State University of Londrina
Center of Technology and Urbanization
Department of Electrical Engineering
Electrical Engineering Master Program

Giovanni Maciel Ferreira Silva

Throughput and Latency Q-Learning-based Random Access Protocols for mMTC Systems

Londrina,
January 13, 2022

Giovanni Maciel Ferreira Silva

Throughput and Latency Q-Learning-based Random Access Protocols for mMTC Systems

A Dissertation submitted to the Electrical Engineering Graduate Program at the State University of Londrina in fulfillment of the requirements for the degree of Master of Science in Electrical Engineering.

Area: Telecommunications Systems

Supervisor: Prof. Dr. Taufik Abrão

Londrina,
January 13, 2022

Silva, Giovanni Maciel Ferreira

Throughput and Latency Q-Learning-based Random Access Protocols for mMTC Systems. Londrina, 2022. 111 p.

Supervisor: Prof. Dr. Taufik Abrão

Dissertation (Master of Science) – Department of Electrical Engineering – State University of Londrina

1. mMTC, 2. Q-Learning, 3. Random Access, 4. NOMA, 5. Latency

Giovanni Maciel Ferreira Silva

Throughput and Latency Q-Learning-based Random Access Protocols for mMTC Systems

A Dissertation submitted to the Electrical Engineering Graduate Program at the State University of Londrina in fulfillment of the requirements for the degree of Master of Science in Electrical Engineering.

Area: Telecommunications Systems

Examination Board

Prof. Dr. Glauber Gomes de Oliveira Brante
Federal University of Technology - Parana
Member

Prof. Dr. Marcello Gonçalves Costa
State University of Londrina
Member

Prof. Dr. Taufik Abrão
State University of Londrina
Supervisor

Londrina,
January 13, 2022

Acknowledgements

I would like to thank God for giving me strength in the most difficult moments. Special thanks: to my family for the support; to Prof. Dr. Taufik Abrão for the very important role of guiding me to the best paths that culminated in this work; to colleagues from the Telecomm&SP laboratory for sharing knowledge and moments of relaxation; to the Effortech Technology team for investing in my technical improvement; to CAPES for funding and for believing in the science developed in the our country; and to UEL for providing the space and material needed.

”Let it grow, let it blossom, let it flow.”

Eric Clapton

Abstract

Massive machine-type communication (mMTC) networks will play a key role in sixth generation wireless communication systems (6G). Thousands of devices compete for network resources for sending packets, such as a sensor network in a farm or an automated factory in Industry 4.0. In this scenario, the random access (RA) problem arises, in which devices randomly select network resources and collisions occur frequently. One of the promising ways to solve this problem is to use Q-learning (QL) algorithms. In this work, some machine learning-based techniques available in the literature such as independent and collaborative QL algorithms are analyzed in terms of system throughput and latency. An improvement in the QL collaborative technique and a low-complexity distributed packet-based algorithm are also proposed. Finally, the power disparity between devices in a cell is analyzed using non-orthogonal multiple access (NOMA) with a central node applying successive interference cancellation (SIC) to reduce collisions. A QL algorithm with multi-power levels that increases throughput and reduces latency in NOMA mMTC scenarios is proposed.

Keywords: mMTC, Q-learning, random access, NOMA, latency, successive cancellation interference.

Resumo

As redes de comunicação do tipo máquina massiva (mMTC - *massive machine-type communication*) terão um papel fundamental nos sistemas de sexta geração de comunicação sem fio (6G). Neste modo de uso da rede, milhares de dispositivos disputam os recursos de acesso disponíveis para o envio de pacotes, como por exemplo uma rede de sensores no campo ou uma fábrica automatizada da indústria 4.0. Nesse cenário, surge o problema de acesso aleatório, no qual os dispositivos selecionam aleatoriamente os recursos da rede e colisões ocorrem com frequência. Uma das formas promissoras de resolver esse problema é utilizar algoritmos de aprendizado por reforço QL (*Q-learning*). Neste trabalho, algumas técnicas presentes na literatura como os algoritmos QL-Independente e QL-Colaborativo são analisadas em termos de vazão e latência. Também são propostos um algoritmo distribuído baseado em pacotes de baixa complexidade, bem como melhorias na técnica QL-Colaborativa. Finalmente, analisa-se a disparidade de potência entre dispositivos em uma célula com a utilização de acesso múltiplo não ortogonal (NOMA - *non-orthogonal multiple access*) com um nó central aplicando o cancelamento sucessivo de interferência (SIC - *successive interference cancellation*) para reduzir a probabilidade de colisões. Neste contexto, é proposto um algoritmo QL com múltiplos níveis de potência capaz de aumentar a vazão enquanto reduz a latência de redes NOMA mMTC.

Palavras-chave: mMTC, Q-learning, acesso aleatório, NOMA, latência, cancelamento sucessivo de interferência.

List of Acronyms

5G	5th generation of wireless communications
6G	6th generation of wireless communications
AAoI	average age of information
AI	artificial intelligence
AP	access point
App	application
AWGN	additive white Gaussian noise
BS	base station
CSI	channel state information
D2D	device-to-device
DL	downlink
EB	exabytes
eMBB	enhanced mobile broadband
GB	gigabytes
Gbps	gigabytes per second
HD	high definition
HTC	human-type communication
i.i.d.	independent and identically distributed
IoT	internet of things
Kbps	kilobytes per second
KPI	key performance indicator
LAS	likelihood ascent search
LEO	low-earth orbit
LPWAN	low-power wide-area network

Mbps	megabytes per second
MDP	Markov decision process
ML	machine learning
mMIMO	massive multiple-input multiple-output
mMTC	massive machine-type communication
MPL	multi-power level
Msg	message
mULC	massive ultra-reliable low-latency communication
NOMA	non-orthogonal multiple access
PSD	power spectral density
QoS	quality of service
QL	Q-learning
RA	random access
RAR	random access response
RL	reinforcement learning
RRC	radio resource control
SINR	signal-to-interference-plus-noise ratio
SIC	successive interference cancellation
SMS	short message service
SUCRe	strongest-user collision resolution
Tbps	terabytes per second
UAV	unmanned aerial vehicle
UL	uplink
ULBC	ultra-reliable low-latency broadband communication
uMBB	ubiquitous mobile broadband
URLLC	ultra-reliable and low-latency communication

UT	user terminal
V2X	vehicle-to-everything
XL-MIMO	extra-large massive multiple-input multiple-output

List of Notations

$x \approx a$	x is approximately equals to a
$x >> a$	x is much greater than a
$x \in \mathcal{A}$	x belongs to set $\mathcal{A}$
$x \to a$	x tends to a
$x \leftarrow a$	assigns the value a to the variable x
$\mathcal{A} = \{1, \ldots, n\}$	set $\mathcal{A}$ contains all natural numbers between 1 and n
$\forall x \in \mathcal{A}$	for all x that belongs to set A
$\|\mathcal{A}\|$	cardinality of the set $\mathcal{A}$
$x \sim \mathcal{CN}(\mu, \sigma^2)$	x is a complex Gaussian random variable with mean μ and variance σ^2
$x \sim \mathcal{U}(a, b)$	x is a continuous uniform random variable on the interval (a, b)
$\max\limits_{i}(x_i)$	max value of x in i-th dimension
$\log_b(x)$	logarithm of x to base b
$\sum_{i=1}^{N}(x_i)$	summation of all x_i from $i = 1$ to $i = N$
$\lim_{x \to \infty}(f(x))$	limit of $f(x)$ when x tends to ∞
$\mathcal{M}_b\{x\}$	quantization of x using b bits
$\mathbb{P}(a\|b)$	conditional probability of a given b

List of Symbols

N_{reps}	number of Monte-Carlo realizations
N	number of machine-type devices;
K	number of time-slots;
L	number of transmitting packets;
p	number of payload bits for packet transmission on uplink;
b	number of header bits for reward on downlink;
κ_n	time-slot selected for the n-th device
$\mathcal{L}$	loading factor
$\mathcal{N}$	set of the device indexes
$\mathcal{K}$	set of the time-slot indexes
$\mathcal{C}_n$	set of the time-slots with maximum Q-values
ψ_k	set of interfering devices at k-th time-slot;
$Q_{n,k,p}$	Q-value from n-th device at k-th time-slot transmitting with the p-th power;
$R_{n,k,p}$	reward sent to n-th device at k-th time-slot for the p-th transmission power;
λ	action policy
α	learning rate;
C_k	congestion level at k-th time-slot;
ℓ_n	number of remaining packets of the n-th device
ν_n	convergence factor for n-th device
S	number of successful transmissions
T	number of time-slots spent at convergence
δ	number of frames until convergence (latency)
$\mathcal{T}$	normalized throughput
$\mathcal{T}_\infty$	asymptotic throughput
y_k	received signal at k-th time-slot
$\tilde{x}_n$	transmitted signal by n-th device
$x_{n,k}$	attenuated transmitted signal by n-th device at k-th time-slot
$h_{n,k}$	Rayleigh fading channel between n-th device and central node at k-th time-slot
w_k	AWGN sample in the time-slot

σ_w^2	AWGN noise power
N_0	noise PSD
P_t	transmitted power
$P_{\max}$	maximum transmitted power
$V_{\max}$	maximum transmitted signal voltage
$P_{n,k}$	received power from n-th device at k-th time-slot
$\bar{P}_n$	mean power of n-th device modeled by log-distance path loss
$\bar{P}_{d_0}$	mean power at reference distance
$\mathcal{P}$	number of transmission power levels
η	path loss exponent
d_n	distance between n-th device and central node
d_0	reference distance
r	cell radius
c	speed of light
π	pi constant
f_c	central frequency
B	bandwidth
$\gamma_{n,k}^{\text{NOMA}}$	instantaneous received SINR from n-th device at k-th time-slot
$\tilde{\gamma}_{n,k}^{\text{NOMA}}$	instantaneous received SINR from n-th device at k-th time-slot after imperfect SIC
$\bar{\gamma}$	SINR threshold
β	SIC error factor
τ	spectral efficiency

List of Figures

List of Tables

Contents

1 Introduction

1.1 Motivation

1.1.1 5G and beyond

The rapid advancement in the fields of artificial intelligence (AI) and internet of things (IoT) has caused the amount of data generated by various devices on the wireless network to increase exponentially. It was determined that in the sixth generation of wireless communication (6G) a rate of 1 Tbps should be reached, and the traffic volume is estimated to be in the order of 250 GB/month per subscription, so that the global traffic volume reaches 5000 EB/month (CHOWDHURY et al., 2020).

All this data will be used and shared by a large interconnected ecosystem of services such as ultra smart cities, multi-dimensional reality, haptic communication, telemedicine and tactile internet (BHAT; ALQAHTANI, 2021). Some of the technologies that will enable all these services to operate in the same architecture are: massive multiple-input-multiple-output (mMIMO), device-to-device communication (D2D), cell-free and vehicle-to-everything networks (V2X).

Fig. 1.1 shows the evolution of wireless communication generations, indicating what are the predictions of what will be implemented in 6G. The use of massive machine-type communication (mMTC) will allow for ubiquitous communication with a massive network of IoT devices to provide advances in healthcare, industrial machinery and transport and logistics services.

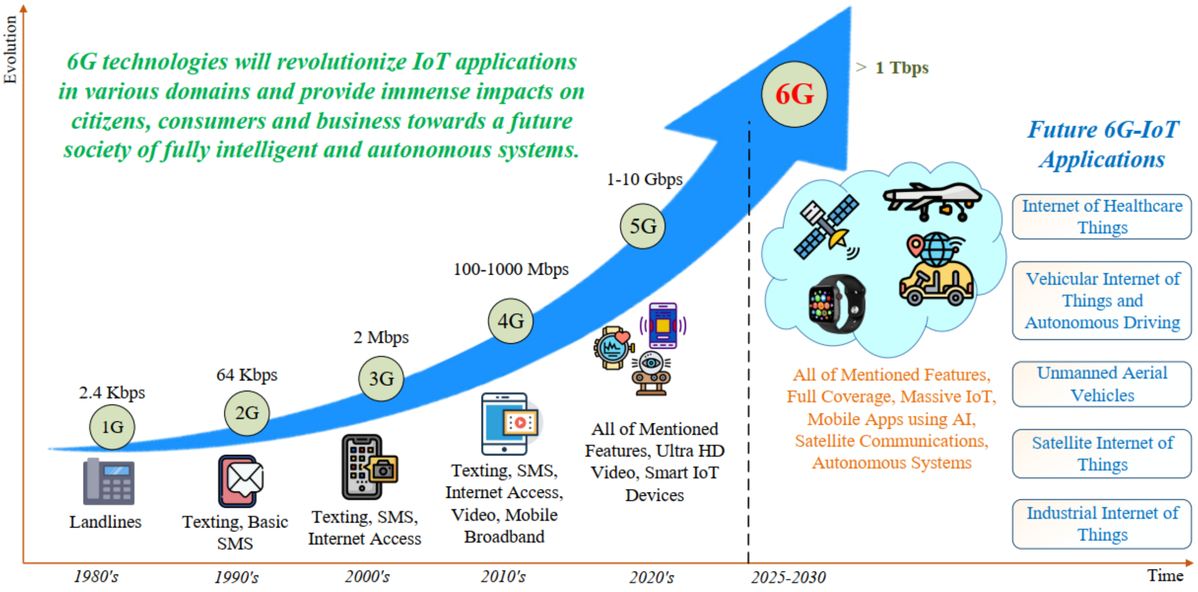

Evolution
6G technologies will revolutionize IoT applications in various domains and provide immense impacts on citizens, consumers and business towards a future society of fully intelligent and autonomous systems.
> 1 Tbps
6G
1-10 Gbps
5G
100-1000 Mbps
4G
2 Mbps
3G
64 Kbps
2G
2.4 Kbps
1G
Landlines
Texting, Basic SMS
Texting, SMS, Internet Access
Texting, SMS, Internet Access, Video, Mobile Broadband
All of Mentioned Features, Ultra HD Video, Smart IoT Devices
All of Mentioned Features, Full Coverage, Massive IoT, Mobile Apps using AI, Satellite Communications, Autonomous Systems
Future 6G-IoT Applications
Internet of Healthcare Things
Vehicular Internet of Things and Autonomous Driving
Unmanned Aerial Vehicles
Satellite Internet of Things
Industrial Internet of Things
1980's
1990's
2000's
2010's
2020's
2025-2030
Time

Figure 1.1 – The evolution of wireless networks (NGUYEN et al., 2021).

The aim of the next generations of wireless communication is to make the network increasingly dense, as the large number of devices will be able to provide massive interconnectivity (LEE et al., 2021).

1.1.2 6G services

The current wireless communications scenario encompasses several types of services operating concurrently, such as thousands of sensors sporadically sending readings in short packets and mobile users watching high-definition video streams. Due to the difference between the use of resources by these services, the proposal of the fifth generation of wireless communications (5G) was to divide into three main types of services (POPOVSKI et al., 2018): enhanced mobile broadband (eMBB), ultra-reliable and low-latency communication (URLLC), and mMTC.

In eMBB mode, broadband is used to provide high data rates to users, which guarantees access to streaming services, for example. In the URLLC mode, the focus goes out of high rates and goes to the reliability of receiving the service 99.999% of the time with a reduced latency to less than 1 millisecond. This type of service is indispensable in applications that involve risks such as remote surgeries and autonomous vehicles. Finally, the mMTC mode, which is the focus of this work, allows a high connectivity of devices accessing the network sporadically to send short blocks of data. Examples of this type of application are a network of devices on the farm obtaining data from sensors and agricultural machinery.

Although these services are already being implemented in the tests with 5G around the world, there are already some works in the literature that use these cited services to design the new services that will guide the scientific discussions until the implementation of the 6G, foreseen for 2030. In (JIANG et al., 2021), the intersections between the three 5G services are discussed and three more are created: ultra-reliable low-latency broadband communication (ULBC), ubiquitous mobile broadband (uMBB), and massive ultra-reliable low-latency communication (mULC), as shown in Fig. 1.2.

ULBC combines the high data rate of eMBB with the low latency of URLLC to enable features such as human-type communication (HTC) in immersive gaming, where human movements are mapped to insert the player into the game's graphics. Similarly, uMBB combines the high availability of data at high rates of eMBB with the high connectivity of mMTC devices to ensure ubiquitous communication, allowing machine learning techniques on devices using data present on the Internet. Finally, mULC relies on high device density and low latency to enable services such as the use of risk actuators in the vertical integration of industries 4.0.

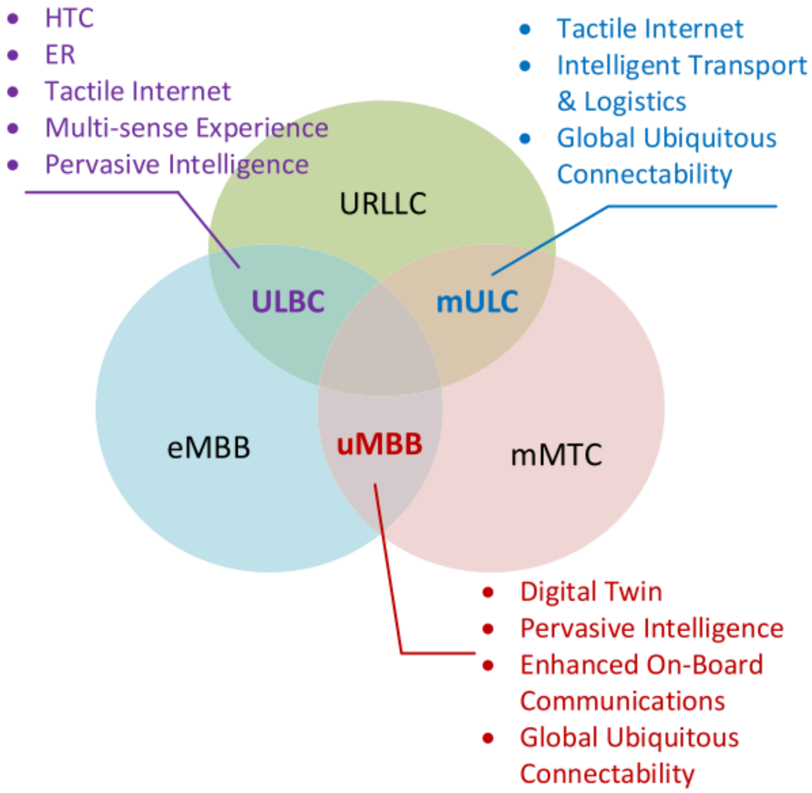


Figure 1.2 – 6G services (JIANG et al., 2021).

1.1.3 Massive machine-type communications

The concept of a machine-type device is to rely on little or no human interaction during its entire operation (BUI et al., 2019). In 5G systems and beyond, these devices will play an important role in services such as unmanned aerial vehicles (UAV) and self-driving cars. Key performance indicators (KPI) for mMTC networks are: enable massive connectivity of 10^6 devices per square kilometer in urban areas and ensure a 10-year battery lifetime (POKHREL et al., 2020).

One of the main characteristics of these networks is that devices use resources randomly and sporadically. It is common to use the concepts of active and inactive devices. We say that a device is active when it uses network resources to transmit data packets or preambles at a given instant of time. Fig. 1.3 shows a typical mMTC network that contains both active and inactive devices.

One of the main requirements for devices in a mMTC network is that they are energy-efficient (ULLAH et al., 2021). As human interaction with the devices can be null, then repair processes must be done autonomously. Therefore, it is extremely important that the devices use low-complexity algorithms so that an optimization of battery life is possible.

Examples of devices that need to be energy-efficient are a remote sensor in a farm communicating over low-power wide-area network (LPWAN) and a marine weather station

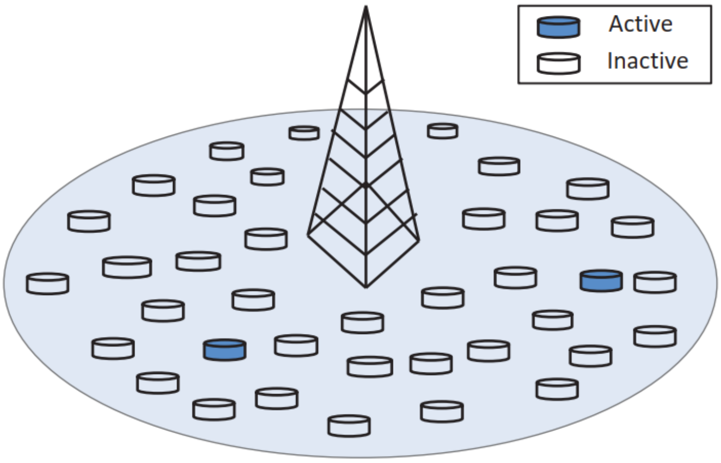


Figure 1.3 – mMTC network with active and inactive devices (ALAM; ZHANG, 2018).

sending data via low-earth orbit satellites (LEO). As these devices can be in hard-to-reach places, the battery should last long enough so that no human interaction is required.

1.1.4 Random access in mMTC

The focus of this work is on the study of the effects of the high density of mMTC devices, and the analysis can be extended with the same validity for future mULC and uMBB scenarios. In mMTC mode, there are thousands of devices that send short data packets sporadically to a central node or base station (BS), which we call central node throughout this work. This sporadic access causes the problem of random access (RA), in which two or more devices can randomly select the same network resources to transmit their data packets, resulting in a collision. This problem is intensified when the density of devices increases and when resources are limited in the system, such as a low number of pilot sequences, intensive reuse of narrow frequency bands or time-slot restriction.

Figure 1.4 shows an example of a random access protocol based on preamble selection. Before sending payload data, the device needs to send the preamble and get an acknowledgment response from the BS. After that, the device sends a radio resource control based on the previous preamble setting. As the orthogonality feature of this protocol is the use of orthogonal preambles, so if two or more devices select the same preamble, a collision occurs. It is worth noting that this protocol is grant-based, as it depends on the permission of the BS to guarantee the device's access to the network.

Collisions can also occur in resource blocks based on time and frequency domain, as in the example in Figure 1.5 that shows the mMTC service intermixed with the URLLC; the latter is represented by the virtual reality users. When the two services select the same resource block, a collision is characterized.

There are several ways to solve the collision problem. The simplest ones are those based on ALOHA, where the central node asks the devices to retransmit their packets. In

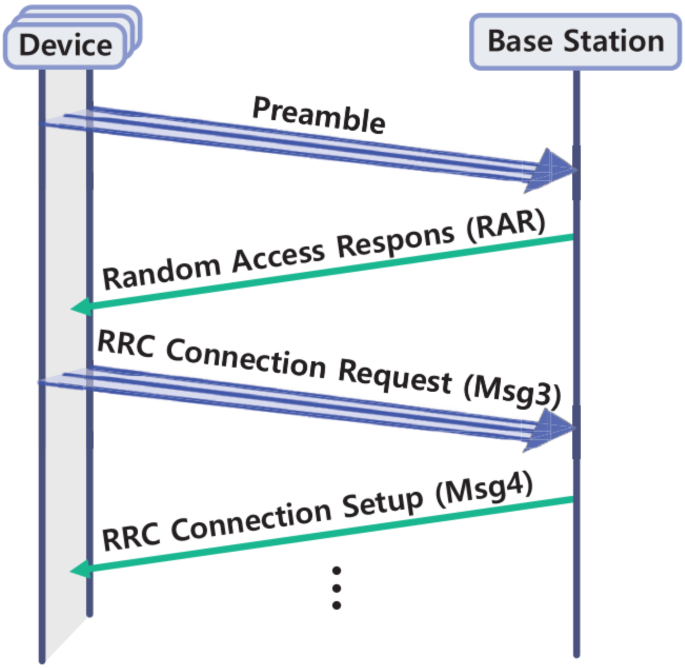

Device
Base Station
Preamble
Random Access Respons (RAR)
RRC Connection Request (Msg3)
RRC Connection Setup (Msg4)
⋮

Figure 1.4 – An example of RA protocol (PIAO et al., 2021).

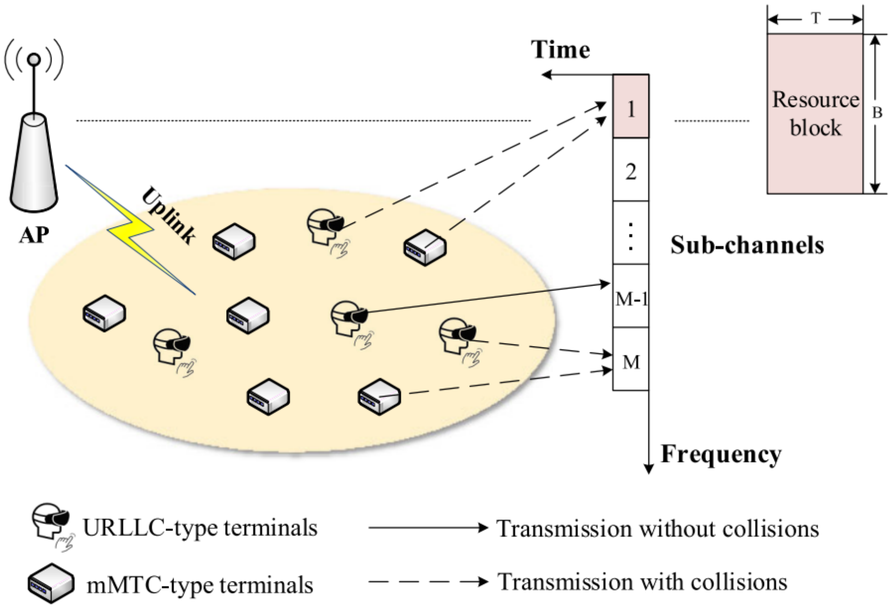


Figure 1.5 – URLLC and mMTC collisions (QI et al., 2020).

retransmission step, the chance of the colliding devices to select the same resources again is reduced. Although ALOHA-based techniques are not very complex to be implemented in machine-type devices, transmission latency is greatly increased. Strongest-user collision resolution (SUCRe), for example, is a method that uses properties of mMIMO (BJÖRNSON et al., 2017) and extra-large mMIMO (XL-MIMO) (NISHIMURA et al., 2020) for collision resolution in a distributed manner, with performance superior to ALOHA. In (ZHANG et al., 2019), it is proposed a feedback-free solution for decoding among asynchronous transmitters, by jointly employ *physical layer network coding* to deal with collisions at the receiver side and protocol sequences to define an *medium access control* (MAC) scheme at the transmitter side. Authors compare the proposed method in terms of detecting delay and the energy consumption with two schemes, one is a TDMA-based scheme and other is a Fountain code-based scheme to solve asynchronous transmitter collisions. Fountain-based scheme generates higher energy consumption cost when the number of transmitters grows.

Another alternative is the use of reinforcement learning (RL) techniques, in which devices can learn what are the best resources available in the network for transmitting their packets based on rewards sent by the central node. In RL-based techniques, the devices of the mMTC network forcibly learn which policy causes the maximization of the obtained rewards. Therefore, there is no dataset that is pre-provided to devices. When considering RL, devices can use grant-free protocols, which reduce communication latency with the central node since it is not necessary to ask permission to access network resources. Unlike the techniques proposed in (ZHANG et al., 2019) and (NISHIMURA et al., 2020), RL algorithms do not resolve collisions, but only make devices retransmit packets that collided in another resource block. With this, the device can store which were the resource blocks used that provide the greatest probability of success, in order to decrease the total latency, which is desired in mMTC networks.

1.1.5 Reinforcement learning and Q-learning

With the increase in data rates and the amount of information available on the networks, machine learning (ML) techniques emerged in 5G systems for process automation based on previous experience. So far, perspectives for 6G systems follow the same trend. Many algorithms have already been proposed to guarantee the strict requirements of reliability and latency.

In mMTC mode, the application of RL is more common. The RL is a variant of ML in which there is an interaction between an agent who performs an action and receives a reward from the environment, as shown in Fig. 1.6. The agent's goal is to maximize the reward obtained. The RL is based on the Markov decision process (MDP). The MDP is defined by a set of states, a set of actions, a probability of transition from states and a reward received after the transition. Markov's property is respected because the probabilities of future states only depend on the current state (SUTTON; BARTO, 2018).

Fig. 1.7 shows an illustration of an agent's state transition in an MDP. At time t, the agent is in a state s_t. It can perform action a_t and move to state s_{t+1} with probability $\mathbb{P}(s_{t+1}|s_t, a_t)$. This process generates the reward R_{t+1} with probability $\mathbb{P}(R_{t+1}|s_t, a_t)$. Various real-life problems can be modeled as an MDP. Two examples are the modeling of a

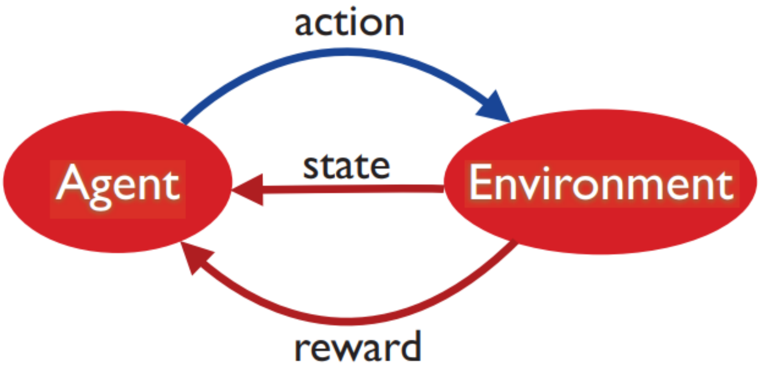


Figure 1.6 – RL model (MOHRI et al., 2018).

queue of customers seeking a service in a commercial store and the organization of inflows and outflows from a company's stock (WHITE, 1993).

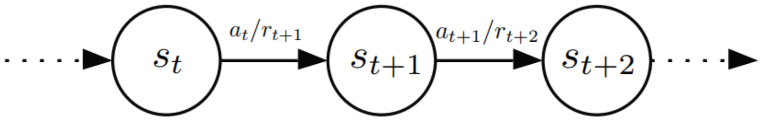


Figure 1.7 – MDP transitions (MOHRI et al., 2018).

The agent needs to find the action policy λ that will guide it to take the best actions in each state. The goal will always be to maximize the expected reward. Within the set of possible actions A, Bellman's optimality condition (MOHRI et al., 2018) guarantees that an action policy is optimal when:

$$a \in \arg\max_{a' \in A} Q_\lambda(s, a'). \tag{1.1}$$

$Q_\lambda(s, a)$ is called the state-action value function Q, or simply Q-function, and is defined as the expected return from taking an action a in state s following the policy λ. As there is a deterministic optimal policy for every MDP, then there is an optimal maximum Q-value for the Q-function.

Within the various existing RL techniques, in this work we focus on the Q-learning (QL) methodology, a RL-based algorithm. In QL algorithms, it is not necessary for the agent to know the policy λ and the probability models $\mathbb{P}(s_{t+1}|s_t, a_t)$ and $\mathbb{P}(R_{t+1}|s_t, a_t)$ to estimate the state transitions after performing actions. An initial Q-value is set for all possible state-action pairs and, based on trial and error, the agent performs actions, obtains rewards from the environment and updates the Q-values. After exploring the data obtained from the environment for a while, the device starts using the data to make clearer decisions, based on an exploration-exploitation philosophy.

Because the QL is a model-free learning method, it is feasible to implement it on machine-type devices due to its low complexity. In mMTC mode, the agents are the devices and the environment is the BS or the central node. The action that the devices take is to send their data packets in the uplink, and the reward is the response of the central node in the downlink, providing information about the result of the transmission.

Based on the rewards received after each transmission attempt, the devices learn what are the best network resources to be used to avoid collision, reducing the total latency of the packet transmission. In (SHARMA; WANG, 2019), devices randomly select a time-slot to transmit and the reward indicates which are the least congested time-slots resources. In (SILVA et al., 2020), devices transmit at different powers (power domain resource) in a non-orthogonal multiple access (NOMA) scenario. The QL algorithm indicates to the devices what is the best power to transmit in order to avoid collision.

It is known that the Q-learning algorithm converges when state-action pairs are visited infinitely many times (MOHRI et al., 2018). In practice, it is not necessary to wait for complete convergence of the algorithm, as the number of packets that devices need to transmit is finite and the access to network resources is random and sporadic.

When devices are mobile, they can enter and leave the system randomly, so it is possible to estimate the sending of packets with a traffic model, as used in (BUI et al., 2019) and (WEERASINGHE et al., 2021). In this scenario, collisions eventually occur. As a result, packets can arrive late at the central node. An important metric to assess the freshness of received packets is the average age of information (AAoI), as evaluated in works (SAHA et al., 2021) and (YU et al., 2021). The goal is to minimize the AAoI choosing properly the projected system parameters and the algorithms used.

The study of QL in this scenario is very promising because the more information the the central node is able to include in the reward, the better the learning of the devices will be, thus being able to increase throughput or reduce latency to guarantee the requirements of 6G. Some solutions already exist in the literature for 5G systems, however it is challenging to ensure ubiquitous communication in 6G when increasing the number of devices. The scenario becomes even more challenging when the successive interference cancellation (SIC) at the receiver is imperfect. The impact of more realistic scenarios on RA problem deploying QL algorithms has not yet been discussed in the literature.

1.2 Objectives

1.2.1 General

The general objective of this work is to propose improvements and evaluate the performance of promising QL algorithms that mitigate the deleterious effects of the RA problem in mMTC networks with strict throughput and latency requirements in 5G/6G system scenarios.

1.2.2 Specific

- To propose improvements in the protocol model of the QL algorithm and discuss the steps from the initialization of the necessary parameters until reaching the convergence;
- Numerically evaluate the throughput and latency in terms of time-slots of the main QL algorithms in different scenarios of system loading factor, learning rates, and number of transmitted packets;
- Include a physical layer model with power disparity between the devices to create a NOMA scenario. In this case, we consider SIC in the detection process and we discuss the problems caused by the imperfect SIC process.

1.3 Organization

This work is organized as follows: in Chapter 2, the general wireless communication system model is presented in connection with the RA protocol level description; representative QL algorithms available in the literature are introduced, and a new QL-based RA algorithm is proposed. In Chapter 3, a physical layer model is described considering NOMA transmission scheme and a multi-power level QL algorithm is developed. Chapter 4 points out the main conclusions of this work, as well as the possibilities for future work. The papers generated during the development of the work are presented in Appendices A, B and C. Besides, in Appendix D one can find a base of the Python source code deployed to generate the numerical simulations of our QL-based RA NOMA algorithms.

1.4 Contributions

The main contributions of this work are:

c.1. numerical evaluation of the throughput and latency of different QL-based RA NOMA algorithms available in the literature;

c.2. improvement in the collaborative QL algorithm for RA discussed in (SHARMA; WANG, 2019), aiming to quantify expeditiously the reward value and send a smaller number of bits;

c.3. proposition of a distributed QL RA algorithm whose reward is based on the remainder number of packets to be sent from each device;

C.4. analysis of the impacts on throughput and latency when considering imperfect SIC or even lack of information in the NOMA system model based on the previous literature results in (SILVA et al., 2020);

C.5. proposition of a QL NOMA algorithm that uses different transmission power levels to increase the maximum number of devices that the mMTC system is able to support.

During the development of this work, three full papers were devised, either submitted or published.

1. G. M. F. Silva and T. Abrão. **Throughput and Latency in the Distributed Q-Learning Random Access mMTC Networks**. Accepted on January 10th, 2022 in *Computer Networks*, Elsevier, IF = 4.474 (2020), and reproduced in Appendix B.

2. G. M. F. Silva and T. Abrão. **Multi-Power Level Q-Learning Algorithm for Random Access in NOMA mMTC Systems**. Submitted to *Transactions on Emerging Telecommunications Technologies*, Wiley, IF = 2.638 (2020), and reproduced in Appendix C.

3. G. M. F. Silva, J. C. Marinello F. and T. Abrão. **Adjustable Threshold LAS Massive MIMO Detection Under Imperfect CSI and Spatial Correlation**. Published in *Physical Communication*, Elsevier, IF = 1.810 (2020), vol 38, p. 100971, Feb. 2020. DOI:⟨10.1016/j.phycom.2019.100971⟩, and reproduced in Appendix A. This work is indirectly related with the theme of the dissertation.

2 QL-based random access protocol for mMTC

2.1 Introduction

One of the challenges of 5G systems and beyond is managing the number of devices that use wireless network resources. As many devices are foreseen in smart cities, smart agricultures and industry 4.0 networks, then techniques capable of serving all devices with the highest reliability and lowest latency possible are needed to ensure the necessary quality of service for AI, UAV, D2D, among others. Therefore, it is very important to evaluate figures of merit such as throughput and latency in mMTC systems to compare different techniques and algorithms and assess which ones are most promising for meeting the strict requirements of 5G and beyond.

In mMTC systems with thousands of devices randomly accessing network resources, the implemented algorithms must result in low-complexity, as they are used in machine-type devices with low-processing power; also, such techniques must be able to manage the available resources in such a way that the devices complete their packets transmission with the lowest possible probability of collision.

In this chapter, QL algorithms are presented as a good low-complexity solution for random access mMTC scenarios, as they are model-free learning techniques compared to other ML algorithms. In this chapter, we propose and analyse a distributed packet QL-based algorithm and perform a comparison with two representative QL-based RA methods, the QL-Independent and QL-Collaborative algorithms discussed recently in (SHARMA; WANG, 2019). Our proposed QL-based solution presents a good performance-complexity trade-off in the analyzed mMTC scenarios, with high throughput and low latency.

The mMTC system model at the protocol layer is presented in Section 2.2. The independent and collaborative QL algorithms and the proposed distributed packet-based are presented and discussed in Section 2.3. The numerical results with throughput and latency analyses of QL algorithms are presented in Section 2.4. Finally, the conclusions and findings of this chapter are presented in Section 2.5.

2.2 System model

The system contains N active mMTC devices covered by a central node. The set contains the ordered device indexes. The devices send data packets to the central node on the uplink, and the central node sends rewards to the devices on the downlink, as the example shown in Fig. 2.1.

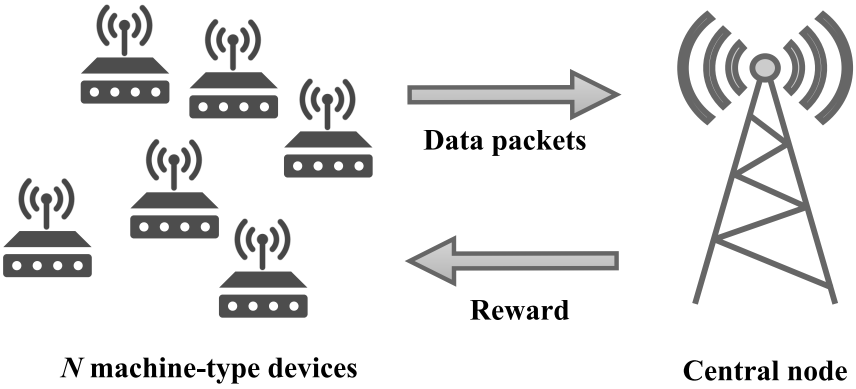


Figure 2.1 – System model.

Initially, a simplified model is considered where all devices transmit on the same frequency, but select different time-slots to transmit. The frame consists of K time-slots for sending uplink packets and a smaller slot for downlink, as shown in Fig. 2.2. By reason of simplification, it is considered that the downlink time-slot is much smaller than the uplink time-slots, approximating the total frame size to K time-slots. The central node performs an adequate power allocation in such a way that it is able to receive the signal and detect the symbols of all devices with the same power. The time spent on training and power allocation does not influence the calculation of total latency. This simplification is considered to focus on performance analysis only at the protocol level. A more realistic model for the physical layer is used in Chapter 3.

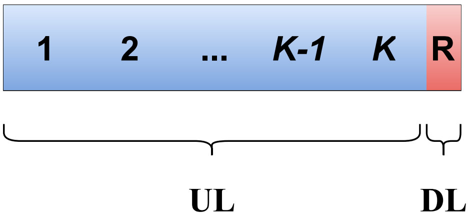


Figure 2.2 – Division of the frame in uplink and downlink time-slots.

Each device has L packets to transmit and only one packet is transmitted per time-slot. At the end of the uplink period, the central node performs a broadcast containing a reward for each device, indicating whether the transmission was successful or not. Success in this scenario is defined as just one device transmitting in a time-slot.

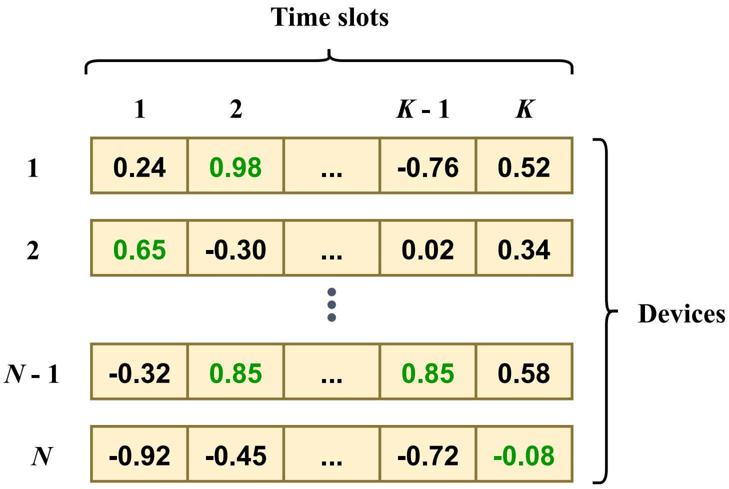
When two or more devices send their packets in the same time-slot, we consider that there was a collision and the transmission failed. If the transmission is successful, the device transmits its next packet. In case of failure, the same packet is retransmitted in the next frame for the interfering devices. We define the set ψ_k as the set of devices that selected the k-th time-slot. For example, if the 3rd and the 5th devices selects the second time-slot, then $\psi_2 = \{3, 5\}$.

The focus is to use QL algorithms so that the devices learn which are the best time-slots to transmit based on the experience of rewards received by the central node.

2.3 QL algorithms

Each device contains a table called Q-table that stores the rewards given the feedback of the central node. The Q-table for each device has K entries, and the value of each entry is called the Q-value. The goal of devices is to choose the best time-slot to transmit packets to reduce collisions.

We can join all the Q-tables $K \times 1$ to form an $N \times K$ matrix containing the tables for all devices in the system. The Q-value $Q_{n,k}$ indicates the preference of the n-th device to transmit in the k-th time-slot. The time-slot selected for transmission κ_n will always be the index of Q-table whose Q-value is maximum. If two or more values are equal to the maximum value, then the selection between these values is random. Fig. 2.3 shows an example with the maximum Q-values highlighted.


Time slots
Devices

	1	2		*K* - 1	*K*
1	0.24	0.98	...	-0.76	0.52
2	0.65	-0.30	...	0.02	0.34
⋮					
N - 1	-0.32	0.85	...	0.85	0.58
N	-0.92	-0.45	...	-0.72	-0.08

Figure 2.3 – Example of an $N \times K$ matrix containing the Q-tables of N devices. The maximum Q-values are in green color.

The Q-table can be initialized in several ways. For the scenario described in this chapter, the Q-table is initialized with all positions equal to zero. Each device updates the Q-value of the chosen time-slot at the end of the frame. The Q-value update from frame t to frame $t+1$ is defined by

$$Q_{n,k}^{(t+1)} = (1-\alpha)Q_{n,k}^{(t)} + \alpha R_{n,k}^{(t)} \tag{2.1}$$

where $R_{n,k}$ is the reward sent by the central node, and $\alpha \in [0,1]$ is the learning rate, which is the weight given by the devices for each reward received.

The QL algorithm converges in two different scenarios: when the N devices transmit all their L packets; or when the N devices find a single time-slot to transmit. In this last scenario there will be no more collisions and it will no longer be necessary to update the algorithm.

The QL algorithms differ in the way they send the reward to the devices, which can contain different information about the system. Some algorithms can transmit a larger number of header bits b in the central node, in relation to the number of payload bits p of the devices. These numbers have an impact on the figures of merit analyzed: throughput and latency. To measure the performance at convergence, we consider total latency, which is the total number of time-slots T required for convergence, and normalized throughput, defined as

$$\mathcal{T} = \left(\frac{p}{b+p}\right)\frac{S}{T} = \left(\frac{p}{b+p}\right)\frac{NL}{T}. \tag{2.2}$$

where S is the total number of successful transmissions. The loading factor $\mathcal{L} = \frac{N}{K}$ will be used to measure performance.

2.3.1 Independent QL

The simplest QL algorithm is the independent one, where the reward sent to the n-th device at the k-th time-slot is

$$R_{n,k}^{\text{IND}} = \begin{cases} +1, & \text{if transmission succeeds,} \\ -1, & \text{otherwise.} \end{cases} \tag{2.3}$$

The complexity is very low because only one header bit is sent ($b = 1$). However, as it does not include any additional information about the state of the system, the performance is also low, as it is shown in the numerical results in Section 2.4.

2.3.2 Collaborative QL

One of the possibilities of including information about the k-th time-slot in the reward is to measure the congestion level. In (SHARMA; WANG, 2019), it is defined that the congestion level can be calculated as

$$C_k = \frac{|\psi_k|}{N}. \tag{2.4}$$

$C_k \rightarrow 0$ when few devices have selected the time-slot, and $C_k \rightarrow 1$ in congested time-slots. As the calculated value is a real number, it is necessary to quantize it in b header bits to perform a fair performance comparison with the other QL algorithms. Therefore, the reward sent by the central node using this algorithm is

$$R_{n,k}^{\text{COL}} = \begin{cases} +1, & \text{if transmission succeeds,} \\ -\mathcal{M}_b\{C_k\}, & \text{otherwise,} \end{cases} \tag{2.5}$$

where $\mathcal{M}_b\{C_k\}$ is the quantized value of C_k using b bits, e.g., if $b = 2$ bits and assuming that the level of congestion varies from 0 to 1, then the reward values can be unambiguously represented by four quantized levels: $\mathcal{M}_b\{C_k\} \in \{0.25, 0.5, 0.75, 1\}$, regardless of the number of devices.

The use of a high number of header bits b impacts both the complexity and the throughput of the QL algorithm, since the ratio between payload bits p of the devices and header bits b in the central node decreases with the increase of b.

2.3.3 Proposed distributed packet-based QL

With the increase in the number of devices and the increase in the probability of collision in crowded mMTC mode, it becomes more difficult for central node to identify the number of interfering devices. Therefore, the advantage of the collaborative Q-learning technique in regions with high density of devices depends on an ideal non-feasible scenario.

In addition, independent and collaborative QL algorithms are not completely fair, as a time-slot becomes unique for one device over the entire learning period, while the other devices continue to collide and expect to randomly find a suitable time-slot to finish transmitting all packets.

Therefore, in this subsection a QL algorithm is proposed that is based on the number of packets that each device still has to transmit in such a way that the devices that still have many pending packets have a greater reward in relation to those that have already transmitted more packets. Defining ℓ_n as the number of packets that the n-th device still has to transmit, then an ν factor is defined by

$$\nu_n = 1 - \frac{\ell_n}{L}. \tag{2.6}$$

$\nu_n \to 0$ when the device still has many packets to transmit, and $\nu_n \to 1$ when the device has already transmitted many packets. The reward model used in this algorithm is the same as the independent one:

$$R_{n,k}^{\text{PAC}} = R_{n,k}^{\text{IND}} = \begin{cases} +1, & \text{if transmission succeeds,} \\ -1, & \text{otherwise.} \end{cases} \tag{2.7}$$

Therefore, only one header bit $b = 1$ is used in the transmission. The ν_n factor does not need to be transmitted, as the device is aware of how many packets are still need to be transmitted. Hence, the update of the Q-value is carried out depending on the reward received:

$$Q_{n,k}^{t+1} = \begin{cases} Q_{n,k}^{t} + \alpha(R_{n,k}^{\text{PAC}} - Q_{n,k}^{t}), & \text{if Tx succeeds,} \\ Q_{n,k}^{t} + \alpha(\nu_n R_{n,k}^{\text{PAC}} - Q_{n,k}^{t}), & \text{otherwise.} \end{cases} \tag{2.8}$$

$$= \begin{cases} Q_{n,k}^{t} + \alpha(1 - Q_{n,k}^{t}), & \text{if Tx succeeds,} \\ Q_{n,k}^{t} - \alpha(\nu_n + Q_{n,k}^{t}), & \text{otherwise.} \end{cases} \tag{2.9}$$

We call it a distributed algorithm because the calculation of the reward that goes into updating the Q-value is transferred from the central node to the devices, based on the number of packets in which each device has yet to transmit. The pseudo-code of the operation of the proposed method is present in Algorithm 1. The algorithm does not rely on any recursion. We define $\mathcal{N} = \{1, 2, \ldots, n, \ldots, N-1, N\}$ as the set of device indexes, $\mathcal{K} = \{1, 2, \ldots, k, \ldots, K-1, K\}$ as the set of available time-slots, and $\mathcal{C}_n$ as the set for n-th device that contains the time-slots with maximum Q-values.

Algorithm 1 **Distributed Packet-Based RA**

```
Initialize $Q_{n,k} = 0,\ \forall n \in \mathcal{N},\ \forall k \in \mathcal{K}$
Initialize $\ell_n = L,\ \forall n \in \mathcal{N};\quad T = 0,\ S = 0$
while $\sum_{n=1}^{N} \ell_n > 0$ do
    Initialize $\kappa_n = 0,\ \forall n \in \mathcal{N}$
    for $n = 1 : N$ do
        if $\ell_n > 0$ then
            $\mathcal{C}_n = \{k \in \mathcal{K} \mid Q_{n,k} = \max_k\{Q_{n,k}\}\}$
            Select randomly: $\kappa_n \in \mathcal{C}_n$
    for $k = 1 : K$ do
        $T \leftarrow T + 1$
        $\psi_k = \{n \in \mathcal{N} \mid \kappa_n = k\}$
        if $|\psi_k| = 1$ then
            $S \leftarrow S + 1$
            $R^{\text{PAC}}_{n,k} = +1,\ \forall n \in \psi_k$
            $Q_{n,k} \leftarrow Q_{n,k} + \alpha(1 - Q_{n,k}),\ \forall n \in \psi_k$
            $\ell_n \leftarrow \ell_n - 1,\ \forall n \in \psi_k$
        else if $|\psi_k| > 1$ then
            $\nu_n = 1 - \frac{\ell_n}{L},\ \forall n \in \psi_k$
            $R^{\text{PAC}}_{n,k} = -1,\ \forall n \in \psi_k$
            $Q_{n,k} \leftarrow Q_{n,k} - \alpha(\nu_n + Q_{n,k}),\ \forall n \in \psi_k$
```

2.4 Numerical results

In this section, the algorithms described in Subsections 2.3.1, 2.3.2 and 2.3.3 are validated numerically using the Monte-Carlo simulation method using Python. Ten thousand realizations were considered in the Monte-Carlo simulation to obtain a good mean of the variables with random distribution. The numerical parameters used are present in Table 2.1. The figures of merit evaluated are the normalized throughput and the number of time-slots spent for the convergence of the algorithm.

2.4.1 Number of bits of collaborative QL

As discussed in Subsection 2.3.2, a scalar uniform quantization for the level of congestion is proposed in this work, because in (SHARMA; WANG, 2019) the effect of

Table 2.1 – Numerical parameters for protocol layer QL.

Parameter	**Value**
Monte-Carlo realizations	$N_{\text{reps}} = 10{,}000$
Time-slots per frame	$K = 400$
Network loading factor	$\mathcal{L} = \frac{N}{K} \in [0.25;\ 3.00]$
Packets per device	$L \in [50; 500]$
Learning rate	$\alpha \in [0.05; 0.5]$
Header bits (collab.)	$b \in [1; 2; 4; 8; 16]$ bits
Payload bits	$p \in [1; 2; 4; 8; \ldots; 256]$ bits

the number of bits of the collaborative algorithm in relation to the independent one was not discussed. Therefore, in this subsection, the effect that the quantization of the reward causes on throughput is evaluated, for a scenario with different numbers of quantization bits and different loading factors. The result for this analysis is shown in Fig. 2.4.

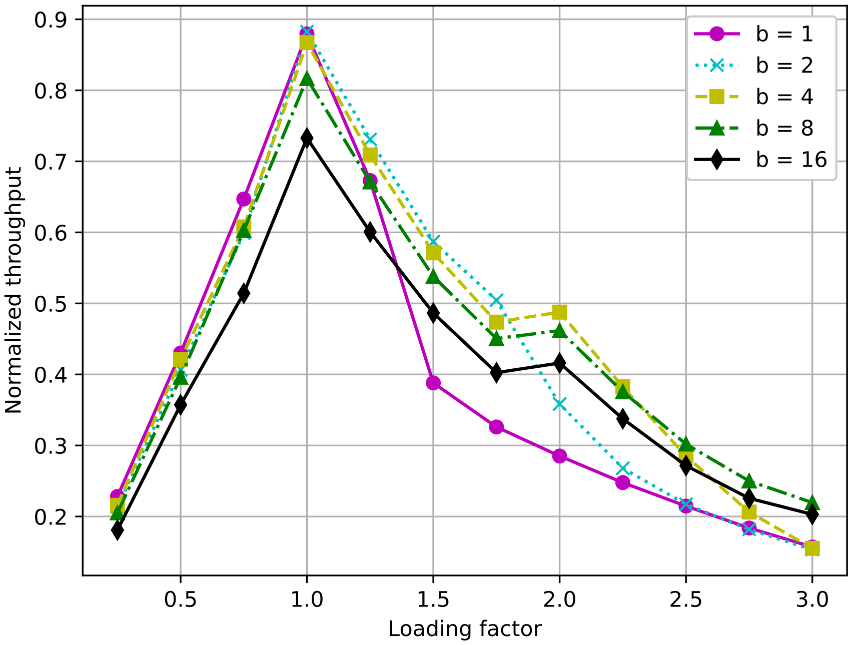


Figure 2.4 – Throughput for collaborative QL varying the loading factor, considering p = 64, L = 100, and α = 0.1.

When considering a very low number of bits, such as $b = 1$, it is observed that the throughput falls considerably in overloaded scenarios ($1 \leq \mathcal{L} \leq 3$). This is because in this scenario where there are more devices than time-slots, the evaluation of the congestion level becomes important to be include in the reward. However, with just one bit it is not possible to accurately calculate the congestion level, so throughput is reduced.

On the other hand, in the scenario where the number of bits is very high ($b = 16$), the throughput is low in scenarios where $\mathcal{L} \leq 1$, because a very large complexity is spent in scenarios where the congestion level naturally tends to zero. Hence, it is considered that $b = 4$ is a good number of bits to provide a good throughput for different loading scenarios. This value will be used in the results that are presented from that subsection.

2.4.2 Normalized throughput

The normalized throughput was evaluated for the three types of QL algorithms in scenarios with different loading factors. The result is present in Fig. 2.5. For the three techniques analyzed, the maximum throughput occurs when $\mathcal{L} = 1$, because in this scenario there is a perfect allocation of the number of time-slots required for the number of devices present. Throughput is lower for scenarios where $\mathcal{L} \neq 1$, because $\mathcal{L} < 1$ indicates that the system is overestimated and more devices could be used to take advantage of the system, as there are more time-slots than devices; $\mathcal{L} > 1$ underestimates the system, where there are many devices colliding and disputing the existing time-slots. We will focus on the discussion for scenarios where $\mathcal{L} > 1$ because it is a realistic and typical environment of crowded mMTC systems.

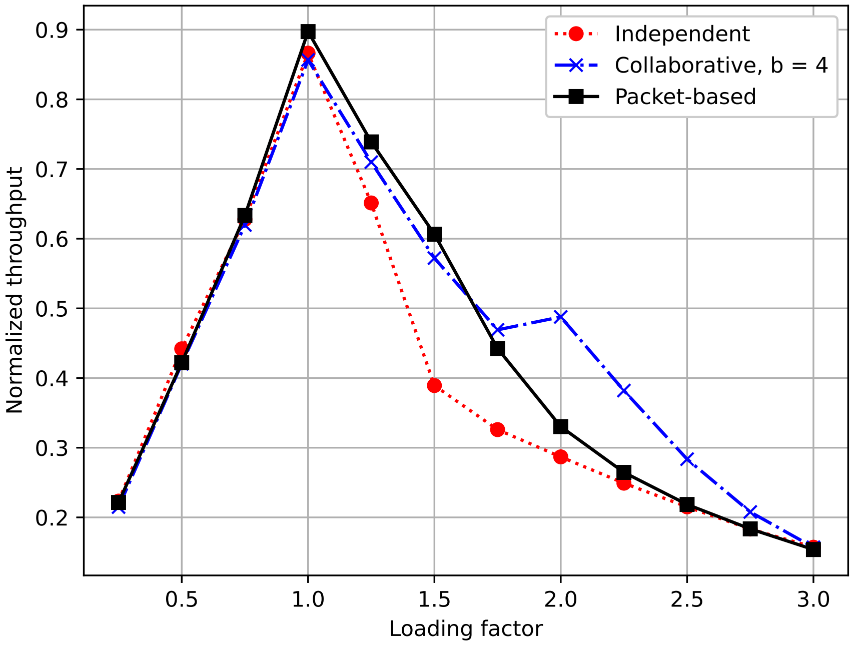


Figure 2.5 – Normalized throughput in function of loading factor for independent, collaborative, and packet-based Q-Learning, considering p = 64, L = 100, K = 400, and α = 0.1.

Among the analyzed algorithms, the independent one has the lowest throughput in crowded scenarios. As the reward is binary and simplified, so there is not much information that helps the process of learning the devices, causing the throughput to decrease. The collaborative algorithm has higher throughput because, as the congestion level is informed in the reward, the devices learn to transmit their packets in less congested time-slots, improving the throughput.

The performance of the packet-based algorithm is superior to the other techniques up to $\mathcal{L} = 1.6$. From that point on, the collaborative algorithm is the superior between

$1.6 \leq \mathcal{L} \leq 3.0$, and then all the algorithms converge to the same throughput value at $\mathcal{L} = 3.0$. It is expected that the collaborative algorithm will perform better in this scenario because the reward sent includes the level of congestion, which facilitates the devices to learning the best time-slots to transmit. However, the packet-based algorithm proved to be superior to the independent one and still presents a lower complexity in relation to the collaborative one, since fewer bits are used in the calculation of the reward, distributing the processing among the devices and facilitating the implementation of the central node, since it does not need to know the number of devices that collided in each time-slot.

2.4.3 Asymptotic throughput

In the previous results, the number of packets was fixed at $L = 100$ to analyze the effect of the loading factor on the system. However, through Eq. 2.2, it can be seen that throughput is also a function of the number of packets. Therefore, in this subsection, the effect of changing the number of packets on throughput was analyzed. Fig. 2.6 shows the result obtained when measuring throughput by changing the number of packets from $L = 50$ to $L = 500$.

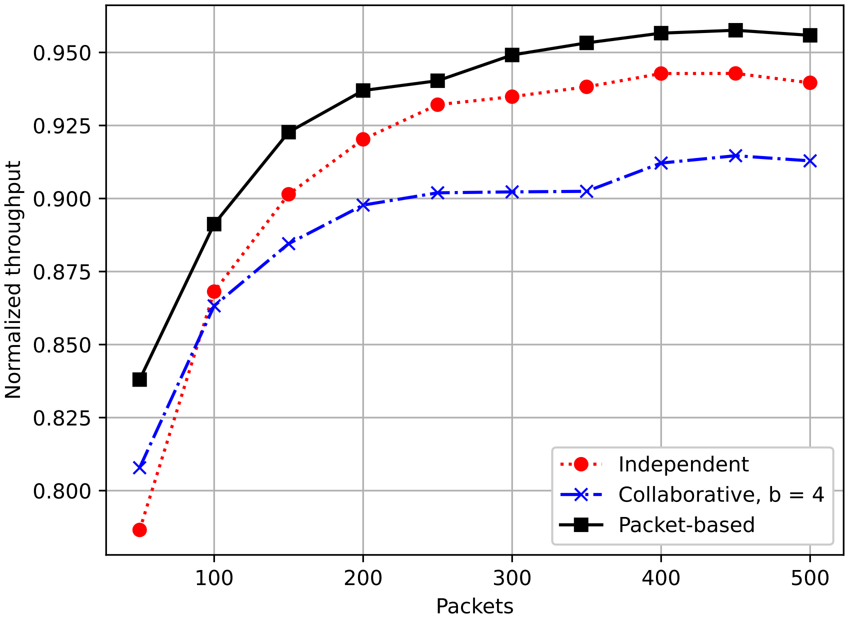


Figure 2.6 – Throughput as a function of the number of packets, considering loading factor $\mathcal{L} = 1$, $K = 400$ time-slots, $p = 64$ bits, and $\alpha = 0.1$.

Up to $L = 400$, the increase in the number of packets causes an increase in throughput, since the number of successes obtained increases at a greater rate than the latency to transmit them. However, for $L > 400$, it was observed that throughput converges to a constant value, since from that point on, transmitting more packets causes an increase in the number of time-slots in the same proportion. Therefore, we define asymptotic throughput as the value of throughput when the number of packets and time-slots tend to infinity:

$$\mathcal{T}_\infty(\mathcal{L}) = \lim_{L,T\to\infty}\left[\left(\frac{p}{b+p}\right)\frac{NL}{T}\right], \tag{2.10}$$

In the analyzed scenario, it is observed that: $\mathcal{T}_\infty^{\text{PAC}}(1) \approx 0.965$; $\mathcal{T}_\infty^{\text{IND}}(1) \approx 0.940$; $\mathcal{T}_\infty^{\text{COL}}(1) \approx 0.915$. The proposed packet-based algorithm presented the best asymptotic throughput in the result obtained.

2.4.4 Payload bits

In addition to analyzing the effect of the number of packets on throughput, we analyzed the effect of increasing the number of payload bits p. In Fig. 2.7, the result of the throughput changing the number of payload bits from $p = 1$ to $p = 256$ is shown.

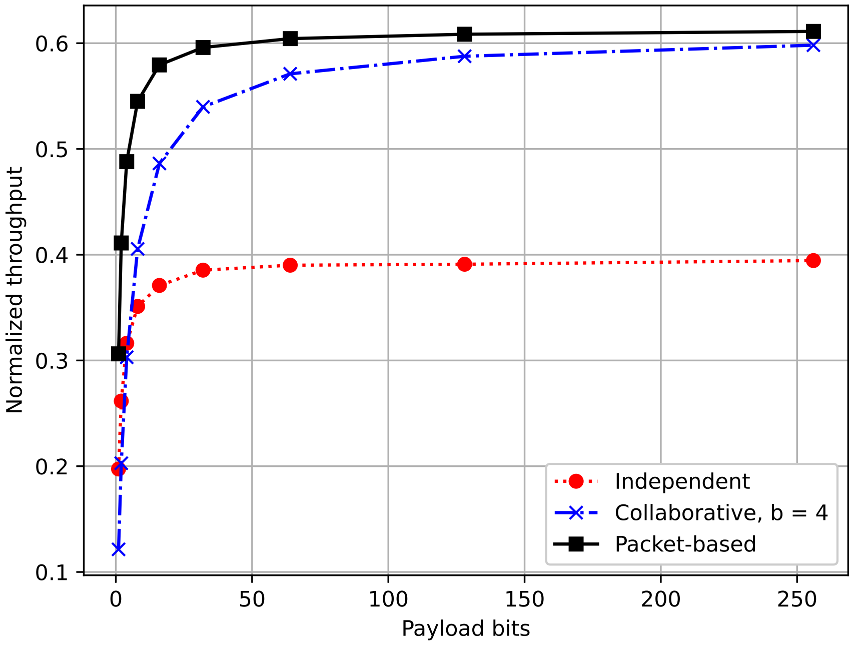


Figure 2.7 – Normalized throughput as as function of payload bits, considering $\mathcal{L} = 1.5$, $\alpha = 0.1$, and $L = 100$.

Throughput increases with the increase in the number of bits up to $p = 64$. There is a convergence to a ceiling throughput from that value. This value $p = 64$ was considered in the other simulations present in this work. As the collaborative technique has a larger number of header bits ($b = 4$), then it depends on a larger number of payload bits to present the same throughput as the packet-based technique. For example, to achieve a normalized throughput of $\mathcal{T} = 0.5$, the collaborative technique needs 16 bits of payload, while the packet-based one needs 4 bits in the analyzed scenario. The reduction in the number of payload bits can be an advantage in simplifying the process in which a bunch of devices randomly access the channel and transmit their packets.

2.4.5 Latency

As low latency is one of the main requirements of 5G/6G systems, we analyzed the number of time-slots needed to obtain the convergence of each QL algorithm as a figure of merit. The result of Fig. 2.8 shows the behavior of the latency of the algorithms with the increase of the loading factor.

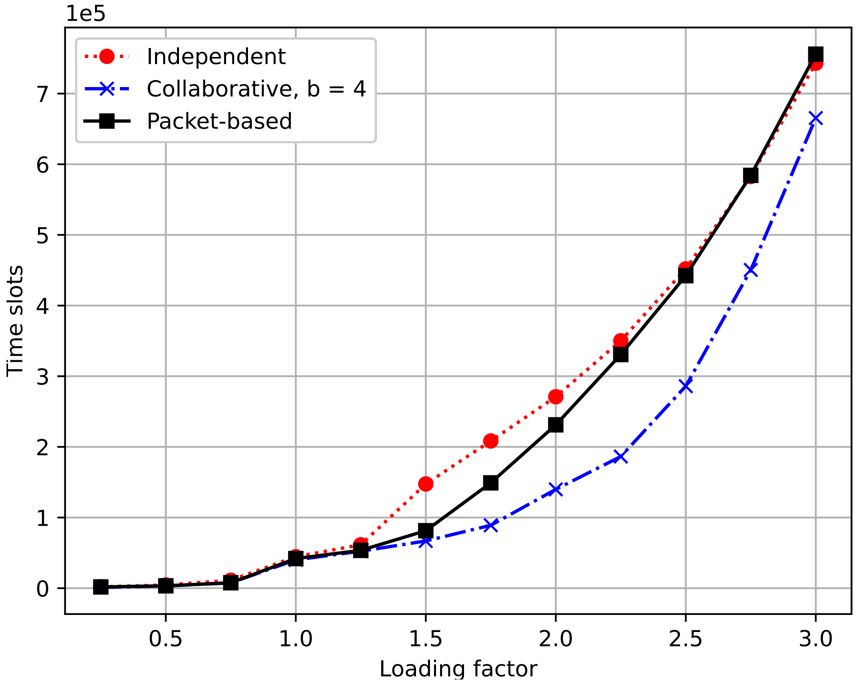


Figure 2.8 – Total number of time-slots as a function of loading factor considering $L = 100$, and $\alpha = 0.1$.

It is clearly observed that the increase in the loading factor impacts the increase in latency, as the probability of collision increases as the system overloads itself. Up to $\mathcal{L} = 1$, the latency is the same for all analyzed algorithms. Between $1.25 \leq \mathcal{L} \leq 2.25$, the independent technique has the highest latency, as there is a greater probability of collision since the reward is the simplest of all the algorithms. Finally, for $\mathcal{L} \geq 2.5$, the packet-based algorithm has the same latency as the independent one.

The collaborative algorithm has the lowest latency of the analyzed scenario, however, it must be taken into account that it is also the most complex algorithm, since the central node needs to know which are all the devices that collided in each time-slot, in addition to informing the value obtained in the reward with more than one bit. Therefore, it is possible to state that, in the scenario, the algorithm presents a good trade-off between throughput, latency and complexity compared to the other two algorithms.

2.4.6 Learning rate

The learning rate α is the weight given to the rewards when the device updates the Q-values in the Q-table, as shown in Eq. 3.7. Here in this subsection, the adopted value for the learning rate $\alpha = 0.1$ is justified. For that, latency was evaluated according to the learning rate, as shown in Fig. 2.9.

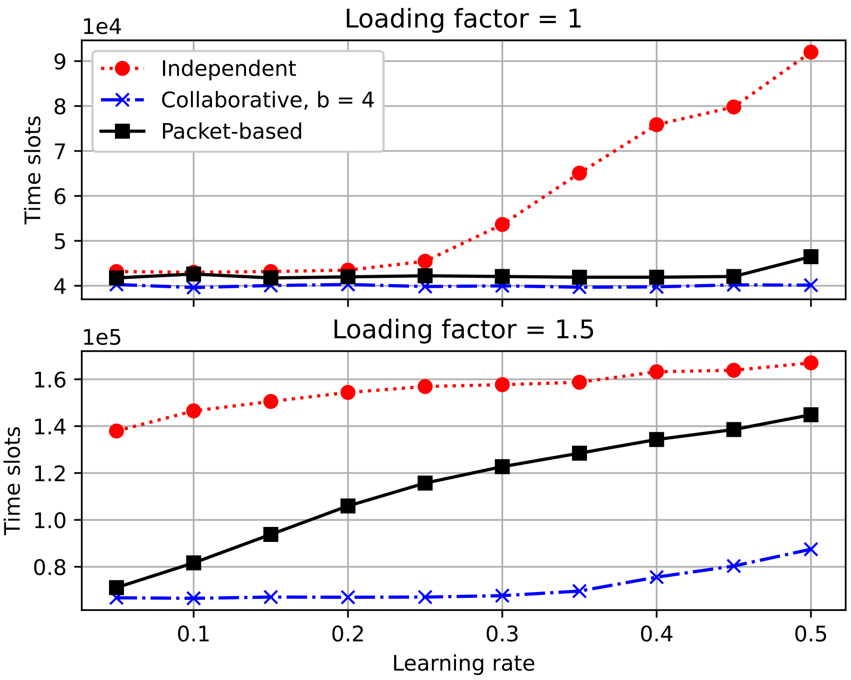


Figure 2.9 – Total number of time-slots as a function of learning rate considering $L = 100$.

It is observed that the increase in the learning rate impacts the increase in latency necessary for the QL algorithms to converge. When the learning rate is high, the weight that the devices give to the reward of the central node is greater. Hence, in more congested scenarios, e.g. $\mathcal{L} \geq 1.5$, more negative than positive rewards can be expected from the central node. Therefore, when devices give greater weight to negative rewards, the latency of the technique increases. This behavior is observed when $\mathcal{L} = 1.5$, as the latency increases significantly with the increase in the learning rate. When $\mathcal{L} = 1$, this behavior is smoothed, since the increase in latency only occurs when $\alpha = 0.5$ for the collaborative and packet-based algorithms.

2.5 Conclusions

In this chapter, we analyzed the performance of QL-based random access algorithms at the protocol layer in different loading factor mMTC scenarios, reaching $\mathcal{L} = 3$, with three devices, in average, disputing for a resource block (time-slot) in crowded mMTC typical scenario. A distributed packet-based algorithm aided by reinforcement Q-learning is proposed, in which the reward sent by the central node is binary and the complexity to updating the Q-table is transferred to the devices side in a distributed way.

The proposed QL-based algorithm presented a higher throughput and a lower latency when compared to the QL-Independent algorithm, even with both algorithms considering the transmission of binary reward in the central node. The superiority is justified by the different way the calculation and updating of the Q-table in the devices, since in the proposed algorithm the devices take into account the amount of remaining packets, which favors the devices having delayed-packets transmission .

When comparing the distributed packet-based QL algorithm with the QL-Collaborative, a superiority of the former method was observed in under- and slightly-crowded scenarios, *i.e.*, $0.25 \leq \mathcal{L} \leq 1.5$. The proposed algorithm uses fewer bits in the reward transmission, which makes the throughput greater in this scenario. Under over-crowded scenarios, $\mathcal{L} \approx 3$, the proposed and QL-Collaborative method presented the same throughput. Therefore, it is possible to conclude that the proposed distributed packet-based algorithm revealed the best performance-complexity trade-off among the analyzed algorithms, as the throughput is higher and the complexity is lower when compared to the QL-collaborative one.

3 Power domain Q-learning for random access in NOMA mMTC systems

3.1 Introduction

In Chapter 2, the mMTC system was analyzed with a focus on the protocol layer to evaluate the performance of different QL algorithms. In this chapter, the physical layer model will be considered in order to use the power diversity between the devices aiding to apply NOMA technique in the central node. Initially, this chapter is inspired in the results of (SILVA et al., 2020).

When the signal from two devices arrives with an acceptable power disparity at the central node, it is possible to apply the SIC to remove interference from the signal from the strongest device and detect the signal from the weakest device. By including the power domain, two or more devices can select the same time-slot without collision, which impacts on a decrease in latency, which is one of the main objectives of mMTC systems. In this scenario, it is possible to use a decentralized Q-learning algorithm in which the devices learn what is the best transmission power to transmit their packets, exempting the central node from spending complexity with power allocation. The fact that the technique is decentralized is an advantage in mMTC systems where there are thousands of devices being served by a single central node.

In this chapter, we analyze the performance of QL algorithms when added to the physical layer with path loss and fading power loss models. This scenario is more realistic because successful transmission is only counted at the central node when the SINR reaches an acceptable threshold, which is what happens in practice, since the receivers have an operating sensitivity that only correctly detects the information when the received power is high enough.

We propose the multi-power level QL (MPL-QL) algorithm, which also performs learning in the power domain. Considering that devices can transmit at different power levels to generate power disparity in the transmitter, then it is possible to apply QL so that devices find out what is the suitable power level to increase the probability of successful packet transmission. The proposed algorithm proved to be superior to other algorithms in the literature.

This chapter is divided as follows: the physical layer system model is described in Section 3.2; the operation of the proposed MPL-QL algorithm is presented in Section 3.3; the numerical results of the MPL-QL compared to other QL algorithms are presented in

Section 3.4; in Section 3.5, the final conclusions of the chapter are presented.

3.2 System model

There are N mMTC devices sending uplink (UL) packets to a central node in a circular cell with radius r. The frequency resources used are a carrier f_c and a bandwidth B. The n-th device is d_n meters away from the central node and it transmits with power $P_{t,n}$.

The transmit frame in the UL is divided into K time-slots, while a downlink (DL) time-slot at the end is deployed for central node broadcast. The devices randomly select a time-slot to transmit. The set ψ_k contains the indexes of all devices that selected k-th time-slot, $k \in \{1, \dots, K\}$. Furthermore, each device has L packets to transmit. The end of system transmission occurs when all devices successfully transmit all of their L packets. At the end, we define the total latency δ as the total number of spent frames to attain convergence, *i.e.*, all packets transmitted successfully by all devices. Assuming that the DL slot is much smaller than the UL slot, it is possible to approximate the length of a frame to K time-slots and the total number of time-slots until the end is δK. Fig. 3.1 shows how transmission frames are divided.

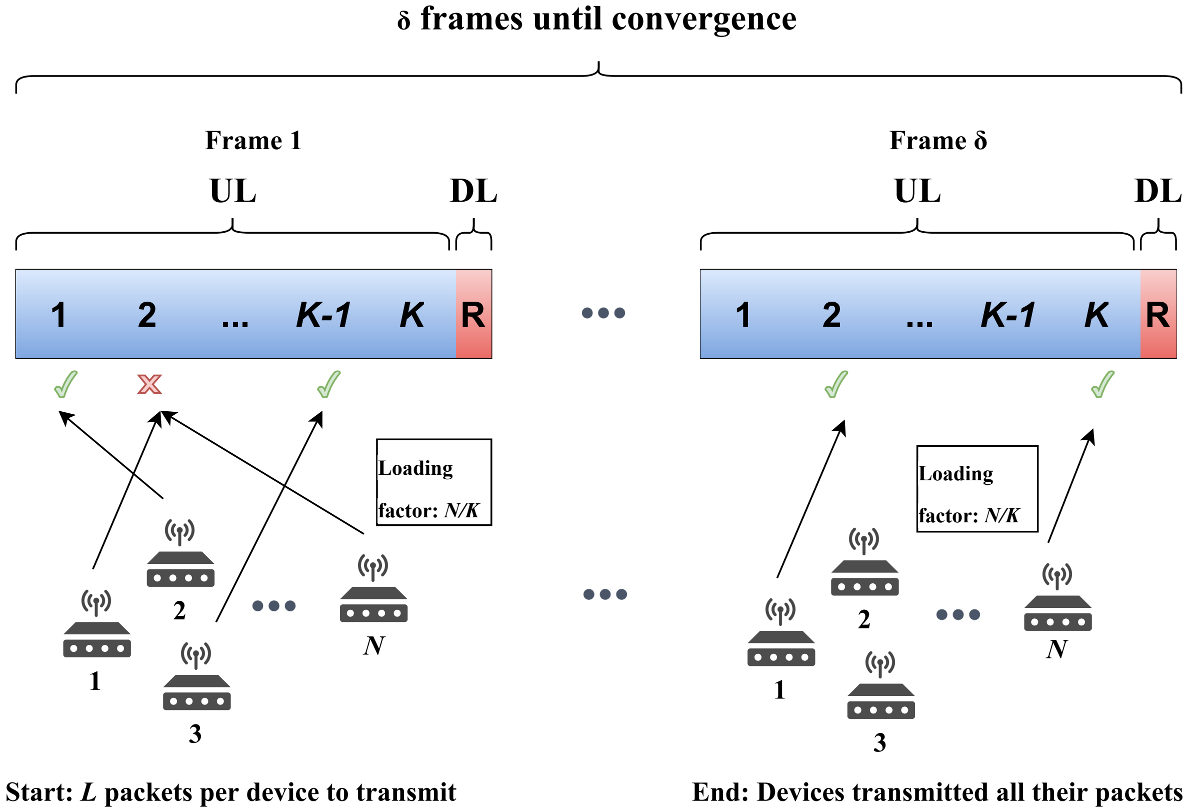

δ frames until convergence
Frame 1
UL
DL
1
2
...
K-1
K
R
Loading factor: N/K
1
2
3
N
Frame δ
UL
DL
1
2
...
K-1
K
R
Loading factor: N/K
1
2
3
N
Start: L packets per device to transmit
End: Devices transmitted all their packets

Figure 3.1 – Frame in UL and DL time-slots until the end of system transmission (all devices), namely convergence of transmission process.

The received signal in the central node at the k-th time-slot is simply defined as:

$$y_k = \sum_{\forall n \in \psi_k} x_{n,k} + w_k, \tag{3.1}$$

where $x_{n,k}$ is the attenuated signal transmitted by the n-th device at the k-th time-slot, and $w_k \sim \mathcal{CN}(0, N_0 B)$ is the additive white Gaussian noise (AWGN) at the receiver in the k-th time-slot with power spectral density N_0. The signal transmitted by the n-th device is $\tilde{x}_n$, and the attenuated received signal before AWGN $x_{n,k}$ takes into account the path loss and short-term fading effects.

It is necessary that there is CSI in the transmitter (device) and in the receiver (central node) to detect the information correctly in both communication directions. However, as the most relevant information are the uplink data packets, a low-complexity channel estimation technique is enough to detect downlinks in devices.

Let's consider that $h_{n,k}$ is an independent and identically distributed zero mean and unit variance Rayleigh fading of the n-th device at k-th time-slot. Therefore, the instantaneous signal-to-interference-plus-noise ratio (SINR) received from the n-th device at the k-th time-slot can be defined as

$$\gamma_{n,k} = \frac{P_{n,k}}{\sum_{\forall j \in \psi_k, j \neq n} P_{j,k} + w_k^2}, \tag{3.2}$$

where $P_{n,k} = h_{n,k}^2 \bar{P}_n$ is the instantaneous power of the n-th device at k-th time-slot. $\bar{P}_n$ is calculated based on the log-distance path loss model:

$$\bar{P}_n = P_{t,n} + \bar{P}_{d_0} - 10\eta \log_{10}\left(\frac{d_n}{d_0}\right), \quad [\text{dB}] \tag{3.3}$$

where η is the path loss exponent, d_0 is a reference distance, and $\bar{P}_{d_0}$ is a reference constant power given by

$$\bar{P}_{d_0} = 20 \log_{10}\left(\frac{c}{4\pi d_0 f_c}\right). \quad [\text{dB}] \tag{3.4}$$

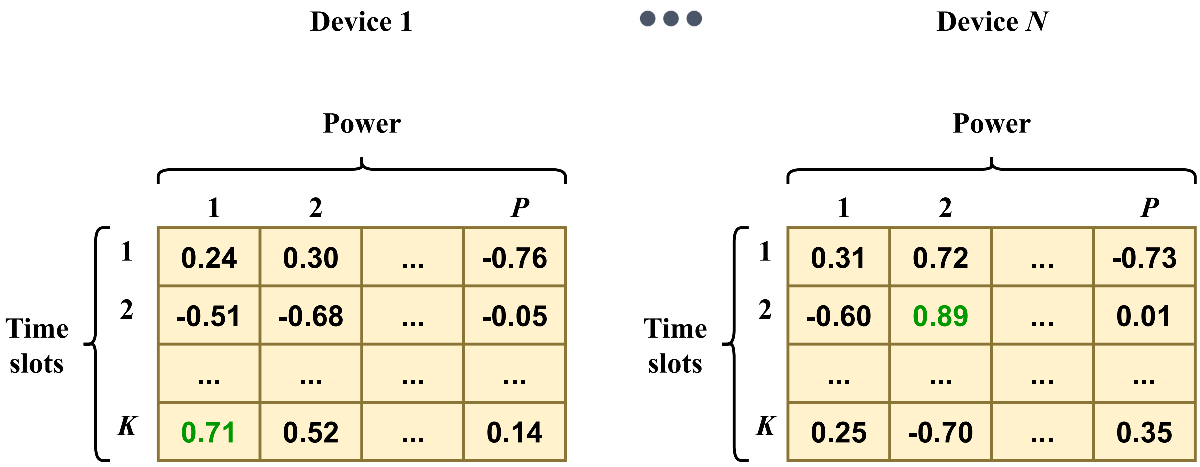
Assuming that the devices have the same quality of service (QoS) requirements, we can set a threshold SINR $\bar{\gamma}$ at the receiver to ensure the packet can be detected. The packet transmitted by the n-th device at k-th time-slot can be successfully received at the central node when $\gamma_{n,k} \geq \bar{\gamma}$.

3.3 Multi-power level Q-learning algorithm

This section describes the proposed MPL-QL grant-free RA procedure. Each device can transmit with a maximum power $P_{\max}$. The transmitted power $P_{t,n}$ of the n-th device can assume $\mathcal{P}$ equidistant power levels between 0 and $P_{\max}$, *e.g.* for $\mathcal{P} = 4$:

$$P_{t,n} \in \left\{\frac{P_{\max}}{4}, \frac{P_{\max}}{2}, \frac{3P_{\max}}{4}, P_{\max}\right\} \qquad [W].$$

The selection of which time-slot and power level the device will transmit is based on the Q-table indices whose Q-value is maximum. When there are two or more values equal to the maximum, the device randomly selects between them. Fig. 3.2 depicts the structure of the power level and time-slot selection based on the Q-table.



Device 1

Time slots	1	2	...	*P*
1	0.24	0.30	...	-0.76
2	-0.51	-0.68	...	-0.05
	...	...	...	...
K	0.71	0.52	...	0.14

Device *N*

Time slots	1	2	...	*P*
1	0.31	0.72	...	-0.73
2	-0.60	0.89	...	0.01
	...	...	...	...
K	0.25	-0.70	...	0.35

Figure 3.2 – Q-table for each device.

As the devices present a power disparity given by the differences in distances and transmission powers, then the central node can apply a *successive interference cancellation* (SIC) procedure to remove the interference from the devices that collided in the same time-slot. With this, the SINR considering NOMA becomes:

$$\gamma_{n,k}^{\text{NOMA}} = \frac{P_{n,k}}{\sum_{j=n+1}^{|\psi|} P_{j,k} + w_k^2}. \tag{3.5}$$

The transmission of the n-th device is successful if

$$R_{n,k,p} = \begin{cases} +1, & \text{if } \gamma_{n,k}^{\text{NOMA}} \geq \bar{\gamma} \\ -1, & \text{otherwise.} \end{cases} \tag{3.6}$$

With the reward received, the device updates its Q-table (SUTTON; BARTO, 2018):

$$Q_{n,k,p}^{(t+1)} = Q_{n,k,p}^{(t)} + \alpha(R_{n,k,p} - Q_{n,k,p}^{(t)}). \tag{3.7}$$

ℓ_n is the number of packets that the n-th device still has to transmit. The devices continue transmitting until all of their L packets are transmitted. Total latency δ is the number of frames required for the complete transmission of packets until the algorithm converges. Algorithm 2 indicates the pseudo-code step-by-step of the proposed MPL-QL operation.

3.4 Numerical results

In this section, the results of simulations of QL algorithms in RA mMTC scenarios are presented. The results were obtained using the Monte-Carlo simulation method. The

Algorithm 2 MPL-QL algorithm

```
Initialize $Q_{n,k,p} \sim \mathcal{U}[-1,1]\ \forall n,k,p$;
Initialize $\ell_n = L,\ \forall n$;
Initialize $\delta = 0,\ S = 0$
while $\sum_{n=1}^{N} \ell_n > 0$ do
    for all devices that $\ell_n > 0$ do
        Search $k$ and $p$ where
        $Q_{n,k,p} = \max_{k,p}\{Q_{n,k,p}\}$
    for all time-slots $k = 1 : K$ do
        if $|\psi_k| > 0$ then
            Calculate $\gamma_{n,k}^{\text{NOMA}}$ using Eq. (3.5) $\forall n \in \psi_k$
            if $\gamma_{n,k}^{\text{NOMA}} \geq \bar{\gamma}$ then
                Success: $S \leftarrow S + 1,\ \ell_n \leftarrow \ell_n - 1$
                $R_{n,k,p} = 1$
            else
                $R_{n,k,p} =$ -1
            Update: $Q_{n,k,p}^{(t+1)} = Q_{n,k,p}^{(t)} + \alpha(R_{n,k,p} - Q_{n,k,p}^{(t)})$
    Increment a frame: $\delta \leftarrow \delta + 1$
```

source code of the simulations was developed in Python language and is presented in a summarized form in Appendix D.

The numerical parameters used in this section are presented in Table 3.1. The value of the chosen parameters of frequency and cell size are typical of IoT scenarios, and the amount of devices represents a crowded NOMA mMTC scenario. The typical SINR threshold was selected as $\bar{\gamma} = 3$, considering that the outage probability is a suitable metric for the communication system performance evaluation (SILVA et al., 2020). Hence, if the SINR is greater than or equal to the Shannon capacity, *i.e.*

$$\gamma_{n,k}^{\text{NOMA}} \geq 2^{\tau} - 1, \tag{3.8}$$

where τ is the spectral efficiency in bits/s/Hz, then a desired (adopted) spectral efficiency of $\tau = 2$ bits/s/Hz makes $\bar{\gamma} = 3$.

3.4.1 Throughput and latency of MPL-QL

The results presented in this subsection summarize a characterization of the MPL-QL algorithm in terms of normalized throughput and latency for different power levels $\mathcal{P}$. The aim is to assess which is the suitable number of power levels that provides a good performance-complexity trade-off. A large number of levels increases the order of the Q-table stored in the devices, as shown in Fig. 3.2.

The normalized throughput used in this section is an approximation of the throughput calculated by Eq. 2.2. We considered that the number of payload bits is much higher than the number of header bits ($p >> b$). Because of that, the normalized throughput

Table 3.1 – Numerical parameters for power domain NOMA QL.

Parameter	**Value**
Monte-Carlo realizations	$N_{\text{reps}} = 10000$
Time-slots per frame	$K = 100$
Network loading factor	$\mathcal{L} = \frac{N}{K} \in [0.25;\ 10]$
Packets per device	$L \in [50;\, 100]$
Learning rate	$\alpha = 0.1$
SINR threshold	$\bar{\gamma} = 3$
Transmit power levels	$\mathcal{P} \in [2; 4; 8; 12; 16]$
Cell radius	$r = 200$ m
Reference distance	$d_0 = 1$ m
Bandwidth	$B = 125$ kHz
Carrier frequency	$f_c = 915$ MHz
Path loss exponent	$\eta = 3$
Noise PSD	$N_0 = -150$ dBm/Hz
Maximum power	$P_{\max} = 1$ mW

becomes

$$\mathcal{T} \approx \frac{S}{\delta K}. \tag{3.9}$$

This approximation is valid because the maximum number of header bits b needed to obtain good throughput is $b = 4$ for the collaborative QL algorithm and $b = 1$ for the others, as analyzed in Subsection 2.4.1. Considering that devices can transmit packets with payloads of 128 or even 256 bits, then the correction term $\dfrac{p}{b+p}$ in Eq. 2.2 becomes negligible in this NOMA scenario.

In Fig. 3.3, the normalized throughput behavior as a function of the loading factor $\mathcal{L}$ is shown for different values of power levels $\mathcal{P}$. We considered $L = 100$ packets and $K = 100$ time-slots to obtain this result.

We can see that increasing the number of power levels from $\mathcal{P} = 2$ to $\mathcal{P} = 12$ causes an increase in normalized throughput. With higher power levels, there is a greater difference between the signal from the desired device and the signal from interfering devices, which allows the SIC to detect more packets, increasing the SINR.

However, when increasing $\mathcal{P}$ from 12 to above, it is noticed that the throughput gain becomes marginal. This indicates that there is an interference limit in the system in which it is not possible to detect more packets in the receiver, as the power difference between the desired signal and the interferers becomes smaller and smaller, and it is not possible to reach the SINR threshold to ensure quality of the system.

There is a $\mathcal{P}$ value between 8 and 12 that ensures a good trade-off between performance and complexity, as it ensures good throughput without increasing the complexity of Q-table storage on devices.

The latency δ of the MPL-QL was also analyzed, being the total number of frames needed for the convergence of the algorithm. The same simulation parameters used in the

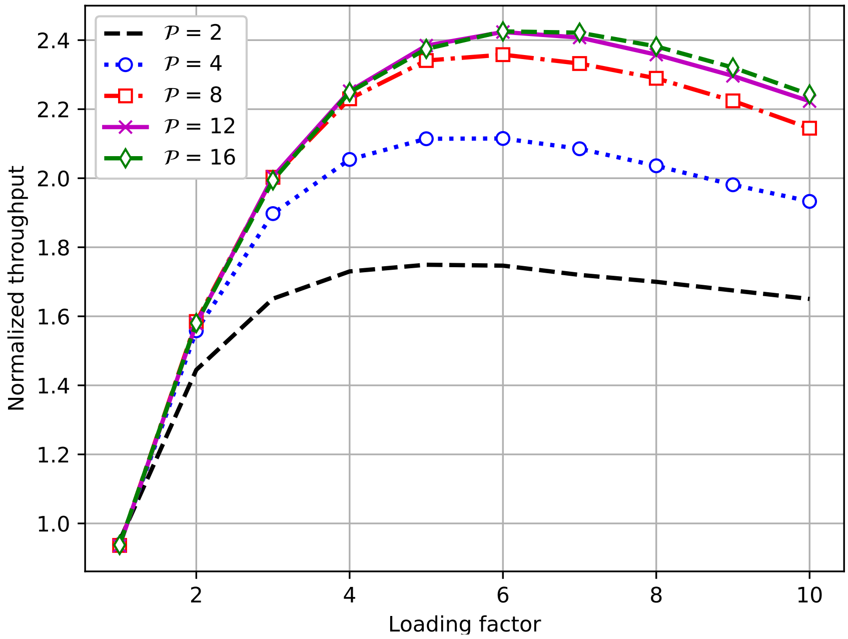


Figure 3.3 – MPL-QL throughput for different power levels $\mathcal{P}$.

result of Fig. 3.3 were considered. The result of latency as a function of the loading factor is shown in Fig. 3.4.

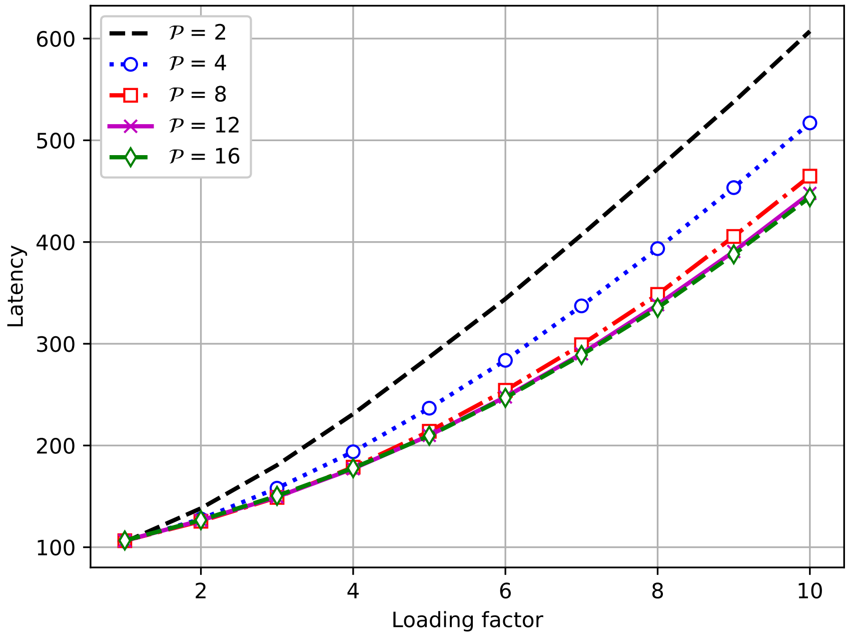


Figure 3.4 – MPL-QL latency (total number of frames).

Latency decreases with increasing power levels. For a loading factor $\mathcal{L} = 6$, the MPL-QL with $\mathcal{P} = 8$ has 30% lower latency compared to $\mathcal{P} = 2$. With more transmitter power levels, the higher the SINR, as discussed also in the throughput result of Fig. 3.3. This causes an increase in the number of successes that occurs within a frame, which makes the devices transmit their packets faster and the algorithm to finish in a shorter time.

It is also observed that the latency reaches a lower limit when $\mathcal{P}$ is increased from 8 to 16. As the number of levels increases, the granularity increases, which makes it difficult for the SIC in the receiver to remove the interfering devices, decreasing the number of successes and achieving maximum performance limited by system interference.

It is considered in the following simulations that the MPL-QL operates with $\mathcal{P}$ = 8 power levels, as the results in Figures 3.3 and 3.4 indicate that it is a good value to guarantee a good trade-off between throughput, latency and complexity, excluding transmitter hardware complexity in providing eight power levels. This $\mathcal{P}$ value is used in comparison with other QL algorithms present in the literature.

3.4.2 Convergence of MPL-QL

In this subsection, we analyze the convergence of the MPL-QL for two different power levels: $\mathcal{P}$ = 2 and $\mathcal{P}$ = 8. The figures of merit are evaluated in function of the number of frames. The results were obtained for the n-th device, but the behavior observed in the average is the same for all devices in the system. The figures of merit are calculated over $\mathcal{M}$ realizations.

The first figure of merit considered in the convergence analysis is the interference that the n-th device suffers when selecting a time-slot to transmit. Interference is calculated as

$$I_{n,k} = \sum_{j=n+1}^{|\psi_k|} P_{j,k}. \qquad [W] \tag{3.10}$$

The calculation is performed by the receiver after the SIC. In addition to interference, we also evaluate the convergence factor ν, previously defined in Eq. 2.6. The convergence factor indicates how close the n-th device is to completing the complete transmission of its packets. At the beginning of the algorithm, $\nu = 0$. When the device finishes the transmission of all packets, $\nu = 1$.

Fig. 3.5 shows the convergence of the MPL-QL algorithm in terms of interference and convergence factor for $\mathcal{P}$ = 2 and $\mathcal{P}$ = 8 considering different loading factors $\mathcal{L}$ and total number of packets per device L.

At the beginning of the algorithm, all devices are colliding in time-slots and learning which ones are the best to transmit. That is why an interference oscillation is observed until reaching L frames, *i.e.* $L = 50$ in Fig. 3.5a) and $L = 100$ in Fig. 3.5b). After L frames, the devices that selected the best time-slots with the lowest congestion levels finish transmitting the L packets, so they exit the algorithm. Because of that, the interference observed in the n-th device starts to reduce steadily until it becomes zero. It can be seen in Figures 3.5b) and Fig. 3.5c) that the instant at which the interference is zero is the same at which the convergence factor is one, because at this point where there is no more interference, the n-th device ends the transmission of its packets.

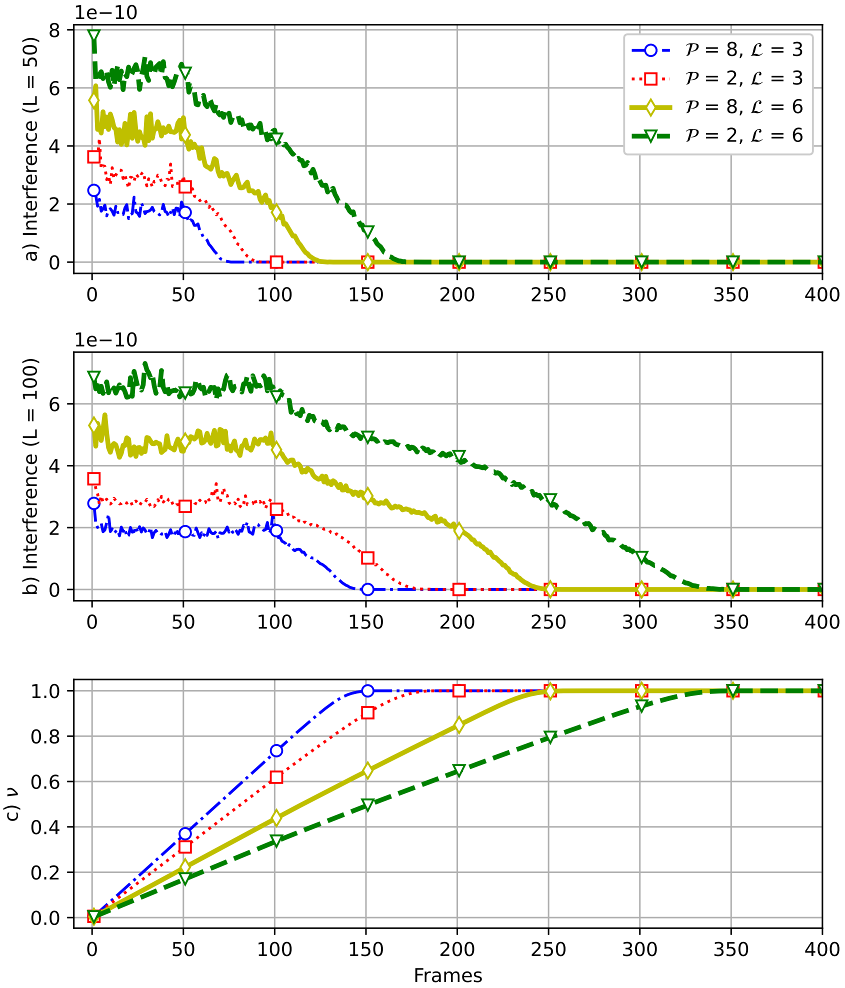


Figure 3.5 – Interference and convergence factor of the MPL-QL algorithm considering the n-th device. a) $L = 50$ packets; b) $L = 100$ packets; c) convergence factor under $L = 100$ packets.

Increasing the loading factor causes more devices to compete for available time-slots in the frame. That is why interference is higher when $\mathcal{L} = 6$ compared to $\mathcal{L} = 3$. Also, convergence is slower for a higher loading factor, as with higher interference, more collisions occur, which forces devices to retransmit the same packet more times until the successful transmission.

Finally, the increase in the number of power levels causes the interference to decrease between devices, as it increases the granularity of the division between the levels, which consequently increases the probability of interfering devices to select lower power levels in relation to the desired device. This makes the initial interference lower for $\mathcal{P} = 8$ compared to $\mathcal{P} = 2$. This makes convergence also occur faster, as the interference becomes zero quicker, which increases the SINR and more packets are successfully transmitted.

3.4.3 Comparison with other QL algorithms

The MPL-QL algorithm with $\mathcal{P} = 8$ is compared in this section with other algorithms: a) slotted ALOHA (SA), in which there is no central node reward and the device does not learn the best time-slot to transmit, always selecting randomly; b) independent QL; c) collaborative QL; and d) distributed packet-based QL.

The MPL-QL is the only algorithm that depends on a two-dimensional learning, being both in the domain of time-slots and in the domain of power levels. The other techniques perform learning only in the time-slots domain, so the Q-table of these algorithms is a $(K \times 1)$ vector, in contrast to the two-dimensional $(K \times \mathcal{P})$ Q-table of MPL-QL. Besides, all devices transmit with $P_{\max}$ in the other QL algorithms, as it is a simple way to compare with the adaptive power of MPL-QL. For a broad comparison, it would be more representative to aggregate a centralized (or distributed) power allocation policy by determining the optimal power for the other algorithms. However, such analysis under a specific power allocation strategy has not been carried out in the current work, being left for future works.

Fig. 3.6 shows the result of comparing the throughput and latency of the analyzed algorithms as a function of the loading factor $\mathcal{L}$, considering $K = 100$ time-slots/frame and $L = 100$ packets/device. The algorithm with the lowest throughput is SA, as it is the most simplified form of sending the packets without getting feedback from the central node. The device does not take advantage of the less congested time-slots, which means that even when success is obtained at the receiver, a new time-slot is randomly selected by the device. Latency was not analyzed for SA. There is no feedback from the central node, so packet transmission always ends after L frames for all devices.

From (SHARMA; WANG, 2019) and what was discussed in Chapter 2, it is noted that there is a difference in throughput and latency of independent, collaborative, and packet-based QL algorithms. In some scenarios, packet-based was superior, while collaborative was superior in others. However, in the scenario analyzed in Fig. 3.6, it is noted that the performance of the three algorithms is quite similar. The path loss and fading power loss effects of the realistic scenario analyzed in this chapter degrade the performance of the algorithms and the gains that were observed previously become marginal because of the severity of the fading.

The MPL-QL proved to be an algorithm that deals well with the effects of RA in NOMA mMTC scenarios, as it presented the best throughput and the lowest latency among all the discussed algorithms. As the algorithm generates extra power diversity by allowing more than one transmit power level, then devices that may be close can transmit at different power levels, which causes it to generate a power difference at the receiver in such a way that the SIC can be performed, increasing the SINR and consequently the number of successes obtained.

It can be seen in Fig. 3.6 that, considering a loading factor $\mathcal{L} = 4$, which means that

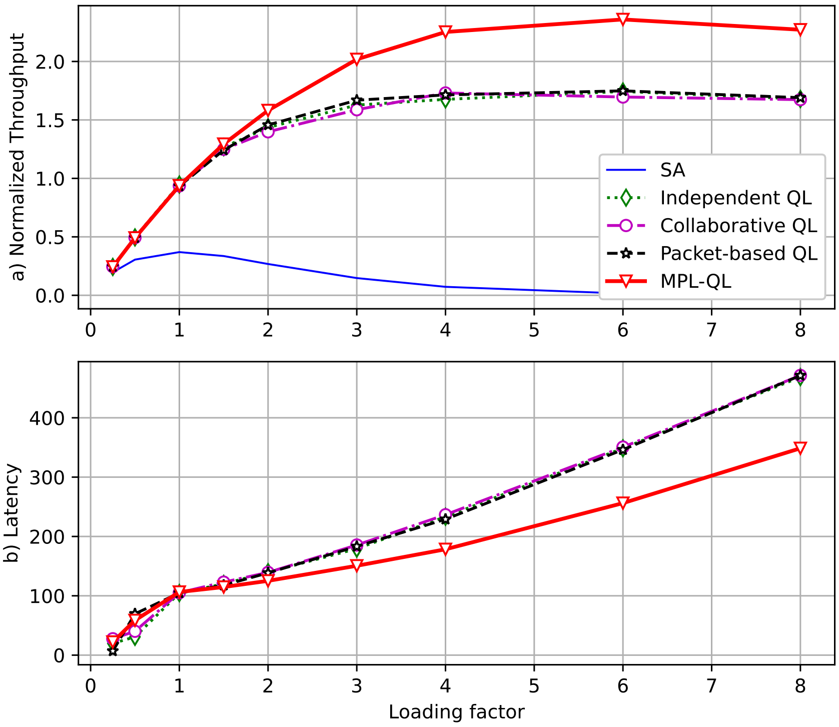


Figure 3.6 – a) Throughput, and b) Latency for the SA and four QL-based algorithms, with $\mathcal{P} = 8$ for the proposed MPL-QL. $L = 100$ and $K = 100$.

there are 4 active devices competing for a time-slot, the MPL-QL provides approximately 2.3 successes/frame, while the collaborative QL provides 1.7 successes/frame. This indicates that MPL-QL allows more devices within the system to transmit successfully, being the most suitable for NOMA mMTC scenarios in which the number of devices can reach tens of thousands.

3.4.4 QL algorithms with imperfect SIC

In all the results obtained in this chapter, it was considered that the central node is capable of performing a perfect SIC during the reception process. This scenario is ideal, because when the loading factor increases, more devices compete for the same time-slot and it becomes more difficult for the central node to cancel the interference perfectly, without generating a residual error that is propagated through the devices.

To evaluate the behavior of the proposed MPL-QL algorithm in more realistic scenarios, we consider an imperfect SIC model based on (KARA; KAYA, 2020) that modifies the SINR calculation:

$$\tilde{\gamma}_{n,k}^{\text{NOMA}} = \frac{P_{n,k}}{\beta \sum_{j=0}^{n-1} P_{j,k} + \sum_{j=n+1}^{|\psi|} P_{j,k} + w_k^2}. \tag{3.11}$$

β is the SIC error factor. When $\beta = 0$, the SIC is performed perfectly. On the other hand, when $\beta = 1$, there is no SIC at the receiver. Typical values for the SIC error factor are in the range $\beta \in \{0.01;\ 0.30\}$, following the values adopted in (KARA; KAYA, 2020).

In Fig. 3.7, it is shown the effect on throughput of the MPL-QL, independent QL, and packet-based QL algorithms by changing the SIC error factor between $\beta = 0$, $\beta = 0.01$ and $\beta = 0.02$. Collaborative QL was omitted in this scenario as it has already been shown to be performs very similarly to independent and packet-based algorithms. $K = 100$ time-slots/frame and $L = 100$ packets/device were considered.

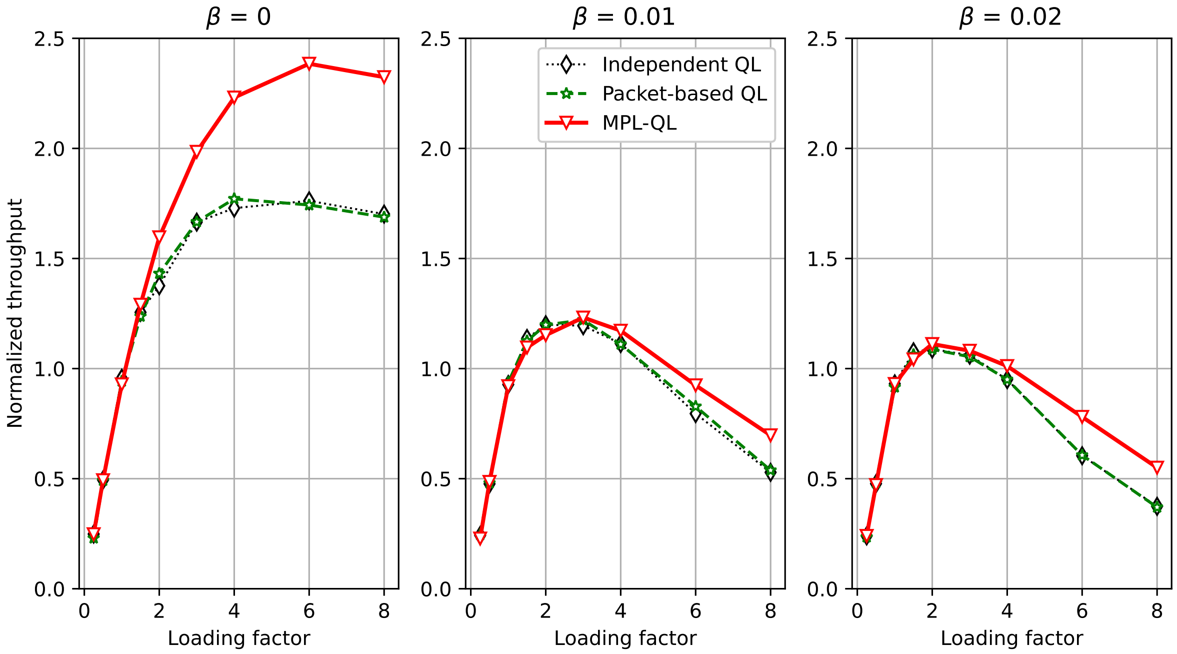


Figure 3.7 – Independent QL, Packet-based QL and MPL-QL under SIC imperfection: a) $\beta = 0$; b) $\beta = 0.01$; $\beta = 0.02$. We considered $L = 100$ and $K = 100$.

The increase in β increases the interference of the system, because for $\beta > 0$, there is a residue of interference from devices with higher powers that was not perfectly canceled out. Increasing interference decreases SINR, which consequently increases latency until algorithm convergence, decreasing throughput.

For the MPL-QL algorithm, for $\beta = 0$, the maximum throughput is reached when $\mathcal{L} = 6$, that is, with 6 devices disputing a time-slot. For $\beta = 0.01$ and $\beta = 0.02$, the maximum throughput is obtained for $\mathcal{L} = 3$ and $\mathcal{L} = 2$, respectively, which indicates that the maximum number of devices that the system supports is reduced.

The algorithm that presents the best throughput among the analyzed algorithms is the MPL-QL. As multiple power levels are considered in the transmitter, the power is received with disparity. Even with an imperfect SIC, the central node can more easily remove interference from devices that have selected the lower power levels. This does not happen for the other algorithms, since the transmitted power is the same for all devices. Therefore, it is concluded that the MPL-QL is the most suitable algorithm in the NOMA mMTC scenario as it provides a greater power granularity and allows more devices to be serviced using the same time-slot resources. Indeed, to further mitigate the impact of SIC imperfection under crowded mMTC scenarios, it would be possible to use low-complexity unbalanced power allocation techniques. Hence, the powers could arrive at the receiver side with pre-defined disparities, facilitating the central node's work to cancel the interfering signals.

3.5 Conclusions

When considering a realistic power model in QL algorithms, it was observed that the performance of the independent, collaborative and packet-based algorithms presented in Chapter 2 had the same performance, as time-domain learning alone is not enough to deal with collisions that occur when two devices transmit at the same power in a given time-slot.

Therefore, the MPL-QL algorithm was proposed, which considers power diversity in the transmitter, making the QL algorithm perform learning both in power and time domain. It was observed that 8 power levels present the best performance-complexity trade-off, as the increase in throughput from that level is marginal, and with more levels, the complexity increases as the order of the Q-table is increased.

In the other results, considering the MPL-QL with 8 power levels, the throughput and latency were compared with the other QL algorithms, and it was observed that the MPL-QL has the best performance, since the power diversity at the transmitter causes the power disparity to be greater at the receiver, increasing the average SINR, which consequently increases throughput. The superiority of the MPL-QL is maintained when the realistic imperfect SIC model was added, in which interference is not completely eliminated in the receiver.

4 Conclusions

The results obtained in this work showed the performance in terms of throughput and latency of different QL algorithms for mMTC networks. The improvement proposed in the collaborative QL algorithm revealed that the number of reward quantization bits cannot be too low or too high because it directly affects the accuracy of reward value, the decision of device in access the time-slot resource, and as a consequence the final system throughput. The trade-off value found was 4 bits.

The proposed packet-based algorithm has demonstrated be promising in terms of (higher) throughput regarding the collaborative algorithm in some scenarios of application and always superior to the independent QL algorithm. Although the latency of the collaborative algorithm is lower when compared to the one proposed, its complexity is higher because it is necessary for the central node to know the number of devices that collided in each time-slot resource. On the other hand, the packet-based algorithm results in a simplified reward procedure, just like the independent QL algorithm, while the processing remains distributed among the devices, which is an crucial advantage in crowded mMTC scenarios.

When considering the power domain in the NOMA mMTC scenario, it was noticed that the performance of the QL algorithms was different compared to the protocol layer scenario only. The throughput and latency of the independent, collaborative, and packet-based algorithms were similar. Therefore, the MPL-QL random access protocol was proposed, which uses different power levels at the transmitter side, increasing the SINR at the receiver side and makes more successes occur in the same time-slot. The MPL-QL with eight power levels has revealed the best trade-off between throughput, latency and complexity among all analyzed algorithms.

4.1 Future research directions

In this section, topics for future research are presented for the discussions covered in this Master's Dissertation:

- **Traffic models and age of information:** The mMTC network can include active and inactive devices with a typical probability of activation around 1%. This differs from what is analyzed in this work in which all devices are active and sending packets. Considering that devices can be activated sporadically, there are some new challenging issues that require innovative traffic models for mMTC use mode. Because in such scenarios, devices can enter and leave the transmission system with different mobilities and probability of activation; besides, some devices may have many packets to transmit while others may enter to transmit a few packets. In such distinct cases, modeling the input/output of devices in the system, monitoring the number of packets to be sent, and the rate of success of transmission could be challenging. Furthermore, age of information can be a valuable figure of merit to evaluate the delay of arrival of packets at the receiver when there is a collision between two or more devices considering the same block of resources;

- **Low-complexity ML techniques:** the QL algorithms used in this work have intermixed learning and testing phases, which can generate an exploration-exploitation problem, as the device needs to decide between continuing to explore the system or exploiting the data already collected. The same figures of merit evaluated in this work can be applied in low-complexity ML techniques, such as linear machine learning discussed in (MEI et al., 2021) and the two-class neural network present in (ABDELMOUMIN et al., 2021), which already provide devices with a set of learning data in a preliminary phase, and data testing is performed in a second stage.

Bibliography

ABDELMOUMIN, G.; RAWAT, D. B.; RAHMAN, A. On the performance of machine learning models for anomaly-based intelligent intrusion detection systems for the internet of things. *IEEE Internet of Things Journal*, p. 1–1, 2021. Cited on page 68.

ALAM, M.; ZHANG, Q. Novel codebook-based mc-cdma with compressive sensing multiuser detection for sporadic mmtc. In: *2018 IEEE Globecom Workshops (GC Wkshps)*. [S.l.: s.n.], 2018. p. 1–6. Cited 2 times on page(s) 23 and 32.

BHAT, J. R.; ALQAHTANI, S. A. 6G ecosystem: Current status and future perspective. *IEEE Access*, v. 9, p. 43134–43167, 2021. Cited on page 29.

BJÖRNSON, E.; CARVALHO, E. de; SØRENSEN, J. H.; LARSSON, E. G.; POPOVSKI, P. A Random Access Protocol for Pilot Allocation in Crowded Massive MIMO Systems. *IEEE Transactions on Wireless Communications*, v. 16, n. 4, p. 2220–2234, 2017. Cited on page 33.

BUI, A.-T. H.; NGUYEN, C. T.; THANG, T. C.; PHAM, A. T. A comprehensive distributed queue-based random access framework for mmtc in lte/lte-a networks with mixed-type traffic. *IEEE Transactions on Vehicular Technology*, v. 68, n. 12, p. 12107–12120, 2019. Cited 2 times on page(s) 31 and 36.

CHOWDHURY, M. Z.; SHAHJALAL, M.; AHMED, S.; JANG, Y. M. 6G wireless communication systems: Applications, requirements, technologies, challenges, and research directions. *IEEE Open Journal of the Communications Society*, v. 1, p. 957–975, 2020. Cited on page 29.

JIANG, W.; HAN, B.; HABIBI, M. A.; SCHOTTEN, H. D. The Road Towards 6G: A Comprehensive Survey. *IEEE Open Journal of the Communications Society*, v. 2, p. 334–366, 2021. Cited 3 times on page(s) 23, 30, and 31.

KARA, F.; KAYA, H. Improved User Fairness in Decode-Forward Relaying Non-Orthogonal Multiple Access Schemes With Imperfect SIC and CSI. *IEEE Access*, v. 8, p. 97540–97556, 2020. Cited on page 63.

LEE, Y. L.; QIN, D.; WANG, L.-C.; SIM, G. H. 6G massive radio access networks: Key applications, requirements and challenges. *IEEE Open Journal of Vehicular Technology*, v. 2, p. 54–66, 2021. Cited on page 30.

MEI, K.; LIU, J.; ZHANG, X.; CAO, K.; RAJATHEVA, N.; WEI, J. A low complexity learning-based channel estimation for OFDM systems with online training. *IEEE Transactions on Communications*, v. 69, n. 10, p. 6722–6733, 2021. Cited on page 68.

MOHRI, M.; ROSTAMIZADEH, A.; TALWALKAR, A. *Foundations of Machine Learning*. 2. ed. Cambridge: The MIT Press, 2018. Cited 3 times on page(s) 23, 35, and 36.

NGUYEN, D. C.; DING, M.; PATHIRANA, P. N.; SENEVIRATNE, A.; LI, J.; NIYATO, D.; DOBRE, O.; POOR, H. V. 6G internet of things: A comprehensive survey. *IEEE Internet of Things Journal*, p. 1–1, 2021. Cited 2 times on page(s) 23 and 29.

NISHIMURA, O. S.; MARINELLO, J. C.; ABRÃO, T. A Grant-Based Random Access Protocol in Extra-Large Massive MIMO System. *IEEE Communications Letters*, v. 24, n. 11, p. 2478–2482, 2020. Cited 2 times on page(s) 33 and 34.

PIAO, Y.; KIM, Y.; LEE, T.-J. Random power back-off for random access in 5g networks. *IEEE Access*, v. 9, p. 121561–121569, 2021. Cited 2 times on page(s) 23 and 33.

POKHREL, S. R.; DING, J.; PARK, J.; PARK, O.-S.; CHOI, J. Towards enabling critical mmtc: A review of urllc within mmtc. *IEEE Access*, v. 8, p. 131796–131813, 2020. Cited on page 31.

POPOVSKI, P.; TRILLINGSGAARD, K. F.; SIMEONE, O.; DURISI, G. 5G Wireless Network Slicing for eMBB, URLLC, and mMTC: A Communication-Theoretic View. *IEEE Access*, v. 6, p. 55765–55779, 2018. Cited on page 30.

QI, R.; CHI, X.; ZHAO, L.; YANG, W. Martingales-based aloha-type grant-free access algorithms for multi-channel networks with mmtc/urllc terminals co-existence. *IEEE Access*, v. 8, p. 37608–37620, 2020. Cited 2 times on page(s) 23 and 33.

SAHA, S.; SUKUMARAN, V. B.; MURTHY, C. R. On the minimum average age of information in irsa for grant-free mmtc. *IEEE Journal on Selected Areas in Communications*, v. 39, n. 5, p. 1441–1455, 2021. Cited on page 36.

SHARMA, S. K.; WANG, X. Collaborative Distributed Q-Learning for RACH Congestion Minimization in Cellular IoT Networks. *IEEE Communications Letters*, v. 23, n. 4, p. 600–603, 2019. Cited 6 times on page(s) 36, 37, 39, 42, 44, and 62.

SILVA, M. V. da; SOUZA, R. D.; ALVES, H.; ABRÃO, T. A NOMA-Based Q-Learning Random Access Method for Machine Type Communications. *IEEE Wireless Communications Letters*, v. 9, n. 10, p. 1720–1724, 2020. Cited 4 times on page(s) 36, 38, 53, and 57.

SUTTON, R. S.; BARTO, A. G. *Reinforcement Learning: An Introduction*. 2. ed. Cambridge: The MIT Press, 2018. Cited 2 times on page(s) 34 and 56.

ULLAH, M. A.; MIKHAYLOV, K.; ALVES, H. Enabling mmtc in remote areas: Lorawan and leo satellite integration for offshore wind farms monitoring. *IEEE Transactions on Industrial Informatics*, p. 1–1, 2021. Cited on page 31.

WEERASINGHE, T. N.; CASARES-GINER, V.; BALAPUWADUGE, I. A. M.; LI, F. Y. Priority enabled grant-free access with dynamic slot allocation for heterogeneous mmtc traffic in 5g nr networks. *IEEE Transactions on Communications*, v. 69, n. 5, p. 3192–3206, 2021. Cited on page 36.

WHITE, D. J. *Markov Decision Processes*. 1. ed. Chichester: John Wiley & Sons, 1993. Cited on page 35.

YU, B.; CAI, Y.; WU, D. Joint access control and resource allocation for short-packet-based mmtc in status update systems. *IEEE Journal on Selected Areas in Communications*, v. 39, n. 3, p. 851–865, 2021. Cited on page 36.

ZHANG, Y.; LO, Y.-H.; LU, L.; SHU, F.; WONG, W. S. Protocol sequences for asynchronous multiple access with physical-layer network coding. *IEEE Wireless Communications Letters*, v. 8, n. 4, p. 980–983, 2019. Cited on page 34.

Appendix

APPENDIX A – Full paper published in the journal ”Physical Communication”

Title: Adjustable threshold LAS massive MIMO detection under imperfect CSI and spatial correlation.
Authors: Giovanni Maciel Ferreira Silva, José Carlos Marinello Filho, Taufik Abrão.
Journal: Physical Communication, ISSN 1874-4907, Volume 38, p. 100971.
Publication Date: Feb 2020.

Physical Communication 38 (2020) 100971

Contents lists available at ScienceDirect

Physical Communication

journal homepage: www.elsevier.com/locate/phycom

Full length article

Adjustable threshold LAS massive MIMO detection under imperfect CSI and spatial correlation

Giovanni Maciel Ferreira Silva, Jose Carlos Marinello Filho, Taufik Abrão [*]

Department of Electrical Engineering, State University of Londrina (DEEL-UEL). Rod. Celso Garcia Cid - PR445, Po. Box 10.011. CEP: 86057-970, Londrina, PR, Brazil

ARTICLE INFO

Article history:
Received 5 June 2019
Received in revised form 17 November 2019
Accepted 8 December 2019
Available online 13 December 2019

Keywords:
Massive MIMO
5G
Likelihood ascent search (LAS)
Adjustable threshold
Imperfect CSI
ULA correlation

ABSTRACT

In this paper, a likelihood ascent search (LAS) detector with adjustable threshold (ρ-LAS) is proposed for uplink (UL) massive multiple-input-multiple-output (M-MIMO) systems. The performance-complexity tradeoff for the ρ-LAS-based detector is extensively analyzed and compared with the conventional LAS M-MIMO detector by means of Monte Carlo simulations (MCS). Adjusting a threshold associated with the likelihood function, for each system and channel scenario, we found that the ρ-LAS is able to achieve better performance than the conventional LAS detector without complexity increment. Considering practical scenarios deteriorated by antenna correlation and imperfect channel state information (CSI) in M-MIMO systems, ρ-LAS detector has proven to be superior than the LAS detector in terms of performance while requires a fixed but very marginal additional number of computations. In addition, ρ-LAS provided a much better performance-complexity tradeoff in scenario with medium signal-to-noise ratio (SNR) and high number of antennas, a common operation scenario in M-MIMO systems. Finally, the ρ-LAS M-MIMO and three representative M-MIMO detection methods, namely the polynomial expansion (PE), the dual band Newton inversion (DBNI) and the iterative sequential detector (ISD), are compared. The results indicate a substantial performance-complexity tradeoff improvement for our proposed ρ-LAS detector.

1. Introduction

Low latency, high data rates, high quality of service (QoS) and high energy efficiency are the goals of the fifth generation (5G) of wireless communication systems [1]. One of the proposed technologies to integrate 5G is the massive multiple-input-multiple-output (M-MIMO) system [2], in which a large array of antennas can guarantee the demand for high data dates of users in 5G. M-MIMO systems provide a high gain in spectral efficiency (SE) and energy efficiency (EE) using only linear processing [3]. Improvements in SE occurs due to the high multiplexing gain, and EE is improved since the transmitter antennas can concentrate power only in the receiver direction [4,5].

In [6], it is demonstrated that the fast fading and uncorrelated noise effects disappear as the number of antennas grows without limit. However, other problems and effects still remain in such systems. One of the problems in M-MIMO systems is the uplink detection. With a large number of transmit antennas, low complexity detectors such as matched filtering (MF) have a poor performance. Detectors with matrix inversions such as zero forcing (ZF) and minimum mean square error (MMSE) become impracticable due to large matrix dimensions. Thus, detectors with good performance-complexity tradeoff are needed in M-MIMO systems.

One detector that combines low complexity with good performance in M-MIMO systems is the likelihood ascent search (LAS) detector [7]. LAS uses a low-complexity linear detector as a initial solution and changes iteratively the estimated symbols aiming to increase the likelihood function [8].

There is a wide family of LAS detectors [9–11], differing in how they choose the symbol candidates to be changed aiming to decrease the cost function or equivalently increase the likelihood function. For example, the way to change symbols can be sequential (SLAS), parallel (PLAS), global (GLAS) or in multistages (MLAS). In this work, we considered SLAS for simplicity. Some works [12–14] apply an optimization in the known LAS detector to improve performance or to decrease complexity order. Our focus in this work is to change the updating rule in order to improve performance. The updating rule from LAS detector is defined as the decision process to change a data symbol aiming at increasing the likelihood function or equivalently decrease the cost function. It sometimes can be more strict or smooth; hence, a parameter ρ can be aggregated to adjust such updating rule in order to decrease the bit error rate (BER) [15]. The choice of ρ value

[*] Corresponding author.
E-mail address: taufik@uel.br (T. Abrão).

https://doi.org/10.1016/j.phycom.2019.100971
1874-4907/© 2019 Elsevier B.V. All rights reserved.

depends on the system and channel operation characteristics, including signal-to-noise ratio (SNR) level, number of antennas, modulation order, and quality of channel state information (CSI) estimation. When the choice of ρ value reflects the best LAS bit-error rate (BER) performance, we called this detector as *adjustable threshold LAS* (ρ-LAS).

Recently, many detection techniques have emerged as low-complexity suitable-performance alternatives for M-MIMO systems. In [16,17], a polynomial expansion (PE) is used to estimate the inverse of channel matrix with less complexity than MMSE. In [18], it is considered a dual band Newton inversion (DBNI) method, where some non-diagonal elements of the channel matrix are disregarded. The PE and DBNI detectors estimate the transmitted symbol vector considering an low computational cost approximation for the inverse channel matrix $\mathbf{H}$. In this sense, the PE detector considers a polynomial approximation with K_{PE} order, whereas the DBNI detector considers a diagonal in band matrix, where the E_{DBNI} elements neighboring the main diagonal of the matrix $\mathbf{H}$ are considered. Besides, an iterative sequential detector (ISD) approach capable of removing interference from unwanted users in the received signal vector has been proposed in [19]. The ISD detector is an iterative detector similar to LAS that considers k_{ISD} iterations in its detection process.

MIMO detectors based on matrix inversion approximations present improved BER performance in systems operating with low antenna loading factor, *i.e.*, when the ratio between the number of transmitter antennas[1] and receiver antennas, $\mathcal{L} = \frac{n_T}{n_R} \ll 1$. However, under boundary conditions where the number of transmitting antennas becomes close to the receiving antennas, $\mathcal{L} \approx 1$, the performance of such detectors is remarkably worsened. In addition, the robustness of the M-MIMO detectors against spatial correlation due to antenna array is not discussed in the literature despite being a recurrent problem in M-MIMO.

In [20], the authors discuss an expectation propagation approximation (EPA) to reduce complexity in M-MIMO while maintaining good performance. To accomplish this, the authors explore channel hardening by considering very low antenna loading factor to avoid matrix inversion and reduce complexity. However, when considering channel matrix degradation by correlation or estimation error, channel hardening properties vanish. The authors do not analyze the performance of these detectors in such M-MIMO degrading scenarios. Moreover, in [21] the authors present a low-complexity detector that considers a reliability-feedback ordering aiming at improving performance regarding the linear techniques, such as ZF and MMSE, but being more complex than these detectors. Our technique can be less complex than linear detectors in certain scenarios because it does not apply any matrix inversion.

The authors in [22] discuss the LAS detector performance under channel estimation error scenarios. To improve bit-error-rate, the authors propose the use of the equivalent noise covariance matrix. However, it is still necessary to invert the covariance matrix, which also increases the complexity, resulting in a burden computation regarding our proposed technique.

The *contribution* of this work is twofold: firstly, we find a numerical factor ρ that adjusts the exchange of symbols in different signal-to-noise power ratio (SNR) scenarios and number of antennas, maximizing the M-MIMO system performance. The proposed procedure follows a different way from what was proposed in [15], where the ρ parameter was only generated for specific SNR and number of antennas scenarios. Secondly, the robustness of the proposed ρ-LAS detector is analyzed numerically in scenarios much more realistic, including antenna spatial correlation, caused by the arrangement of the antenna array, in opposition to [15–19] that only consider an imperfect channel state information caused by errors in the estimation process. Moreover, the analyzed detectors' performance takes into account practical scenarios with low and high antenna loading factor. Finally, we also evaluate the performance-complexity tradeoff for the proposed detector, while comparing with recent M-MIMO techniques proposed in [16–19].

The remainder of this paper is organized as follows. Section 2 presents the M-MIMO system model and detection techniques used in this work. In Section 3 we describe the proposed ρ-LAS M-MIMO detector. The numerical simulation analysis, including performance, complexity and comparison with other representative M-MIMO detector are driven in Section 4. Final remarks and conclusions are offered in Section 5.

2. System model

We consider a point-to-point M-MIMO uplink system with n_T antennas at the transmitter and n_R antennas at the receiver. The jth transmitter antenna transmits the symbol x_j. The ith receiver antenna receives the symbol x_j multiplied by a complex Gaussian channel gain h_{ij} with zero mean and unit variance. The received signal is also deteriorated by an additive white Gaussian noise (AWGN) sample n_i, with zero mean and power spectral density N_0. Considering all the n_T antennas, the symbol y_i received on the ith antenna is given by

$$y_i = \sum_{j=1}^{n_T} h_{ij} x_j + n_i. \tag{1}$$

We can rewrite Eq. (1) in the matrix notation to cover all the n_R receiving antennas:

$$\mathbf{y_c} = \mathbf{H_c}\mathbf{x_c} + \mathbf{n_c}, \tag{2}$$

where $\mathbf{y_c}$ is the $n_R \times 1$ BS received signal complex vector, $\mathbf{x_c}$ is the $n_T \times 1$ UE transmitted signal complex vector, $\mathbf{H_c}$ represents the $n_R \times n_T$ complex channel gain matrix between UEs and BS and $\mathbf{n_c}$ is the $n_R \times 1$ complex AWGN vector.

Each transmitted symbol x_j can assume a complex value given by the complex constellation set $\mathcal{S}_c$. For the M-QAM modulation, the symbol set is defined as $\mathcal{S}_c = \{a + jb \mid a, b \in \{-\sqrt{M} + 1, -\sqrt{M} + 3, \ldots, -3, -1, +1, +3, \ldots, \sqrt{M} - 3, \sqrt{M} - 1\}\}$.

It is usual to transform the complex system model in Eq. (2) into a real model to facilitate the analysis of detection techniques [8]. The dimensions of the vectors and matrices are doubled, separating the complex variables into their real and imaginary parts. The vectors $\mathbf{y_c}$, $\mathbf{x_c}$ and $\mathbf{n_c}$ are transformed into $\mathbf{y_r} = [\mathbb{R}(\mathbf{y_c})^T \ \mathbb{I}(\mathbf{y_c})^T]^T$, $\mathbf{x_r} = [\mathbb{R}(\mathbf{x_c})^T \ \mathbb{I}(\mathbf{x_c})^T]^T$ and $\mathbf{n_r} = [\mathbb{R}(\mathbf{n_c})^T \ \mathbb{I}(\mathbf{n_c})^T]^T$. The matrix $\mathbf{H_c}$ becomes $\mathbf{H_r}$ given by

$$\mathbf{H_r} = \begin{bmatrix} \mathbb{R}(\mathbf{H_c}) & -\mathbb{I}(\mathbf{H_c}) \\ \mathbb{I}(\mathbf{H_c}) & \mathbb{R}(\mathbf{H_c}) \end{bmatrix}. \tag{3}$$

Thus, the new model for the received signal vector is given by

$$\mathbf{y_r} = \mathbf{H_r}\mathbf{x_r} + \mathbf{n_r}. \tag{4}$$

For convenience, the subscript r of Eq. (4) is disregarded. Now, the complex set $\mathcal{S}_c$ for the M-QAM modulation becomes the real equivalent set $\mathcal{S}$ for $\sqrt{M}$-PAM modulation, given by $\mathcal{S} = \{w \mid w \in \{-\sqrt{M} + 1, -\sqrt{M} + 3, \ldots, -3, -1, +1, +3, \ldots, \sqrt{M} - 3, \sqrt{M} - 1\}\}$.

Along this work, we have assumed a power allocation technique that reverses the effect of path-loss. Hence, we can consider only the small-scale fading for simplicity while including the long-term effect on the SNR variation.

[1] Or equivalently, the number of users equipped with a single transmitting antenna.

2.1. Maximum likelihood (ML) detector

The optimal detector which finds the signal vector with the maximum probability of being transmitted based on the observed received signal, described by Eq. (4), is the maximum likelihood (ML) detector. The equation that represents the estimated signal by ML is [23]

$$\widehat{\mathbf{x}}_{\text{ML}} = \underset{\mathbf{x}\in\mathcal{S}^{2n_T}}{\arg\min}\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2, \tag{5}$$

where $\mathcal{S}$ is the real equivalent symbol set depending on the modulation order. ML detector searches exhaustively the best solution in the symbol set. When considering M-QAM modulation, for example, it is observed that the complexity of ML, $\mathfrak{C}_{\text{ML}}$, is exponential with the number of transmitting antennas n_T regarding the modulation order M, i.e., $\mathfrak{C}_{\text{ML}} \propto M^{n_T}$ [24–26], which is very large when the number of antennas and modulation order increase, becoming impracticable in M-MIMO systems because its large number of antennas.

2.2. Linear detectors

Sub-optimal linear detectors with reduced complexity can be used, such as matched filtering (MF) and minimum mean square error (MMSE). The estimated signal by a linear detector can be written as

$$\widehat{\mathbf{x}} = \mathbf{W}\mathbf{y}, \tag{6}$$

where $\mathbf{W}$ is the linear transformation matrix. The corresponding transformation matrices for the MF and MMSE MIMO detectors are given, respectively, by [8]

$$\mathbf{W}_{\text{MF}} = \mathbf{H}^T, \tag{7}$$

and

$$\mathbf{W}_{\text{MMSE}} = \left(\mathbf{H}^T\mathbf{H} + \frac{N_0}{E_s}\mathbf{I}\right)^{-1}\mathbf{H}^T, \tag{8}$$

where E_s is the average symbol energy. As the MF MIMO detector is not able to eliminate all the multi-antenna inter-users interference, it presents a BER floor and a poor performance in M-MIMO scenarios. On the other hand, the MMSE detector applies matrix inversion, which becomes impracticable as the number of antennas increases. Recent proposed M-MIMO detectors such as PE [16,17] and DBNI [18] utilize an approximation of the inverse matrix aiming to reduce complexity. However, such low-complexity detection approaches have a drawback of performance loss in comparison to the exact inversion matrix performed by the pure MMSE.

2.3. Likelihood ascent search (LAS) detector

In M-MIMO systems, linear detectors lose performance. To circumvent this loss, iterative detectors can be used as an alternative between linear detectors and ML. One of the most deployed iterative detectors is the *likelihood ascent search* (LAS) detector. LAS is based on ML detector. Considering the cost function decreasing goal in Eq. (5), and only the terms dependent of $\mathbf{x}$, one can write the likelihood function of $\mathbf{x}$ as the negative of the cost function [27]:

$$\Lambda(\mathbf{x}) = 2\mathbf{x}^T\mathbf{H}^T\mathbf{y} - \mathbf{x}^T\mathbf{H}^T\mathbf{H}\mathbf{x}. \tag{9}$$

Using a linear detector as the initial solution, the goal is to increase the likelihood function at every single step m of the algorithm, *i.e.*, $\Lambda(\mathbf{x}^{(m+1)}) - \Lambda(\mathbf{x}^{(m)}) \geq 0$, until a fixed number of steps n_{F} has been performed. The selected symbols to increase likelihood function can be chosen in a few ways, typically applying the *bit flip* rule. In this work, we consider the sequential LAS (SLAS) procedure. It is sequential because the bit flip rule is performed in all antennas from the first one to the last one, sequentially. After a complete sequential search is done, the search returns to the first antenna and keeps circularly until a fixed number of steps n_{F} is reached. A factor k can be defined to represent the ratio between the maximum number of steps and the number of antennas:

$$k = \frac{n_{\text{F}}}{2n_T}. \tag{10}$$

If $k > 1$, the update rule will be performed more than one time in each antenna. If $k < 1$, the search will stop before applying the bit flip rule over all the antennas.

One can apply the gradient principle to verify the likelihood function increasing values, which is a well-known method to obtain the greatest increasing direction of the function. Hence, defining:

$$\mathbf{y}_{\text{eff}} = 2\mathbf{H}^H\mathbf{y}, \tag{11}$$

$$\mathbf{H}_{\text{eff}} = \mathbf{H}^T\mathbf{H}, \tag{12}$$

and

$$\mathbf{H}_{\text{real}} = 2\mathbb{R}(\mathbf{H}_{\text{eff}}) = 2\mathbf{H}_{\text{eff}}, \tag{13}$$

we can find the gradient $\mathbf{g}^{(m)}$ simply:

$$\mathbf{g}^{(m)} = \frac{\partial(\Lambda(\mathbf{x}^{(m)}))}{\partial(\mathbf{x}^{(m)})} = \mathbf{y}_{\text{eff}} - \mathbf{H}_{\text{real}}\mathbf{x}^{(m)}. \tag{14}$$

The second derivative

$$\mathbf{g}'^{(m)} = \frac{\partial^2(\Lambda(\mathbf{x}^{(m)}))}{\partial(\mathbf{x}^{(m)})^2} = -\mathbf{H}_{\text{real}}, \tag{15}$$

indicates whether there is a local maximum or minimum in the neighborhood of the local search. For good operation of the LAS detector, it is necessary that the likelihood function never decreases after a symbol update. In [28,29], it can be seen that, if the gradient entry of the jth antenna, namely $g_j^{(m)}$, exceeds a threshold ζ_j in a bit flip update, it is guaranteed that the likelihood function will increase if this symbol is updated. The threshold value may change depending on the type of search performed. A suitable threshold ζ_j value for SLAS is given by the absolute value of the $\mathbf{g}'^{(m)}$ applied to the jth antenna, as considered in [7]:

$$\zeta_j = |(\mathbf{H}_{\text{real}})_{j,j}|. \tag{16}$$

As a result, the updating bit flip rule for the symbol from the jth antenna, x_j, can be written as

$$x_j^{(m+1)} = \begin{cases} x_j^{(m)} + 2, & \text{if } g_j^{(m)} > \zeta_j, \\ x_j^{(m)} - 2, & \text{if } g_j^{(m)} < -\zeta_j, \\ x_j^{(m)}, & \text{otherwise,} \end{cases} \tag{17}$$

taking care to maintain $x_j^{(m+1)}$ in the symbol set $\mathcal{S}$. The algorithm runs until $m = n_{\text{F}}$. Then, $\mathbf{x}^{(n_{\text{F}})}$ is the final data vector estimated by the LAS algorithm.

The symbol exchange rule applied in Eq. (17) has been adapted for convenience to $\sqrt{M}$-PAM modulations. However, as the gradient $\mathbf{g}^{(m)}$ indicates the direction of highest growth of the likelihood function given a vector of symbols defined in the complex plane, it is possible to deploy Eq. (17) to any modulation provided that the constellation symbols can be separated into their real and imaginary parts.

3. Proposed adjustable threshold LAS (ρ-LAS)

In some scenarios with variable SNR, channel conditions, number and spatial correlation of antennas, the bit flip rule needs to be more smooth or more strict [15]. Hence, in this work, a factor ρ is included by fitting means aiming to better adjust the threshold factor depending on the scenario parameters such as SNR and number of antennas. Based on (16), the new proposed threshold factor $\widetilde{\zeta_j}$ is written as

$$\widetilde{\zeta_j} = \rho \ |(\mathbf{H}_{\mathrm{real}})_{j,j}|, \tag{18}$$

where ρ is the factor that gives the best improvement in performance. As demonstrated in Section 3.1, the optimal ρ value fluctuates around 1.0, typically in the range $\rho \in [0.7; \ldots; 1.3]$, depending on the number of antennas n, SNR region and fading channel condition. The LAS in [7] consider only $\rho = 1$; however, we demonstrate numerically in Section 3.1 that finding the appropriate ρ factor constitutes a way to improve systematically the BER performance with no further complexity increment. Besides, it is possible to find the optimal threshold $\widetilde{\zeta_j}$ using the suitable ρ factor for the chosen scenario. A pseudo-code for the proposed ρ-LAS detector is depicted in Algorithm 1.

Algorithm 1 ρ-LAS M-QAM M-MIMO detector

```
Input: $\mathbf{y}$, $\mathbf{H}$, $\mathbf{x}^{(0)}$, $k$, $\rho_{\mathrm{fit}}^{(2,4)}$, $M$, $n_T$, $n_R$
Output: $\mathbf{x}^{(n_{\mathrm{F}})}$
Calculate $n_{\mathrm{F}}$, Eq. (10)
Calculate $\mathbf{y}_{\mathrm{eff}}$, Eq. (11)
Calculate $\mathbf{H}_{\mathrm{real}}$, Eq. (13)
$j$ = 1, $m$ = 0
while $m < n_{\mathrm{F}}$ do
    if $j > n_T$ then
        $j \leftarrow 1$
    else
        Calculate $g_j^{(m)} = (\mathbf{y}_{\mathrm{eff}})_j - (\mathbf{H}_{\mathrm{real}})_{(j,:)}\mathbf{x}^{(m)}$, Eq. (14)
        Calculate threshold $\widetilde{\zeta_j}$, Eq. (18)
        if $g_j^{(m)} > \widetilde{\zeta_j}$ then
            $x_j^{(m+1)} \leftarrow x_j^{(m)} + 2$
            if $x_j^{(m+1)} > \sqrt{M} - 1$ then
                $x_j^{(m+1)} \leftarrow \sqrt{M} - 1$
            end if
        else if $g_j^{(m)} < -\widetilde{\zeta_j}$ then
            $x_j^{(m+1)} \leftarrow x_j^{(m)} - 2$
            if $x_j^{(m+1)} < -\sqrt{M} + 1$ then
                $x_j^{(m+1)} \leftarrow -\sqrt{M} + 1$
            end if
        else
            $x_j^{(m+1)} \leftarrow x_j^{(m)}$
        end if
        $j \leftarrow j + 1$
        $m \leftarrow m + 1$
    end if
end while
Solution: $\mathbf{x}^{(n_{\mathrm{F}})}$
End
```

3.1. Threshold fitting

A fitting process was carried out aiming at finding a polynomial equation that approximates the optimal ρ factor values for the ρ-LAS M-MIMO detector operating under different channel and system scenarios. As one can see in [15], ρ depends on the analyzed scenario, especially SNR level and number of antennas, n. Hence, we have evaluated the BER performance for γ from 0 to 20 dB, $n_T = n_R = n$ from 15 to 225, while factor ρ was chosen in Eq. (18) in the sense that gives the smallest BER. Although it is generally considered $n_R \gg n_T$ on Massive MIMO systems to suppress intra-cellular interference, $n_R = n_T$ has been considered in this work for simplify of the analysis, and at same time representing the worst case detection performance scenario. Notice that although even the simplest detectors are able to perform suitably with $n_R \gg n_T$, the scenario $n_R = n_T$ requires much more advanced detection strategies. After that, we fitted a polynomial surface to indicate the suitable ρ as a function of SNR (γ) and the number of antennas (n). It was chosen the smallest polynomial fit by selecting the 2nd order dependency in γ and 4th order in n, as an attempt to hold low-complexity implementation combined with good performance improvement. The fitted surface is depicted in Fig. 2 and the associated polynomial equation is given by

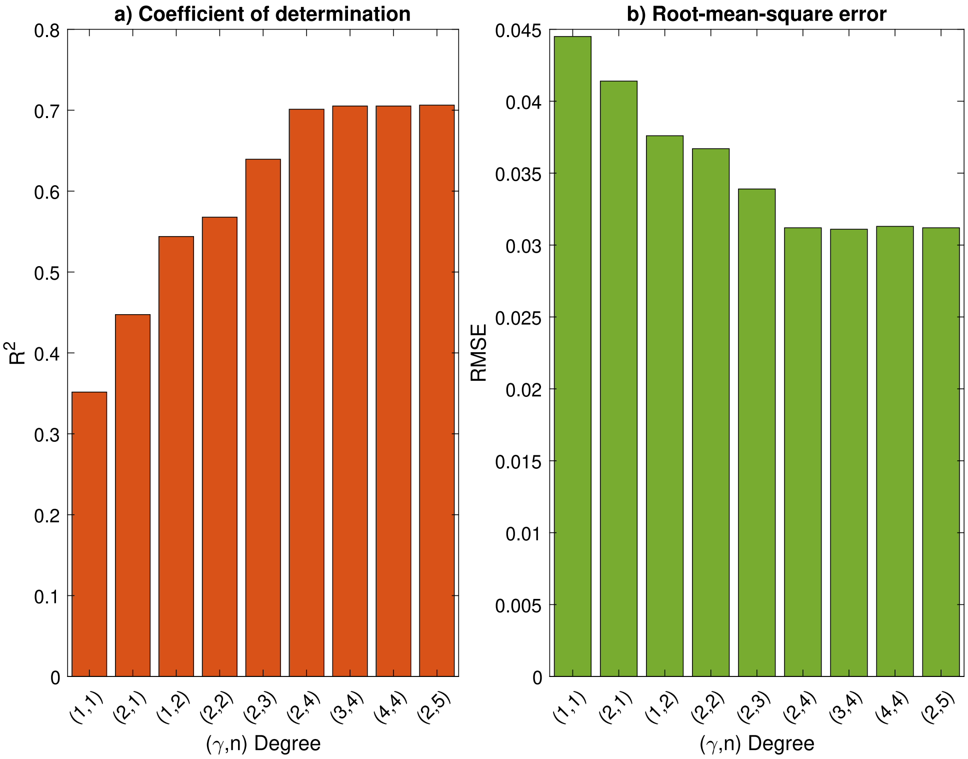
$$\begin{aligned}\rho_{\mathrm{fit}}^{(2,4)}(\gamma, n) = {} & p_{00} + p_{10}\gamma + p_{01}n + p_{11}\gamma n \\ & + p_{20}\gamma^2 + p_{02}n^2 + p_{21}\gamma^2 n + p_{12}\gamma n^2 \\ & + p_{22}\gamma^2 n^2 + p_{03}n^3 + p_{13}\gamma n^3 + p_{04}n^4\end{aligned} \tag{19}$$

Table 1
Fit parameters.

Parameter	Value
p_{00}	0.831
p_{10}	0.004336
p_{01}	0.002674
p_{20}	-0.0007697
p_{11}	0.0001584
p_{02}	$-3.56 \cdot 10^{-5}$
p_{21}	$1.357 \cdot 10^{-5}$
p_{12}	$-2.49 \cdot 10^{-6}$
p_{03}	$2.297 \cdot 10^{-7}$
p_{22}	$-6.106 \cdot 10^{-8}$
p_{13}	$9.971 \cdot 10^{-9}$
p_{04}	$-5.632 \cdot 10^{-10}$
R-square	0.7012
RMSE	0.03118

with coefficients values given in Table 1. Although the R-squared value resulted a little bit below the ideal value, the BER performance of the proposed ρ-LAS detector has resulted substantially improved by employing the fitted ρ values, as demonstrated in Section 4. Moreover, increasing the fit order did not lead to significant improvements in the coefficient of determination (R^2) and root-mean-square error (RMSE). Therefore, we selected the 2nd order for SNR γ and 4th order for number of antennas n, which proved to be sufficient to hold reduced the fitting complexity and simultaneously improve BER performance regarding the conventional LAS detector. One can infer on this choice (2nd polynomial degree for γ and 4th order for n) inspecting Fig. 1, where the RMSE and R^2 were determined for different polynomial degrees in γ and n. Indeed, there are no substantial improvement on the selected quality parameters beyond (2, 4) polynomial degree for (γ, n). Therefore, this pair of degree is chosen for all numerical simulations as the minimum degree that preserve the quality of fitting for all data points.

In Fig. 2(a), it can be seen that ρ is between 0.7 and 0.8 for low number of antennas n, it increases until 1.0 or 1.05 for a medium n and start to decrease in M-MIMO scenarios. For γ, one can confirm a similar tendency, starting with low values for low γ, increasing for medium γ and then decreasing for high γ, just as a quadratic function. In the remainder numerical results presented in this work, Eq. (19) has been deployed in simulations as the best ρ factor values for the ρ-LAS. Besides, in Fig. 2(b) the ρ variation


a) Coefficient of determination

b) Root-mean-square error

Fig. 1. Quality parameters of the polynomial surface fitting as a function of degrees in γ and n: (a) Coefficient of determination (R^2); (b) Root-mean-square error (RMSE).

can be examined in a typical M-MIMO scenario, where there is a low-medium SNR on the receiver and a high number of antennas at the transmitter. When the number of antennas tends to 200 and the SNR tends to 0 dB, it is observed that the factor $\rho \approx 0.8$, being much different than the value of 1.0 preconized in [7].

To corroborate how near the $\rho_{\text{fit}}(\gamma, n)$ obtained by fitting, Eq. (19), is from the optimum ρ_{opt} found exhaustively by numerical simulation, the BER performance of both ρ-LAS detectors are exhibited in Fig. 4 and discussed in details in the next section. To summarize, the gap in terms of BER performance is insignificant for high number of antennas 60 × 60 and a wide SNR range, $\gamma = [0\ 1\ 2\ \ldots, 9\ 10]$ dB in Fig. 4(a), demonstrating the effectiveness of the proposed method to expeditiously find suitable near-optimal $\rho(\gamma, n)$ values.

4. Numerical results: Performance and complexity

We considered Monte-Carlo simulation (MCS) method to analyze performance and complexity of ρ-LAS compared with other representative low-complexity MIMO detectors. The numerical results are classified into four groups analysis:

(i) ρ-LAS convergence;
(ii) ρ-LAS BER performance analysis;
(iii) Realistic scenarios analysis, including antenna array correlation and imperfect channel estimates;
(iv) BER performance-complexity tradeoff.

Extensive simulation results are provided by means of MCS method, considering at least 500 realizations for suitable bit error rate averages; for each realization, random information, additive noise, and short-term fading samples are generated. The antenna arrangement is changed by the number of receiving antennas n_R and a loading factor $\mathcal{L} = \frac{n_T}{n_R}$. When $\mathcal{L} = 1$, then we denote $n = n_R = n_T$. Besides, in the following numerical results, we set the arrangement of the number of system antennas in the format $n_R \times n_T$. To make the scenario more realistic, we considered an uniform linear array (ULA) with correlation index κ and imperfect CSI with estimation error index τ. Both effects and parameters are modeled and analyzed in Section 4.3.

Table 2
MCS parameters.

Parameter	Adopted value
Adjusting threshold factor	$\rho = [0.7\ 0.8\ 0.9\ 1.0\ 1.1\ 1.2\ 1.3]$
Number of Rx antennas	$n_R = [10\ 15\ 20\ \cdots\ 215\ 220\ 225]$
Loading factor, $\frac{n_T}{n_R}$	$\mathcal{L}_{\%} = [6.25\ 12.5\ 25\ 50\ 100]\%$
Correlation index	$\kappa = [0.0\ 0.1\ 0.2]$
CSI error index	$\tau = [0.0\ 0.1\ 0.3]$
BPSK/M-QAM order	$M = [2\ 4\ 16\ 64]$
SNR	$\gamma = [0\ 1\ 2\ \cdots\ 18\ 19\ 20]$ dB
MIMO detector parameters	
Steps of ρ-LAS	$k = [1\ 2\ 3\ \cdots\ 9\ 10]$
Initial solution $\mathbf{x}^{(0)}$	MF detector output
PE maximum order	$K_{\text{PE}} = 2$
DBNI bandwidth	$E_{\text{DBNI}} = 2$
ISD iterations	$k_{\text{ISD}} = 4$
MCS realizations	$\mathcal{I} = 5 \cdot [10^2\ 10^3\ 10^4\ 10^5\ 10^6]$ trials

As a comparison to the proposed technique, some recent detection schemes were considered, such as PE [16,17], DBNI [18] and ISD [19]. In this work, we fix $k_{\text{ISD}} = 4$, $K_{\text{PE}} = 2$ and $E_{\text{DBNI}} = 2$. Different M-QAM modulation orders, sequential LAS and a wide range of number of antennas and loading factors have been considered in the numerical analysis. The vector for the first iteration of LAS and ρ-LAS, $\mathbf{x}^{(0)}$, is obtained from MF, Eq. (7). Table 2 summarizes the main simulation parameters values adopted across this section, considering values similar to those used in [7,25,30].

4.1. ρ-LAS convergence

To analyze the ρ-LAS convergence, we fixed the loading factor $\mathcal{L}_{\%} = 100\%$ for some γ values, while varying the iteration factor k

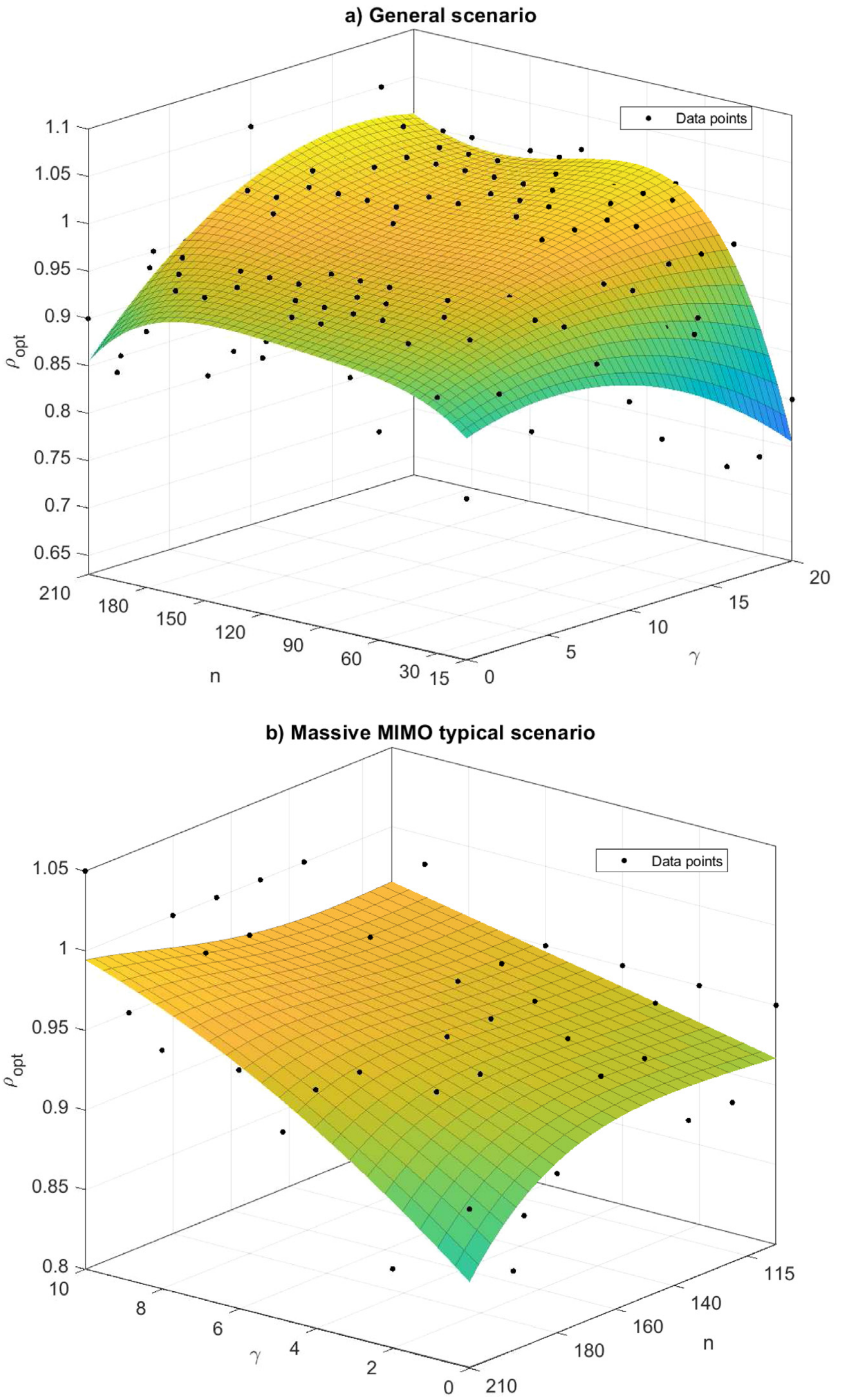


Fig. 2. ρ polynomial surface as a function of SNR γ and number of antennas n, considering $k = 5$ and $\mathbf{x}^{(0)}$ given by matched filter (MF) detector. (a) General scenario covering low to high SNR regions and number of antennas; (b) Zoom in on a typical M-MIMO scenario with low SNR and high number of antennas.

from 1 to 10 to identify how many iterations the ρ-LAS needs to achieve full convergence. Fig. 3 exhibits BER convergence results for two scenarios: (a) $\gamma = 5$ dB and changing n for three arrangements: 10×10, 40×40 and 70×70 antennas; (b) 70×70 for three SNR values, $\gamma = [0\ 5\ 10]$ dB. From Fig. 3 one can infer that, by increasing γ and n, the BER decreases asymptotically until the full algorithm's convergence. Also, from Fig. 3(a) one can see that the ρ-LAS algorithm achieves full convergence after $k = 4$ steps for all the three n antenna arrangements. Notice that increasing n to 70×70 (M-MIMO scenario) just increases k marginally. The same occurs changing the SNR in Fig. 3(b). It is noted that the ρ-LAS detector outperforms conventional LAS without requiring further steps, since both detectors converge at $k \approx 4$. Therefore, one can conclude that $k = 4$ steps is enough to guarantee the ρ-LAS full convergence in all scenarios. Hence, we have fixed $k = 4$ in all remainder numerical results and analysis.

4.2. BER performance vs. SNR for different number of antennas

To corroborate the performance gain of the proposed ρ-LAS M-MIMO detector, this subsection analyzes the BER as a function of γ and n. We split such analysis into two case scenarios. First, we considered the BER × γ analysis fixing n. Secondly, we considered the BER × n analysis fixing γ.

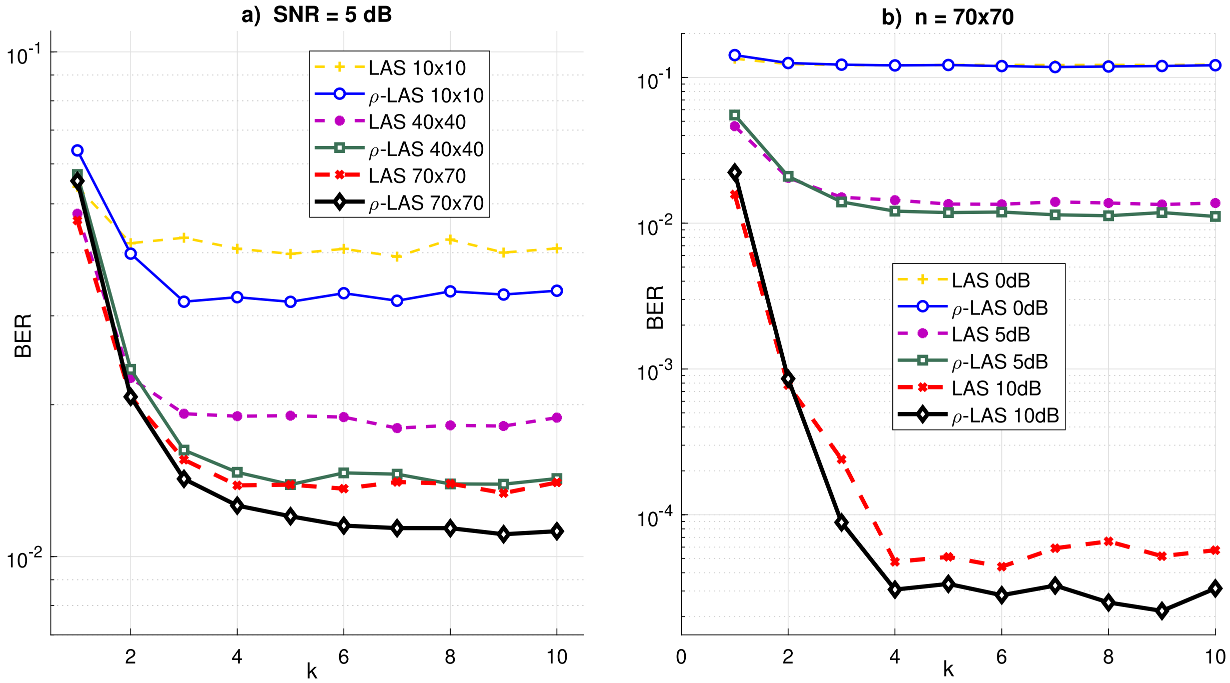


Fig. 3. LAS and ρ-LAS convergence analysis. (a) fixing γ and changing n. (b) fixing n and changing γ. BPSK modulation and MF as the initial solution of ρ-LAS.

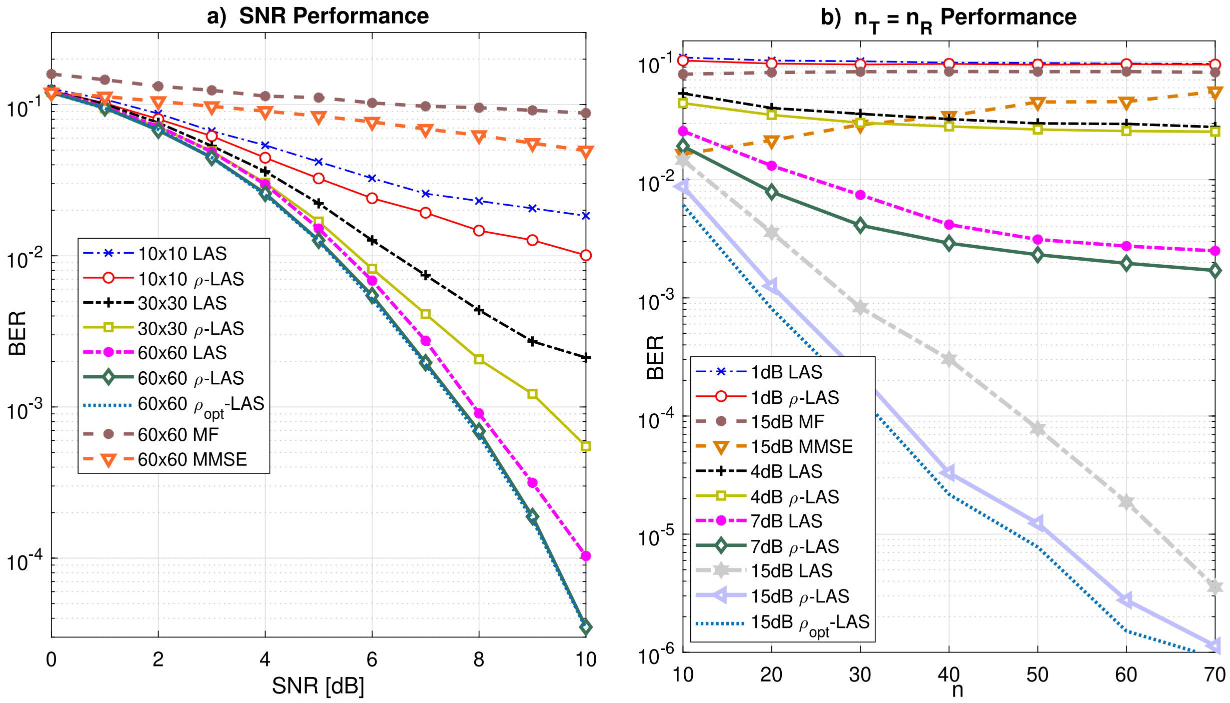


Fig. 4. Performance comparison between LAS and ρ-LAS in two arrangements: (a) BER$\times\gamma$; (b) BER$\times n$. It was considered $k = 4$ and MF detector as the initial solution ($\mathbf{x}^{(0)}$).

Fig. 4(a) depicts the BER performance comparison of M-MIMO detectors with the conventional LAS and linear MF and MMSE detectors against the proposed ρ-LAS considering $n = 10 \times 10$, 30×30 and 60×60 antennas while changing γ from 0 to 10 dB. For reference, the ρ_{opt}-LAS detector, which hypothetically find the ρ factor by exhaustive search in the range of $\rho \in [0.5 : 0.1 : 1.5]$, is included in Fig. 4 as the achievable lower bound for BER. Notice that linear detectors perform poorly considering a large number of antennas due to the inter-antenna interference. For $n = 60$ antennas and with increasing SNR, it is possible to observe that the MF presents a BER floor, because it cannot eliminate the interference between the antennas. It is observed that MMSE outperforms MF in the high SNR region.

ρ-LAS notably provides a superior BER performance w.r.t. the LAS detector for all considered γ range and n arrangements. As expected, when the number of antennas increases, e.g., M-MIMO scenarios, BER decreases. For instance, in quasi M-MIMO scenarios (60×60), this feature becomes interesting. ρ-LAS provides a BER of $3 \cdot 10^{-5}$ with a moderate SNR ≈ 10 dB, representing a SNR gain of ≈ 1 dB w.r.t conventional LAS.

Fig. 4(b) depicts the BER scenarios for changing the number of antennas n from 10 to 70 considering low, medium and high SNRs, i.e. $\gamma \in [1\ 7\ 15]$ dB. Linear detectors, even under high SNR of $\gamma = 15$ dB, tend to perform similarly to the conventional LAS detector operating with only $\gamma = 1$ dB under large number of antennas, i.e., above 60×60 antennas. Besides, the MMSE detector presents a BER $\approx 10^{-2}$ under 10×10 antennas. But its performance worsens with the increase in the number of antennas, reaching a BER of 5×10^{-2} under 70×70 antennas.

To corroborate the suitable performance gain of ρ-LAS regarding conventional LAS, it can be seen on Fig. 4(b) that ρ-LAS can provide a BER of 10^{-5} requiring 13 antennas less than LAS detector in a moderate-high SNR regime ($\gamma = 15$ dB). Therefore, ρ-LAS can be considered an appropriate choice for the uplink M-MIMO detection problem by providing a substantial BER performance gains in such large antenna scenarios.

It is worth noticing that applying the surface fitting procedure for ρ, Eq. (19), has resulted a performance very close to ρ_{opt}. In Fig. 4(a), the BER curves of ρ_{opt} and ρ_{fit} for 60×60 antennas are almost overlapping, presenting no significant difference in performance. For a higher SNR ($\gamma = 15$ dB), as in Fig. 4(b), ρ_{opt}-LAS detector performs better than the fitted one ρ-LAS. However, since ρ_{opt} is only found by an exhaustive search, then the adjusted factor ρ_{fit} is used to reduce the complexity of ρ-LAS detector, while guaranteeing a substantial performance improvement over the conventional LAS M-MIMO detector.

4.2.1. Low-complexity M-MIMO detectors comparison

The performance of ρ-LAS detector was compared to the PE, DBNI, MMSE and ISD detectors. To enjoy the benefits of channel hardening, where effective channel gains tend to be deterministic as the number of receiving antennas increases [8], and to take advantage of the favorable propagation that occurs when channel vectors tend to be orthogonal when n_R increases [31], the number of receiving antennas was set to $n_R = 64$ and the number of transmitting antennas was fixed in $n_T = [4,\ 8,\ 16]$, resulting in loading factors $\mathcal{L} \in \left[\frac{1}{16},\ \frac{1}{8},\ \frac{1}{4}\right]$, respectively. Fig. 5 depicts the BER performance in terms of SNR values. As the number of transmitting antennas increases, the degradation of BER in all detectors increases too. In all scenarios, it is observed that the ρ-LAS detector presents the best performance among the analyzed detectors, even if marginally, as observed in the scenario with the highest load factor (1/4). With this result, it is possible to observe that, even in scenarios with low antenna loading factors ($n_R \gg n_T$), the proposed detector is able to present a satisfactory and superior performance to the other low-complexity MIMO detectors. Because the performance of detectors based on channel matrix inversion (PE, DBNI and MMSE) are highly dependent on good favorable propagation, the BER is strongly degraded in realistic high loading factor scenarios.

The performance was also analyzed as a function of the modulation order. The antenna configuration was set at 64×16 and modulation order was changed to 4, 16, and 64-QAM. Fig. 6 shows the analyzed scenarios. Under high order modulation (64-QAM), the MMSE detector can eliminate multi-antenna interference between users. Therefore, it can decrease BER monotonically, while other detectors present a BER floor. However, it needs a high SNR to exceed the iterative detectors ISD, LAS and ρ-LAS. The PE and DBNI detectors consider an approximation of the inverse channel matrix, while the ISD detector considers only the diagonal elements of the matrix; on the other hand, the both LAS-based detectors use the MF as the initial solution. Interestingly, for low and medium modulation order (4 and 16-QAM), it is observed that the proposed ρ-LAS detector has resulted the best performance regarding the other low-complexity LS MIMO detectors.

4.3. Channel estimation error and antenna correlation effects

Correlation between the antennas and errors in channel estimation should be considered in practical M-MIMO scenarios. Herein, we have analyzed two arrangements that model such impairments in M-MIMO schemes. Firstly, we included imperfect CSI at the receiver side. Secondly, a correlation index modeling the spatial channel correlation caused by insufficient distance between the antennas was considered.

The channel matrix estimation can be modeled as [30]

$$\widehat{\mathbf{H}} = \sqrt{1-\tau^2}\mathbf{H} + \tau \mathbf{\Delta} \boldsymbol{H}, \tag{20}$$

where $\Delta\mathbf{H}$ is the estimation error matrix whose entries are modeled by a complex Gaussian variable with zero mean and unit variance. When $\tau = 0$, the channel gain matrix is perfectly estimated; besides, τ in the range $]0;\ 0.3]$ has been considered in this subsection to describe more realistic M-MIMO detection scenarios.

The correlation between receiver and transmitter antennas has been modeled considering a ULA in which antennas are disposed linearly in the axis. Under the ULA, the correlation between the ith and jth antenna-elements is modeled by [25]:

$$c_{ij} = \kappa^{(i-j)^2}, \tag{21}$$

where κ is the correlation index; when $\kappa = 0$, there is no correlation between the two antennas.

The matrices $\mathbf{C}_{\text{Rx}}$ and $\mathbf{C}_{\text{Tx}}$ describe the correlation between the antenna-elements at the receiver and the transmitter side, respectively, being composed by matrix entries written in Eq. (21). The overall correlated channel gain matrix $\tilde{\mathbf{H}}$ is given by

$$\tilde{\mathbf{H}} = \sqrt{\mathbf{C}_{\text{Rx}}}\mathbf{H}\sqrt{\mathbf{C}_{\text{Tx}}}, \tag{22}$$

where $\mathbf{H}$ is the uncorrelated channel matrix. In the special case of $n = n_T = n_R$, and assuming that κ is the same at both transmitter and receiver, the correlation matrices of the receiver and the transmitter are equal and given by [25]

$$\mathbf{C} = \begin{bmatrix} 1 & \kappa & \kappa^4 & \dots & \kappa^{(n-1)^2} \\ \kappa & 1 & \kappa & \dots & \vdots \\ \kappa^4 & \kappa & 1 & \dots & \kappa^4 \\ \vdots & \vdots & \vdots & \ddots & \kappa \\ \kappa^{(n-1)^2} & \dots & \kappa^4 & \kappa & 1 \end{bmatrix}. \tag{23}$$

Fig. 7 depicts BER performance for three uncorrelated scenarios with different values for the channel estimation error index, $\tau = [0,\ 0.1,\ 0.3]$ for Fig. 7(a), (b) and (c), respectively. Generically, the increasing in the channel error estimation causes a BER performance deterioration in all the analyzed detectors. Specifically, the PE and DBNI detectors consider a matrix inversion approximation; hence, both detectors are not able to completely eliminate the interference between the antennas, presenting a strong BER deterioration w.r.t. SNR, combined to an irreducible BER floor, which is accentuated with the τ index increasing. However, in all analyzed scenarios of Fig. 7, the ρ-LAS M-MIMO detector presented the best performance which indicates that, even if the ρ factor was found for a scenario with perfect CSI (Table 1), the same fitting procedure could be deployed to further improve the ρ-LAS performance achieved in Fig. 7.

The scenario analyzed in Fig. 8 considers perfect CSI but with spatial correlation between the antennas, with correlation indexes $\kappa = [0\ 0.1\ 0.2]$ in Fig. 8(a), (b) and (c), respectively. The PE and DBNI detectors are quite sensitive to channel correlation, resulting in a strong BER performance degradation in scenarios with low-medium spatial correlation, since they consider an inverse

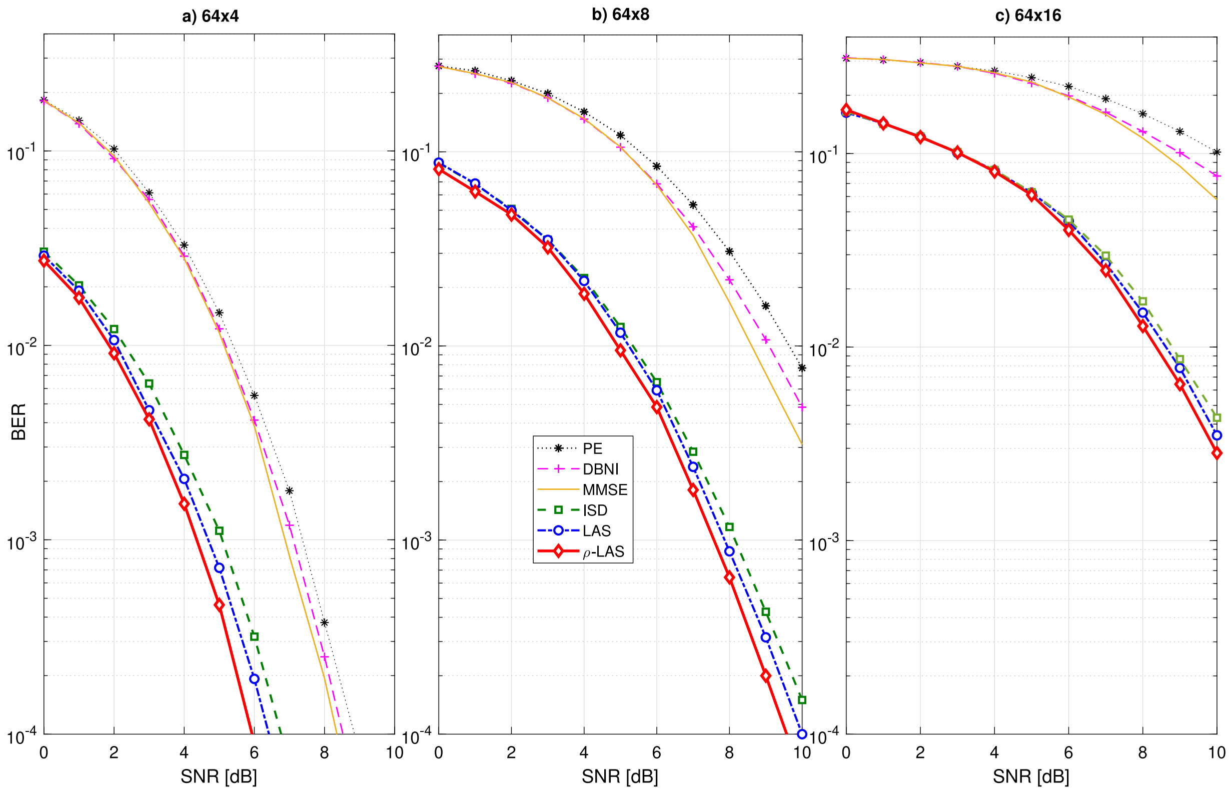


Fig. 5. BER performance in three $n_R \times n_T$ antenna configurations: (a) 64 × 4; (b) 64 × 8; (c) 64 × 16. It was considered 16-QAM, $k = 4$ and MF detector as the initial solution $(\mathbf{x}^{(0)})$ of LAS and ρ-LAS.

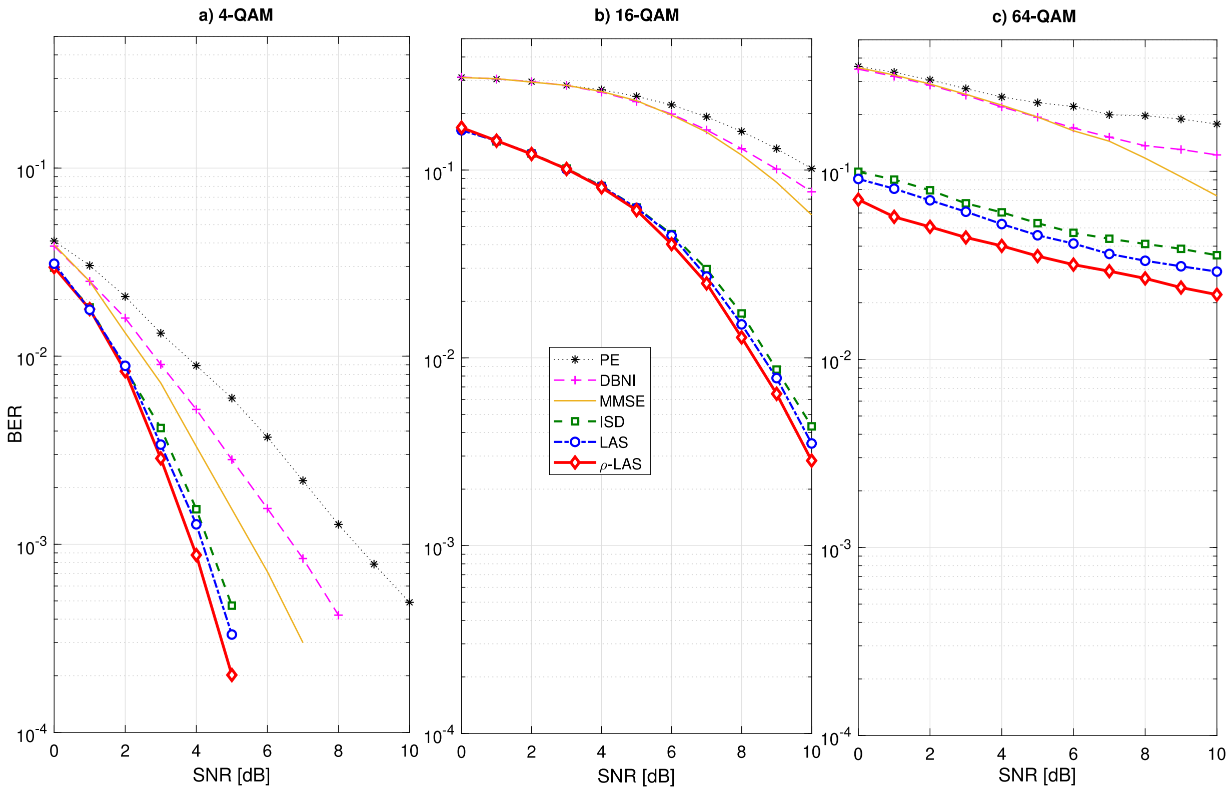


Fig. 6. BER performance under three modulation orders: (a) 4-QAM; (b) 16-QAM; (c) 64-QAM. It was considered 64 × 16 antennas, $k = 4$ and MF detector as the initial solution $(\mathbf{x}^{(0)})$ of LAS and ρ-LAS.

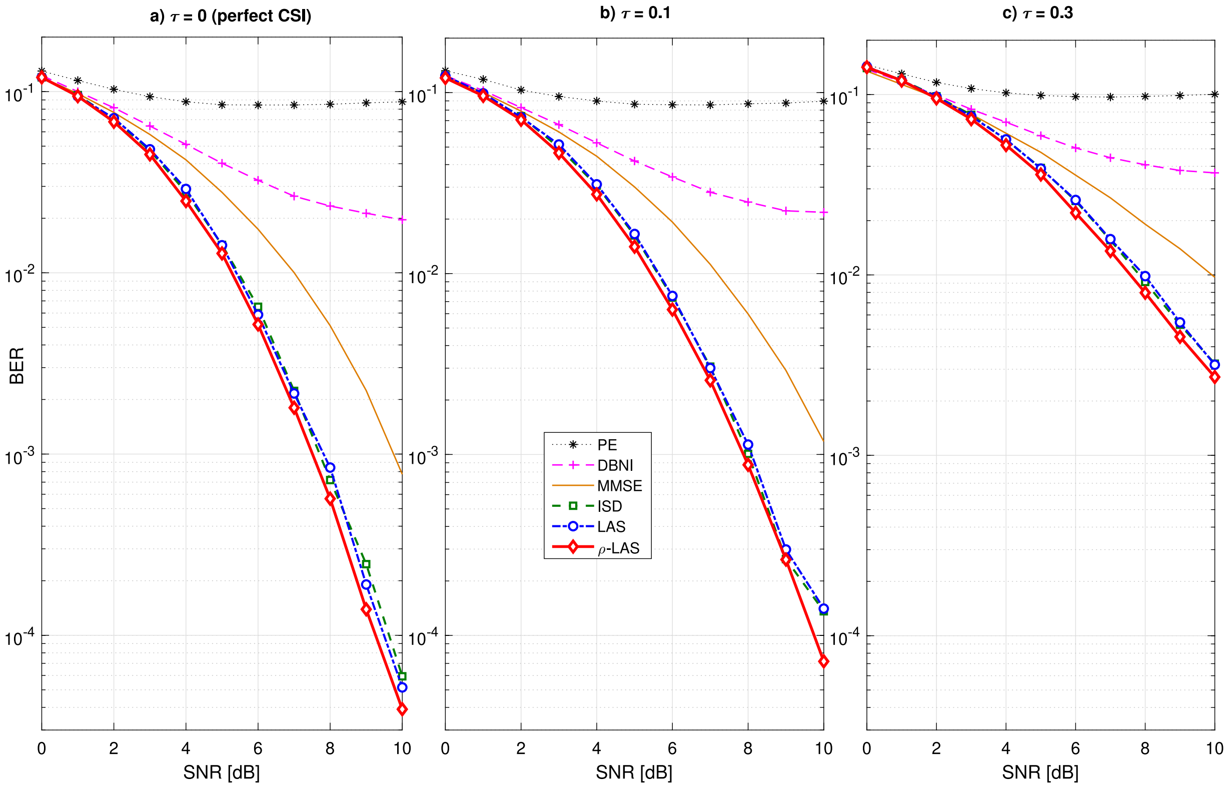


Fig. 7. BER performance for three channel error estimate scenarios (τ): (a) $\tau = 0$ (perfect CSI); (b) $\tau = 0.1$ (medium channel estimation error); (c) $\tau = 0.3$ (high channel error condition). 64 × 32 antennas, 4-QAM, $k = 4$ and MF detector as the initial solution for the LAS and ρ-LAS.

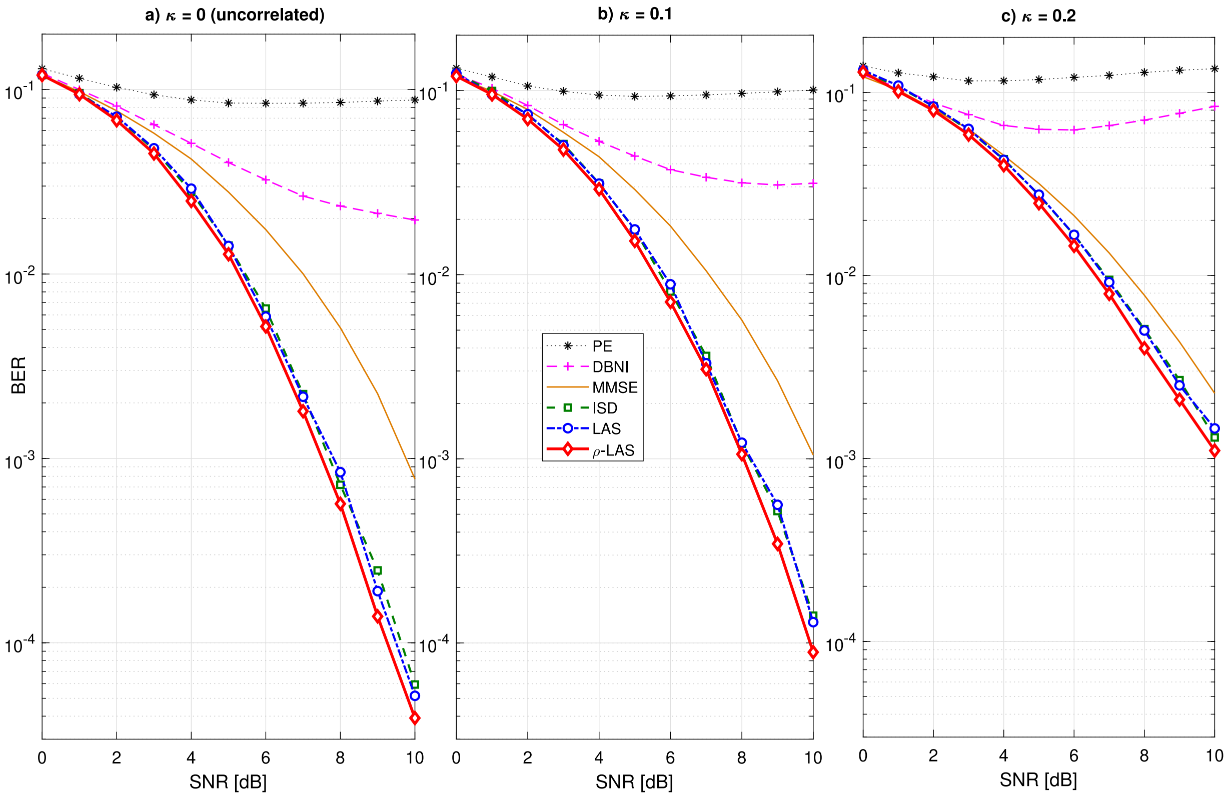


Fig. 8. Performance analysis considering three correlation indexes: (a) $\kappa = 0$ (uncorrelated); (b) $\kappa = 0.1$; (c) $\kappa = 0.2$. It was considered 64 × 32 antennas, 4-QAM, $k = 4$ and MF detector as the initial solution of LAS and ρ-LAS.

approximation of the channel matrix, which is strongly affected by correlated channel gains effect. With the increase of the correlation index, the performance of the iterative LAS and ISD detectors tend to approximate the MMSE detector performance. However, since MMSE inverts completely the channel matrix to proceed with the detection, its complexity tends to be larger than the iterative ones. Therefore, in scenarios with medium-high spatial antenna correlation, iterative LAS and ISD M-MIMO detectors are a good choice in terms of BER performance-complexity tradeoff. Among the analyzed iterative detectors, the proposed ρ-LAS detector presents the best performance.

4.4. Point-to-point downlink performance analysis

In the downlink of an M-MIMO system, generically there is low spatial correlation between users, since it is usual assuming the terminals are sufficiently far from each other so that the channel coefficients can be assumed independent. Therefore, an identity correlation matrix at the receiver side ($\mathbf{C}_{\text{Rx}} = \mathbf{I}$) can be assumed. Hence, it is possible to apply the proposed low-complexity detection technique as long as the receiver has multiple antennas. In addition, precoding techniques may be used at the transmitter side to further alleviate the receiver burden processing. As the focus of this work is to improve the LAS detection performance, the processing at the transmitter side has not been analyzed.

In Fig. 9, the performance of the detectors is analyzed in a scenario where there is only spatial antenna correlation at the transmitter side, considering $\kappa = 0$, 0.1 and 0.2 indices. Schemes with 64 transmitter and 256 receiver antennas were examined, as well as perfect CSI ($\tau = 0$) and 4-QAM modulation have been assumed. The proposed detector is still superior to the other techniques analyzed in the downlink, presenting the best performance even in the scenarios with spatial correlation. As one can infer, the proposed ρ-LAS detector is more robust to the effect of spatial correlation, since it is able to attain a BER of 10^{-3}, representing $\approx$ 3.5 dB more in SNR regarding the case with no correlation ($\kappa = 0$), and $\approx$ 4.5 dB when $\kappa = 0.2$.

4.5. Complexity analysis

In this subsection we have carried out a computational complexity analysis considering the number of floating-point operations (*flops*), as well as a performance-complexity tradeoff evaluation, given by the figure of merit ν of Eq. (24) for the linear detectors and iterative LAS and ISD detectors in the context of M-MIMO detection. A calculation of the number of flops is performed for each detection process.

Without loss of generality, we analyzed the $\mathcal{L} = 1$ scenario, i.e., $n = n_T = n_R$. We also considered vectors and matrices with lengths $n \times 1$ and $n \times n$, respectively. Thus we can define that the multiplication between two real-valued vectors (inner product), the multiplication between one real-valued vector and one real-valued matrix and the multiplication of two real-valued matrices requires $2n$, $2n^2$ and $2n^3$ flops [32], respectively. If the variables are complex, then a complex addition spends 2 flops and a complex multiplication spends 6 flops. We also assume that a matrix inversion is made by Gauss elimination, which spends $\frac{2}{3}n^3$ flops, and we disregard some computational operations such as allocation, memory access and permutation. Using Eq. (7) and Eq. (8) for the linear detectors, Algorithm 1 for the LAS detector and ρ determination in Eq. (19), the overall complexity for each M-MIMO detection technique according to the number of flops is summarized in Table 3, where $\mathfrak{C}_{\text{MF}}$ is the complexity (flops) of a linear detector (MF) and $\mathfrak{C}_{\rho_{\text{fit}}}$ is the number of flops due to the ρ surface fitting procedure.

Table 3
Detector complexities in terms of flops, $\mathfrak{C}$.

Detector	Complexity (Flops)
MMSE	$\frac{112}{3}n^3 + 8n^2$
PE	$16n^3 + 8n^2(K_{\text{PE}} + 2) + 12n$
DBNI	$16n^3 + 64n^2 + 32n\left(E_{\text{DBNI}} + \frac{1}{8}\right)$
ISD	$16n^3 + 16n^2 + 4n(k_{\text{ISD}} + 2)$
LAS	$\mathfrak{C}_{\text{MF}} + 16n^3 + 4n^2 + 4(n_{\text{F}} + 1)$
ρ-LAS	$\mathfrak{C}_{\text{MF}} + 16n^3 + 4n^2 + 4(n_{\text{F}} + 1) + \mathfrak{C}_{\rho_{\text{fit}}}$
MF	$\mathfrak{C}_{\text{MF}} = 8n^2$

$\mathfrak{C}_{\rho_{\text{fit}}}$ is the number of flops needed to calculate the expression of the polynomial fit given by Eq. (19), which is performed as input parameter for the ρ-LAS algorithm. This number is constant and has a lower weight than third-order terms in the complexity of ρ-LAS.

In addition, we calculated the number of flops of PE, DBNI and ISD detectors according to the operations presented in [17,18] and [19], respectively. K_{PE} is the order of the polynomial expansion of PE detector, E_{DBNI} is the number of neighboring elements of the diagonal matrix to be considered in DBNI detection and k_{ISD} is the number of iterations per antenna in ISD.

The matched filter (MF) procedure represents the smallest complexity, with the number of flops in the second order of n. The MMSE detector presents terms in the third order since it involves matrix inversion. PE and DBNI have the same order of complexity of the MMSE detector, but the complexity is reduced because it does not contain matrix inversion. Instead, the maximum order of the polynomial expansion K_{PE} of the PE detector and the number of elements E_{DBNI} of the bandwidth diagonal matrix of the DBNI detector affect their complexity. Because of that, the complexity of PE and DBNI detectors have terms depending of maximum polynomial order ($K_{\text{PE}} = 2$) and bandwidth ($E_{\text{DBNI}} = 2$), respectively. The ISD detector, which also has third-order terms, presents reduced complexity regarding the MMSE M-MIMO detector, since it requires a low-number of iterations (k_{ISD}).

Finally, the LAS detectors have the same order of complexity as the MMSE, only including third and second order terms dependent on the number of iterations, initial detection, and the complexity from the ρ fitting procedure. Moreover, the complexity difference between LAS and ρ-LAS is due to the ρ fitting procedure with a fixed number of flops called $\mathfrak{C}_{\rho_{\text{fit}}}$. However, the fitting procedure is counted a single time, when the physical scenario characteristics changes. Therefore, its complexity can be neglected; hence, the LAS and ρ-LAS M-MIMO detectors complexities are practically the same. Even with the same order of complexity of the MMSE detector, the LAS detector has a superior performance, and it is improved with the increasing number of antennas, which is the opposite behavior for the MMSE detector in Fig. 4(b). Therefore, a superior performance-complexity tradeoff is attained for both LAS-based M-MIMO detectors.

To elaborate further, one can define a simple figure of merit ν to quantify the complexity-performance tradeoff of the M-MIMO detectors:

$$\nu = \frac{1}{\mathfrak{C} \times \mathfrak{B}} \tag{24}$$

where $\mathfrak{C}$ is the number of flops spent by the M-MIMO detector to attain a target BER performance, $\mathfrak{B}$. A decrease in BER performance provides an increase of ν. Also, the higher the complexity, the smaller the value of ν.

To analyze the performance-complexity tradeoff of the studied low-complexity M-MIMO detectors, we start examining the behavior of the number of flops and ν for an improving number

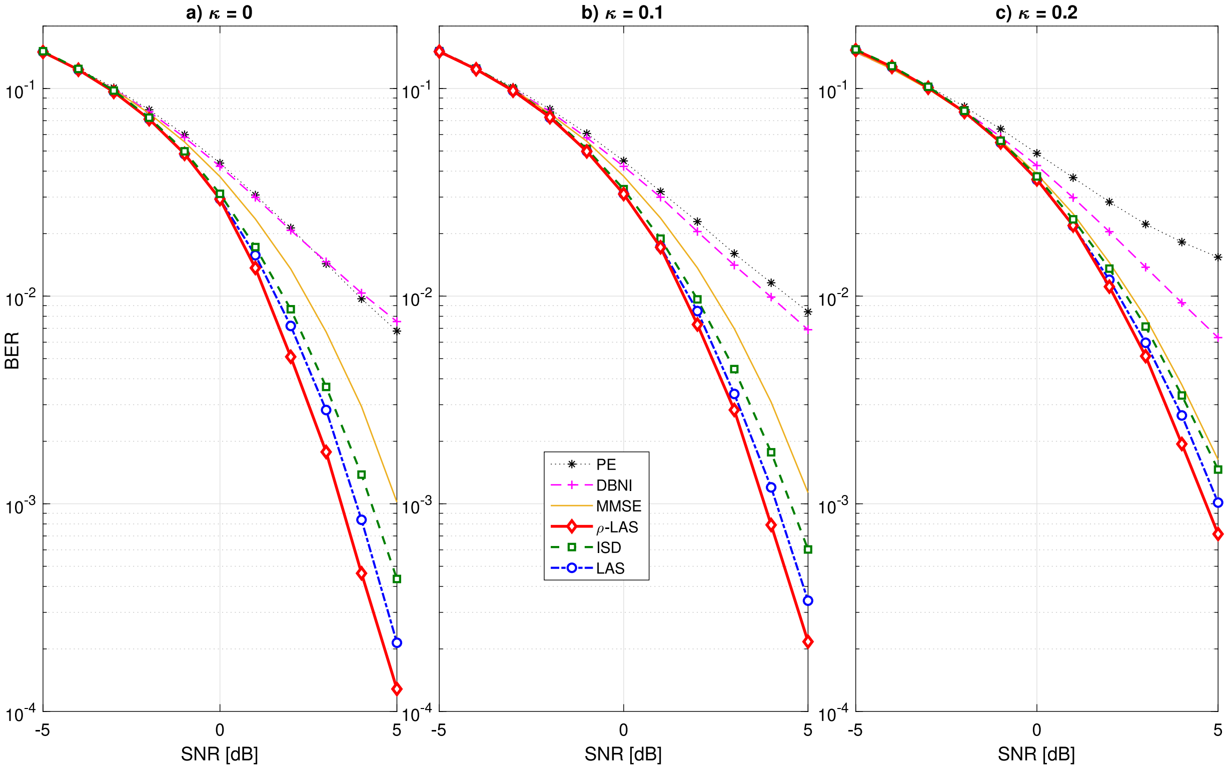
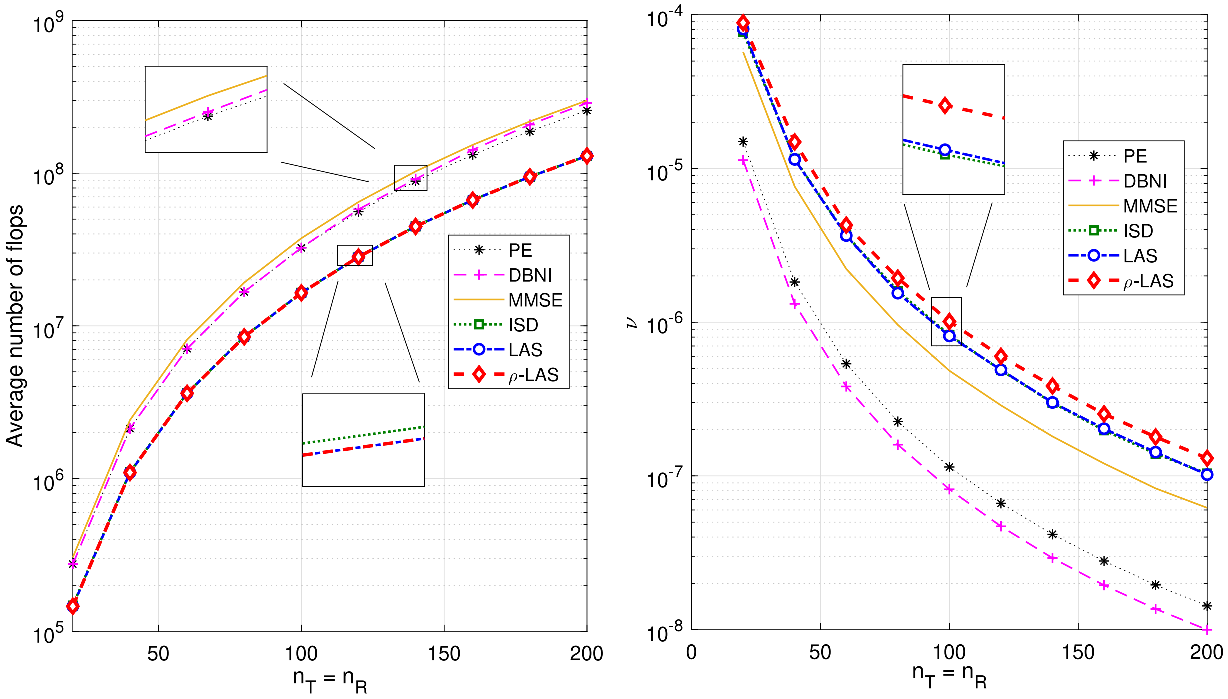


Fig. 9. BER performance of detectors in a point-to-point downlink scenario with 64 transmitter antennas and 256 receiver antennas. We considered $\tau = 0$ (perfect CSI), 4-QAM, and considering three correlation indexes at the transmitter: (a) $\kappa = 0$ (uncorrelated); (b) $\kappa = 0.1$; (c) $\kappa = 0.2$.



a) Average number of flops ($\mathfrak{C}$);

b) Performance-complexity tradeoff (ν).

Fig. 10. Complexity analysis changing the number of antennas $n = n_T = n_R$, considering $\gamma = 10$ dB, 4-QAM, $k = 4$ and $\mathbf{x}^{(0)}$ given by MF output.

of antennas n. Fig. 10 shows the correspondent curves related to the PE, DBNI, MMSE, ISD, LAS and ρ-LAS detectors.

The complexity in terms of the number of flops *vs.* $n \in [20;\ 200]$ antennas is presented in Fig. 10(a). It is observed that the most complex detector is the MMSE, since it considers the exact channel matrix inversion. Among the analyzed linear detectors, the DBNI and PE M-MIMO detectors are less complex because they consider approximations for the inverse matrix. Furthermore, the ISD, LAS and ρ-LAS detectors are less complex because they are iterative and do not depend on matrix inversion. It is observed that the ρ-LAS and LAS complexity curves are superimposed and less complex than ISD detector, since ρ-LAS presents almost the same complexity as the LAS detector because the calculation of ρ_{opt} factor adds a marginal complexity to the algorithm.

To corroborate the superior performance-complexity tradeoff for the proposed ρ-LAS approach, the figure of merit ν in Fig. 10(b) is larger for the ρ-LAS detector, which presents almost the same complexity as the conventional LAS detector and superior BER performance. ISD detector also has resulted in a suitable ν-tradeoff, but because its higher complexity regarding the LAS detectors, its tradeoff is marginally inferior to the conventional LAS detector. Finally, the linear PE, DBNI and MMSE detectors combine high-complexity with low-performance for M-MIMO scenarios with unitary loading factor ($n_T = n_R$). Therefore, they resulted in the worst performance-complexity tradeoffs with respect to the iterative ISD, LAS and ρ-LAS detectors.

5. Conclusions

The proposed ρ-LAS detector suitable for large-scale MIMO systems outperforms the conventional LAS detector from the literature. It was possible to generate a polynomial surface that adjusts well the optimal points of ρ which guarantees improved BER performance regarding the conventional LAS MIMO detector. As the convergence of ρ-LAS M-MIMO detector occurs with the same number of iterations as the LAS, then the marginal complexity increment occurs only due to the calculation of the ρ factor.

The BER performance of the ρ-LAS MIMO detector resulted superior regarding the conventional LAS and linear low-complexity MIMO detector with the increasing in the number of antennas and SNR. Moreover, under spatial antenna correlation and imperfect CSI estimates, even with performance degradation caused by channel array configuration, the ρ-LAS MIMO detector was able to perform better than LAS and much better than the linear PE, DBNI and even MMSE detectors, while continuing to converge with the same number of iterations w.r.t. the uncorrelated antenna and perfect CSI estimates scenarios. Finally, in MIMO system scenarios with high number of antennas, medium SNRs regime and high loading factor $\mathcal{L} \approx 1$, the proposed ρ-LAS detector demonstrated the best complexity-performance tradeoff among the analyzed linear and iterative MIMO detectors, being a promising detector option for M-MIMO systems.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Giovanni Maciel Ferreira Silva: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Writing - draft, Writing - review & editing. **Jose Carlos Marinello Filho:** Formal analysis, Investigation, Methodology, Resources, Software, Supervision, Validation, Writing - draft, Writing - review & editing. **Taufik Abrão:** Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Supervision, Validation, Writing - draft, Writing - review & editing.

Acknowledgments

This work was supported in part by the National Council for Scientific and Technological Development (CNPq) of Brazil under Grants 304066/2015-0, and in part by CAPES - Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Brazil (scholarship), and by the Londrina State University - Paraná State Government (UEL).

References

[1] M.H. Alsharif, R. Nordin, N.F. Abdullah, A.H. Kelechi, How to make key 5G wireless technologies environmental friendly: A review, Trans. Emerg. Telecommun. Technol. 29 (1) (2017) e3254, e3254 ett.3251 [Online] Available: https://onlinelibrary.wiley.com/doi/abs/101002/ett.3254.

[2] G. Xiaohu, W. Haichao, Z. Ran, L. Qiang, N. Qiang, 5G multimedia massive MIMO communications systems, Wirel. Commun. Mob. Comput. 16 (11) (2016) 1377–1388, [Online] Available: https://onlinelibrary.wiley.com/doi/abs/101002/wcm.2704.

[3] L. Lu, G.Y. Li, A.L. Swindlehurst, A. Ashikhmin, R. Zhang, An overview of massive MIMO: Benefits and challenges, IEEE J. Sel. Top. Sign. Proces. 8 (5) (2014) 742–758.

[4] K. Zheng, L. Zhao, J. Mei, B. Shao, W. Xiang, L. Hanzo, Survey of large-scale MIMO systems, IEEE Commun. Surv. Tutor. 17 (3) (2015) 1738–1760, thirdquarter.

[5] D.C. Araujo, T. Maksymyuk, A.L.F. de Almeida, T. Maciel, J.C.M. Mota, M. Jo, Massive MIMO: survey and future research topics, IET Commun. 10 (15) (2016) 1938–1946.

[6] T.L. Marzetta, Noncooperative cellular wireless with unlimited numbers of base station antennas, IEEE Trans. Wireless Commun. 9 (11) (2010) 3590–3600.

[7] K. Vardhan, S. Mohammed, A. Chockalingam, A low-complexity detector for large MIMO systems and multicarrier CDMA systems, IEEE J. Sel. Areas Commun. 26 (3) (2008) 473–485.

[8] A. Chockalingam, B.S. Rajan, Large MIMO Systems, Cambridge University Press, 2014.

[9] Y. Sun, A family of likelihood ascent search multiuser detectors: an upper bound of bit error rate and a lower bound of asymptotic multiuser efficiency, IEEE Trans. Commun. 57 (6) (2009) 1743–1752.

[10] A.K. Sah, A.K. Chaturvedi, Sequential and global likelihood ascent search based detection in large MIMO systems, IEEE Trans. Commun. PP (99) (2017) 1.

[11] A.A. Pereira, R. Sampaio-Neto, Low-complexity soft-output MIMO uplink detection for large systems iterative detection and decoding, Trans. Emerg. Telecommun. Technol. 29 (2) (2018) e3251, e3251 ett.3251 [Online] Available: https://onlinelibrary.wiley.com/doi/abs/101002/ett.3251.

[12] I. Chihaoui, M.L. Ammari, Suited architecture for massive MIMO detector based on antenna selection and LAS algorithm, in: International Symposium on Signal, Image, Video and Communications (ISIVC), 2016, pp. 126–130.

[13] P. Li, R.D. Murch, Multiple output selection-LAS algorithm in large MIMO systems, IEEE Commun. Lett. 14 (5) (2010) 399–401.

[14] Z. Linbo, Z. Zhuo, L. Tong, S. Shanshan, Optimal LAS rceiver in massive MIMO system, in: IEEE International Conference on Electronic Information and Communication Technology, 2016, pp. 522–525.

[15] G.M.F. Silva, J.C.M. Filho, T. Abrao, Sequential likelihood ascent search detector for massive MIMO systems, AEU - Int. J. Electron. Commun. 96 (2018) 30–39, [Online] Available: http://www.sciencedirect.com/science/article/pii/S1434841118300621.

[16] N. Shariati, E. Bjornson, M. Bengtsson, M. Debbah, Low-complexity polynomial channel estimation in large-scale MIMO with arbitrary statistics, IEEE J. Sel. Top. Sign. Proces. 8 (5) (2014) 815–830.

[17] M. Wu, B. Yin, G. Wang, C. Dick, J.R. Cavallaro, C. Studer, Large-scale MIMO detection for 3GPP LTE: Algorithms and FPGA implementations, IEEE J. Sel. Top. Sign. Proces. 8 (5) (2014) 916–929.

[18] C. Tang, C. Liu, L. Yuan, Z. Xing, High precision low complexity matrix inversion based on newton iteration for data detection in the massive MIMO, IEEE Commun. Lett. 20 (3) (2016) 490–493.

[19] M. Mandloi, V. Bhatia, Low-complexity near-optimal iterative sequential detection for uplink massive MIMO systems, IEEE Commun. Lett. 21 (3) (2017) 568–571.

[20] X. Tan, Y. Ueng, Z. Zhang, X. You, C. Zhang, A low-complexity massive MIMO detection based on approximate expectation propagation, IEEE Trans. Veh. Technol. 68 (8) (2019) 7260–7272.

[21] A. Datta, M. Mandloi, V. Bhatia, Reliability feedback-aided low-complexity detection in uplink massive MIMO systems, Int. J. Commun. Syst. 32 (15) (2019) e4085, e4085 dac.4085 [Online] Available: https://onlinelibrary.wiley.com/doi/abs/101002/dac.4085.

[22] I. Chihaoui, M.L. Ammari, P. Fortier, Improved LAS detector for MIMO systems with imperfect channel state information, IET Commun. 13 (9) (2019) 1297–1303.

[23] M.A. Albreem, M. Juntti, S. Shahabuddin, Massive MIMO detection techniques: A survey, IEEE Commun. Surv. Tutor. (2019) 1.

[24] R.T. Kobayashi, F. Ciriaco, T. Abrão, Efficient near-optimum detectors for large MIMO systems under correlated channels, Wirel. Pers. Commun. 83 (2) (2015) 1287–1311, [Online] Available: https://doi.org/10.1007/s11277-015-2450-y.

[25] J.L. Negrão, T. Abrão, Efficient detection in uniform linear and planar arrays MIMO systems under spatial correlated channels, Int. J. Commun. Syst. 31 (11) (2018) e3697, e3697 dac.3697 [Online] Available: https://onlinelibrary.wiley.com/doi/abs/101002/dac.3697.

[26] J. Chen, A low complexity data detection algorithm for uplink multiuser massive MIMO systems, IEEE J. Sel. Areas Commun. 35 (8) (2017) 1701–1714.

[27] S. Verdu, Multiuser Detection, Cambridge University Press, 1998.

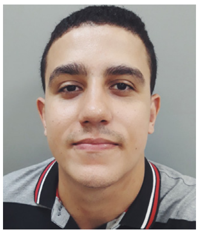
[28] Y. Sun, A family of linear complexity likelihood ascent search multiuser detectors for CDMA communications, in: Proc IEEE 6th Intl Symp on Spread Spectrum Tech and App, 2000.

[29] Y. Sun, Eliminating-highest-error and fastest-metric-descent criteria and iterative algorithms for bit-synchronous CDMA multiuser detection, in: Proc IEEE ICC '98, 1998, pp. 1576–1580.

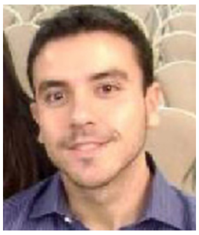
[30] M.N. Boroujerdi, S. Haghighatshoar, G. Caire, Low-complexity statistically robust precoder/detector computation for massive MIMO systems, IEEE Trans. Wireless Commun. 17 (10) (2018) 6516–6530.

[31] X. Wu, N.C. Beaulieu, D. Liu, On favorable propagation in massive MIMO systems and different antenna configurations, IEEE Access 5 (2017) 5578–5593.

[32] G.H. Golub, C.F.V. Loan, Matrix Computations, third ed., The Johns Hopkins University Press, 1996.



Giovanni Maciel Ferreira Silva received his B.S. degree in Electrical Engineering from Londrina State University, Londrina, Brazil in December 2018. He is currently working towards his M.S. in Electrical Engineering at Londrina State University, Londrina, Brazil. His research interests lie in communications and signal processing, especially in physical-layer of wireless communication networks, including optimization for wireless communication systems, massive MIMO detection and channel estimation techniques for 5G MIMO systems.



Jose Carlos Marinello Filho. Received his B.S. and M.S. degree in Electrical Engineering (the first with Summa Cum Laude) from Londrina State University, PR, Brazil, in December 2012 and Sept 2014, respectively, and his Ph.D. degree in electrical engineering from the Polytechnic School of the University of São Paulo, São Paulo, Brazil, in 2018. His research interests include signal processing and wireless communications, especially massive multiple-antenna precoding/decoding techniques, acquisition of channel-state information, multicarrier modulation, cross-layer optimization of MIMO/OFDM systems, interference management and 5G.

Taufik Abrao (IEEE-SM'12, SM-SBrT) received the B.S., M.Sc., and Ph.D. degrees in electrical engineering from the Polytechnic School of the University of São Paulo, São Paulo, Brazil, in 1992, 1996, and 2001, respectively. Since March 1997, he has been with the Communications Group, Department of Electrical Engineering, Londrina State University, Paraná, Brazil, where he is currently an Associate Professor in Telecommunications and the Head of the Telecomm. & Signal Processing Lab. He is a Productivity Researcher from the CNPq Brazilian Agency (Pq-1D) . From July–October 2018 he was with the Connectivity section, Aalborg University as a Guest Researcher. In 2012, he was an Academic Visitor with the Southampton Wireless Research Group, University of Southampton, Southampton, U.K. From 2007 to 2008, he was a Post-doctoral Researcher with the Department of Signal Theory and Communications, Polytechnic University of Catalonia (TSC/UPC), Barcelona, Spain. He has participated in several projects funded by government agencies and industrial companies. He is involved in editorial board activities of several journals in the telecommunications area and has served as TPC member in several symposiums and conferences. He has also served as an Associate Editor for the IEEE ACCESS since 2016, the IET Journal of Engineering since 2014, the IET Signal Processing since Dec-2018, and JCIS-SBrT journal since 2018. Previously, he served as AE of the IEEE Communication Surveys & Tutorials (2013–2017). Moreover, Prof. Abrao has been served as Executive Editor of the ETT-Wiley journal since 2016. He is a member of SBrT and a senior member of IEEE. His current research interests include communications and signal processing, especially massive MIMO, ultra-reliable low latency communications, detection and estimation, multicarrier systems, cooperative communication and relaying, resource allocation, as well as heuristic and convex optimization aspects of 5G wireless systems. He has supervised 27 M.Sc. and 4 Ph.D. students, as well as 3 postdocs, co-authored twelve book chapters on mobile radio communications and +280 research papers published in international journals and conferences.

APPENDIX B – Full paper accepted for publication in the journal ”Computer Networks”

Title: Throughput and Latency in the Distributed Q-Learning Random Access mMTC Networks.
Authors: Giovanni Maciel Ferreira Silva, Taufik Abrão.
Journal: Computer Networks, ISSN 1389-1286.
Acceptance Date: Jan 2022.

Throughput and Latency in the Distributed Q-Learning Random Access mMTC Networks

Giovanni Maciel Ferreira Silva, Taufik Abrão

***Abstract*—In mMTC mode, where thousands of devices try to access network resources sporadically, the problem of random access (RA) and collisions between devices that select the same resources arise. A promising approach to solve the RA problem is the use of learning mechanisms, specially Q-learning algorithm, where the devices learn about the best time-slot periods to transmit through rewards sent by the central node. In this work, we propose a distributed packet-based learning method of varying the reward given by the central node that favors devices having a larger number of remaining packets to transmit. The numerical results indicated that the proposed distributed packet-based Q-learning method attains a better throughput-latency trade-off than the independent and collaborative techniques in practical scenarios, while the number of payload bits of the packet-based technique is reduced regarding the collaborative Q-learning RA technique for achieving the same normalized throughput.**

***Keywords* – mMTC, random access, throughput, latency, Q-learning.**

I. Introduction

Since the beginning of studies on the fifth generation of wireless communications (5G), it is known that the paradigm is not simply to increase transmission rates [1]. With the demand for services such as the Internet of Things (IoT), smart cities, smart homes, among others, we seek to solve the problem of ensuring connectivity for thousands of devices at an access point. Because of this, 5G has been divided into three main modes: enhanced mobile broadband (eMBB), to guarantee high rates for mobile users; ultra reliable and low latency communications (URLLC), to guarantee a low latency and high reliability connection to certain services such as remote surgery, and massive machine-type communications (mMTC), to connect thousands of machine-type devices to the network.

In mMTC mode, thousands or even tens of thousands machine-type devices access network resources sporadically, i.e., a sensor network that sends data every minute. In this way, the transmission rate is not one of the most important figures of merit, but throughput, which evaluates the number of successful transmissions in a given time interval, and the probability of collision, which is the number of collisions that occurs at a certain time interval [2], [3].

The problem that arises in this mode is to solve the random access (RA), since the devices randomly select the time-slot to transmit and collisions of two or more devices may occur, making communication impossible. Traditionally, the slotted ALOHA (SA) is used to solve the RA problem. In this technique, colliding devices retransmit after a fixed time-slot window, which reduces the probability of a new collision. However, it has been shown [4] that SA has a high probability of collision in highly congested scenarios, where the number of devices is greater than or equal to the number of time-slots in a frame. Many solutions to solve the RA problem in mMTC are present in the literature, based on several different techniques, such as the grant-free ALOHA [5], the unequal access latency (UAL) [6], the sparse signal recovery [7], and the distributed queue-based framework [8].

An alternative and promising way to solve the RA problem is the use of machine learning (ML) techniques [9], where the devices themselves learn to choose the best time-slots to transmit, avoiding collisions and increasing throughput. Q-Learning, being model-free, is a viable solution in these scenarios since machine-type devices must carry out learning in a simplified and not very complex way. Machine learning-based techniques will play an important role in technologies foreseen for the sixth generation of wireless communication (6G) [10]–[12].

Recently, techniques based on Q-Learning are present in many works in the field of wireless communications. Some applications include network slicing [13], spectrum access [14], non-orthogonal multiple access (NOMA) [15], and geographic routing for unmanned robot networks (URNs) [16].

One of the simplest ways to use Q-Learning to mitigate the collision problem in RA protocols is the independent technique, where central node sends a binary reward to the devices, informing if the transmission was successful or if there was a collision between two or more devices. Such technique does not perform well in scenarios where the number of devices is equal to or greater than the number of available time-slots. In contrast to it, the collaborative Q-Learning method is suggested, where the reward sent to the colliding devices is the level of congestion in the time-slot. In this case, the devices learn to choose the least disputed time-slots, increasing the throughput of the system [4].

An alternative to the independent Q-Learning technique is the collaborative approach, which considers the level of congestion in the reward sent from the central node to the devices. The throughput is higher for this technique than the independent technique, however the reward needs to be sent in more than one bit and the central node needs to know the number of devices that collided in a given time-slot.

Both techniques mentioned are not fair, as the devices that randomly select the least disputed time-slots will transmit their packages more quickly, as the learning method will provide them with unique time-slots. On the other hand,

G. Maciel and T. Abrão are with Department of Electrical Engineering, State University of Londrina, Parana, Brazil. E-mail: giomaciel.fs@gmail.com, taufik@uel.br

devices that collide frequently will take longer to transmit all of their packets.

The *Contribution* of this work is to propose a Q-Learning RA technique that does not detract from the devices that select the most congested time-slots at the beginning of the transmission, as occurs in the collaborative technique proposed in [4]. The proposed distributed packet-based Q-Learning technique benefits devices that still have many packets to transmit, sending them a greater reward. The technique in [4] sends larger rewards to devices that have uniquely selected time-slots, causing some devices to end transmission very quickly, while others take longer. In general, the distributed packet-based method reduces the latency variance. Also, we have proposed an improvement in the collaborative Q-Learning technique aiming at establishing a reasonable level of congestion with a finite number of bits, and as result, reducing the header when sending the reward.

The remainder of the work is composed of the system model in Section II, the proposed distributed packet-based Q-Learning reward method in Section III; numerical results are analyzed in Section IV; the main conclusions and final remarks are presented in Section V.

II. System model

Let's consider an mMTC network, where there are N machine-type devices transmitting packets with p bits of payload to a central node. A frame is made up of K time-slots, and the N devices select one of the slots to transmit. Each device has L packets to transmit, with only one packet being transmitted per time-slot. The loading factor is given by the ratio between the number of active devices and the number of time-slots within a frame, $\mathcal{L} = \frac{N}{K}$. The indexes for each device and each time-slot are sorted in sets $\mathcal{N} = \{1, \ldots, N\}$ and $\mathcal{K} = \{1, \ldots, K\}$, respectively. In addition, we define the set ψ_k indicating which devices have chosen the k-th time-slot. For example, if devices 2 and 5 selected the 3rd time-slot, then $\psi_3 = \{2, 5\}$.

The transmission of a packet is considered successful when only one device selects the k-th time-slot, *i.e* it results in cardinality one, $|\psi_k| = 1$. Otherwise, if two or more devices choose the same time-slot, $|\psi_k| > 1$, a collision occurs, and the transmission is considered a failure.

For simplicity of analysis, the effects of physical channel losses such as multipath fading and AWGN (high SNR regime) are not considered. As the focus of the work is on developing the reward sending mechanisms in Q-Learning-based RA protocols aiming to improve the learning process of the devices, we assume, as in [4], [9], that the signal from all devices arrive with the same power at the central node. The random variables are defined by the channel/slot selection in each device. In addition, central node does not apply any collision resolution method to decide which device wins. When two or more devices select the same channel to transmit, central node considers it a collision and requests that the devices retransmit the packet.

At the end of the frame, central node sends a *reward* signaling to the devices composed by b bits indicating whether the transmission was successful or not. The devices use the reward information to learn over the transmissions which are the best time-slots subset to transmit. The end of transmission occurs when all devices transmit all their packets.

To illustrate the transmission process across $K = 6$ time-slots, Fig. 1 depicts the RA-based network considering $N = 8$ devices. In this simple example, only devices 3, 4 and 7 select time-slots exclusive to them. Therefore, they are the only ones to receive a positive reward from the central node. On the other hand, the other devices collide with each other, and therefore the reward sent by the central node is negative.

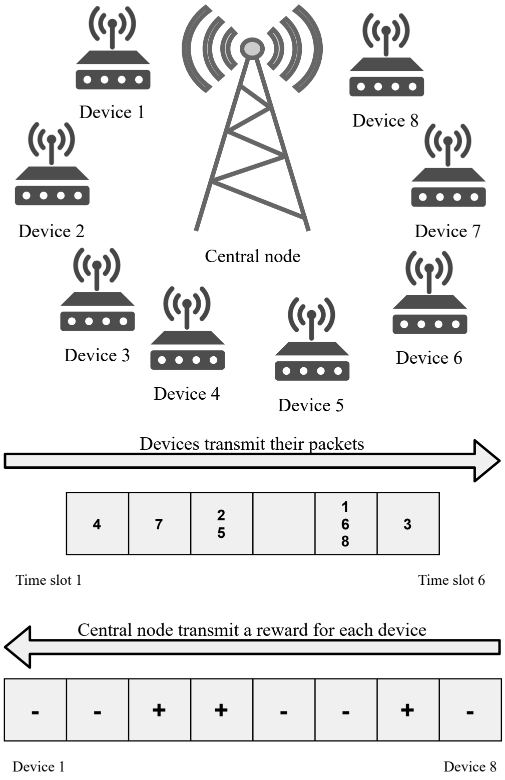


Figure 1. Reward-based RA network with $N = 8$ devices and $K = 6$ time-slots.

III. Random access with Q-Learning

Q-Learning is a type of machine learning (ML), which is model-free and it can be implemented in a distributed way and with low complexity. The advantage of using Q-Learning to solve the RA problem is that it is easily implemented on thousands of mMTC devices due to its low complexity, while the devices decide in a distributed and decentralized way the best time-slots to transmit based on previous transmissions. The learning method of each device can be modeled as a Markov decision process (MDP), where the change to a future state depends on the factors: the current state, the transition probability function and the reward value [17].

The n-th device has a Q-value, namely $Q^t_{n,k}$, that indicates the preference to transmit in the k-th time-slot and step t. All Q-values make up a Q-table of N rows and K columns. Initially, the entire Q-table is set to zero: $Q_{n,k} = 0, \forall n \in 1, \ldots, N, \ \forall k \in 1, \ldots, K$. Hence, in order to transmit a packet, the device selects the time-slot with the highest Q-value from its Q-table. If there is more than one time-slot whose Q-value is the maximum, then the choice is random among these Q-values.

At the end of the frame, the central node sends a reward to each device indicating whether the transmission was successful or not in a given time-slot. Thus the Q-value in the next step of the n-th device and k-th time-slot is updated to

$$Q^{t+1}_{n,k} = Q^{t}_{n,k} + \alpha(R_{n,k} - Q^{t}_{n,k}) \tag{1}$$

where $R_{n,k}$ is the reward transmitted by central node, and α is the learning rate. The learning rate is a weight value in the range $\alpha \in [0; 1]$. In this work, α is assumed fixed and equal for all devices in the system.

The Q-table update, and the packet transmission are performed subsequently until each device transmits all of its packets. The reward-based RA algorithm is considered to have converged when all devices have transmitted all their packets. In the convergence process, it is defined that the total number of successes is S, the total number of failures is F and the total number of time-slots spent is T.

A. Independent Q-Learning

The independent Q-Learning technique requires that central node send only one bit (b = 1) for each device. The reward sent to the n-th device that chose the k-th time-slot is simply defined as: [4]:

$$R^{\text{IND}}_{n,k} = \begin{cases} +1, & \text{if transmission succeeds,} \\ -1, & \text{otherwise.} \end{cases} \tag{2}$$

Therefore, if only the n-th device has chosen the k-th slot, the transmission is successful and the reward is +1. If two or more devices choose the k-th slot, a collision occurs and the reward is -1 for all of them. Reward $R^{\text{IND}}_{n,k}$ is used to update the Q-table for all devices and time-slots through Eq. (1).

B. Collaborative Q-Learning

Assuming that central node is aware of the number of devices that tried to access the k-th slot, it is possible to define a congestion level C_k in slot k, given by

$$C_k = \frac{|\psi_k|}{N}, \tag{3}$$

As a result, the reward sent by central node to the n-th device that chose the k-th time-slot is given by [4]

$$R^{\text{COL}}_{n,k} = \begin{cases} +1, & \text{if transmission succeeds,} \\ -\mathcal{M}_b\{C_k\}, & \text{otherwise,} \end{cases} \tag{4}$$

where $\mathcal{M}_b\{C_k\}$ is a quantized value of C_k based on the number of bits b available for the header, e.g., if $b = 2$ bits and assuming that the level of congestion varies from 0 to 1, then the reward values can be unambiguously represented by four quantized levels, $\mathcal{M}_b\{C_k\} \in \{0.25, 0.5, 0.75, 1\}$.

As C_k in this case is a real number, then the central node should transmit a quantized version of such real number, decreasing the spectral efficiency of transmission, and the devices will have to use a certain number of quantization bits b to represent this real value. Therefore, there is a trade-off between bandwidth overhead and accuracy when quantized version (limited number of bits) is transmitted by the central node and the true value of the reward. Hence, the fixed number of bit of quantization must be selected carefully.

The advantage of the collaborative method over the independent one is that the devices learn to choose the time-slots with lower levels of congestion to transmit their packages. The disadvantage is that the central node needs to know the number of interfering devices. In addition, the reward becomes a real number and no longer a bit, as in the independent method.

C. Distributed Packet-based RA for Crowded MTC Scenarios

With the increase in the number of devices and the increase in the probability of collision in crowded mMTC mode, it becomes more difficult for central node to identify the number of interfering devices. Therefore, the advantage of the collaborative Q-learning technique in regions with high density of devices depends on an ideal non-feasible scenario. In addition, independent and collaborative Q-learning techniques are not completely fair, as a time-slot becomes unique for one device over the entire learning period, while the other devices continue to collide and expect to randomly find a suitable time-slot to finish transmitting all packets.

Therefore, this work proposes a distributed packet-based Q-Learning random access technique where the Q-table updating takes into account the number of remaining packet that each device still has to transmit in that frame. The higher this number, the greater is the respective reward, increasing the frequency of transmission attempt in that time-slot; hence, it is expected that on average all devices finish transmitting their packets at the same time.

Let's define the factor ϵ_n for each device as:

$$\epsilon_n = 1 - \frac{\ell_n}{L}, \tag{5}$$

where ℓ_n is the number of remaining packets to be transmitted by the n-th device; hence, when the device has already transmitted a large number of packets, ϵ_n tends to 1.

In the proposed Q-learning-based RA method, the reward sent by central node to the n-th device at the k-th time-slot is defined in a same way as in the independent Q-learning method:

$$R^{\text{PAC}}_{n,k} = R^{\text{IND}}_{n,k} = \begin{cases} +1, & \text{if transmission succeeds,} \\ -1, & \text{otherwise.} \end{cases} \tag{6}$$

However, since the proposed method is totally distributed, the reward processing is utterly done by the devices. Hence, under this method, the Q-Table updating takes into account the number of packets that the device still has to transmit results:

$$Q^{t+1}_{n,k} = \begin{cases} Q^{t}_{n,k} + \alpha(R^{\text{PAC}}_{n,k} - Q^{t}_{n,k}), & \text{if Tx succeeds,} \\ Q^{t}_{n,k} + \alpha(\epsilon_n R^{\text{PAC}}_{n,k} - Q^{t}_{n,k}), & \text{otherwise.} \end{cases} \tag{7}$$

$$= \begin{cases} Q^{t}_{n,k} + \alpha(1 - Q^{t}_{n,k}), & \text{if Tx succeeds,} \\ Q^{t}_{n,k} - \alpha(\epsilon_n + Q^{t}_{n,k}), & \text{otherwise.} \end{cases} \tag{8}$$

where eq. (8) can be obtained by substituting (6) into (7).

Notice that in the distributed packet-based RA method, the central node does not need to know the number of devices that has collided in a given time-slot. Therefore, the reward to be transmitted is binary ($b = 1$), requiring the same infra-structure than the independent technique. In addition, among the devices that collided, devices that need to transmit more packets are privileged with a more positive reward compared to those devices with less packets remaining to be transmitted, making the technique more appropriate to attain improved throughput-complexity tradeoff when compared to collaborative and independent-like methods. The pseudo code for the proposed distributed packet-based technique is present in Algorithm 1.

Algorithm 1 Distributed Packet-Based RA Method

```
Initialize $Q_{n,k} = 0,\ \forall n \in \mathcal{N},\ \forall k \in \mathcal{K}$
Initialize $\ell_n = L,\ \forall n \in \mathcal{N}; \quad T = 0,\ S = 0$
while $\sum_{n=1}^{N} \ell_n > 0$ do
    Initialize $c_n = 0,\ \forall n \in \mathcal{N}$
    for $n = 1 : N$ do
        if $\ell_n > 0$ then
            $\mathcal{C}_n = \{k \in \mathcal{K} \mid Q_{n,k} = \max_k\{Q_{n,k}\}\}$
            Select randomly: $c_n \in \mathcal{C}_n$
    for $k = 1 : K$ do
        $T \leftarrow T + 1$
        $\psi_k = \{n \in \mathcal{N} \mid c_n = k\}$
        if $|\psi_k| = 1$ then
            $S \leftarrow S + 1$
            $R^{\text{PAC}}_{n,k} = +1,\ \forall n \in \psi_k$
            $Q_{n,k} \leftarrow Q_{n,k} + \alpha(1 - Q_{n,k}),\ \forall n \in \psi_k$
            $\ell_n \leftarrow \ell_n - 1,\ \forall n \in \psi_k$
        else if $|\psi_k| > 1$ then
            $\epsilon_n = 1 - \frac{\ell_n}{L},\ \forall n \in \psi_k$
            $R^{\text{PAC}}_{n,k} = -1,\ \forall n \in \psi_k$
            $Q_{n,k} \leftarrow Q_{n,k} - \alpha(\epsilon_n + Q_{n,k}),\ \forall n \in \psi_k$
```

IV. Numerical results

In this section, proposed Q-Learning RA technique is numerically validated via computer simulations, and compared with the independent and collaborative learning methods. In order to guarantee an average behavior of the number of transmissions carried out successfully, 10^4 realizations for each experiment were considered. The main simulation parameter values used are shown in Table I.

Table I
Numerical parameters.

Parameter	Value
Monte-Carlo realizations	N_{reps} = 10,000
Time-slots per frame	K = 400
Network Loading factor	$\mathcal{L} = \frac{N}{K} \in [0.25;\ 3.00]$
Packets per device	$L \in [50; 500]$
Learning rate	$\alpha \in [0.05; 0.5]$
Header bits (collab.)	$b \in [1; 2; 4; 8; 16]$ bits
Payload bits	$p \in [1; 2; 4; 8; \ldots; 256]$ bits

An important figure of merit is the normalized throughput, defined as the ratio between the number of successful packet transmissions, S, and the corresponding number of time-slots required, T. However, as not all bits in the transmission are data from the devices, so the ratio between the payload bits and reward bits should be taken into account. Hence, the normalized throughput is defined as

$$\mathcal{T} = \left(\frac{p}{b+p}\right)\frac{S}{T} = \left(\frac{p}{b+p}\right)\frac{NL}{T}. \tag{9}$$

The calculation of normalized throughput is performed after the convergence of the algorithm, when all devices transmit all their packets, and it indicates how efficiently the time-frames have being used in each RA method.

A. Number of bits of quantized collaborative reward

To find the smallest number of bits that results in a suitable accuracy in representing the actual number of the congestion level in the collaborative technique without reducing the throughput, Fig. 2 depicts the average throughput calculated as a function of the loading factor $\mathcal{L}$. The result shows that, within the analyzed scenario, a suitable tradeoff choice for the number of quantization bits that maximize the mean throughput in the collaborative Q-Learning technique is $b = 4$ bits. By deploying four bits, it is possible to attain a good level of quantization for the real number of the reward, but without reducing the throughput due to the increase in header bits; hereafter, this value was adopted in all simulations of the collaborative technique.

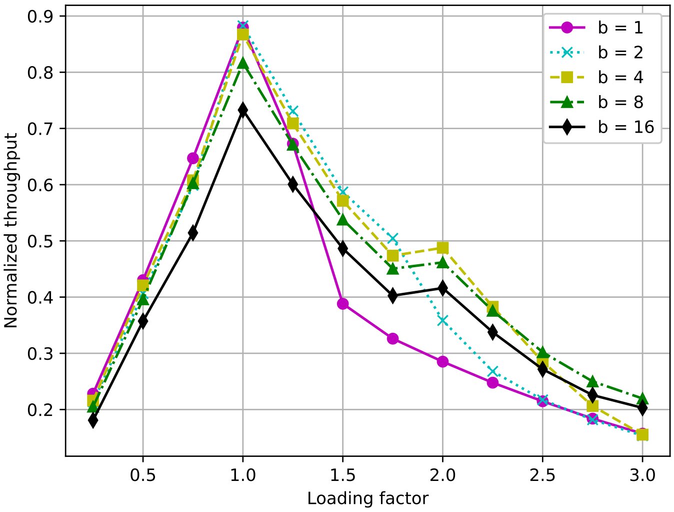


Figure 2. Throughput for collaborative method varying the loading factor, considering p = 64, L = 100, and α = 0.1.

B. Normalized Throughput

In Fig. 3, the throughput is analyzed as a function of the loading factor. It is observed that the maximum throughput is obtained when $\mathcal{L} = 1$ for all techniques, because in this scenario, the frame is being used with greater efficiency, where in average there is a time-slot for each device. Hence, as expected, the throughput is lower in underloaded and overloaded scenarios, *i.e.,* $\mathcal{L} \neq 1$, where there are fewer or many devices than time-slots and is not the ideal scenario, as more and more devices could be allocated on the network to increase the spectral efficiency. In particular, we are interest in crowded MTC scenarios.

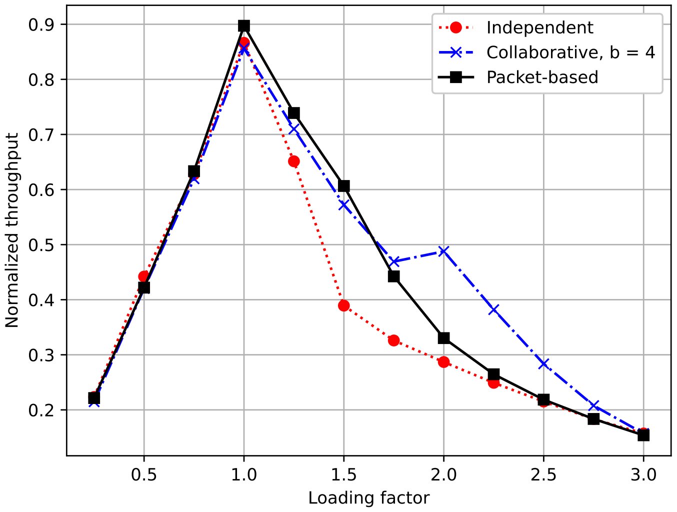


Figure 3. Normalized throughput in function of loading factor for independent, collaborative, and packet-based Q-Learning, considering p = 64, L = 100, K = 400, and α = 0.1.

In the $\mathcal{L} > 1$ scenario, the RA techniques start to have a worse throughput because, when there are more devices than time-slots, the probability of collision increases substantially and unavoidably, which consequently reduces the success probability and throughput. The difference between independent and collaborative Q-Learning RA techniques stands out in this important scenario of practical interest. The collaborative technique has greater throughput because the central node indicates to the devices which time-slots have the highest congestion level, through the information sent as a reward. The devices learn to transmit in the least congested time-slots, thus reducing the probability of collision and, consequently, increasing throughput.

The performance of the packet-based technique is superior to other techniques up to $\mathcal{L} = 1.6$. From that point on, the collaborative technique becomes superior in the interval $1.6 \leq \mathcal{L} \leq 3.0$, and then the techniques converge to the same throughput value. It is expected that the collaborative technique presents a higher throughput in relation to the others in the medium-high congestion scenarios $(1.75 \leq \mathcal{L} \leq 3.0)$ because the reward sent by the central node provides more details about the level of congestion of each time-slot. However, the packet-based technique still proves to be superior to the independent one in this scenario, in addition to being less complex than the collaborative one in relation to the central node, since the reward sent is binary.

C. Asymptotic Throughput

In Fig. 4, the throughput for the three reward techniques was analyzed with the change in the number of packets that each device has to transmit, from $L = 50$ to $L = 500$ packets. For this result, we consider $\mathcal{L} = 1$, payload $p = 64$ bits, and learning rate $\alpha = 0.1$.

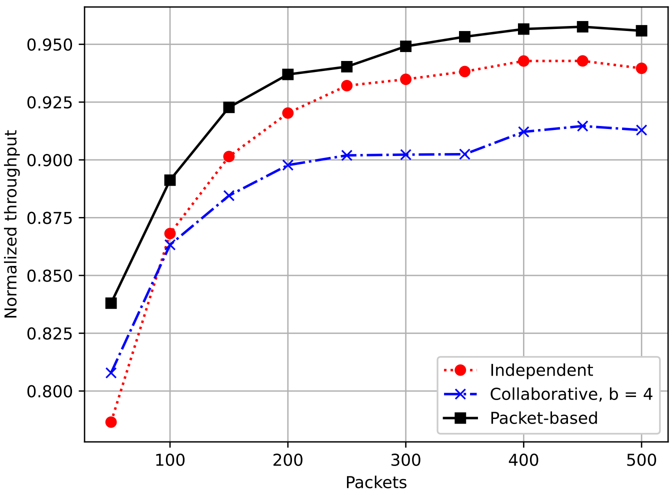


Figure 4. Throughput as a function of the number of packets, considering loading factor $\mathcal{L}$ = 1, K = 400 time-slots, p = 64 bits, and α = 0.1.

It is possible to conclude that the throughput increases with the increase in the number of packets. This is because the number of successes increases, without having a significant increase in the number of time-slots needed to transmit all packets. However, the curves begin to converge to a constant value. This indicates that, even if the number of packets increases, the time to transmit them in the same proportion is increased, which makes the throughput constant. The proposed distributed packet-based RA method reveals a superior *asymptotic normalized throughput*:

$$\mathcal{T}_{\infty}(\mathcal{L}) = \lim_{L,T\to\infty} \left(\frac{p}{b+p}\right) \frac{NL}{T},$$

resulting for the specific network loading factor: $\mathcal{T}_{\infty}^{\text{PAC}}(1) \approx 0.965;\quad \mathcal{T}_{\infty}^{\text{IND}}(1) \approx 0.940;\quad \mathcal{T}_{\infty}^{\text{COL}}(1) \approx 0.915$

D. Payload bits

Fig. 5 shows the result of the throughput as a function of the number of payload bits b. The numerical result indicates that when the number of payload bits is small, with a value close to the number of header bits, the throughput is low. As the number of payload bits increases, the throughput increases until it converges to a ceiling value. This convergence occurs in our configuration setup close to $p = 64$, and for this reason, this payload value was considered in the rest of the simulations in this work.

As the collaborative technique has a larger number of header bits (b = 4), then it depends on a larger number of payload bits to present the same throughput as the packet-based technique. For example, to achieve a normalized throughput of $\mathcal{T}_p = 0.5$, the collaborative technique needs 16 bits of payload, while the packet-based one needs 4 bits in the analyzed scenario. The reduction in the number of payload bits can be an advantage in simplifying the process in which a bunch of devices randomly access the channel and transmitting their packets.

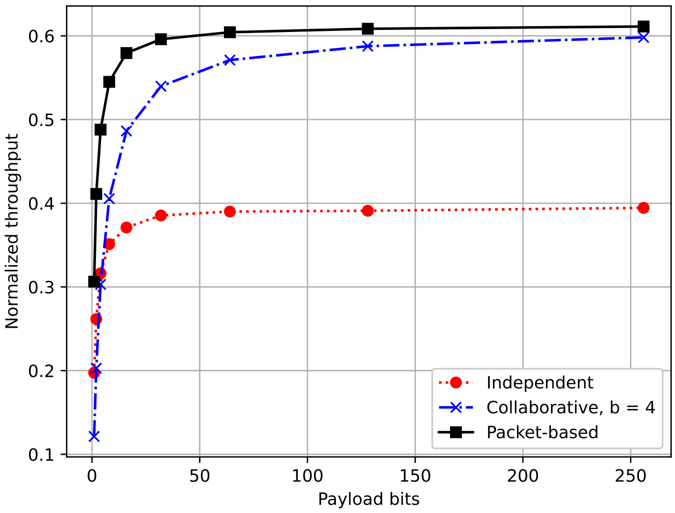


Figure 5. Normalized throughput as as function of payload bits, considering $\mathcal{L}$ = 1.5, α = 0.1, and L = 100.

E. Latency

Latency in this work is defined as the total amount of time-slots T that all devices need to transmit a fixed number of packets. In Fig. 6, there is an analysis of the total number of time-slots required for the complete transmission of $L = 100$ packets/device according to an increasing in the loading factor of the system.

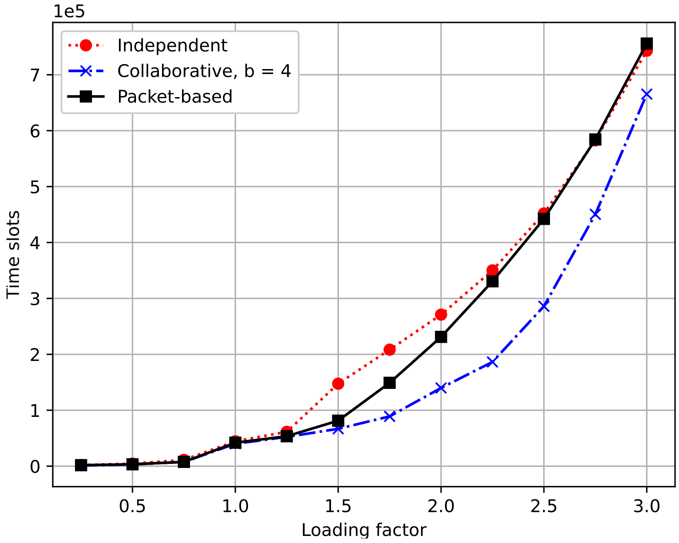


Figure 6. Total number of time-slots as a function of loading factor considering L = 100, and α = 0.1.

As expected, latency in terms of total time-slots required increases when the loading factor increases, as the system becomes more congested and the number of collisions increases, requiring more retransmissions. The result can be analyzed in three different scenarios. For a low-medium loading factor, $\mathcal{L} \leq 1.0$, all techniques have the same latency. For slightly-crowded and crowded scenarios, *i.e.*, $1.2 \leq \mathcal{L} \leq 2.5$,, the independent technique has the highest latency in relation to the others. Finally, for high-loading over-crowded scenarios, the Q-Learning packet-based RA technique approaches the latency of the independent technique, while the collaborative technique holds the lowest latency.

Hence, from Fig. 3, 4, 5, and 6, one can infer that the proposed distributed packet-based RA method attains the best throughput-latency trade-off for a wide range loading factors, $0.75 \leq \mathcal{L} \leq 2.5$, mainly in typically (over)crowded scenarios.

F. Learning Rate

Finally, the adopted value for the learning rate is justified. For that, latency was evaluated according to the learning rate, as shown in Fig. 7. The adoption of an increasing value for the learning rate negatively affects the performance of reward-based RA techniques in crowded scenarios, as the latency to achieve convergence increases. When the learning rate is high, the weight given to the reward of the central node is greater. Hence, in more congested scenarios, more negative than positive rewards can be expected. Therefore, when devices give greater weight to negative rewards, the latency of the technique increases. This behavior is observed when $\mathcal{L}$ = 1.5, as the latency increases significantly with the increase in the learning rate. When $\mathcal{L}$ = 1, this behavior is smoothed, since the increase in latency only occurs when $\alpha = 0.5$ for the collaborative and packet-based techniques.

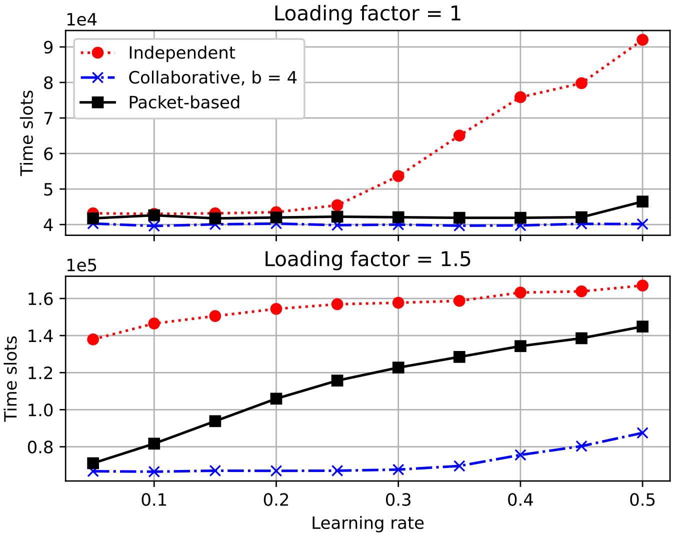


Figure 7. Total number of time-slots as a function of learning rate considering L = 100.

V. Conclusions

Q-Learning-based random access methods for mMTC networks have been investigated in terms of throughput and latency. The numerical results and analyses presented in this work have demonstrated that the proposed distributed packet-based RA method attains higher throughput and lower latency than the independent Q-Learning RA technique, even with the central node transmitting only a bit of reward for both techniques. In addition, the distributed packet-based method presented the best throughput-latency trade-off regarding the independent and collaborative techniques for different loading factor scenarios ($0.25 \leq \mathcal{L} \leq 1.5$). These results are due to the reduction in the number of bits transmitted by the central node to one, while distributing the processing among the devices. Finally, in highly congested scenarios, *e.g.*, $\mathcal{L} \approx 3$, the throughput of the proposed technique is the same as that of the collaborative technique.

References

[1] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view," *IEEE Access*, vol. 6, pp. 55 765–55 779, 2018.

[2] C. Bockelmann, N. Pratas, H. Nikopour, K. Au, T. Svensson, C. Stefanovic, P. Popovski, and A. Dekorsy, "Massive machine-type communications in 5G: physical and MAC-layer solutions," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 59–65, Sep. 2016.

[3] C. Bockelmann, N. K. Pratas, G. Wunder, S. Saur, M. Navarro, D. Gregoratti, G. Vivier, E. De Carvalho, Y. Ji, C. Stefanovic, P. Popovski, Q. Wang, M. Schellmann, E. Kosmatos, P. Demestichas, M. Raceala-Motoc, P. Jung, S. Stanczak, and A. Dekorsy, "Towards massive connectivity support for scalable mMTC communications in 5G networks," *IEEE Access*, vol. 6, pp. 28 969–28 992, 2018.

[4] S. K. Sharma and X. Wang, "Collaborative distributed Q-learning for RACH congestion minimization in cellular IoT networks," *IEEE Communications Letters*, vol. 23, no. 4, pp. 600–603, April 2019.

[5] R. Qi, X. Chi, L. Zhao, and W. Yang, "Martingales-based ALOHA-type grant-free access algorithms for multi-channel networks with mMTC/URLLC terminals co-existence," *IEEE Access*, vol. 8, pp. 37 608–37 620, 2020.

[6] J. Jiao, L. Xu, S. Wu, Y. Wang, R. Lu, and Q. Zhang, "Unequal access latency random access protocol for massive machine-type communications," *IEEE Transactions on Wireless Communications*, vol. 19, no. 9, pp. 5924–5937, 2020.

[7] Y. Cui, S. Li, and W. Zhang, "Jointly sparse signal recovery and support recovery via deep learning with applications in MIMO-based grant-free random access," *IEEE Journal on Selected Areas in Communications*, pp. 1–1, 2020.

[8] A. H. Bui, C. T. Nguyen, T. C. Thang, and A. T. Pham, "A comprehensive distributed queue-based random access framework for mMTC in LTE/LTE-A networks with mixed-type traffic," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 12, pp. 12 107–12 120, 2019.

[9] S. K. Sharma and X. Wang, "Towards massive machine type communications in ultra-dense cellular IoT networks: Current issues and machine learning-assisted solutions," *IEEE Communications Surveys Tutorials*, pp. 1–1, 2019.

[10] M. Z. Chowdhury, M. Shahjalal, S. Ahmed, and Y. M. Jang, "6G wireless communication systems: Applications, requirements, technologies, challenges, and research directions," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 957–975, 2020.

[11] Z. Zhang, Y. Xiao, Z. Ma, M. Xiao, Z. Ding, X. Lei, G. K. Karagiannidis, and P. Fan, "6G wireless networks: Vision, requirements, architecture, and key technologies," *IEEE Vehicular Technology Magazine*, vol. 14, no. 3, pp. 28–41, 2019.

[12] L. U. Khan, I. Yaqoob, M. Imran, Z. Han, and C. S. Hong, "6G wireless systems: A vision, architectural elements, and future directions," *IEEE Access*, vol. 8, pp. 147 029–147 044, 2020.

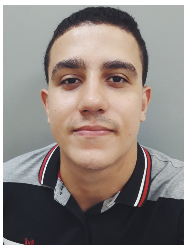
[13] T. Li, X. Zhu, and X. Liu, "An end-to-end network slicing algorithm based on deep q-learning for 5G network," *IEEE Access*, vol. 8, pp. 122 229–122 240, 2020.

[14] Z. Su, M. Dai, Q. Xu, R. Li, and S. Fu, "Q-learning-based spectrum access for content delivery in mobile networks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 1, pp. 35–47, 2020.

[15] M. V. da Silva, R. D. Souza, H. Alves, and T. Abrao, "A NOMA-based q-learning random access method for machine type communications," *IEEE Wireless Communications Letters*, vol. 9, no. 10, pp. 1720–1724, 2020.

[16] W. Jin, R. Gu, and Y. Ji, "Reward function learning for q-learning-based geographic routing protocol," *IEEE Communications Letters*, vol. 23, no. 7, pp. 1236–1239, 2019.

[17] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge: The MIT Press, 2018.



Giovanni Maciel Ferreira Silva received his B.S. degree in Electrical Engineering from Londrina State University, Londrina, Brazil in December 2018. He is currently working towards his M.S. in Electrical Engineering at Londrina State University, Londrina, Brazil. His research interests lie in communications and signal processing, especially in physical-layer of wireless communication networks, including optimization for wireless communication systems, random access methods for mMTC networks, and massive MIMO detection.

Taufik Abrão (SM'12, SM-SBrT) received the B.S., M.Sc., and Ph.D. degrees in electrical engineering from the Polytechnic School of the University of São Paulo, Brazil, in 1992, 1996, and 2001, respectively. Since March 1997, he has been with the Communications Group, Department of Electrical Engineering, Londrina State University, Paraná, Brazil, where he is currently an Associate Professor in Telecommunications and the Head of the Telecomm. & Signal Processing Lab. He is a Productivity Researcher from the CNPq Brazilian Agency (Pq-1D). From July-October 2018 he was with the Connectivity section, Aalborg University as a Guest Researcher. In 2012, he was an Academic Visitor with the Southampton Wireless Research Group, University of Southampton, Southampton, U.K. From 2007 to 2008, he was a Post-doctoral Researcher with the Department of Signal Theory and Communications, Polytechnic University of Catalonia (TSC/UPC), Barcelona, Spain. He has participated in several projects funded by government agencies and industrial companies. He has served as an Associate Editor for the IEEE Systems Journal (2020), the IEEE Access (2016-2018), IEEE Communication Surveys & Tutorials (2013-2017), the AEUe-Elsevier (2020), the IET Signal Processing (2019), and JCIS-SBrT (2018-2020). Prof. Abrao has been served as Executive Editor of the ETT-Wiley journal since 2016. His current research interests include communications and signal processing, especially massive MIMO, XL-MIMO, URLLC, mMTC, optimization methods, detection and estimation, multicarrier systems, cooperative communication and relaying, resource allocation. He has supervised +30 M.Sc., +10 Ph.D. students, as well as 3 postdocs, co-authored twelve book chapters on mobile radio communications and +280 research papers published in international journals and conferences.

APPENDIX C – Full paper submitted to the journal ”Transactions on Emerging Telecommunications Technologies”

Title: Multi-Power Level Q-Learning Algorithm for Random Access in NOMA mMTC Systems.
Authors: Giovanni Maciel Ferreira Silva, Taufik Abrão.
Journal: Transactions on Emerging Telecommunications Technologies, ISSN 2161-3915.
Submission Date: Nov 2021.

Multi-Power Level Q-Learning Algorithm for Random Access in NOMA mMTC Systems

Giovanni Maciel Ferreira Silva, Taufik Abrão

***Abstract*—The massive machine-type communications (mMTC) service will be part of new services planned to integrate the beyond fifth generation of wireless communication (B5G). In mMTC, thousands of devices sporadically access available resource blocks on the network. In this scenario, the massive random access (RA) problem arises when two or more devices collide when selecting the same resource block. There are several techniques to deal with this problem. One of them deploys Q-learning (QL), in which devices store in their Q-table the rewards sent by the central node that indicate the quality of the transmission performed. The device learns which are the best resource blocks to select and transmit in order to avoid collisions. We propose a multi-power level QL (MPL-QL) algorithm that uses non-orthogonal multiple access (NOMA) transmit scheme to generate transmission power diversity and allow accommodate more than one device in the same time-slot as long as the signal-to-interference-plus-noise ratio (SINR) exceeds a threshold value. The numerical results reveal that the best performance-complexity trade-off is obtained by using a higher power levels, typically eight levels. The proposed MPL-QL can deliver better throughput and lower latency when compared to other recent QL-based algorithms found in the literature.**

***Keywords* – NOMA, mMTC, Q-Learning; random access; power allocation.**

I. Introduction

Machine-type wireless communication will be more widely used in applications such as internet of things (IoT), smart house, virtual reality, etc [1], [2]. The goal of the B5G wireless communications involves achieve ubiquitous communication in networks with ultra-dense devices allocation [3]–[5]. A data consumption of nearly 5 zettabytes per month is estimated across 17 billion devices [6]. In addition, due to the outbreak of the COVID-19 pandemic, there has been an remarkable increase in remote activities in work, health and education areas, which will be much more frequent in the post-pandemic environment [7].

Devices connected to the wireless network use different types of service. In the fifth generation of wireless communications (5G) systems, a clear division into three main use modes was defined [8]: enhanced mobile broadband (eMBB) for devices that require high data rates as an augmented reality user; ultra-reliable low-latency communications (URLLC) for applications that require 99.999% communication reliability such as remote surgery, while holding end-to-end latency below 1ms; and massive machine-type communications (mMTC), composed of thousands of devices with low processing power that access network data sporadically.

The study of these services remains relevant for 6G application scenarios. In the new generation of communications, new services will be generated by merging the benefits of existing ones. In [6], massive ultra-reliable low-latency communication (mULC) is presented as a combination of the low latency of URLLC with the high number of mMTC devices, a densification application process. This new use mode can be associated with the intelligent transport, where high reliability is required for traffic safety and various traffic sensors and monitors send data about the vehicle's condition. Besides, the ubiquitous mobile broadband (uMBB) is also suggested as a use of high eMBB rates in mMTC devices to enable applications such as ubiquitous networking and digital twin.

As the mMTC scenarios studied in 5G could be associated to the 6G systems, the analysis of the main problems that affect this service is still relevant. One is the random access (RA) procedure. To reduce latency in communication with devices, it is common to use grant-free RA techniques, in which devices do not need a training step with pilot sequences before sending data packets. With the increase in the number of devices and the data rate starvation with new applications, the problem of RA is aggravated. As the device access to the network is sporadic but there is a crowded number of inactive users in the network, then it is common for two or more devices to select the same resource block to transmit data, which is characterized as a collision.

There are several techniques that mitigate the massive RA problem [9], [10]. One of the simplest and least complex is the performed by the slotted ALOHA (SA) protocol, which makes the device resend the col-

G. Maciel and T. Abrão are with Department of Electrical Engineering, State University of Londrina, Parana, Brazil. E-mail: giomaciel.fs@gmail.com, taufik@uel.br

lided packet after a fixed time window. There is also the strongest-user collision resolution (SUCRe) protocol [11], which solves the collision problem by calculating, in a distributed way in each user terminal (UT), the strongest user signal. Another possibility to mitigate the RA issue in (over)-crowded networks consists in deploying reinforcement learning (RL) techniques. In RL, devices take actions and receive rewards from the central node indicating the quality of actions taken. RL has an advantage over more traditional machine learning (ML) techniques in such RA crowded complex scenario, as it is not necessary to passively receive a dataset [12].

A more simplified yet effective RL model for this scenario is the Q-Learning (QL) which is a model-free RL [13]. The device learns which are the best resource blocks it should transmit based on its Q-table storage of the rewards sent by the central node. The low-complexity of QL makes it suitable to operate in crowded RA scenarios with many devices randomly transmitting short packets [14], [15]. In [16], an independent QL technique with a binary reward and a collaborative technique in which the device receives information on the congestion level of each time-slot are proposed. In [17], a NOMA-based QL algorithm is proposed in which the device can transmit at up to three different power levels to generate power diversity at the receiver while increase throughput. Recently, [18], proposed a packet-based QL scheme that can benefit devices that still have many packets to transmit, sending them a bigger reward.

The *contribution* of this work is twofold: first, we propose a multi-power levels QL algorithm (MPL-QL), evaluating the impact of increasing power levels on the throughput and latency, differing from what was done in [17] where only three power levels are proposed, which does not exploit the full benefit of the power domain of NOMA. Second, we compare performance metrics, such as throughput, of the proposed MPL-QL protocol with four well-established RA protocols, the SA, the independent QL [16], the collaborative QL [16], and the packet-based QL [18].

The remainder of the letter is composed of the system model described in Section II; the proposed MPL-QL algorithm is presented in Section III; numerical results are analyzed in Section IV; the final remarks in Section V closes the letter.

II. System model

There are N mMTC devices sending uplink (UL) packets to a central node in a circular cell with radius r. The frequency resources used are a carrier f_c and a bandwidth B. The n-th device is d_n meters away from the central node and it transmits with power P_t. The distribution of devices within the circular cell is shown in Fig. 1.

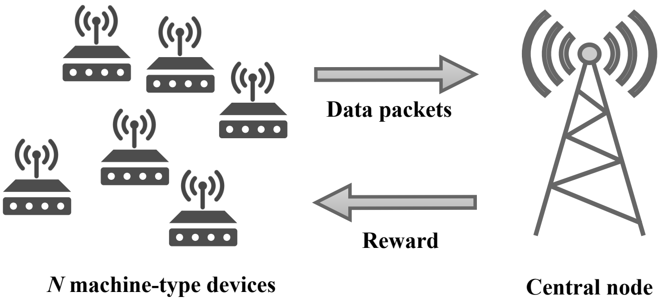


Figure 1. System model.

The transmit frame in the UL is divided into K time-slots, while a downlink (DL) time-slot at the end is deployed for central node broadcast. The devices randomly select a time-slot to transmit. The set ψ_k contains the indexes of all devices that selected k-th time-slot, $k \in \{1, \dots, K\}$. Furthermore, each device has L packets to transmit. The end of system transmission occurs when all devices successfully transmit all of their L packets. At the end, we define the total latency δ as the total number of spent frames to attain convergence, *i.e.*, all packets transmitted successfully by all devices. Assuming that the DL slot is much smaller than the UL slot, it is possible to approximate the length of a frame to K time-slots and the total number of time-slots until the end is δK. Fig. 2 shows how transmission frames are divided.

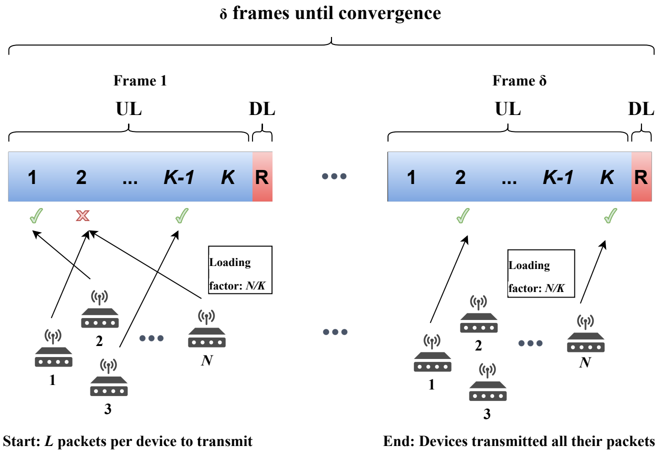

δ frames until convergence
Frame 1
UL
DL
1
2
...
K-1
K
R
✓
✗
✓
Loading factor: N/K
1
2
3
N
Frame δ
UL
DL
1
2
...
K-1
K
R
✓
✓
Loading factor: N/K
1
2
3
N
Start: L packets per device to transmit
End: Devices transmitted all their packets

Figure 2. Frame in UL and DL time-slots until the end of system transmission (all devices), namely convergence of transmission process.

The received signal in the central node at the k-th time-slot is simply defined as:

$$y_k = \sum_{\forall n \in \psi_k} x_{n,k} + w_k, \tag{1}$$

where $x_{n,k}$ is the attenuated signal transmitted by the n-th device at the k-th time-slot, and $w_k \sim \mathcal{CN}(0, N_0 B)$ is the additive white Gaussian noise (AWGN) at the receiver in the k-th time-slot with power spectral density N_0.

Let's consider that $h_{n,k}$ is an independent and identically distributed zero mean and unit variance Rayleigh fading of the n-th device at k-th time-slot. Therefore, the instantaneous signal-to-interference-plus-noise ratio (SINR) received from the n-th device at the k-th time-slot can be defined as

$$\gamma_{n,k} = \frac{P_{n,k}}{\sum_{\forall j \in \psi_k, j \neq n} P_{j,k} + w_k^2}, \tag{2}$$

where $P_{n,k} = h_{n,k}^2 \bar{P}_n$ is the instantaneous power of the n-th device at k-th time-slot. $\bar{P}_n$ is calculated based on the log-distance path loss model:

$$\bar{P}_n = P_t + \bar{P}_{d_0} - 10\eta \log_{10}\left(\frac{d_n}{d_0}\right), \quad \text{[dB]} \tag{3}$$

where η is the path loss exponent, d_0 is a reference distance, and $\bar{P}_{d_0}$ is a reference constant power given by

$$\bar{P}_{d_0} = 20 \log_{10}\left(\frac{c}{4\pi d_0 f_c}\right). \quad \text{[dB]} \tag{4}$$

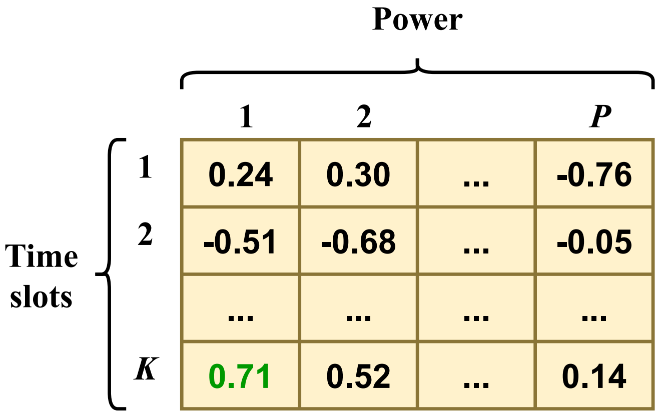
Assuming that the devices have the same quality of service (QoS) requirements, we can set a threshold SINR $\bar{\gamma}$ at the receiver to ensure the packet can be detected. The packet transmitted by the n-th device at k-th time-slot can be successfully received at the central node when $\gamma_{n,k} \geq \bar{\gamma}$.

III. Multi-Power Level Q-Learning Algorithm

This section describes the proposed multi-power level Q-Learning based grant-free RA procedure. Each device can transmit with a maximum power $P_{\max}$. The transmitted symbol is then assumed to have maximum amplitude $V_{\max}$. The symbol $\tilde{x}_n$ transmitted by the device can assume $\mathcal{P}$ equidistant amplitude levels between $-V_{\max}$ and $V_{\max}$, *e.g.*, for $\mathcal{P}$ = 4,

$$\tilde{x} \in \{-V_{\max}, -\frac{V_{\max}}{3}, \frac{V_{\max}}{3}, V_{\max}\}.$$

The selection of which time-slot and power level the device will transmit is based on the Q-table indices whose Q-value is maximum. When there are two or more values equal to the maximum, the device randomly selects between them. Fig. 3 depicts the structure of the power level and time-slot selection based on the Q-table.


Power
Time slots

	1	2		P
1	0.24	0.30	...	-0.76
2	-0.51	-0.68	...	-0.05
	...	...	...	...
K	0.71	0.52	...	0.14

Figure 3. Q-table for each device.

As the devices present a power disparity given by the differences in distances and transmission powers, then the central node can apply a *successive interference cancellation* (SIC) procedure to remove the interference from the devices that collided in the same time-slot. With this, the SINR considering NOMA becomes:

$$\gamma_{n,k}^{\text{NOMA}} = \frac{P_{n,k}}{\sum_{j=n+1}^{|\psi|} P_{j,k} + w_k^2}. \tag{5}$$

The transmission of the n-th device is successful if

$$R_{n,k,p} = \begin{cases} +1, & \text{if } \gamma_{n,k}^{\text{NOMA}} \geq \bar{\gamma} \\ -1, & \text{otherwise.} \end{cases} \tag{6}$$

With the reward received, the device updates its Q-table [19]:

$$Q_{n,k,p}^{(t+1)} = Q_{n,k,p}^{(t)} + \alpha(R_{n,k,p} - Q_{n,k,p}^{(t)}). \tag{7}$$

ℓ_n is the number of packets that the n-th device still has to transmit. The devices continue transmitting until all of their L packets are transmitted. Total latency δ is the number of frames required for the complete transmission of packets until the algorithm converges. Algorithm 1 indicates the pseudo-code step-by-step of the proposed MPL-QL operation.

IV. Numerical Results

In this section, the performance and convergence of the MPL-QL algorithm are analyzed. Performance is measured in terms of throughput and latency, and convergence is analyzed by interference per device and convergence factor. The system simulations for the QL algorithms were coded in Python language [20], with Table I presenting a summary of the parameter values adopted along this section.

Algorithm 1 MPL-QL algorithm

```
Initialize $Q_{n,k,p} \sim \mathcal{U}[-1,1]\ \forall n,k,p$;
Initialize $\ell_n = L,\ \forall n$;
Initialize $\delta = 0,\ S = 0$
while $\sum_{n=1}^{N} \ell_n > 0$ do
    for all devices that $\ell_n > 0$ do
        Search $k$ and $p$ where
            $Q_{n,k,p} = \max_{k,p}\{Q_{n,k,p}\}$
    for all time-slots $k = 1 : K$ do
        if $|\psi_k| > 0$ then
            Calculate $\gamma_{n,k}^{\text{NOMA}}$ using Eq. (5) $\forall n \in \psi_k$
            if $\gamma_{n,k}^{\text{NOMA}} \geq \bar{\gamma}$ then
                Success: $S \leftarrow S + 1,\ \ell_n \leftarrow \ell_n - 1$
                $R_{n,k,p}$ = 1
            else
                $R_{n,k,p}$ = -1
            Update: $Q^{(t+1)}_{n,k,p} = Q^{(t)}_{n,k,p} + \alpha(R_{n,k,p} - Q^{(t)}_{n,k,p})$
    Increment a frame: $\delta \leftarrow \delta + 1$
```

Table I
NUMERICAL PARAMETERS.

Parameter	Value
Monte-Carlo realizations	$\mathcal{M} = 10000$
Time-slots per frame	K = 100
Network loading factor	$\mathcal{L} = \frac{N}{K} \in [0.25;\ 10]$
Packets per device	$L \in [50;\ 100]$
Learning rate	α = 0.1
SINR threshold	$\bar{\gamma}$ = 3
Transmit power levels	$\mathcal{P} \in [2; 4; 8; 12; 16]$
Cell radius	r = 200 m
Reference distance	d_0 = 1 m
Bandwidth	B = 125 kHz
Carrier frequency	f_c = 915 MHz
Path loss exponent	η = 3
Noise PSD	$N_0 = -150$ dBm/Hz
Maximum power	$P_{\max}$ = 1 mW

A. *Throughput and Latency of MPL-QL algorithm*

Throughput τ is calculated as the ratio between the total number of successes S and the total number of time-slots required for the algorithm to converge:

$$\tau = \frac{S}{\delta K}, \qquad \left[\frac{\text{success}}{\text{time-slot}}\right] \tag{8}$$

Throughput indicates on average how many devices can successfully transmit their packets within a time-slot. In Fig. 4, the throughput of the MPL-QL technique is analyzed as a function of the loading factor for different power levels ($\mathcal{P} \in \{2, 4, 8, 12, 16\}$).

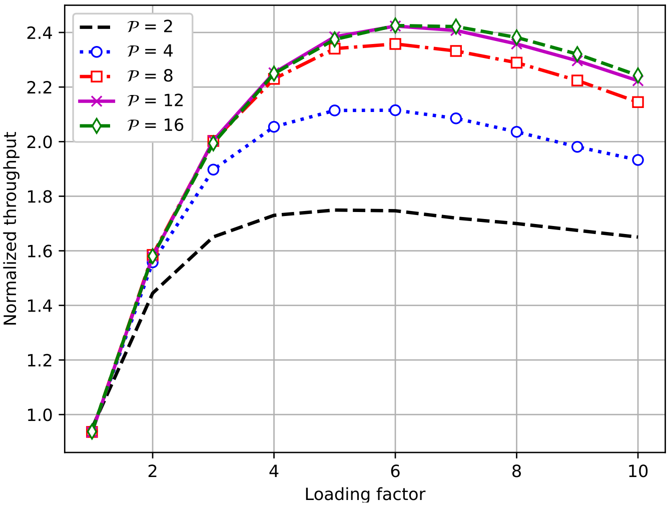


Figure 4. MPL-QL throughput for different power levels $\mathcal{P}$.

Note that the higher the power levels, the higher the throughput. With higher power levels available to the transmitter, the greater the power difference between the desired signal and the interferers at the receiver. This makes the receiver's SIC able to detect more packets successfully.

When increasing the power levels from 12 to 16, a marginal gain in throughput was observed. In this scenario, there is no significant increase in the power disparity that arrives at the receiver, so a maximum number of possible successes is reached after SIC detection. As the increase in the number of power levels causes an increase in the size of the Q-table that the device needs to store, it is possible to say that a good number for power levels is between 8 and 12, as a good performance-complexity trade-off is guaranteed to devices.

In Fig. 5, the latency δ needed to obtain the convergence of the algorithm was analyzed considering the same power levels used in Fig. 4.

Latency decreases with increasing power levels. Considering a loading factor $\mathcal{L}$ = 6, the latency for $\mathcal{P}$ = 8 is $\approx$ 30% lower compared to $\mathcal{P}$ = 2. Analogously to what was discussed in Fig. 4, with a higher value of $\mathcal{P}$, the disparity of power between the device of interest and the interfering devices increase, which makes the occurrence of collisions smaller, consequently decreasing the latency.

The differences (reduction) of latency for power levels $\mathcal{P} > 8$ become marginal, which again indicates that a suitable power level guaranteeing a good performance-complexity trade-off is close to 8. This is because, by increasing the power levels, the granularity is increased. As a result, two or more devices that collide in the same time-slot will have similar power levels. Therefore, the SINR will result higher when the granularity is improved (more power levels), which decreases the number of

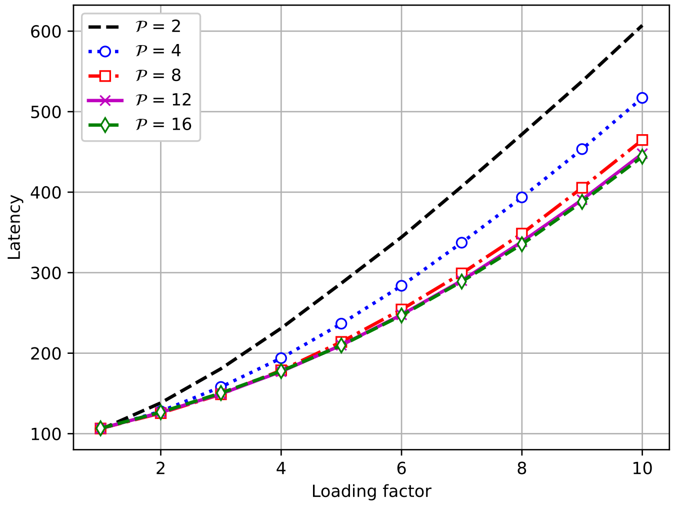


Figure 5. MPL-QL latency (total number of frames).

successes the algorithm is able to attain. The value of $\mathcal{P} = 8$ power-levels was used in the remainder of this section.

B. Convergence of MPL-QL algorithm

The convergence of the MPL-QL algorithm is analyzed by two figures of merit: interference per device and convergence factor per device. The analysis was performed only for the n-th device, but on average the figures of merit for all devices reveal the same behavior.

The interference $I_{n,k}$ of the n-th device at k-th time-slot is calculated as the sum of the powers of the interfering devices that selected the same time-slot, defined by the subset ψ_k; after SIC detection such interference can be calculated as

$$I_{n,k} = \sum_{j=n+1}^{|\psi_k|} P_{j,k}, \qquad [W] \tag{9}$$

while the convergence factor is defined as

$$\nu_n = \frac{L - \ell_n}{L}. \tag{10}$$

At the beginning of the algorithm execution, the n-th device transmitted only a few packets successfully, so $\nu_n \to 0$. When the algorithm is close to the convergence, the n-th device has already transmitted most of its packets, making $\nu_n \to 1$. Fig. 6 depicts the interference along the frames for the n-th device considering a) $L = 50$, b) $L = 100$ [packets/device], and c) convergence factor *vs.* the frame (δ) evolution until convergence ($\nu_n = 1$).

In the beginning of transmission frames, the MPL-QL algorithm can make wrong decisions in choosing the best time-slots and power levels to transmit. Hence, there is an oscillating behaviour of the high interference in the early frames. After a latency $\delta \approx L$ frames, *i.e.*, $L = 50$ and 100 frames in Fig. 6, device interference starts to decrease steadily as devices have already passed the initial learning phase and begin to discover better time-slots and power levels to transmit their packets with greater probability of success. Indeed, when the number of frames is equal to L, a good part of the devices have already transmitted all their packets, as they initially selected the least congested time-slots. Therefore, more empty time-slots start to appear for the n-th device, which makes their interference monotonically decrease after L frames.

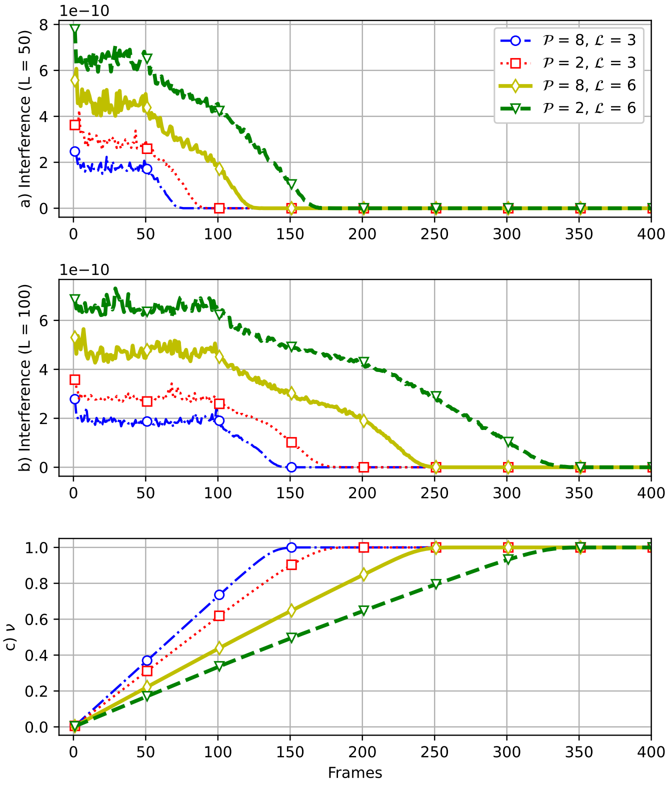


Figure 6. Interference and convergence factor of the MPL-QL algorithm considering the n-th device. a) L = 50 packets; b) L = 100 packets; c) convergence factor under L = 100 packets.

Increasing the loading factor causes more devices to collide in the same time-slot, which causes an increasing in the average interference per device. For this reason, in Fig. 6a) and 6b), the two scenarios with a loading factor $\mathcal{L}$ = 6 have greater interference compared to $\mathcal{L}$ = 3. Moreover, increasing power levels makes the average interference lower. With more power levels, the greater the signal power disparity of the devices that collide in the same time-slot, making the difference between the power of interest and the interfering one greater.

By increasing the loading factor $\mathcal{L}$, more devices are transmitting in the available time-slots. This causes the interference to increase, causing more collisions to happen, increasing the convergence time of the algorithm. For this reason it is possible to observe that the curves for $\mathcal{L}$ = 3 converge faster than the curves for $\mathcal{L}$ = 6.

Increasing the number of power levels also makes convergence occur faster. This is because more available power levels generate a power disparity between two or more devices that collide in the same time-slot, making the signal from the device with the highest power level even greater in relation to the interfering ones. In Fig. 6c), it can be seen that the number of power levels $\mathcal{P}$ = 8 converged faster than the $\mathcal{P}$ = 2 in both loading factor scenarios.

C. Comparison with Other RA Methods

The performance of the proposed MPL-QL algorithm was compared with other methods available in the literature, specifically: ***a***) Slotted Aloha (SA), where there is no feedback from the central node to the devices. The devices only send all their UL packets and the number of successes is obtained when there is no collision; ***b***) Independent QL [16]; ***c***) Collaborative QL [16]; and Packet-Based QL [18].

As the aforementioned QL techniques do not consider different transmitter power levels, then the transmitted power is the same for all devices. This impacts on the evolution of the QL algorithm; hence, as the Q-table reveals in Fig. 3, it does not present the dimension of the powers for such techniques, being considered only the time-slots dimension. Thus, the device learning process is performed only to find the best time-slot for transmission with minimal probability of collision.

The difference between the three QL-based algorithms deployed in the comparison is in the way in which the reward is done by the central node. Hence, in the *independent QL* algorithm, the reward sent by the central node is defined as:

$$R_{n,k}^{\text{IND}} = \begin{cases} +1, & \text{if } \gamma_{n,k}^{\text{NOMA}} \geq \bar{\gamma} \\ -1, & \text{otherwise.} \end{cases} \tag{11}$$

It is a binary reward, similar to the MPL-QL, but it is performed only in the time-slot dimension. On the other hand, for the collaborative QL, the congestion level of the time-slot is defined as:

$$C_k = \frac{|\psi_k|}{N}; \tag{12}$$

and included in the negative reward of *collaborative QL*:

$$R_{n,k}^{\text{COL}} = \begin{cases} +1, & \text{if } \gamma_{n,k}^{\text{NOMA}} \geq \bar{\gamma} \\ -C_k, & \text{otherwise.} \end{cases} \tag{13}$$

As a result, the collaborative QL algorithm is more complex than independent QL, since the central node needs to know the number of devices that collided in each time slot. However, the performance is superior as more information related to the system state is sent during the execution of the algorithm [16].

The *packet-based QL* considers the convergence factor of Eq. (10) and includes such factor in the its reward:

$$R_{n,k}^{\text{PAC}} = \begin{cases} +1, & \text{if } \gamma_{n,k}^{\text{NOMA}} \geq \bar{\gamma} \\ -\dfrac{L - \ell_n}{L}, & \text{otherwise.} \end{cases} \tag{14}$$

Packet-based QL random access strategy favors devices that still have a lot of packets to transmit, sending them a greater reward w.r.t. devices that are already close to convergence [18].

Fig. 7.a) shows the normalized throughput and Fig. 7.b) depicts the latency against loading factor $\mathcal{L}$ for the proposed MPL-QL algorithm, the Slotted Aloha, and the other three QL-based algorithms in the literature. For these results, K = 100 time-slots/frame and L = 100 packets per device were considered. SA is the simplest RA protocol, since devices randomly select a time-slot with no reward sent by the central node to indicate transmission quality. For this reason, the SA throughput is the worst among all the analyzed techniques. Independent and collaborative QL techniques have higher throughput than SA as central node rewards are used for devices to better select which time-slots to transmit. In [16], it is noted that the collaborative QL has a better performance than the independent QL. However, by adding the power domain in NOMA scenarios, the performance of the techniques becomes the same.

The proposed MPL-QL random access method has presented the substantial increasing in the throughput and simulteneously decreasing in the latency for higher loading factors $\mathcal{L}$. Such system throughput and latency improvements can be explained by the fact that in the realistic scenario where devices are subject to the effects of path loss and fading, increasing the power diversity at the transmitter allows nearby devices to transmit at different powers, which increases the SINR after the SIC, at the receiver side. Therefore, more successes are expected to occur when using MPL-QL compared to other QL algorithms, increasing the throughput.

Elaborating further, when analyzing the throughput in Fig. 7a) and latency in Fig. 7b), one infers that the power domain can be advantageous to allocate more devices in a time-slot. The QL-based algorithms in the literature exploring only the time-slot domain do not take advantage of the power diversity in the transmitter to avoid collisions, so they tend to converge more slowly.

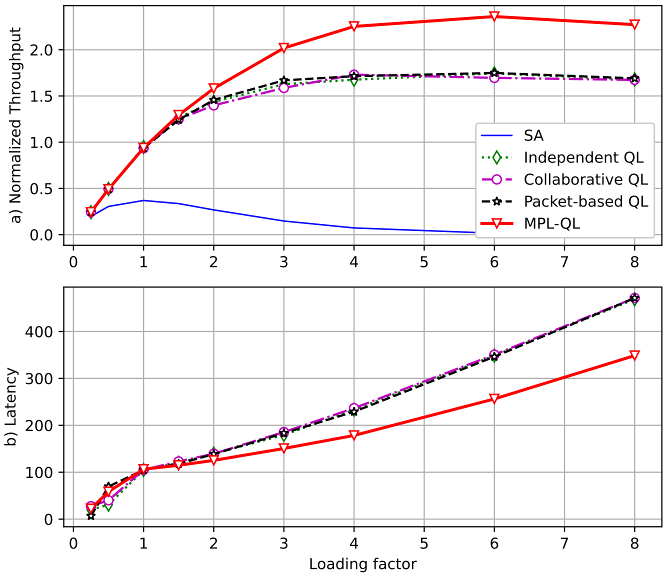


Figure 7. a) Throughput, and b) Latency for the SA and four QL-based algorithms, with $\mathcal{P}$ = 8 for the proposed MPL-QL. L = 100 and K = 100.

Hence, for a loading $\mathcal{L} = 4$, where there are on average 4 devices transmitting per time-slot, the SA protocol is capable of generating little more than $\frac{1}{10}$ success. On the other hand, independent and collaborative QL-based RA algorithms are able to generate 1.7 successes, while the proposed MPL-QL generates 2.3 successes. Therefore, it is shown that, on average, the MPL-QL is able to better deal with the collisions between devices, mainly when the loading factor increasing beyond 3.

D. *QL-based RA Techniques with Imperfect SIC*

The previous results were obtained considering that the central node applies a perfect SIC when receiving packets. However, error-free cancellation is difficult to achieve in crowded mMTC scenarios due to the different levels of interference affecting each signal device. In this subsection, we consider an imperfect SIC model in which there is a residue of the powers of devices that have already passed through the SIC modeling the imperfect signal cancelling effect. Hence, considering NOMA, the new SINR with the imperfect SIC can be written as:

$$\tilde{\gamma}_{n,k}^{\text{NOMA}} = \frac{P_{n,k}}{\beta \sum_{j=0}^{n-1} P_{j,k} + \sum_{j=n+1}^{|\psi|} P_{j,k} + w_k^2}. \qquad (15)$$

where $\beta \in [0, 1]$ is the SIC error factor. $\beta = 0$ indicates that the interference is perfectly cancelled, collapsing in Eq. (5), while when $\beta = 1$, models the absence of SIC procedure at the central node. Typical realistic values for the error factor are in range $\beta \in \{0.01;\ 0.30\}$, depending on the level of interference and of course the received power disparities distribution.

Fig. 8 depicts a comparison of the throughput of QL-based techniques considering $\beta \in [0; 0.1; 0.2]$. As expected, increasing β worsens the throughput of all algorithms, as interference increases, which consequently increases collision and latency until the algorithm attain convergence. Notice that by increasing the value of β decreases the maximum number of devices that the system serves. MPL-QL achieves maximum throughput for a loading factor $\mathcal{L}$ = 6 operating under perfect SIC, *i.e.,* β = 0. For β = .01 and β = 0.02, the loading factor to achieve maximum throughput is reduced to $\mathcal{L} = 3$ and $\mathcal{L} = 2$, respectively. Such decrease is also due to increased latency due to increased interference that can not be cancelled ($\beta \neq 0$). The MPL-QL method proved to be superior to the other QL-based algorithms in all RA crowded scenarios, thanks to the greater power-level granularity, allowing the power differences between the desired device and the interferers larger, reducing collisions.

V. Conclusions

The performance and convergence of the proposed MPL-QL method for different power-levels granularity have been characterized and compared with other QL-based RA algorithms. It was observed that the best number of power levels that guarantee a good performance-complexity trade-off is $\mathcal{P} = 8$ levels. This value was used to compare the throughput with other recent grant-free RA algorithms, namely the independent, collaborative, and packed-based QL-based algorithms, and the classical SA method. The 8-levels MPL-QL technique has revealed the best performance compared to the other analyzed RA techniques, due to the enough power diversity generated by the MPL-QL technique, improving the SINR at the receiver side, while increasing the chance of successful transmissions of a greater number of devices in crowded RA scenarios.

The proposed MPL-QL method demonstrated superiority in both throughput and latency regarding the other QL-based algorithms in all RA crowded scenarios analyzed, due to the greater power-level granularity, allowing the power differences between the desired device and the interferers larger, reducing collisions.

References

[1] H.-M. Chen, J. Liu, H. Su, S. Lin, J. Zhu, and L. Chen, "Towards energy and resource efficient design for scalable mMTC with a distributed energy-restricted cluster based transmission scheme," in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, 2020, pp. 1309–1313.

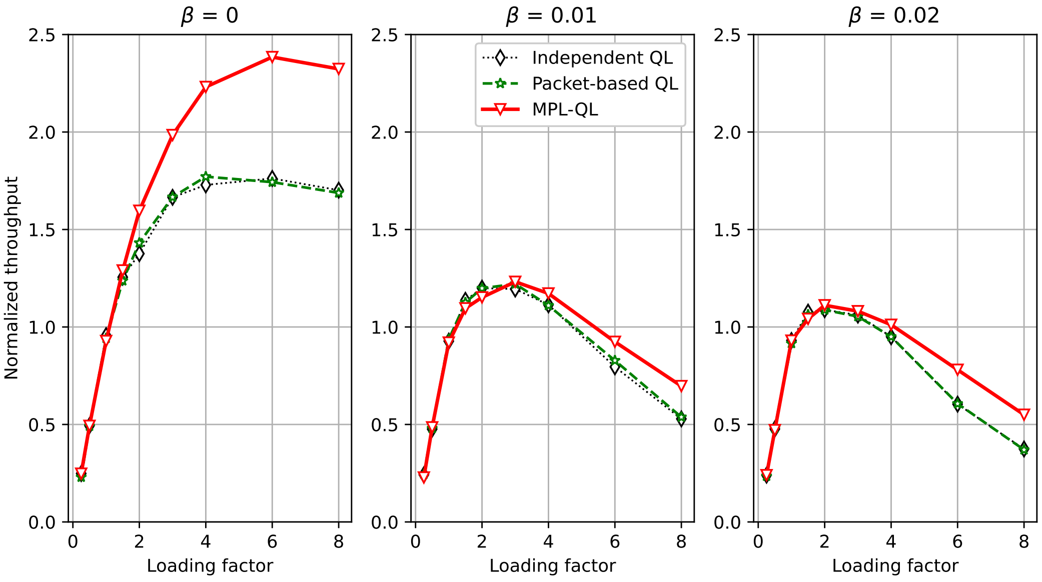


Figure 8. Independent QL, Packet-based QL and MPL-QL under SIC imperfection: a) β = 0; b) β = 0.01; β = 0.02. We considered L = 100 and K = 100.

[2] C. Kalalas and J. Alonso-Zarate, “Massive connectivity in 5G and beyond: Technical enablers for the energy and automotive verticals,” in *2020 2nd 6G Wireless Summit (6G SUMMIT)*, 2020, pp. 1–5.

[3] M. Z. Chowdhury, M. Shahjalal, S. Ahmed, and Y. M. Jang, “6G wireless communication systems: Applications, requirements, technologies, challenges, and research directions,” *IEEE Open Journal of the Communications Society*, vol. 1, pp. 957–975, 2020.

[4] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, D. Niyato, O. Dobre, and H. V. Poor, “6G internet of things: A comprehensive survey,” *IEEE Internet of Things Journal*, pp. 1–1, 2021.

[5] Y. L. Lee, D. Qin, L.-C. Wang, and G. H. Sim, “6G massive radio access networks: Key applications, requirements and challenges,” *IEEE Open Journal of Vehicular Technology*, vol. 2, pp. 54–66, 2021.

[6] W. Jiang, B. Han, M. A. Habibi, and H. D. Schotten, “The road towards 6G: A comprehensive survey,” *IEEE Open Journal of the Communications Society*, vol. 2, pp. 334–366, 2021.

[7] J. R. Bhat and S. A. Alqahtani, “6G ecosystem: Current status and future perspective,” *IEEE Access*, vol. 9, pp. 43 134–43 167, 2021.

[8] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, “5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view,” *IEEE Access*, vol. 6, pp. 55 765–55 779, 2018.

[9] J. Jiao, L. Xu, S. Wu, R. Lu, and Q. Zhang, “MSPA: Multi-slot pilot allocation random access protocol for mMTC-enabled IoT system,” *IEEE Internet of Things Journal*, pp. 1–1, 2021.

[10] T. Wang, Y. Wang, C. Wang, Z. Yang, and J. Cheng, “Group-based random access and data transmission scheme for massive mtc networks,” *IEEE Transactions on Communications*, pp. 1–1, 2021.

[11] O. S. Nishimura, J. C. Marinello, and T. Abrão, “A grant-based random access protocol in extra-large massive MIMO system,” *IEEE Communications Letters*, vol. 24, no. 11, pp. 2478–2482, 2020.

[12] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*, 2nd ed. Cambridge: The MIT Press, 2018.

[13] M. Wiering and M. van Otterlo, *Reinforcement Learning: State-of-the-Art*, 1st ed. Berlin: Springer-Verlag, 2012.

[14] O. J. Pandey, T. Yuvaraj, J. K. Paul, H. H. Nguyen, K. Gundepudi, and M. K. Shukla, “Improving energy efficiency and QoS of LPWANs for IoT using Q-Learning based data routing,” *IEEE Transactions on Cognitive Communications and Networking*, pp. 1–1, 2021.

[15] D.-D. Tran, S. K. Sharma, S. Chatzinotas, and I. Woungang, “Q-Learning-Based SCMA for efficient random access in mMTC networks with short packets,” in *2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2021, pp. 1334–1338.

[16] S. K. Sharma and X. Wang, “Collaborative distributed Q-learning for RACH congestion minimization in cellular IoT networks,” *IEEE Communications Letters*, vol. 23, no. 4, pp. 600–603, 2019.

[17] M. V. da Silva, R. D. Souza, H. Alves, and T. Abrão, “A NOMA-based Q-learning random access method for machine type communications,” *IEEE Wireless Communications Letters*, vol. 9, no. 10, pp. 1720–1724, 2020.

[18] G. M. F. Silva and T. Abrão, “Throughput and latency in the distributed Q-Learning random access mMTC networks,” arXiv, Nov. 2021.

[19] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge: The MIT Press, 2018.

[20] N. Habib, *Hands-On Q-Learning with Python*, 1st ed. Birmingham: Packt Publishing Ltd, 2019.

APPENDIX D – Python Code for NOMA QL

The following Python source code can be used to measure the normalized throughput of the NOMA QL algorithm when considering a perfect SIC at the receiver:

```
######## import ########
import numpy as np
from numpy import random as rand
from numpy.core.function_base import linspace
import matplotlib.pyplot as plt
import datetime as dt
import os

######## parallel algorithm ########
def noma_algorithm(NumDevices: int, NumTimeSlots: int, NumPackets: int, \
    NumTransmitPower: int, SINR_th: float, LearningRate: float, \
    Pmax: int, method: str):

    # number of power levels needs to be even
    if (NumTransmitPower % 2) == 1: return 0
    # check for valid algorithm
    if method != 'SA' and method != 'NOMA-QL' and method != 'Ind-QL' \
        and method != 'Col-QL' and method != 'Pac-QL': method = 'SA'

    # init variables
    PathLossExp = 3
    CellRadius = 200
    Gtx_dB = 0
    Grx_dB = 0
    NoisePSD_dBm = -150
    Freq = 915e6
    Bandwidth = 125e3
    c = 3e8
    d0 = 1

    # normalizes the power to the number of levels (NOMA-QL only)
    if method != 'NOMA-QL':
        NumTransmitPower = 1
        PossibleTxPower = Pmax
        MeanPower = Pmax
    else:
        PossibleAmplitudes = linspace(-np.sqrt(Pmax),np.sqrt(Pmax),\
            NumTransmitPower)
    PossibleTxPower = PossibleAmplitudes**2
    MeanPower = np.sum(PossibleTxPower)/NumTransmitPower

NoisePSD = (10**(NoisePSD_dBm/10))/1e3
NoisePower = NoisePSD*Bandwidth
# friis equation
ReferencePower_dB = Gtx_dB + Grx_dB + 20*np.log10(c/(4*d0*Freq*np.pi))
# raondom device position in a circular cell
d = CellRadius*np.sqrt(rand.uniform(0,1,(NumDevices,1)))
RemainingPackets = NumPackets*np.ones((NumDevices,1),dtype=int)
Qtable = rand.uniform(-1,1,(NumDevices,NumTimeSlots,NumTransmitPower))
Reward = np.zeros((NumDevices,NumTimeSlots,NumTransmitPower))
flag_unique_slots = False
Successes = 0
TotalLatency = 0

# q-learning algorithm
while (np.sum(RemainingPackets) > 0):

    # exit if devices have found unique time slots to transmit
    if (flag_unique_slots == True): break

    if method != 'SA':
        flag_unique_slots = True

    # generate channel samples
    ChannelSamples = (rand.normal(0,1,[NumDevices,1]) +
        1j*rand.normal(0,1,[NumDevices,1]))/np.sqrt(2)
    ChannelPower = abs(ChannelSamples)**2

    # select time slots
    SelectedTimeSlots = np.zeros((NumDevices,1))
    SelectedPower = np.zeros((NumDevices,1))
    TransmitPower = np.zeros((NumDevices,1))
    for n in range (0,NumDevices):
        if RemainingPackets[n] > 0:
            if method == 'SA':
                SelectedTimeSlots[n] = rand.choice(\
                    range(0,NumTimeSlots)) + 1
                TransmitPower[n] = PossibleTxPower
            else:
                # get max values from Q-Table
                Qvaluemax = np.amax(Qtable[n,:,:],keepdims=True)
                Qvalueindexes = np.where(Qtable[n,:,:] == Qvaluemax)
                SelectedTimeSlots[n] = rand.choice(\
                    Qvalueindexes[0]) + 1
                SelectedPower[n] = rand.choice(Qvalueindexes[1])
            if method == 'NOMA-QL':
                TransmitPower[n] = PossibleTxPower[\
                    int(SelectedPower[n])]
            else:
                TransmitPower[n] = PossibleTxPower

# calculate transmit powers
AveragePower_dB = ReferencePower_dB - \
    10*PathLossExp*np.log10(d/d0)
AveragePower = 10**(AveragePower_dB/10)
AveragePower *= TransmitPower/MeanPower
Power = ChannelPower*AveragePower

# search for interferent devices among all time slots
for k in range(0, NumTimeSlots):
    InterfDevices = np.where(SelectedTimeSlots == (k+1))[0]
    if (len(InterfDevices) > 0):
        if method != 'SA':
            # calculate SINR
            SINR = np.zeros((len(InterfDevices),1))
            SortedPower = sorted(Power[InterfDevices],reverse=True)
            SortedPower_Index = sorted(range(len(\
                Power[InterfDevices])),\
                key=Power[InterfDevices].__getitem__,\
                reverse=True)

            InterfPower = np.sum(SortedPower)
            for m in range(0,len(InterfDevices)):
                n = InterfDevices[int(SortedPower_Index[m])]
                NoiseSample = (rand.normal(0,1) +
                    1j*rand.normal(0,1))*np.sqrt(NoisePower/2)
                InstantNoisePower = abs(NoiseSample)**2

                if method == 'NOMA-QL':
                    p = int(SelectedPower[n])
                else:
                    p = 0
                InterfPower -= SortedPower[m]
                SINR[m] = SortedPower[m]/(InterfPower + \
                    InstantNoisePower)
                # check for a success transmission
                if(SINR[m] > SINR_th):
                    if method == 'NOMA-QL' or \
                        method == 'Ind-QL' or \
                        method == 'Col-QL' or \
                        method == 'Pac-QL':
                        RemainingPackets[n] -= 1
```

```
                            Successes += 1
                            Reward[n,k,p] = 1
                        else:
                            if method == 'Col-QL':
                                Reward[n,k,p] = \
                                    -len(InterfDevices)/NumDevices
                            elif method == 'Pac-QL':
                                epsilon = 1 - \
                                    (RemainingPackets[n]/NumPackets)
                                Reward[n,k,p] = -epsilon
                            else:
                                Reward[n,k,p] = -1

                        if method == 'NOMA-QL' or method == 'Ind-QL' or \
                            method == 'Col-QL' or method == 'Pac-QL':
                            Qtable[n,k,p] = (1-\
                                LearningRate)*Qtable[n,k,p] \
                                + LearningRate*Reward[n,k,p]

                    if method == 'SA':
                        RemainingPackets[InterfDevices] -= 1
                else:
                    if len(InterfDevices) == 1:
                        Successes += 1
                    RemainingPackets[InterfDevices] -= 1

            if (len(InterfDevices) > 1):
                flag_unique_slots = False

        # increase latency
        TotalLatency += NumTimeSlots

    # calculate throughput after the algorithm ends
    Throughput = Successes/TotalLatency
    return Throughput

######## parameters ########
NumSimulations = 1
NumDevices = np.arange(100,600,100)
# Algorithms: 'SA','Ind-QL','Col-QL','Pac-QL', and 'NOMA-QL'
QL_Algorithm = 'Ind-QL'
NumTimeSlots = 100
NumPackets = 100
NumPowerLevels = 8
MaximumTxPower = 1e-3
LearningRate = 0.1
SINR_threshold = 3
```

```
######## calculate throughput for each N ########
os.system('cls')
LoadingFactor = NumDevices/NumTimeSlots
MeanThroughput = np.zeros((len(NumDevices),1))
current_time = dt.datetime.now().strftime("%d/%m/%Y %H:%M:%S")
print(current_time)
print("Simulating...")
for ind in range(0,len(NumDevices)):
    Throughput = []
    for irep in range(0,NumSimulations):
        Throughput.append(noma_algorithm(NumDevices[ind],NumTimeSlots,\
            NumPackets,NumPowerLevels,SINR_threshold,LearningRate,\
                MaximumTxPower,QL_Algorithm))
    MeanThroughput[ind] = np.mean(Throughput)
    print('NumDevices = %d, Throughput = %.04f' % (NumDevices[ind],\
        MeanThroughput[ind]))
current_time = dt.datetime.now().strftime("%d/%m/%Y %H:%M:%S")
print(current_time)

######## plot figure ########
fig, ax = plt.subplots()
ax.plot(LoadingFactor,MeanThroughput,'bo-.',label=QL_Algorithm)
ax.set(xlabel='Loading factor', ylabel='Normalized throughput')
ax.grid()
ax.legend()
plt.show()
```