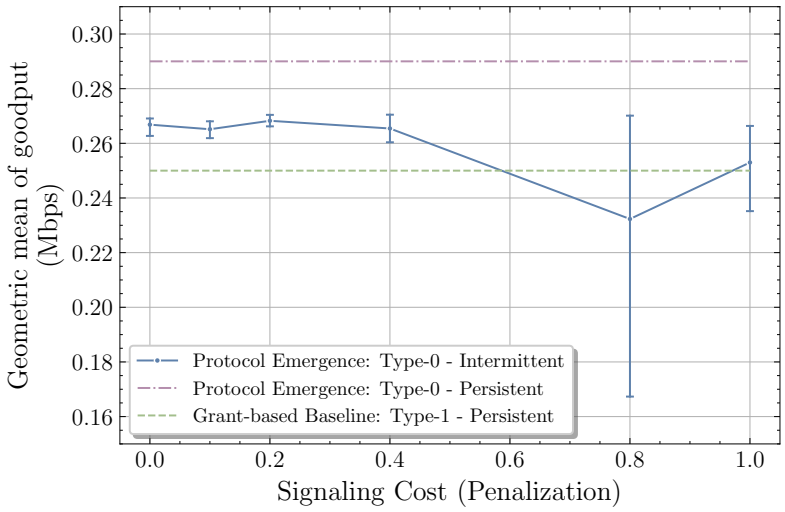
Parameter	**Symbol**	**Value**
Memory length	k	3
Replay buffer size		10^5
Batch size		1024
Number of neurons per hidden layer		64
Interval between updating policies		96
Number hidden layers		4
Activation function of hidden neurons		ReLU
Interval between updating policies		96
Optimizer algorithm		Adam
Learning rate		10^{-3}
Discount factor		0.9
Policy regularizing factor		10^{-3}
Gumbel-softmax temperature factor	ζ	1
Target networks soft-update factor		10^{-3}

can select the best performing protocol on the training episodes, which is the historical best across all repetitions. This selection step can be seen as a *survival-of-the-fittest* approach because only one protocol of the population of N_{rep} is chosen going forward. A summary of the main simulation parameters is provided in table 5.1, while the parameters of the MARL training algorithms is listed in table 5.2.

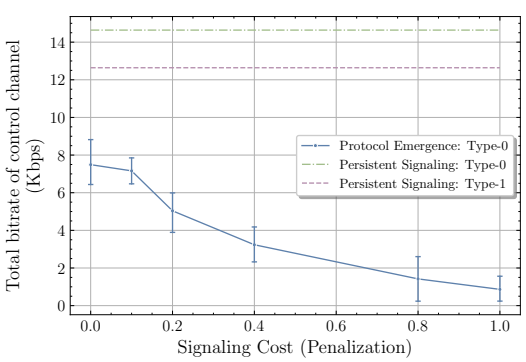
5.3.3 Results

In the following set of experiments, we train a population of protocols for different values of signaling cost. In fig. 5.3, the effect of the signaling cost is highlighted in contrast to different metrics. The main conclusions we can draw from this set of experiments are:

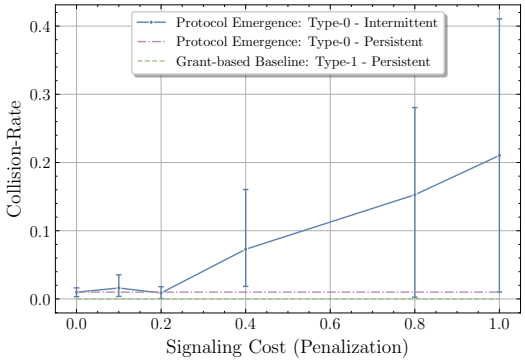
- Protocol emergence with intermittent signaling can produce protocols with good performance in terms of fairness as illustrated in fig. 5.3a, however, increasing



(a) Comparison in terms of control bit-rate.



(b) Comparison in terms of collision-rate.



(c) Comparison in terms of collision-rate.

Figure 5.3: Performance evaluation for different values of the signaling cost. Parameters: Number of UEs $U = 4$; number of resources $M = 4$; episode length $T = 24$; UL and DL vocabulary sizes are $V_{\text{UCM}} = 5$ and $V_{\text{DCM}} = 16$.

the signaling cost leads to a decrease in performance and a higher variance.

- Protocol emergence with persistent signaling outperforms both the protocol emergence with intermittent signaling in terms of geometric mean of the goodput.
- Increasing the signaling cost leads to reducing the signaling used by the emerged protocol, as illustrated in fig. 5.3b, moreover, even with zero cost, the emerged protocol still uses intermittent signaling instead of a persistent signaling.

- The control channel bit-rate decreases with the collision rate, as shown in fig. 5.3c. This is natural, as reducing the amount of control exchanges leads to a lower coordination, which can be perceived by the increase in collisions.

An interesting observation from fig. 5.3b is that even at zero cost, the emerged protocol does not use persistent signalling. This happens because there is no incentive to do so, as it achieves a low collision-rate and a good performance, even if below the persistent-signalling approach, thus, choosing only to communicate when needed.

5.4 Chapter Summary

In this chapter, we have proposed a RL formulation for protocol emergence where the nodes can decide if they will transmit a control message, which can reduce the control channel bit-rate. The scenario studied is a non-continguous resource allocation.

The results indicate the capabilities of the MARL framework to produce protocols with reduced signaling, which can be controlled by the reward function with a signaling cost weight. However, reducing the signaling bit-rate leads to a lower level of coordination, reducing the overall network performance and increasing the collision-rates. The proposed idea together with a vocabulary size variation can lead to the design of protocols with very low signaling overhead, which might be of interest in some scenarios with reduced bandwidth, reducing the amount of resources used for control signaling and increasing the resources used for data transmission.

Chapter 6

Conclusions

Contents

6.1 Thesis Summary

The main goals of this thesis was to provide the foundations to protocol emergence, highlighting the challenges involved with it, study its learning capabilities and its ability to control the amount of signaling used. In addition, it also aimed at providing an overview of multiple-access and the procedures involved with it in fifth generation (5G) new radio (NR).

Regarding the proposed studies and their performance evaluations, on one hand, chapter 3 provided a framework for protocol emergence based on multi-agent reinforcement learning (MARL), with results highlighting its superior performance when compared with other protocol design techniques. Besides, it also described methods for performance evaluation, interpretation, cross-node coordination and signaling over-

head quantification, with the results reinforcing the utility of such framework. On the other hand, chapter 4 applied the protocol emergence idea to a more challenging problem of contiguous resource allocation, showing its robust learning capabilities for different traffic densities and for different objective functions. Moreover, it showed its scaling capabilities in terms of user equipments (UEs) and resource blocks (RBs), while also showing that providing a reasonable degree of signaling granularity helps producing a better set of protocols. Lastly, chapter 5 proposed a intermittent signaling formulation, in which the nodes decide if a control message will be sent, instead of always sending a control message. The results showed that the proposed MARL framework can produce protocols with reduced signaling and the signaling throughput can be controlled by a signaling cost weight added to the reward function, although reducing the signaling throughput can lead to lower coordination and worst performance.

6.2 Future Directions

In terms of future perspectives, the idea of jointly learning the control signaling and the network control optimization can be extended to some different problems in wireless communications. Such problems include: adaptive modulation and coding (AMC) [108]–[110], uplink power control [111], [112], link adaptation [113], uplink transmission precoder selection [114] and beam management [115]–[117]. These problems and the signaling used to support it are highlighted in table 6.1.

Besides applying to protocol emergence to new problems, scheduling is still an interesting avenue for protocol emergence, where it can be applied to scenarios such as non-orthogonal multiple access [56], [118], ultra-reliable and low-latency communications (uRLLC) [119], vehicle-to-everything (V2X) [120] and extended reality (XR) [121], [122]. Moreover, protocol emergence can be extended to goal-oriented communica-

Table 6.1: Possible problems and the signaling involved.

Wireless Problem	**Signaling Messages**
Uplink power control	Power headroom report (PHR) and transmit power control (TPC) command
Adaptive modulation and coding	Channel quality indicator (CQI) and modulation and coding scheme (MCS) index
Downlink link adaptation	Channel quality indicator (CQI), rank indicator (RI), precoding matrix indicator (PMI) and modulation and coding scheme (MCS)
Uplink precoder selection	Transmit precoding matrix indicator (TPMI)
Beam management	Sounding reference signal (SRS) and channel state information (CSI) requests

tions [123], [124] providing the means to produce goal-oriented protocols [125].

Another axis for perspectives relates to the fundamentals for protocol emergence and on the study of protocols. Possible directions include:

1. Studying coordination: Propose additional metrics for evaluation of the control-plane or use such metrics during training to achieve better coordination, as proposed by [100].
2. Profiling the protocols: Compare the effect of different reward functions and different system models on the characteristics of the protocol.
3. Protocol interpretation: Focusing on better tools to interpret the emerged communication, such as explainable artificial intelligence (XAI) [126] based on Shapley values [127] or causal models [128].

Scalability is currently a challenge in MARL [107], [129], both in terms of number of agents and size of the state and action spaces. Naturally, this is also a challenge for protocol emergence, which calls for learning algorithms able to handle this challenge or engineering techniques employed to deal with scalability.

Moreover, the system models addressed in this thesis are simplistic, as it assumes a dedicated control channel for all UEs. For a low number of UEs, that assumption is possible to be made, as the amount of control channel information exchanged is much smaller than the data exchanged, thus, very few resources in the system need to be separated for control. However, a more realistic scenario can be interesting, where the total number resources is shared between control and data and this approach can be compared with a separation approach, which would ensure dedicated resources for the control signalling.

On terms of reinforcement learning (RL) methods, other architectures can be proposed to deal better with the intricacies of wireless protocols. Of particular interest, we highlight attention mechanisms [130], decision transformers [131] and model-based RL [132], [133].

6.3 Main Contributions

The present thesis makes advancements in wireless communications and protocol design through the exploration of protocol emergence. It provides the fundamentals for protocol emergence through a framework for producing and studying such protocols. The ability to autonomously optimize protocols is pertinent to the development of self-organizing and self-optimizing networks, while also being crucial for adapting to dynamic and complex wireless environments. Moreover, insights from this research can contribute to future wireless communication standards, as the end goal of protocol emergence research would be the generation of a fully learned communication system.

Additionally, it studies the learning capabilities, limitations, and scalability challenges for the proposed protocol emergence, while also highlighting its performance benefits when applied to resource allocation. The improved resource management

and signaling utilization is fundamental to enhancing the performance and capacity of wireless systems. Furthermore, the adaptive nature of protocol emergence can be key to handling diverse traffic scenarios and different objective functions, making it a versatile solution to be used across a wide range of scenarios.

It also makes contribution to control signaling optimization, as protocol emergence provides the means to produce a optimized control-plane, reducing signaling overhead while maintaining an effective performance. Reducing signaling overhead is important both to conserve energy and to allow more resources to be used for data transmission.

In summary, this thesis contributes to advancing the autonomy, efficiency, and adaptability of wireless protocols. It provides understanding of protocol emergence by autonomously learning effective control signaling schemes, addressing protocol design challenges, and paving the way for future research in emergent protocols. As such, this research may be of interest for both telecommunications and artificial intelligence communities.

References

[1] J. Erfanian, D. Lister, Q. Zhao, G. Wikström, and Y. Chen, "6G vision & analysis of potential use cases," *IEEE Communications Magazine*, vol. 61, no. 4, pp. 12–14, 2023.

[2] M. Z. Asghar, S. A. Memon, and J. Hämäläinen, "Evolution of wireless communication to 6G: Potential applications and research directions," *Sustainability*, vol. 14, no. 10, 2022, ISSN: 2071-1050. DOI: 10.3390/su14106356. [Online]. Available: https://www.mdpi.com/2071-1050/14/10/6356.

[3] A. F. M. Shahen Shah, "A Survey From 1G to 5G Including the Advent of 6G: Architectures, Multiple Access Techniques, and Emerging Technologies," in *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, 2022, pp. 1117–1123. DOI: 10.1109/CCWC54503.2022.9720781.

[4] J. Yang, N. Tin, and A. Khandani, "Adaptive modulation and coding in 3G wireless systems," in *Proceedings IEEE 56th Vehicular Technology Conference*, vol. 1, 2002, 544–548 vol.1. DOI: 10.1109/VETECF.2002.1040403.

[5] F. J. Mullany, "High-speed downlink access in 3G systems: A portent for the evolution of 4g systems?" *Wireless Personal Communications*, vol. 17, no. 2-3, pp. 225–235, 2001.

[6] ETSI, "Overview of 3GPP Release 5," European Telecommunications Standards Institute (ETSI), Tech. Rep., 2003. [Online]. Available: https://www.3gpp.org/ftp/tsg_t/TSG_T/TSGT_21/Docs/PDF/TP-030215.pdf.

[7] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Layer Procedures," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 36.213, Mar. 2019, Version 15.5.0. [Online]. Available: http://www.3gpp.org/DynaReport/36213.htm.

[8] 3GPP, "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and channel coding," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 36.213, Nov. 2008, Version 8.7.0. [Online]. Available: http://www.3gpp.org/DynaReport/36212.htm.

[9] 3GPP, "NR; Multiplexing and Channel Coding," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.212, Oct. 2023, Version 17.6.0. [Online]. Available: http://www.3gpp.org/DynaReport/38212.htm.

[10] 3GPP, "LTE; Interface between the Control plane Plane and the User Plane of EPC Nodes," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 29.244, Jul. 2017, Version 14.0.0. [Online]. Available: http://www.3gpp.org/DynaReport/29244.htm.

[11] A. Lazaridou and M. Baroni, "Emergent multi-agent communication in the deep learning era," *arXiv preprint arXiv:2006.02419*, 2020.

[12] T. Kasai, H. Tenmoto, and A. Kamiya, "Learning of communication codes in multi-agent reinforcement learning problem," in *2008 IEEE conference on soft computing in industrial applications*, IEEE, 2008, pp. 1–6.

[13] P. Varshavskaya, L. P. Kaelbling, and D. Rus, "Efficient distributed reinforcement learning through agreement," *Distributed Autonomous Robotic Systems 8*, pp. 367–378, 2009.

[14] J. N. Foerster, Y. M. Assael, N. de Freitas, and S. Whiteson, "Learning to communicate with Deep multi-agent reinforcement learning," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 2016, pp. 2145–2153.

[15] S. Sukhbaatar, A. Szlam, and R. Fergus, "Learning multiagent communication with backpropagation," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 2016, pp. 2252–2260.

[16] A. F. M. S. Shah, A. N. Qasim, M. A. Karabulut, H. Ilhan, and M. B. Islam, "Survey and performance evaluation of multiple access schemes for next-generation wireless communication systems," *IEEE Access*, vol. 9, pp. 113 428–113 442, 2021. DOI: 10.1109/ACCESS.2021.3104509.

[17] R. Moosavi and E. G. Larsson, "Reducing physical layer control signaling using mobile-assisted scheduling," *IEEE transactions on wireless communications*, vol. 12, no. 1, pp. 368–379, 2012.

[18] R. Moosavi, J. Eriksson, and E. G. Larsson, "Differential signaling of scheduling information in wireless multiple access systems," in *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, IEEE, 2010, pp. 1–6.

[19] E. Azarnasab, R.-R. Chen, K. H. Teo, Z. Tao, and B. Farhang-Boroujeny, "Medium access control signaling for reliable spectrum agile radios," in *GLOBECOM 2009-2009 IEEE Global Telecommunications Conference*, IEEE, 2009, pp. 1–5.

[20] J. Jiang and Z. Lu, "Learning attentional communication for multi-agent cooperation," *Advances in neural information processing systems*, vol. 31, 2018.

[21] A. Das *et al.*, "Tarmac: Targeted multi-agent communication," in *International Conference on Machine Learning*, PMLR, 2019, pp. 1538–1546.

[22] A. Singh, T. Jain, and S. Sukhbaatar, "Learning when to communicate at scale in multi-agent cooperative and competitive tasks," in *International Conference on Learning Representations*, 2018.

[23] R. Lowe, J. Foerster, Y.-L. Boureau, J. Pineau, and Y. Dauphin, "On the pitfalls of measuring emergent communication," in *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, 2019, pp. 693–701.

[24] T. Eccles, Y. Bachrach, G. Lever, A. Lazaridou, and T. Graepel, "Biases for emergent communication in multi-agent reinforcement learning," *Advances in neural information processing systems*, vol. 32, 2019.

[25] A. Mostaani, O. Simeone, S. Chatzinotas, and B. Ottersten, "Learning-based physical layer communications for multiagent collaboration," in *2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2019, pp. 1–6. DOI: 10.1109/PIMRC.2019.8904190.

[26] J. S. P. Roig and D. Gündüz, "Remote reinforcement learning over a noisy channel," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1–6. DOI: 10.1109/GLOBECOM42002.2020.9322408.

[27] T.-Y. Tung, S. Kobus, J. P. Roig, and D. Gündüz, "Effective communications: A joint learning and communication framework for multi-agent reinforcement learning over noisy channels," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 8, pp. 2590–2603, 2021.

[28] G. He, S. Cui, Y. Dai, and T. Jiang, "Learning task-oriented channel allocation for multi-agent communication," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 11, pp. 12 016–12 029, 2022.

[29] M. A. Jadoon, A. Pastore, M. Navarro, and F. Perez-Cruz, "Deep reinforcement learning for random access in machine-type communication," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*, 2022, pp. 2553–2558. DOI: 10.1109/WCNC51071.2022.9771953.

[30] Z. Chen and D. B. Smith, "Heterogeneous machine-type communications in cellular networks: Random access optimization by deep reinforcement learning," in *2018 IEEE*

International Conference on Communications (ICC), 2018, pp. 1–6. DOI: 10.1109/ICC.2018.8422775.

[31] J.-H. Lee, H. Seo, J. Park, M. Bennis, and Y.-C. Ko, "Learning emergent random access protocol for LEO satellite networks," *IEEE Transactions on Wireless Communications*, pp. 1–1, 2022. DOI: 10.1109/TWC.2022.3192365.

[32] M. Zhang, L. de Alfaro, and J. Garcia-Luna-Aceves, "Using reinforcement learning in slotted aloha for ad-hoc networks," in *Proceedings of the 23rd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2020, pp. 245–252.

[33] Z. Guo, Z. Chen, P. Liu, J. Luo, X. Yang, and X. Sun, "Multi-agent reinforcement learning based distributed channel access for next generation wireless networks," *IEEE Journal on Selected Areas in Communications*, pp. 1–1, 2022. DOI: 10.1109/JSAC.2022.3143251.

[34] A. Farago, A. Myers, V. Syrotiuk, and G. Zaruba, "Meta-MAC protocols: Automatic combination of MAC protocols to optimize performance for unknown conditions," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 9, pp. 1670–1681, 2000. DOI: 10.1109/49.872955.

[35] I. Chlamtac, A. Farago, A. Myers, V. Syrotiuk, and G. Zaruba, "Adapt: A dynamically self-adjusting media access control protocol for ad hoc-networks," in *Seamless Interconnection for Universal Services. Global Telecommunications Conference. GLOBECOM'99.(Cat. No. 99CH37042)*, IEEE, vol. 1, 1999, pp. 11–15.

[36] Q. Ye and W. Zhuang, "Distributed and adaptive medium access control for internet-of-things-enabled mobile networks," *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 446–460, 2017. DOI: 10.1109/JIOT.2016.2566659.

[37] A. Gomes, D. F. Macedo, and L. F. Vieira, "Automatic MAC protocol selection in wireless networks based on reinforcement learning," *Computer Communications*, vol. 149, pp. 312–323, 2020.

[38] H. B. Pasandi and T. Nadeem, "Towards a learning-based framework for self-driving design of networking protocols," *IEEE Access*, vol. 9, pp. 34 829–34 844, 2021. DOI: 10.1109/ACCESS.2021.3061729.

[39] A. Valcarce and J. Hoydis, "Towards joint learning of optimal MAC signaling and wireless channel access," *IEEE Transactions on Cognitive Communications and Networking*, pp. 1–1, 2021. DOI: 10.1109/TCCN.2021.3080677.

[40] L. Miuccio, S. Riolo, S. Samarakoony, D. Panno, and M. Bennis, "Learning generalized wireless MAC communication protocols via abstraction," *arXiv preprint arXiv:2206.06331*, 2022.

[41] M. P. Mota, A. Valcarce, J.-M. Gorce, and J. Hoydis, "The emergence of wireless MAC protocols with multi-agent reinforcement learning," in *2021 IEEE Globecom Workshops (GC Wkshps)*, IEEE, 2021, pp. 1–6.

[42] M. P. Mota, A. Valcarce, and J.-M. Gorce, "Scalable joint learning of wireless multiple-access policies and their signaling," in *2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)*, IEEE, 2022, pp. 1–5.

[43] S. Seo, J. Park, S.-W. Ko, J. Choi, M. Bennis, and S.-L. Kim, "Toward semantic communication protocols: A probabilistic logic perspective," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 8, pp. 2670–2686, 2023. DOI: 10.1109/JSAC.2023.3288268.

[44] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[45] G. Brockman *et al.*, "Openai gym," *arXiv preprint arXiv:1606.01540*, 2016.

[46] K. Greff, A. Klein, M. Chovanec, F. Hutter, and J. Schmidhuber, "The sacred infrastructure for computational research," in *Proceedings of the 16th python in science conference*, vol. 28, 2017, pp. 49–56.

[47] P. Tarafder and W. Choi, "MAC protocols for mmWave communication: A comparative survey," *Sensors*, vol. 22, no. 10, p. 3853, 2022.

[48] K. Abid, H. Lakhlef, and A. Bouabdallah, "A survey on recent contention-free MAC protocols for static and mobile wireless decentralized networks in IoT," *Computer Networks*, vol. 201, p. 108 583, 2021.

[49] C. Han, X. Zhang, and X. Wang, "On medium access control schemes for wireless networks in the millimeter-wave and terahertz bands," *Nano Communication Networks*, vol. 19, pp. 67–80, 2019.

[50] O. Haddad, "Channel Modeling and Multiple Access Solutions for Medical Wireless Body-Area Networks based on Optical Wireless Technology," Theses, Ecole Centrale Marseille, Jul. 2021. [Online]. Available: https://theses.hal.science/tel-03716999.

[51] S. Gamal, M. Rihan, S. Hussin, A. Zaghloul, and A. A. Salem, "Multiple access in cognitive radio networks: From orthogonal and non-orthogonal to rate-splitting," *IEEE Access*, vol. 9, pp. 95 569–95 584, 2021. DOI: 10.1109/ACCESS.2021.3095142.

[52] S. Jiang, "State-of-the-art medium access control (MAC) protocols for underwater acoustic networks: A survey based on a MAC reference model," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 1, pp. 96–131, 2018. DOI: 10.1109/COMST.2017.2768802.

[53] R. Pickholtz, D. Schilling, and L. Milstein, "Theory of spread-spectrum communications - a tutorial," *IEEE Transactions on Communications*, vol. 30, no. 5, pp. 855–884, 1982. DOI: 10.1109/TCOM.1982.1095533.

[54] Z. Zheng, S. Jiang, R. Feng, L. Ge, and C. Gu, "Survey of reinforcement-learning-based MAC protocols for wireless ad hoc networks with a MAC reference model," *Entropy*, vol. 25, no. 1, p. 101, 2023.

[55] L. G. Roberts, "Dynamic allocation of satellite capacity through packet reservation," in *Proceedings of the June 4-8, 1973, national computer conference and exposition*, 1973, pp. 711–716.

[56] P. Wang, J. Xiao, and L. Ping, "Comparison of orthogonal and non-orthogonal approaches to future wireless cellular systems," *IEEE Vehicular Technology Magazine*, vol. 1, no. 3, pp. 4–11, 2006.

[57] M. Vaezi, Z. Ding, and H. V. Poor, *Multiple access techniques for 5G wireless networks and beyond*. Springer, 2019, vol. 159.

[58] V. K. Garg and Y.-C. Wang, "7 - wireless network access technologies," in *The Electrical Engineering Handbook*, W.-K. CHEN, Ed., Burlington: Academic Press, 2005, pp. 1005–1009, ISBN: 978-0-12-170960-0. DOI: https://doi.org/10.1016/B978-012170960-0/50075-X. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B978012170960050075X.

[59] D. Falconer, F. Adachi, and B. Gudmundson, "Time division multiple access methods for wireless personal communications," *IEEE Communications Magazine*, vol. 33, no. 1, pp. 50–57, 1995. DOI: 10.1109/35.339881.

[60] P. Baier, "CDMA or TDMA? CDMA for GSM?" In *5th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Wireless Networks - Catching the Mobile Future.*, vol. 4, 1994, 1280–1284 vol.4. DOI: 10.1109/WNCMF.1994.529459.

[61] R. T. Derryberry, S. D. Gray, D. M. Ionescu, G. Mandyam, and B. Raghothaman, "Transmit diversity in 3g cdma systems," *IEEE communications magazine*, vol. 40, no. 4, pp. 68–75, 2002.

[62] S. Ahmadi, "Chapter 3 - new radio access physical layer aspects (part 1)," in *5G NR*, S. Ahmadi, Ed., Academic Press, 2019, pp. 285–409, ISBN: 978-0-08-102267-2. DOI: https://doi.org/10.1016/B978-0-08-102267-2.00003-8. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780081022672000038.

[63] H. Wymeersch and A. Eryilmaz, "Chapter 12 - multiple access control in wireless networks," in *Academic Press Library in Mobile and Wireless Communications*, S. K. Wilson, S. Wilson, and E. Biglieri, Eds., Oxford: Academic Press, 2016, pp. 435–465, ISBN: 978-0-12-398281-0. DOI: https://doi.org/10.1016/B978-0-12-398281-0.00012-0. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780123982810000120.

[64] I. Villordo-Jimenez, I. E. Zaldivar-Huerta, and G. Galvan-Tejada, "An overview of SDMA in communications systems," in *2006 49th IEEE International Midwest Symposium on Circuits and Systems*, IEEE, vol. 1, 2006, pp. 168–171.

[65] S. Chen, S. Sun, G. Xu, X. Su, and Y. Cai, "Beam-space multiplexing: Practice, theory, and trends, from 4G TD-LTE, 5G, to 6G and beyond," *IEEE Wireless Communications*, vol. 27, no. 2, pp. 162–172, 2020.

[66] A. Brand and H. Aghvami, *Multiple access protocols for mobile communications: GPRS, UMTS and beyond*. John Wiley & Sons, 2002.

[67] O. Tervo, T. Levanen, K. Pajukoski, J. Hulkkonen, P. Wainio, and M. Valkama, "5G new radio evolution towards sub-THz communications," in *2020 2nd 6G Wireless Summit (6G SUMMIT)*, IEEE, 2020, pp. 1–6.

[68] 3GPP, "NR; NR and NG-RAN Overall description; Stage-2," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.300. [Online]. Available: http://www.3gpp.org/DynaReport/38300.htm.

[69] 3GPP, "NR; Physical Layer Procedures for Control," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.213, Oct. 2023, Version 17.7.0. [Online]. Available: http://www.3gpp.org/DynaReport/38213.htm.

[70] 3GPP, "NR; Physical Layer Procedures for Data," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.214, Jun. 2019, Version 15.6.0. [Online]. Available: http://www.3gpp.org/DynaReport/38214.htm.

[71] A. C. V. Gummalla and J. O. Limb, "Wireless medium access control protocols," *IEEE Communications Surveys & Tutorials*, vol. 3, no. 2, pp. 2–15, 2000. DOI: 10.1109/COMST.2000.5340799.

[72] Z. H. Jaber, D. J. Kadhim, and A. S. Al-Araji, "Medium access control protocol design for wireless communications and networks review," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 2, p. 1711, 2022.

[73] K. Akkarajitsakul, E. Hossain, D. Niyato, and D. I. Kim, "Game theoretic approaches for multiple access in wireless networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 3, pp. 372–395, 2011.

[74] D. Jiang and G. Liu, "An overview of 5G requirements," *5G Mobile Communications*, pp. 3–26, 2016.

[75] M. Manini, "Resource allocation and scheduling in 5th generation networks," Theses, Université Rennes 1, Jan. 2021. [Online]. Available: https://theses.hal.science/tel-03223084.

[76] A. Mamane, M. Fattah, M. El Ghazi, M. El Bekkali, Y. Balboul, and S. Mazer, "Scheduling algorithms for 5g networks and beyond: Classification and survey," *IEEE Access*, vol. 10, pp. 51 643–51 661, 2022.

[77] P. Viswanath, D. Tse, and R. Laroia, "Opportunistic beamforming using dumb antennas," *IEEE Transactions on Information Theory*, vol. 48, no. 6, pp. 1277–1294, 2002. DOI: 10.1109/TIT.2002.1003822.

[78] M. Kountouris and D. Gesbert, "Memory-based opportunistic multi-user beamforming," in *Proceedings. International Symposium on Information Theory, 2005. ISIT 2005.*, 2005, pp. 1426–1430. DOI: 10.1109/ISIT.2005.1523578.

[79] H. Kim and Y. Han, "A proportional fair scheduling for multicarrier transmission systems," *IEEE Communications letters*, vol. 9, no. 3, pp. 210–212, 2005.

[80] "Reinforcement learning: A survey," *Journal of artificial intelligence research*, vol. 4, pp. 237–285, 1996.

[81] C. M. Bishop, *Pattern Recognition and Machine Learning, 5th Edition* (Information Science and Statistics). Springer, 2007, ISBN: 9780387310732. [Online]. Available: http://www.worldcat.org/oclc/71008143.

[82] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[83] K. J. Astrom *et al.*, "Optimal control of Markov processes with incomplete state information," *Journal of mathematical analysis and applications*, vol. 10, no. 1, pp. 174–205, 1965.

[84] T. Jaakkola, S. Singh, and M. Jordan, "Reinforcement learning algorithm for partially observable Markov decision problems," *Advances in neural information processing systems*, vol. 7, 1994.

[85] A. R. Cassandra, L. P. Kaelbling, and M. L. Littman, "Acting optimally in partially observable stochastic domains," in *Aaai*, vol. 94, 1994, pp. 1023–1028.

[86] V. Mnih *et al.*, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[87] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.

[88] T. Ni, B. Eysenbach, and R. Salakhutdinov, "Recurrent model-free rl can be a strong baseline for many pomdps," in *International Conference on Machine Learning*, PMLR, 2022, pp. 16 691–16 723.

[89] J. X. Wang *et al.*, "Learning to reinforcement learn," in *Proceedings of the 39th Annual Meeting of the Cognitive Science Society, CogSci 2017, London, UK, 16-29 July 2017*, G. Gunzelmann, A. Howes, T. Tenbrink, and E. J. Davelaar, Eds., cognitivesciencesociety.org, 2017, ISBN: 978-0-9911967-6-0. [Online]. Available: https://mindmodeling.org/cogsci2017/papers/0252/index.html.

[90] Y. Duan, J. Schulman, X. Chen, P. L. Bartlett, I. Sutskever, and P. Abbeel, "Rl^2: Fast reinforcement learning via slow reinforcement learning," *arXiv preprint arXiv:1611.02779*, 2016.

[91] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction* (Adaptive Computation and Machine Learning series). MIT Press, 2018, ISBN: 9780262039246. [Online]. Available: https://books.google.com.br/books?id=6DKPtQEACAAJ.

[92] F. A. Oliehoek, M. T. Spaan, and N. Vlassis, "Optimal and approximate q-value functions for decentralized POMDPs," *Journal of Artificial Intelligence Research*, vol. 32, pp. 289–353, 2008.

[93] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications," *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 3826–3839, 2020. DOI: 10.1109/TCYB.2020.2977374.

[94] R. Lowe, Y. I. Wu, A. Tamar, J. Harb, O. P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Advances in neural information processing systems*, 2017, pp. 6379–6390.

[95] T. P. Lillicrap *et al.*, "Continuous control with deep reinforcement learning.," in *ICLR*, Y. Bengio and Y. LeCun, Eds., 2016. [Online]. Available: http://dblp.uni-trier.de/db/conf/iclr/iclr2016.html#LillicrapHPHETS15.

[96] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with Gumbel-Softmax," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France,*

April 24-26, 2017, Conference Track Proceedings, OpenReview.net, 2017. [Online]. Available: https://openreview.net/forum?id=rkE3y85ee.

[97] C. Maddison, A. Mnih, and Y. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," in *Proceedings of the international conference on learning Representations*, International Conference on Learning Representations, 2017.

[98] S. Sinha, H. Bharadhwaj, A. Srinivas, and A. Garg, "D2rl: Deep dense architectures in reinforcement learning," *arXiv preprint arXiv:2010.09163*, 2020.

[99] I. Uchendu *et al.*, "Jump-start reinforcement learning," in *International Conference on Machine Learning*, PMLR, 2023, pp. 34 556–34 583.

[100] N. Jaques *et al.*, "Social influence as intrinsic motivation for multi-agent deep reinforcement learning," in *International Conference on Machine Learning*, PMLR, 2019, pp. 3040–3049.

[101] M. Vecerik *et al.*, "Leveraging demonstrations for deep reinforcement learning on robotics problems with sparse rewards," *arXiv preprint arXiv:1707.08817*, 2017.

[102] J. Ho and S. Ermon, "Generative adversarial imitation learning," *Advances in neural information processing systems*, vol. 29, 2016.

[103] R. Lowe, A. Gupta, J. Foerster, D. Kiela, and J. Pineau, "On the interaction between supervision and self-play in emergent communication," in *International Conference on Learning Representations*, 2019.

[104] P. Dent, G. E. Bottomley, and T. Croft, "Jakes fading model revisited," *Electronics letters*, vol. 13, no. 29, pp. 1162–1163, 1993.

[105] R. Jain, D. Chiu, and W. Hawe, *A quantitative measure of fairness and discrimination for resource allocation in shared computer systems*, 1998. arXiv: cs/9809099 [cs.NI].

[106] OpenAI *et al.*, *Dota 2 with large scale deep reinforcement learning*, 2019. arXiv: 1912.06680 [cs.LG].

[107] K. Gogineni, P. Wei, T. Lan, and G. Venkataramani, "Scalability bottlenecks in multi-agent reinforcement learning systems," *arXiv preprint arXiv:2302.05007*, 2023.

[108] A. J. Goldsmith and S.-G. Chua, "Adaptive coded modulation for fading channels," *IEEE Transactions on communications*, vol. 46, no. 5, pp. 595–602, 1998.

[109] F. Blanquez-Casado, G. Gomez, M. d. C. Aguayo-Torres, and J. T. Entrambasaguas, "eOLLA: An enhanced outer loop link adaptation for cellular networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, pp. 1–16, 2016.

[110] R. Fantacci, D. Marabissi, D. Tarchi, and I. Habib, "Adaptive modulation and coding techniques for OFDMA systems," *IEEE Transactions on Wireless Communications*, vol. 8, no. 9, pp. 4876–4883, 2009.

[111] A. Simonsson and A. Furuskar, "Uplink power control in lte-overview and performance, subtitle: Principles and benefits of utilizing rather than compensating for SINR variations," in *2008 IEEE 68th Vehicular Technology Conference*, IEEE, 2008, pp. 1–5.

[112] F. H. C. Neto, D. C. Araújo, M. P. Mota, T. F. Maciel, and A. L. de Almeida, "Uplink power control framework based on reinforcement learning for 5G networks," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 6, pp. 5734–5748, 2021.

[113] F. J. Martın-Vega, J. C. Ruiz-Sicilia, M. C. Aguayo, and G. Gómez, "Emerging tools for link adaptation on 5G NR and beyond: Challenges and opportunities," *IEEE Access*, vol. 9, pp. 126 976–126 987, 2021.

[114] J. Lee, J.-K. Han, and J. Zhang, "MIMO technologies in 3GPP LTE and LTE-advanced," *EURASIP Journal on wireless communications and networking*, vol. 2009, pp. 1–10, 2009.

[115] M. Giordani, M. Polese, A. Roy, D. Castor, and M. Zorzi, "A tutorial on beam management for 3GPP NR at mmWave frequencies," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 173–196, 2018.

[116] M. Q. Khan, A. Gaber, P. Schulz, and G. Fettweis, "Machine learning for millimeter wave and terahertz beam management: A survey and open challenges," *IEEE Access*, vol. 11, pp. 11 880–11 902, 2023.

[117] D. d. S. Brilhante *et al.*, "A literature survey on AI-aided beamforming and beam management for 5g and 6g systems," *Sensors*, vol. 23, no. 9, p. 4359, 2023.

[118] Z. Ding, X. Lei, G. K. Karagiannidis, R. Schober, J. Yuan, and V. K. Bhargava, "A survey on non-orthogonal multiple access for 5G networks: Research challenges and future trends," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 10, pp. 2181–2195, 2017.

[119] A. Azari, M. Ozger, and C. Cavdar, "Risk-aware resource allocation for URLLC: Challenges and strategies with machine learning," *IEEE Communications Magazine*, vol. 57, no. 3, pp. 42–48, 2019. DOI: 10.1109/MCOM.2019.1800610.

[120] M. Christopoulou, S. Barmpounakis, H. Koumaras, and A. Kaloxylos, "Artificial intelligence and machine learning as key enablers for V2X communications: A comprehensive survey," *Vehicular Communications*, p. 100 569, 2022.

[121] P. Lin, Q. Song, F. R. Yu, D. Wang, A. Jamalipour, and L. Guo, "Wireless virtual reality in beyond 5g systems with the internet of intelligence," *IEEE Wireless Communications*, vol. 28, no. 2, pp. 70–77, 2021.

[122] G. Minopoulos and K. E. Psannis, "Opportunities and challenges of tangible XR applications for 5G networks and beyond," *IEEE Consumer Electronics Magazine*, vol. 12, no. 6, pp. 9–19, 2023. DOI: 10.1109/MCE.2022.3156305.

[123] E. C. Strinati and S. Barbarossa, "6G networks: Beyond shannon towards semantic and goal-oriented communications," *Computer Networks*, vol. 190, p. 107 930, 2021.

[124] D. Gündüz *et al.*, "Beyond transmitting bits: Context, semantics, and task-oriented communications," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 1, pp. 5–41, 2023. DOI: 10.1109/JSAC.2022.3223408.

[125] J. Holm, F. Chiariotti, A. E. Kalør, B. Soret, T. B. Pedersen, and P. Popovski, "Goal-oriented scheduling in sensor networks with application timing awareness," *IEEE Transactions on Communications*, vol. 71, no. 8, pp. 4513–4527, 2023. DOI: 10.1109/TCOMM.2023.3282256.

[126] F. Xu, H. Uszkoreit, Y. Du, W. Fan, D. Zhao, and J. Zhu, "Explainable AI: A brief survey on history, research areas, approaches and challenges," in *Natural Language Processing and Chinese Computing: 8th CCF International Conference, NLPCC 2019, Dunhuang, China, October 9–14, 2019, Proceedings, Part II 8*, Springer, 2019, pp. 563–574.

[127] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems 30*, I. Guyon *et al.*, Eds., Curran Associates, Inc., 2017, pp. 4765–4774. [Online]. Available: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

[128] P. Madumal, T. Miller, L. Sonenberg, and F. Vetere, "Explainable reinforcement learning through a causal lens," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, 2020, pp. 2493–2500.

[129] K. Cui *et al.*, "A survey on large-population systems and scalable multi-agent reinforcement learning," *arXiv preprint arXiv:2209.03859*, 2022.

[130] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

[131] L. Chen *et al.*, "Decision transformer: Reinforcement learning via sequence modeling," *Advances in neural information processing systems*, vol. 34, pp. 15 084–15 097, 2021.

[132] D. Willemsen, M. Coppola, and G. C. de Croon, "MAMBPO: Sample-efficient multi-robot reinforcement learning using learned world models," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2021, pp. 5635–5640.

[133] X. Wang, Z. Zhang, and W. Zhang, "Model-based multi-agent reinforcement learning: Recent progress and prospects," *arXiv preprint arXiv:2203.10603*, 2022.

Appendix A

Résumé en Français

A.1 Contexte

L'idée principale de la présente thèse est de laisser émerger le sens des messages de signalisation de manière renforcée au niveau des nœuds du réseau pendant qu'ils effectuent une tâche de communication. Par *sens* des messages, nous entendons une correspondance entre les messages de contrôle reçus au temps $t < t'$ et les actions (actions de signalisation de contrôle, ainsi que les actions d'accès au canal) prises au temps $t \geq t'$. Comme la signalisation est apprise conjointement avec la tâche, un protocole spécifique à la tâche peut être produit de cette manière. Nous nous attendons à ce que les techniques d'apprentissage de protocole contribuent de plus en plus à l'intelligence native de l'interface radio, dont les principales propriétés sont :

- Automatisation: La reproductibilité et la rapidité peuvent être atteintes avec la génération automatisée de protocoles.
- Personnalisation: La capacité de concevoir des protocoles adaptés à une application spécifique.

- Optimalité: Recherche automatisée de l'espace de signalisation, permettant la sélection de protocoles avec le moins de surcharge.

Les principaux objectifs de ce travail sont les suivants :

1. Fournir les fondements de l'émergence de protocoles pour permettre des avancées futures dans ce domaine pour les systèmes de communication de prochaine génération.
2. Introduire l'émergence de protocoles de la couche de contrôle d'accès au support (medium access layer - MAC), mettant en évidence les différents défis et avantages qui y sont associés.
3. Développer un cadre pour l'émergence de protocoles MAC qui permet non seulement de produire de nouveaux protocoles, mais aussi de comparer différents protocoles et d'interpréter les messages de signalisation.
4. Étudier les performances des protocoles MAC dans un problème difficile d'allocation de ressources, évaluant ses capacités d'apprentissage dans différentes conditions système.
5. Fournir une formulation pour contrôler la quantité d'échanges de signalisation, permettant un contrôle fin du débit de signalisation.
6. Indiquer des orientations pour les futures recherches dans le domaine, à la fois sur l'étude de l'émergence de protocoles et sur d'autres applications.

A.2 Cadre pour l'émergence de protocoles MAC

A.2.1 Introduction

Le principal objectif du cadre proposé est d'établir les fondements de l'émergence de protocole, d'investiguer son bénéfice par rapport à d'autres techniques de conception de protocole et de discuter de la manière d'étudier efficacement les protocoles. Ainsi, les nouvelles connaissances obtenues à partir de cela concernent :

- Comparaison de différentes méthodes pour apprendre un protocole.
- **Étude de la taille du vocabulaire**: Nous évaluons les capacités de l'émergence de protocole à réduire la quantité de signalisation utilisée.
- **Caractérisation des protocoles**: Nous proposons de nouveaux indicateurs clés de performance (key performance indicators - KPIs) et des métriques pour comparer les protocoles émergents et les interpréter en utilisant la théorie de l'information.
- **Interprétation du protocole**: Nous étudions l'utilisation des messages de contrôle par le protocole émergent.

L'analyse de performance est effectuée dans un problème d'accès multiple en liaison montante.

A.2.2 Modèle du Système

Dans une cellule unique avec une station de base (base station - BS) desservant U équipements utilisateurs (user equipments - UEs) dans un canal partagé montant à fentes, où chaque UE doit transmettre des données à la BS. On suppose que les tampons de chaque UE sont initialement vides et ont une capacité de B unités de données de service (service data units - SDUs). Le modèle de trafic pour les arrivées de SDU suit

un processus de Bernoulli. La tâche de transmission a un horizon fixe T en termes de créneaux horaires.

Les nœuds du réseau, les UE et les BS, sont considérés comme des preneurs de décision capables d'échanger des informations à l'aide de messages via le canal de contrôle. Ces messages sont transmis via des canaux de contrôle supposés dédiés et exempts d'erreurs. L'ensemble des messages de contrôle possibles en liaison montante et descendante a une cardinalité de V_{UCM} et V_{UCM} respectivement. À chaque pas de temps, la BS peut envoyer un message de contrôle à chaque UE, et chaque UE peut envoyer un message de contrôle à la BS. De plus, les UE peuvent également envoyer des unités de données de protocole (protocol data units - PDUs) via le canal partagé ou supprimer une SDU de leur tampon.

En termes de métriques de performance, le débit utile est la principale métrique de performance d'intérêt. Le débit utile est calculé comme la quantité de données d'information reçue par le BS pendant toute la durée de la tâche de transmission de T créneaux temporels. En plus de cela, le taux de collision et la fiabilité sont également utilisés pour comparer les protocoles. Enfin, deux métriques basées sur l'information mutuelle sont proposées, dont l'une est une nouveauté, afin de mesurer le degré de coordination du protocole. Ces métriques quantifient la relation entre une action d'UE et le message qui a été récemment reçu du BS, ainsi que la relation entre le message envoyé par le BS et les messages reçus par les UEs.

A.2.3 Formulation du Problème comme MARL

Le problème est formulé comme une tâche coopérative d'apprentissage par renforcement multi-agent (multi-agent reinforcement learning - MARL), où les couches MAC des nœuds du réseau sont des agents RL qui doivent apprendre à communiquer

entre eux pour résoudre une tâche de transmission en liaison montante. De plus, l'UE doit apprendre quand envoyer des données par le biais du canal partagé et quand supprimer une SDU. Pour décider comment agir, les agents doivent prendre en compte les messages reçus des autres agents.

En termes de formulation d'apprentissage par renforcement (reinforcement learning - RL), les observations sont définies comme suit:

- Observation de la BS: L'état du canal, c'est-à-dire inactif, occupé ou réception sans collision. L'état occupé se produit lorsque la BS ne peut pas décoder l'information reçue, soit en raison d'une collision, soit en raison des conditions du canal. À son tour, la réception sans collision se produit lorsque la BS peut décoder avec succès la PDU reçue et sa valeur représente de chaque UE la PDU décodée provient.
- Observation de l'UE: Le nombre de SDUs dans le tampon à un moment donné. Il représente l'état du tampon d'une UE donnée.

Alors que l'UE a trois actions possibles à choisir:

- Ne rien faire: Elle ne transmet pas une PDU ni ne supprime une SDU.
- Transmettre: Elle transmet la plus ancienne SDU dans le tampon.
- Supprimer: Elle supprime la plus ancienne SDU dans le tampon.

De plus, les nœuds ont une action supplémentaire, appelée action de communication. Elle est représentée par les messages de contrôle en liaison montante ou descendante à transmettre tout en étant également disponible à l'état de l'autre agent en tant que message reçu. Pour le BS, cette action est un vecteur contenant tous les messages de liaison descendante envoyés à toutes les UE, tandis que pour l'UE, c'est une seule valeur avec le message à envoyer au BS.

La récompense est la même pour tous les agents, car il s'agit d'une tâche coopérative, et elle vise à maximiser le bon débit tout en cherchant à maximiser la fiabilité. La fonction est une somme des contributions de chaque UE, et pour chaque UE, une pénalisation est attribuée si elle supprime une SDU qui n'a pas été reçue par la BS et une récompense est attribuée si la BS reçoit avec succès de nouvelles données de cette UE.

L'algorithme d'apprentissage choisi est le gradient de politique déterministe multi-agent (multi-agent deep deterministic policy gradient - MADDPG). Deux modifications sont ajoutées à cet algorithme:

- Départ RL (jump-start RL): Une méthode pour tirer parti d'une politique d'expert pour améliorer les performances d'apprentissage.
- D2RL: Une modification de l'architecture qui améliore les performances globales des réseaux de neurones RL en utilisant des connexions denses.

A.2.4 Résultats de Simulation

Les techniques d'apprentissage sont comparées à deux baselines basés sur des règles, une baseline sans contention basée sur des autorisations et une baseline basée sur la contention. Un ensemble d'expériences est réalisé avec plusieurs résultats afin d'étudier les avantages de l'émergence de protocole et comment étudier de tels protocoles.

Les techniques d'apprentissage sont comparées à deux références basées sur des règles, une basée sur une attribution sans contention et une autre basée sur la contention. Une série d'expériences est menée avec plusieurs résultats afin d'étudier les avantages de l'émergence de protocole et la manière d'étudier de tels protocoles.

Le premier ensemble d'expériences compare différentes méthodes de conception de protocoles:

1. Émergence de protocole: Le vocabulaire du plan de contrôle, les politiques et les politiques d'accès au canal des UEs et des BS sont appris.
2. Apprentissage de protocole: Les politiques de plan de contrôle et d'accès au canal des UE sont apprises, tandis que la BS suit une référence basée sur une attribution avec une politique de plan de contrôle fixe.
3. Apprentissage d'accès au canal: Seule la politique d'accès au canal des UE est apprise, tandis que leur plan de contrôle et la BS suivent la référence basée sur une attribution.

Les résultats montrent que, bien que l'émergence de protocole prenne plus de temps à apprendre que les autres approches, elle présente des performances supérieures une fois stabilisée par rapport aux autres approches. Les résultats illustrent également l'adaptabilité de l'émergence de protocole dans des scénarios de taux d'erreur réduit. Alors que les autres approches de conception de protocoles atteignent un plateau lors de la réduction du taux d'erreur, les performances de l'émergence de protocole continuent d'augmenter.

Le deuxième ensemble de résultats concerne la production de protocoles avec l'émergence de protocoles. Il compare différentes algorithmes d'entraînement et différentes configurations de vocabulaire. Les points importants sont les suivants :

1. Tirer parti d'un expert peut être bénéfique: cela accélère l'apprentissage, bien que cela ait l'inconvénient de limiter le vocabulaire du plan de contrôle pour qu'il soit identique à celui de l'expert, limitant la recherche dans l'espace de sig-

nalisation.

2. L'émergence de protocoles permet de réduire la signalisation: Comme l'émergence de protocoles peut être utilisée avec différentes quantités de signalisation en modifiant les tailles de vocabulaire, l'avantage de cette technique devient clair. Elle permet de produire des protocoles avec moins de signalisation que les baselines comparées, tout en maintenant les performances.

Cela est suivi par des résultats montrant la comparaison des protocoles en termes de coordination et également en termes de différentes métriques de performance du réseau. Les résultats de la coordination montrent qu'une coordination basée sur une signalisation plus stricte conduit à une performance accrue. La caractérisation des protocoles en termes de métriques vise à fournir un profil des protocoles mettant en évidence les différents compromis tels que le délai, le taux de collision, le débit et la fiabilité. De plus, cela montre également que le profil d'un protocole peut être contrôlé par la fonction de récompense en concevant un protocole qui vise à être plus économe en énergie.

Enfin, la probabilité conditionnelle est utilisée pour étudier et interpréter le sens des messages de contrôle. Cette étude montre qu'il est possible de comprendre le sens de la communication émergente tout en illustrant également que ce sens peut être fluide, le sens d'un message pouvant changer en fonction du contexte. Cette signification contextuelle est ce qui permet aux protocoles émergents de bien performer même avec une signalisation réduite.

A.2.5 Conclusion

En résumé, nous avons proposé un nouveau cadre pour l'étude des protocoles émergents, comprenant la production de protocoles, l'analyse de la coordination, le

profilage des protocoles et leur interprétation. Nous avons démontré comment, en utilisant des techniques d'émergence de protocoles, les nœuds radio peuvent apprendre le vocabulaire et la politique de signalisation, ainsi que la politique d'accès au canal, équivalente au contrôle de la couche physique par les agents. Nous avons également montré que le protocole moyen émergé surpasse deux baselines très différentes (sans contention et basée sur la contention) et mis en évidence les forces de ces méthodes.

A.3 Émergence de Protocole pour l'Allocation de Ressources Contiguës

A.3.1 Introduction

Dans ce chapitre, l'accent a été mis sur l'étude des performances de l'émergence de protocoles dans un problème plus complexe, évaluant ses capacités d'apprentissage, sa scalabilité et ses limitations. Le scénario étudié concerne la planification avec une allocation de ressources dans le domaine fréquentiel de manière contiguë.

A.3.2 Modèle du Système

Dans une seule cellule avec une base station (BS) desservant U UEs sur un canal partagé à fentes en liaison montante, où chaque UE doit transmettre des données à la BS. La bande passante disponible dans le domaine fréquentiel est divisée en blocs de ressources (resource blocks - RBs), tandis que le domaine temporel est supposé être en fentes avec une durée de pas de temps fixe. On suppose que les tampons de chaque UE sont initialement vides et ont une capacité de B bits. Le modèle de trafic pour les arrivées des SDUs suit un processus de Bernoulli, et les SDU ont une taille constante en bits. Un processus de demande de répétition automatique (automatic repeat request - ARQ) est utilisé par les nœuds du réseau pour gérer les tampons, ainsi, lors de la

réception réussie d'un bloc de transport par la BS, l'UE retire ces bits du tampon. La tâche de transmission a un horizon fixe T en termes de fentes temporelles. Le système est conscient du canal, chaque UE ayant un rapport signal/bruit (signal-to-noise ratio - SNR), en supposant un canal plat et corrélé dans le temps avec un évanouissement de Rayleigh.

Les nœuds du réseau, UEs et BS, sont supposés être des décideurs capables d'échanger des informations en utilisant des messages via le canal de contrôle. Ces messages sont transmis par des canaux de contrôle supposés dédiés et sans erreur. L'ensemble des messages de contrôle possibles en liaison montante et en liaison descendante a une cardinalité de V_{UCM} et V_{UCM}. À chaque pas de temps, la BS peut envoyer un message de contrôle à chaque UE, et chaque UE peut envoyer un message de contrôle à la BS. De plus, les UEs peuvent également envoyer un bloc de transport à travers le canal partagé, en décidant quels RB utiliseront pour la transmission. La transmission se fait à travers des RB adjacents, et des collisions peuvent survenir si plusieurs UEs transmettent sur la même ressource, auquel cas le bloc de transport entier est perdu.

En termes de métriques de performance, la moyenne géométrique des débits utiles est la principale métrique de performance d'intérêt, car elle fournit une mesure de l'équité. Le débit utile est calculé comme la quantité de données d'information reçue par la BS pendant toute la tâche de transmission de T fentes temporelles.

A.3.3 Formulation du Problème comme MARL

Le problème est formulé comme une tâche coopérative d'apprentissage par renforcement multi-agent. Les agents UE doivent apprendre quand envoyer des données et dans quels RB. Les UEs et la BS apprennent également comment communiquer entre eux via le canal de contrôle, en envoyant les informations pertinentes pour coopérer et

éviter les collisions, utilisées en entrée par les nœuds du réseau pour décider comment agir.

En termes de formulation RL, les observations sont définies comme suit :

- Observation de la BS: Un vecteur avec les rendements spectraux du schéma de modulation et de codage (modulation and coding scheme - MCS) de chaque UE, la quantité normalisée de données reçues de chaque UE et l'état de chaque RB, c'est-à-dire, libre, occupé ou réception sans collision de quel UE.
- État de l'UE: Le nombre de RB nécessaires pour transmettre l'ensemble des données du tampon à un moment donné, compte tenu de son MCS. Il représente l'état du tampon d'une UE donnée en termes de RB nécessaires.

Alors que l'action de l'UE est divisée en 3 composantes, une action principale et deux paramètres d'action, chacun ayant un nombre de valeurs possibles :

- Action principale: Ne rien faire ou transmettre des données
- Paramètre d'action 1: Indique le RB initial pour l'allocation contiguë.
- Paramètre d'action 2: Représente le nombre de RB alloués pour la transmission.

De plus, les nœuds ont une action supplémentaire, appelée action de communication. Elle est représentée par les messages de contrôle en liaison montante ou descendante à transmettre tout en étant également disponible à l'état de l'autre agent en tant que message reçu. Pour la BS, cette action est un vecteur contenant tous les messages de liaison descendante envoyés à toutes les UEs, tandis que pour l'UE, c'est une seule valeur avec le message à envoyer à la BS.

La récompense est la même pour tous les agents, car il s'agit d'une tâche coopérative, et elle vise à maximiser l'équité. Elle est définie comme la moyenne géométrique

des débits utiles de toutes les UEs.

L'algorithme d'apprentissage choisi est le gradient de politique déterministe multi-agent avec D2RL.

A.3.4 Résultats de Simulation

La solution proposée est comparée à un schéma de planification basé sur l'ordonnanceur à équité proportionnelle. Plusieurs expériences sont menées pour évaluer les capacités d'apprentissage des méthodes proposées.

Les premières expériences évaluent la scalabilité de l'émergence de protocoles en termes de scénarios de trafic, de nombre d'UE et de nombre de RB disponibles. L'étude de la scalabilité du trafic montre que dans les régimes de trafic faible, l'émergence de protocoles a des performances similaires à la baseline, mais la surpasse à mesure que la densité de trafic augmente. Les résultats illustrent également la scalabilité avec la bande passante, l'écart de performance entre l'émergence de protocoles et la baseline étant similaire lors de l'augmentation de la bande passante. La scalabilité avec les UEs est plus difficile, car le protocole basé sur l'octroi est conçu pour bien fonctionner dans des scénarios avec de nombreuses UEs, et la performance dans de tels scénarios est presque la même en termes d'équité.

La deuxième série d'expériences évalue les capacités d'apprentissage pour différentes fonctions objectives. L'ensemble des fonctions objectives comprend : le débit utile moyen, la moyenne géométrique du débit utile, le débit utile minimal et l'indice de justice de Jain (JFI). Ces résultats montrent que la baseline PF est un protocole équilibré avec un débit utile global élevé et une grande équité. Cependant, l'émergence de protocoles permet la conception de protocoles qui la surpassent en termes de débit utile global, de service minimal et d'équité.

La troisième série d'expériences étudie l'effet des tailles de vocabulaire, en concevant des expériences avec des modèles de trafic hétérogènes pour les UEs et un canal variable. Les résultats de ces expériences montrent qu'il est possible de produire des protocoles performants avec n'importe quelle valeur de taille de vocabulaire en liaison montante, mais en général, l'apprentissage est meilleur pour des valeurs entre quatre et dix. Augmenter la taille du vocabulaire au-delà de ce point rend l'apprentissage plus difficile, car cela rend l'espace de signalisation trop grand sans qu'il soit utile. La principale contrainte d'apprentissage est le vocabulaire en liaison descendante, car pour des valeurs très basses, l'émergence de protocoles est incapable de produire des protocoles avec de bonnes performances, ne produisant que des protocoles performants après un certain point. Contrairement au cas en liaison montante, augmenter la taille du vocabulaire en liaison descendante n'entrave pas l'apprentissage.

A.3.5 Conclusion

En résumé, nous avons évalué les capacités de la technique proposée d'émergence de protocoles dans un problème plus complexe, à savoir un problème d'allocation de ressources contiguës. Nous avons fourni une formulation RL pour aborder ce problème, avec une méthode de politique sélectionnant le RB de départ et le nombre total de RB alloués, tout en apprenant également la signalisation nécessaire pour la coordination entre la BS et les UEs. Les résultats mettent en évidence les capacités d'apprentissage et les limitations de la méthode proposée.

A.4 Émergence de Protocole sous Contraintes de Signalisation

A.4.1 Introduction

Dans ce chapitre, l'émergence de protocoles a été utilisée pour adapter la densité de la signalisation de contrôle dans un scénario de planification impliquant une allocation de ressources non contiguë. Ainsi, au lieu d'une signalisation persistante comme dans les chapitres 3 et 4, ce chapitre introduit un scénario de signalisation intermittente où les nœuds décident s'ils vont envoyer un message de contrôle ou non. En conséquence de la signalisation intermittente, il est possible de réduire la quantité de données qui transite par les canaux de contrôle.

A.4.2 Modèle du Système

Dans une seule cellule avec une BS desservant U UEs sur un canal partagé à fentes en liaison montante, où chaque UE doit transmettre des données à la BS. La bande passante disponible dans le domaine fréquentiel est divisée en RBs, qui sont regroupés en groupes de blocs de ressources (resource block group - RBG), tandis que le domaine temporel est supposé être en fentes avec une durée de pas de temps fixe. On suppose que les tampons de chaque UE sont initialement vides et ont une capacité de B bits. Le modèle de trafic pour les arrivées des SDUs suit un processus de Bernoulli, et les SDUs ont une taille constante en bits. Un processus ARQ est utilisé par les nœuds du réseau pour gérer les tampons, ainsi, lors de la réception réussie d'un bloc de transport par la BS, l'UE retire ces bits du tampon. La tâche de transmission a un horizon fixe T en termes de fentes temporelles. Le système est conscient du canal, chaque UE ayant une SNR, en supposant un canal plat et corrélé dans le temps avec un évanouissement de Rayleigh.

Les nœuds du réseau, UEs et BS, sont supposés être des décideurs capables d'échanger des informations en utilisant des messages via le canal de contrôle. Ces messages sont transmis par des canaux de contrôle supposés dédiés et sans erreur. L'ensemble des messages de contrôle possibles en liaison montante et en liaison descendante a une cardinalité de V_{UCM} et V_{UCM}. À chaque pas de temps, la BS peut envoyer un message de contrôle à chaque UE et chaque UE peut envoyer un message de contrôle à la BS. Importamment, les nœuds décident s'ils vont transmettre un message de contrôle ou non, permettant un régime de signalisation intermittente. De plus, les UEs peuvent également envoyer un bloc de transport à travers le canal partagé, en décidant quels RBG elles utiliseront pour la transmission, de manière non contiguë. Des collisions peuvent se produire si plusieurs UEs transmettent sur la même ressource, auquel cas le bloc de transport entier est perdu.

En termes de métriques de performance, la moyenne géométrique des débits utiles est la principale métrique de performance d'intérêt, car elle fournit une mesure de l'équité. Le débit utile est calculé comme la quantité de données d'information reçue par la BS pendant toute la tâche de transmission de T fentes temporelles. Le débit de signalisation est une autre métrique importante à suivre, ainsi que le taux de collision.

A.4.3 Formulation du Problème comme MARL

Le problème est formulé comme une tâche coopérative d'apprentissage par renforcement multi-agent. Les agents UE doivent apprendre quand envoyer des données et dans quels RBG. Les UEs et la BS apprennent également comment et quand communiquer entre eux via le canal de contrôle.

En termes de formulation RL, les observations sont définies comme suit :

- Observation de la BS : Un vecteur avec les rendements spectraux du MCS de

chaque UE, la quantité normalisée de données reçues de chaque UE et l'état de chaque RBG, c'est-à-dire, libre, occupé ou réception sans collision de quel UE.

- État de l'UE : Le nombre de RBG nécessaires pour transmettre l'ensemble des données du tampon à un moment donné, compte tenu de son MCS. Il représente l'état du tampon d'une UE donnée en termes de RBG nécessaires.

L'action d'une UE comprend l'action de transmission de données et l'action de contrôle UCM. L'action de transmission de données est divisée en deux composantes, l'action principale et un paramètre d'action :

- Action principale: Ne rien faire ou transmettre des données
- Paramètre d'action: Une bitmap indiquant si un RBG est utilisé pour la transmission ou non.

L'action de contrôle UCM a deux valeurs possibles, interprétées comme :

- Transmettre l'UCM.
- Ne pas transmettre l'UCM.

La BS n'a que l'action de contrôle DCM, qui est un vecteur contrôlant la transmission de chacun des U DCMs. L'interprétation de chaque élément de ce vecteur d'action est similaire au cas de l'UE.

De plus, les nœuds ont une action supplémentaire, appelée action de communication. Elle est représentée par les messages de contrôle en liaison montante ou descendante à transmettre, en fonction de l'action de contrôle DCM ou UCM, tout en étant également disponible à l'état de l'autre agent en tant que message reçu, s'il a été transmis. Pour la BS, cette action est un vecteur contenant tous les messages de liaison descendante envoyés à toutes les UEs, tandis que pour l'UE, c'est une seule valeur avec

le message à envoyer à la BS.

La récompense est la même pour tous les agents, car il s'agit d'une tâche coopérative, et elle vise à maximiser l'équité tout en compensant la quantité de signalisation utilisée. Elle est définie comme la moyenne géométrique des débits utiles de toutes les UEs, soustraite d'une pénalisation de signalisation. La pénalisation de signalisation est le produit d'un coût de signalisation et de la quantité de messages de contrôle envoyés en une étape temporelle.

A.4.4 Résultats de Simulation

La solution proposée est comparée à un schéma de planification basé sur des octrois équitables proportionnels. Un ensemble d'expériences est mené pour évaluer comment la quantité de signalisation peut être contrôlée et son effet sur les performances.

Les principales conclusions que l'on peut tirer de cet ensemble d'expériences sont :

- L'émergence de protocoles avec signalisation intermittente peut produire des protocoles avec de bonnes performances en termes d'équité. Cependant, l'augmentation du coût de signalisation entraîne une diminution des performances et une plus grande variance.
- L'émergence de protocoles avec signalisation persistante surpasse à la fois l'émergence de protocoles avec signalisation intermittente en termes de moyenne géométrique du débit utile.
- Augmenter le coût de signalisation conduit à une réduction de la signalisation utilisée par le protocole émergent. De plus, même avec un coût nul, le protocole émergent utilise toujours une signalisation intermittente au lieu d'une signali-

sation persistante.

- Le débit binaire du canal de contrôle diminue avec le taux de collision. Cela est naturel, car réduire la quantité d'échanges de contrôle conduit à une coordination plus faible, perceptible par l'augmentation des collisions.

A.4.5 Conclusion

En résumé, une formulation RL a été proposée pour l'émergence de protocoles où les nœuds peuvent décider s'ils vont transmettre un message de contrôle, ce qui peut réduire le débit binaire du canal de contrôle. Les résultats indiquent les capacités du cadre MARL à produire des protocoles avec une signalisation réduite, qui peut être contrôlée par la fonction de récompense avec un poids de coût de signalisation. Cependant, réduire le débit binaire de signalisation conduit à un niveau inférieur de coordination, diminuant les performances globales du réseau et augmentant les taux de collisions.

A.5 Réflexions Finales

Les principaux objectifs de cette thèse étaient de fournir les fondements de l'émergence de protocoles, en mettant en évidence les défis qui y sont associés, d'étudier ses capacités d'apprentissage et sa capacité à contrôler la quantité de signalisation utilisée. De plus, elle visait également à donner un aperçu de l'accès multiple et des procédures qui y sont impliquées dans le cadre de la 5G NR.

En ce qui concerne les études proposées et leurs évaluations de performances, d'une part, le chapter 3 a fourni un cadre pour l'émergence de protocoles basé sur le MARL, avec des résultats mettant en évidence ses performances supérieures par rapport à d'autres techniques de conception de protocoles. De plus, il a décrit des méth-

odes d'évaluation des performances, d'interprétation, de coordination entre les nœuds et de quantification des surcharges de signalisation, les résultats renforçant l'utilité d'un tel cadre. D'autre part, le chapter 4 a appliqué l'idée d'émergence de protocole à un problème plus difficile d'allocation de ressources contiguës, montrant ses robustes capacités d'apprentissage pour différentes densités de trafic et différentes fonctions objectifs. De plus, il a montré ses capacités d'évolution en termes d'UEs et de RBs, tout en montrant qu'une granularité raisonnable de la signalisation aide à produire un meilleur ensemble de protocoles. Enfin, le chapter 5 a proposé une formulation de signalisation intermittente, dans laquelle les nœuds décident si un message de contrôle sera envoyé, au lieu d'envoyer toujours un message de contrôle. Les résultats ont montré que le cadre MARL proposé peut produire des protocoles avec une signalisation réduite et que le débit de signalisation peut être contrôlé par un poids de coût de signalisation ajouté à la fonction de récompense, bien que la réduction du débit de signalisation puisse entraîner une coordination moindre et de moins bonnes performances.

En termes de perspectives futures, l'idée d'apprendre conjointement la signalisation de contrôle et l'optimisation du contrôle du réseau peut être étendue à différents problèmes en communications sans fil. Ces problèmes comprennent: modulation et codage adaptatifs, le contrôle de puissance en liaison montante, la sélection du précodeur de transmission en liaison montante et la gestion de faisceaux.

Cette thèse contribue aux avancées dans les communications sans fil et la conception de protocoles grâce à l'exploration de l'émergence de protocoles. Elle fournit les fondements de l'émergence de protocoles à travers un cadre pour la production et l'étude de tels protocoles. La capacité d'optimiser de manière autonome les protocoles est pertinente pour le développement de réseaux auto-organisés et auto-optimisants,

tout en étant cruciale pour s'adapter à des environnements sans fil dynamiques et complexes. De plus, les idées issues de cette recherche peuvent contribuer aux futures normes de communication sans fil, l'objectif final de la recherche sur l'émergence de protocoles étant la génération d'un système de communication entièrement appris.

FOLIO ADMINISTRATIF

THESE DE L'INSA LYON, MEMBRE DE L'UNIVERSITE DE LYON

NOM : PONTES MOTA DATE de SOUTENANCE : 20/01/2024

Prénoms : Mateus

TITRE : Protocol Emergence with Multi-Agent Reinforcement Learning

NATURE : Doctorat Numéro d'ordre : AAAAISALXXXX

Ecole doctorale : Electronique, Electrotechnique, automatique

Spécialité : Traitement du signal et des images

RESUME : Dans ce travail, nous proposons une technique de conception de protocole automatisée appelée émergence de protocole. Lors de l'émergence d'un protocole, les nœuds du réseau échangent des messages de contrôle afin de se coordonner pour transmettre des données à travers le réseau, mais sans aucun accord préalable sur la signification de ces messages. Cela peut être considéré comme une technique conjointe de signalisation et d'optimisation du réseau, le principal problème de réseau sans fil étudié dans ce travail étant la planification.

Premièrement, les principes fondamentaux de l'émergence des protocoles sont présentés en introduisant un cadre décrivant les méthodes d'évaluation, de caractérisation, de coordination entre nœuds et d'interprétation des performances des protocoles. Ce cadre est étudié dans un problème d'accès multiple slotté.

Dans la deuxième partie de ce travail, nous évaluons les performances d'émergence de protocoles dans un scénario plus difficile impliquant une allocation de ressources contiguës. Ce scénario difficile est utilisé pour évaluer les capacités d'apprentissage et les limites de l'émergence de protocoles, illustrant par exemple la robustesse de certains paramètres et les défis liés à l'évolutivité de l'UE.

Dans la troisième partie de ce travail, nous évaluons l'émergence de protocoles sous contraintes de signalisation, dans un scénario d'allocation non contiguë avec signalisation intermittente. Dans cette étude, nous nous concentrons sur la production d'une méthode dans laquelle le débit binaire de contrôle utilisé par le protocole peut être contrôlé. Les résultats mettent en évidence l'effet du coût de signalisation sur la réduction du débit binaire de contrôle et son effet sur la coordination et les performances.

MOTS-CLÉS : Reinforcement Learning, Wireless Communication, Protocol Emergence

Laboratoire (s) de recherche : CITI

Directeur de thèse: Jean-Marie GORCE

Président de jury :

Composition du jury :

- Veronica BELMEGA (Rapporteur)
- Cedomir STEFANOVIC (Rapporteur)
- Monica NAVARRO (Examinatrice)
- Philippe MARY (Examinateur)
- Jean-Marie GORCE (Directeur de thèse)
- Alvaro VALCARCE (Co-encadrant de thèse)