Uplink Resource Allocation Methods for Next Generation Wireless Networks

Alix Jeannerot

N° d'ordre NNT : 2024ISAL0109

THÈSE DE DOCTORAT DE L'INSA LYON,
Membre de l'Université de Lyon

École Doctorale N° 160
Électronique, Électrotechnique, Automatique

Spécialité/ discipline de doctorat :
Traitement du signal et de l'image

Soutenue publiquement le 16/12/2024, par :

Alix JEANEROT

Uplink Resource Allocation Methods for Next-Generation Wireless Networks

---

Devant le jury composé de :

LOSCRI	Valeria	Directrice de Recherche	Inria Lille	Rapporteuse
LIVA	Gianluigi	Chargé de Recherche, HDR	German Aerospace Center	Rapporteur
FIJALKOW	Inbar	Professeure	ENSEA	Examinatrice
POPOVSKI	Petar	Professeur	Aalborg University	Examinateur
VALCARCE	Alvaro	Ingénieur de Recherche	Nokia Bell Labs	Examinateur
ADJIH	Cédric	Chargé de Recherche	Inria Saclay	Examinateur
GORCE	Jean-Marie	Professeur	INSA Lyon	Directeur de thèse
EGAN	Malcolm	Chargé de Recherche	Inria Lyon	Co-encadrant

INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

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

Bâtiment Charlotte Perriand 3e étage
17, av. des Arts
69621 Villeurbanne Cédex
fedora@insa-lyon.fr

Référence : TH1170_JEANNEROT Alix

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

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

Département FEDORA – INSA Lyon - Écoles Doctorales

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

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

Abstract

Facing the diversity of communication needs of 5G networks and the future 6G, resource allocation is considered as a key enabler to increase the number of devices, the data rate or the reliability of the communication links. In machine-type communications networks, recent work has proposed to adapt the temporal resource allocation as a function of the underlying process driving the activity of the devices. This thesis firstly focuses on the impact of having only limited knowledge of the underlying process, and proposes methods to mitigate the bias induced by the lack of knowledge. Secondly, an algorithm for the joint optimization of the temporal resource allocation and the transmit power of the devices is proposed. The algorithm ensures that devices that are likely to transmit on the same resources do so with a sufficient power diversity to ensure their decodability by the base station. Finally, in networks with an enhanced mobile broadband objective, we propose to jointly optimize the power, the frequency resources used, as well as the number of parallel data streams used by the devices. Our simulation study shows that our joint optimization outperforms current 5G baselines for which these parameters are common to all devices of the cell.

Résumé

Face à la diversité des besoins en communication des réseaux 5G et de la future 6G, l'allocation des ressources disponibles est considérée comme un élément clé pour augmenter la densité de dispositifs, leur débit ou la fiabilité de leurs communications. Dans les réseaux de communication de type machine, des travaux récents ont proposé d'adapter l'allocation des ressources temporelles en fonction du processus sous-jacent qui régit l'activité des dispositifs. Cette thèse se concentre tout d'abord sur l'étude de l'impact d'une connaissance imparfaite de ce processus, et propose des méthodes pour atténuer le biais induit par les connaissances erronées. En second lieu, un algorithme permettant d'optimiser conjointement l'allocation des ressources temporelles et la puissance de transmission des dispositifs est proposé. L'algorithme permet aux dispositifs ayant une forte probabilité de transmettre au même moment, de le faire sur des ressources (temporelles ou de puissance) assurant leur décodabilité. Enfin, dans les réseaux ayant un objectif de haut débit, nous proposons d'optimiser conjointement la puissance, les ressources fréquentielles ainsi que le nombre de flux de données parallèles utilisées par les dispositifs. Notre étude par simulations témoigne que notre optimisation conjointe est plus performante que les méthodes utilisées actuellement en 5G pour lesquelles ces paramètres sont calculés indépendamment les uns des autres.

Contents

List of Figures

List of Tables

Acronyms

With page of first occurrence.

2G	2^{nd} generation	2
3G	3^{rd} generation	2
4G	4^{th} generation	2
5G	5^{th} generation	2
6G	6^{th} generation	2
AWGN	additive white Gaussian noise	10
BCE	binary cross-entropy	85
BS	base station	2
CDF	cumulative distribution function	89
CPLC	closed loop power control	87
CRDSA	collision resolution diversity slotted ALOHA	20
CSA	coded slotted ALOHA	20
CSI	channel state information	4
CSIR	channel state information at receiver	30
CSIT	channel state information at transmitter	30
eMBB	enhanced mobile broadband	2
FSA	frame slotted ALOHA	19
GAMP	generalized approximate message passing	17
GD	gradient descent	44
GFRA	grant-free random access	15
HPPP	homogeneous Poisson point process	66
i.i.d.	independent and identically distributed	23
IoT	Internet of Things	3
IRSA	irregular repetition slotted ALOHA	20
MAC	multiple access control	2
MBB	mobile broadband	2
MIMO	multiple input multiple output	10
MINLP	mixed integer non-linear program	79
ML	machine learning	10
MMSE	minimum mean squared error	30
MRC	maximum ratio combining	30
MTC	machine-type communications	2
MU-MIMO	multiple user multiple input multiple output	28
NB-IoT	Narrowband IoT	13

NLP	non linear program	79
NN	neural network	79
NOMA	non-orthogonal multiple access	12
ODE	ordinary differential equation	130
OFDM	orthogonal frequency division multiplexing	12
OLPC	open loop power control	87
OMA	orthogonal multiple access	12
P2P	Point-to-Point	10
PAPR	peak to power average ratio	82
PD-NOMA	power domain non-orthogonal multiple access	12
PHY	physical	2
PRB	physical resource block	12
PRBG	physical resource block group	12
PSD	power spectral density	78
RA	random access	3
RACH	random access channel	15
RBG	resource block group	12
RB	resource block	12
RE	resource element	12
RG	resource grid	12
RRC	radio ressource control	16
SGD	stochastic gradient descent	5
SIC	successive interference cancellation	20
SIMO	single input multiple output	59
SINR	signal to interference plus noise ratio	27
SNR	signal to noise ratio	27
UE	user equipment	10
UL	Uplink	10
uRLLC	ultra reliable low latency communications	2
ZF	zero forcing	30

CHAPTER

1 Introduction

Contents

1.1 Wireless Cellular Communication: from 2G to the Future 6G

Starting from only voice communication and small text messages, wireless cellular network now also provide high speed access to Internet. This shift of focus has been guided by the 3rd generation partnership project (3GPP), a consortium of standard institutes, companies, telecommunications operators, etc..., which helps the development and standardization of the different generations of cellular communication. It produces different technical specifications that act as basis for the definition of the different standards. Furthermore, it is in charge of defining the requirements and the use cases of the networks, and describes in details how they will be implemented and interoperated.

Through the different generations of wireless cellular communication, each one provided new use cases. The 2nd generation (2G) sparked the democratization of the mobile phone by allowing to make calls and to send small text messages. Then, the 3rd generation (3G) generalized the ability to connect to internet by making possible to send and receive data packets. Afterwards, the 4th generation (4G) further increased the supported data rate which made possible the stream of videos from Internet, video calls, and more generally enabled the versatility of smartphones. The focus of 2G, 3G, and 4G was centered around providing the highest data rate to the devices (known as mobile broadband (MBB)): the more data the better.

In contrast with the previous generations, the 5th generation (5G) was developed around three different axes, each providing different applications for people as well as the industry:

- enhanced mobile broadband (eMBB), with an aim to increase by a factor of 10 to 100 the theoretical data rate compared to 4G; as a consequence, the average monthly data consumption per device went from 5 GB in 2018 to 20 GB in 2024 [33, Fig. 7].
- machine-type communications (MTC), designed to support the maximal number of devices having strict power constraints, with applications to sensor networks or smart factories [110]. Device density could theoretically reach up to 1 million devices per km^2 [50].
- ultra reliable low latency communications (uRLLC), with the purpose to provide fast and reliable delivery of data packets with applications to time-sensitive networks [84]. Ideally, small data packets should be received by the base station (BS) within 1 ms with a probability of $1 - 10^{-5}$[50].

The future 6th generation (6G), expected to be standardized around 2030, is envisioned to pursue the development of the 5G axes with massive machine-type communications (mMTC) [71], extreme uRLLC [80] and provide new services at their intersection [92] like massive uRLLC [60, 115].

To support such a wide variety of use cases, each of these axes have different technical specificities. In particular, the physical (PHY) and multiple access control (MAC) layers have to be designed differently depending on the target application of the network.

1.2 PHY and MAC Layers

At the basis of wireless communication networks lies the PHY and the MAC layers. The PHY layer job is to communicate bits over the wireless channel using, in the case of wireless communication, electromagnetic waves. In this thesis, we focus on communication from devices to the BS (uplink). The PHY layer defines how the bits are encoded into different waveforms and controls the transmit power of the devices according to the data rates required and the regulations.

Lying on top of the PHY layer, the MAC layer orchestrates the communication of multiple devices over the same medium. Because the exact moment a device seeks to transmit data is unknown to the BS, the term *random access (RA)* is used. The aim of the MAC layer is to minimize the loss of data due to interference between devices, also known as collisions. As such, resource allocation is a key part of the MAC layer.

Traditionally, the PHY and MAC layers are viewed as two different building blocks of the network and are optimized independently of each other. However, the different needs of the axes of 5G force to rethink the relationship between the different layers and open the doors of performing cross-layer optimization.

A current critical problem in the development of the next generations of wireless communication networks is being able to efficiently handle communications coming from a possibly large number of devices. Given that the communication needs strongly differ whether the network is eMBB, MTC, or uRLLC, we should seek different solutions for each axis. This thesis thus revolves around optimization of uplink communications at the level of the PHY and MAC layers through power control, and the allocation of resources in a context where multiple devices want to transmit data.

The optimization of the PHY and MAC layers we propose in the different chapters rely on long-term analysis, either because the optimization algorithms we develop converge asymptotically or because the metrics we use are modelling asymptotic behaviors. As a result, we choose to not consider uRLLC networks, which require guarantees at a finite horizon, and solely focus on the MTC and eMBB types of networks.

1.3 Motivations and Challenges in MTC

MTC networks consist of Internet of Things (IoT) devices that operate without the intervention of humans. They send data, which can be measurements or control information, periodically or whenever they sense an event. Well-known applications of MTC networks are for:

- metering, like water or electricity meters
- controlling systems, with applications to smart industry or home automation,
- tracking of objects, used in fleet management or parcel tracking
- monitoring an area (detection of intrusions),

- sensor networks (measuring environmental conditions on large areas).

The vast majority of IoT devices are small, low cost, and battery powered. The promises of MTC networks are to contribute to the digitalization of society, with hopes of improving the efficiency of some activities or industries.

The different application scenarios often require the deployment of a large number of devices. Given that the communication resources are limited, a key problem is to ensure that all these devices can inter-operate and reliably communicate their data. To avoid wasting power in requesting resources, the selection of the resources used for communication is decentralized and made randomly at the device. As such, in principle, no coordination mechanisms are implemented, whether between the device and the BS nor between devices.

1.4 Motivations and Challenges in eMBB

Contrarily to MTC, eMBB networks focus on improving the data rate of the devices. The applications are multifold:

- increase the capacity of the cells by supporting a higher number of devices,
- better support device mobility to enable seamless connectivity in moving vehicles (like trains),
- obtain higher peak data rate, to support applications with high data need (streaming of ultra-high definition videos).

Given that devices have large amounts of data to send, communication of control information and channel state information (CSI) is possible. As a result, the scheduling and usage of the communication resources can be well orchestrated by the BS in a centralized manner. Furthermore, with the help of the CSI, the devices can set their transmit power to match an expected received power at the BS. However, due to the presence of hard constraints in the maximal transmit power, some devices fail to meet the requirements and experience a degraded data rate.

1.5 Contributions of the Thesis and List of Publications

1.5.1 Contributions

Given that devices of MTC networks have typically small data packets to send, coordination between the devices and the BS should be as limited as possible to avoid wasting energy. In the state of the art, it is common to assume that the *a priori* probability that a specific device is active is the same for all devices of the network. In the first two contributions of this thesis, we explore whether the knowledge, either partial (in Chapter 3) or complete (in Chapter 4), of an underlying process driving the activity of the devices can be used as a form of coordination to increase the number of devices that can be supported by the network. The algorithms developed thus lie between decentralized scheduling and centralized orchestration. Furthermore, in Chapter 4, in a form of

PHY-MAC cross-layer optimization, the transmit power of the devices is also controlled to provide new opportunities of decoding collisions.

In eMBB networks, because it is possible to afford more control communication, the usage of the communication resources can be carefully orchestrated to ensure that no collisions occur. Currently, in 5G, the computation by the BS of the transmission parameters used by the devices is based on heuristics and performed independently for each device. In Chapter 5, we propose to jointly design the different transmission parameters of all the devices. We formalize an optimization problem which, based on the CSIs of the devices, returns an allocation of resources and transmit power to each device that maximizes the geometric mean of the rates.

1.5.2 Readers Guideline

The thesis is organized as follows:

- Chapter 2 presents the different models that will be used as well as relevant related work. We present the different channel models, the notion of communication resources, as well as methods to allocate resources to the devices.
- Chapter 3 studies the impact of imperfect user detection when designing resource allocation methods that are a function of the activity statistics of the devices. A method, based on importance sampling, to mitigate the bias introduced by the limited knowledge of the active devices is presented.
- Chapter 4 studies the possibility of performing a joint allocation of slot and power. The algorithm we propose is able to take into account the correlations in the activity of the devices and to assign orthogonal resources (slot or power diversity) to the devices that are highly likely to transmit together.
- Chapter 5 formalizes an optimization problem to jointly assign to all devices of the network their transmit power, the frequencies they use, as well as the number of parallel data streams.
- Chapter 6 summarizes our work, highlights its limitations, and presents some possible future directions.
- Chapter 7 provides a summary in French of the different contributions (Chapter 3, 4 and 5).
- Finally, necessary background in gradient-based optimization is provided in Appendix A. The analysis of the stochastic gradient descent (SGD) algorithm is made through the point of view of stochastic approximations.

1.5.3 Publications

Journals

J1 Malcolm Egan, Alix Jeannerot, Lélio Chetot, Jean-Marie Gorce. "Gradient Boosting for Online Two-Stage Adaptive Group Testing". Submitted to *IEEE Signal Processing Letters*, 2024

J2 Pavan Srinath, Alix Jeannerot, Alvaro Valcarce. "Joint Resource Allocation, UE Rank Selection, and Power Control in Multi-User MIMO Systems with Linear Transceivers". Submitted to *IEEE Transactions on Wireless Communications*, `https://arxiv.org/abs/2407.16483`, 2024

J3 Alix Jeannerot, Malcolm Egan, Jean-Marie Gorce. "Exploiting Device Heterogeneity in Grant-Free Random Access: A Data-Driven Approach". In: *IEEE Transactions on Vehicular Technology*, DOI: `https://doi.org/10.1109/TVT.2024.3396825`, 2024

Conferences

C1 Maxime Vaillant, Alix Jeannerot, Jean-Marie Gorce. "Joint Constellation Shaping Using Gradient Descent Approach for MU-MIMO Broadcast Channel". In: *IEEE SPAWC*, DOI: `https://doi.org/10.1109/SPAWC60668.2024.10694191`, 2024

C2 Alix Jeannerot, Malcolm Egan, Jean-Marie Gorce. "Joint Slot and Power Optimization for Grant Free Random Access with Unknown and Heterogeneous Device Activity". In: *EUSIPCO*, `https://inria.hal.science/hal-0460328`, 2024

C3 Alix Jeannerot, Malcolm Egan, Lélio Chetot, Jean-Marie Gorce. "Mitigating User Identification Errors in Resource Optimization for Grant-Free Random Access". In: *IEEE 97th Vehicular Technology Conference (VTC-Spring)*, DOI: `https://doi.org/10.1109/VTC2023-Spring57618.2023.10200062`, 2023

C4 Alix Jeannerot, Malcolm Egan, Lélio Chetot, Jean-Marie Gorce. "Compensation des Erreurs d'Identification d'Utilisateurs dans l'Allocation Stochastique de Ressources pour Réseaux Aloha Hétérogènes". In: *GRETSI - Colloque sur le traitement du signal et des images*, `https://inria.hal.science/hal-04257479v1`, 2023

Patent

P1 Pavan Srinath, Alix Jeannerot, Alvaro Valcarce. No public title yet, Applicants: Nokia Solutions and Networks Oy, Inria, 2024

1.5.4 Code Repository

The code developed and used for the simulation of Chapter 3 is available at: `https://gitlab.inria.fr/maracas/publications/exploiting-device-heterogeneity-in-gfra`

1.5 Acknowledgments

This thesis was funded through the "Agence nationale de recherche (ANR)" project "IADoc@UdL - ANR-20-THIA-0007" and by the Inria - Nokia Bell Labs Common Laboratory.

We gratefully acknowledge support from the "Pôle Scientifique de Modélisation Numérique" (PSMN) of the ENS de Lyon for the computing resources used for Chapter 3.

Experiments presented in Chapter 4 were carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria and including CNRS, RENATER and several Universities as well as other organizations (see `https://www.grid5000.fr`).

The contributions presented in Chapter 5 were done during a four-month research stay at Nokia Bell Labs, Massy, France.

CHAPTER

2 System Model - State of the Art

Contents

2.1 The Uplink Communication Channel

In cellular networks, we distinguish two types of communications channel: Uplink (UL) and downlink. In the downlink channel, the BS acts as the transmitter and the devices[1] as receivers. The focus of this thesis is on uplink transmissions: when the devices are the transmitters and the BS is the receiver. The main difficulties of uplink transmissions are making sure each device can reliably communicate with the BS and ensuring enough coordination across the devices to guarantee reliable reception of data coming from several devices. From an historical perspective, the first difficulty is a PHY layer problem and the second a MAC layer problem. However, these two problems are highly linked and are thus more and more studied altogether and jointly optimized in new wireless technologies.

Several directions can be taken to enhance and tune the performance of the network with respect to the different axes of the network: eMBB, IoT or uRLLC. Some key research directions are:

- channel coding [91],
- resource allocation [4, 10, 58],
- power control [3, 66, 70],
- scheduling [44],
- multiple input multiple output (MIMO) [108].

Machine learning (ML) and artificial intelligence (AI) are expected to be key enablers for the optimization of different aspects of 6G networks [8, 82, 96, 102, 104]. This thesis mainly deals with different aspects of resource allocation but also takes into accounts the power optimization in a MIMO setting.

Throughout this chapter, we present different channel models and introduce key problems of the PHY and MAC layers. Contrarily to most existing work, we purposely choose to present in parallel the PHY and the MAC layers instead of presenting them sequentially. The goal is to emphasize on their relationship and to gain intuition on how cross-layer optimization can be performed. Starting with a basic Point-to-Point (P2P) additive white Gaussian noise (AWGN) channel to explain the principles of communications, we then move to a multi-user AWGN channel and explain the concept of communication resources and collisions. Afterwards, we present how the selection of communication resources can be made to, ideally, avoid collisions. We then describe the concept of sporadic communication, a key component of Chapter 3 and Chapter 4, and introduce it in the multi-user AWGN model. Next we elaborate on the notion of channel and MIMO communications, and show how power control can be used to decode colliding devices. Finally, we present different metrics to study the performance of a network.

[1] We will use interchangeably use "device", "user", "user equipment (UE)", and "node".

2.2 Communication Models

2.2.1 Transmission without Interference: Single User AWGN

The most basic P2P communication channel is the AWGN. In this model, for a single channel uses, the data sent d lies in $\mathcal{D} \subset \mathbb{C}$, and the noise w is circularly symmetric white Gaussian $w \sim \mathcal{CN}(0, \sigma^2)$. As we will present in 2.2.7, an interesting channel model is the multiplicative channel noise: $f(d) = hd$, $h \in \mathbb{C}$ which we will ignore for the moment. Conceptually, the multiplicative noise can be canceled either by considering an additive noise with a different variance $\tilde{w} \sim \mathcal{CN}(0, \frac{\sigma^2}{h^2})$ or by doing power control (sending $\frac{d}{h}$ instead of d). The AWGN model is thus:

$$y = d + w, \tag{2.1}$$

The data sent d is known as a symbol and $\mathcal{D}$ is the constellation, where often $\mathcal{D}$ has a finite cardinality ($|\mathcal{D}| < \infty$); without loss of generality, we assume that, at each channel use, the symbols are all equally likely to be sent and that $\mathbb{E}_D[d] = 0$ and $\mathbb{E}_D[\|d\|^2] = 1$, meaning that no continuous component is present and the average energy of the constellation is 1. Typical constellations are pulse amplitude modulation (PAM) ($\mathcal{D} = \{-1, 1\}$) and quadrature amplitude modulation (QAM) ($\mathcal{D} = \{-\frac{\sqrt{2}}{2} - j\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2} + j\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2} - j\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2} + j\frac{\sqrt{2}}{2}\}$).

Symbols are the data unit of the PHY Layer and can be grouped into packets (also known as burst) of size L symbols, making the "packet", the data unit of the MAC layer. The value of L varies on the protocol used, but is usually constant over time. In this thesis, we only focus on memoryless channels without inter symbol interference (ISI) (the transmission of a symbol does not affect the subsequent symbols), hence the model of (2.1) can be easily extended to packet transmission:

$$\mathbf{y} = \mathbf{d} + \mathbf{w}, \tag{2.2}$$

where $\mathbf{y} \in \mathbb{C}^L$, $\mathbf{d} \in \mathcal{D}^L$, $\mathbf{w} \sim \mathcal{CN}(0, \sigma^2 I_L)$, with I_L the $L \times L$ identity matrix.

2.2.2 Transmission with Interference: Multi-User AWGN

Throughout this thesis, we assume that all devices of the network are synchronized and that they all start their transmission at the same time. As a result, supposing we have N devices wishing to send a packet to the BS, (2.2) can be extended to multiple device, leading to what is known as the multi-user AWGN channel:

$$\mathbf{y} = \sum_{n=1}^{N} \mathbf{d}_n + \mathbf{w}, \tag{2.3}$$

where $d_n \in \mathcal{D}_n^L$ is the packet of device n and $\mathcal{D}_n$ is the constellation of device n. However, in a model like (2.3), the data of all the devices interferes; it is thus highly likely that the BS won't be able to recover the data of each device.

2.2.3 Communication Resources

Supposing that the BS is not able to decode any of the devices' data if interference is present in (2.3), then BS can thus decode data only if $N = 1$. Such types of channel are known as collision channels. To circumvent this limitation, we can use different communication resources to create different channels (preferably independent of each other) that can be assigned to the devices of the network.

A communication resource is a physical resource that devices have at their disposition to transmit data to the BS. It can have several forms, the most common being time slots (time division multiplexing), waveforms (waveform division multiplexing), carrier frequency (frequency division multiplexing), or power levels (power division multiplexing)...

We distinguish two type of resources: orthogonal and non-orthogonal. Orthogonal resources ensure that two transmissions over two different orthogonal resources will be independent of each other. For example, if two transmissions take place during two disjoint time intervals, one does not interfere with the other. On the other hand, the use of non-orthogonal resources can lead to interference between the transmissions, making necessary the use of advanced decoders at the receiver to separate the different data flows. Example of non-orthogonal resources are overlapping frequency bandwidth, different transmit power... The multiplexing using orthogonal resources, known as orthogonal multiple access (OMA), is the most reliable and should be preferred whenever possible. Using an OMA scheme for a network of N devices will result in the creation of N independent channels (2.1). However, the amount of orthogonal resources is scarce and the number of communication technologies as well as the number of connected devices are increasing, making it harder and harder to attribute a different orthogonal resource to each device. Furthermore, because devices might not need to transmit data at all times, assigning exclusive resources to an inactive device can be considered as a waste. Recent effort has been put toward developing communication over non-orthogonal resources, known as non-orthogonal multiple access (NOMA), which has been first studied for code division multiple access (CDMA) transmissions [95, 105]. In Section 2.2.7 we will describe how non-orthogonal resources (diversity in transmit power) can be used for communication by using power domain non-orthogonal multiple access (PD-NOMA) as an example.

Resources and Frame Structure in 4G and 5G

Since 4G, orthogonal frequency division multiplexing (OFDM) is used to provide efficient and reliable transmission over several orthogonal subcarriers. The time-frequency representation of the OFDM resource grid (RG) can be seen in Fig. 2.1 and Fig. 2.2 (zoom-out version of Fig. 2.1), with in the x-axis the time domain, and in the y-axis the frequency domain. A pair of an OFDM symbol and a subcarrier $e = (\varphi, \tau)$ forms a resource element (RE). The structure of the 5G frames are defined in [1, Chapter 5.1]. In the frequency domain, the subcarriers are grouped into resource blocks (RBs) (also known as physical resource blocks (PRBs)), with $N_{\mathrm{SC}} = 12$ subcarriers per RB. Then, resource blocks are grouped into resource block groups (RBGs) (or physical resource block groups (PRBGs)) which are allocated to devices, one RBG containing N_{block} contiguous RBs, as seen in Fig. 2.2. The

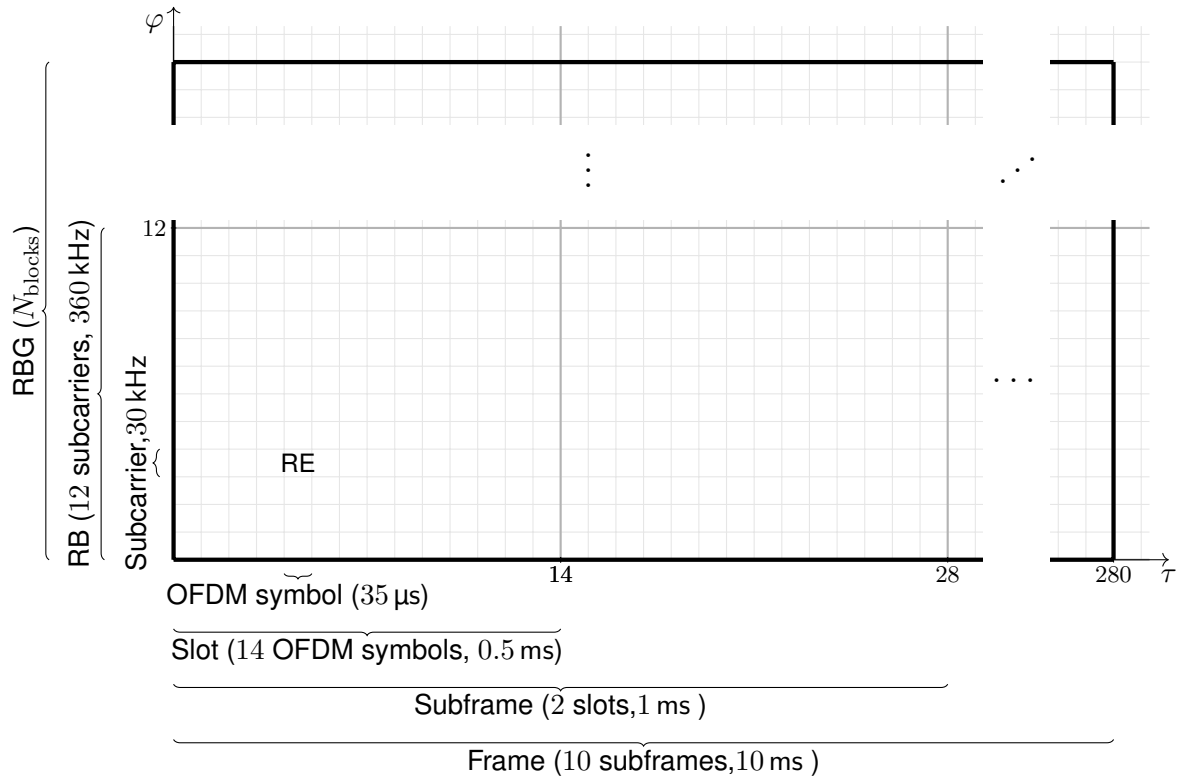


Figure 2.1: Representation of the 5G OFDM RG, at the RE resolution, with the different terminologies in the case of a subcarrier spacing of 30 kHz. On the axes τ represent the symbol index and φ the subcarrier index.

total bandwidth used in 4G is between 5 MHz and 20 MHz and the total 5G bandwidth is noticeably larger, ranging between 5 MHz and 100 MHz. In the time domain, OFDM symbols are grouped into time slots, each consisting of L OFDM symbols ($L = 14$ in 5G and $L = 7$ in 4G). Then, in 5G slots are grouped into subframes of $N_{\text{slot}} \in \{1, 2, 4, 8, 16\}$ depending on the subcarrier spacing S_p (distance between two subcarrier) used, $S_p \in \{15, 30, 60, 120, 240\}$kHz respectively. In 4G, the subcarrier spacing is fixed at 15 kHz, thus each subframes contains 2 slots (because each slot is 7 symbols in 4G). Finally, a radio frame lasts for (10 ms) and consisits of 10 subframes of 1 ms each.

The REs of the RG serve different purposes, depending on the protocols used and their configuration: channel sensing (with transmission of pilots), transmission of control information, transmission of downlink/uplink data...

For the remaining of this chapter as well as in Chapter 3 and in Chapter 4, we use a generic communication model similar to Narrowband IoT (NB-IoT) and suited for MTC application. In NB-IoT, uplink communications occur over a single subcarrier within a narrow bandwidth of 180 kHz, shared by a set of devices that transmit in a single slot per frame. It can operate in two possible subcarrier spacing: 3.75 kHz or 15 kHz. Frames also last for 10 ms and consist in 5 slots of 2 ms in the 3.75 kHz subcarrier spacing, and 20 slots in the 15 kHz subcarrier spacing. Like in 4G, each slot contains $L = 7$ symbols. Chapter 3 and 4 are aimed to provide resource allocation methods within

In Chapter 5, we consider a set of devices co-scheduled in a single slot that transmit their uplink

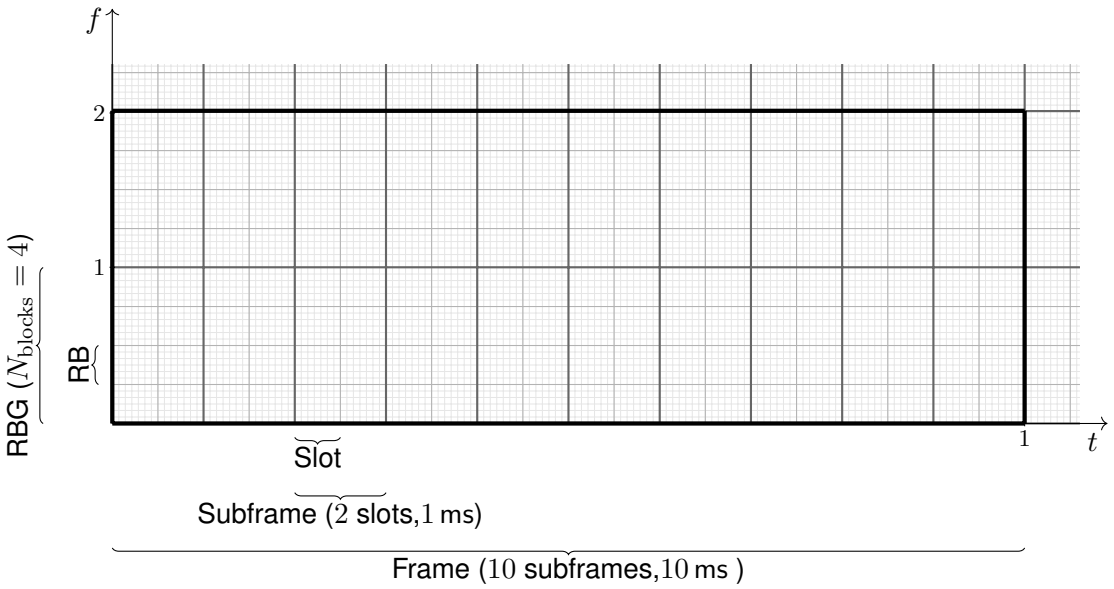


Figure 2.2: Representation of the 5G OFDM RG, at the slot-RB resolution (zoom-out of Fig. 2.1), with the different terminologies in the case of a subcarrier spacing of 30 kHz and with $N_{\text{block}} = 4$. On the axes, t represent the frame index and f the RBG index.

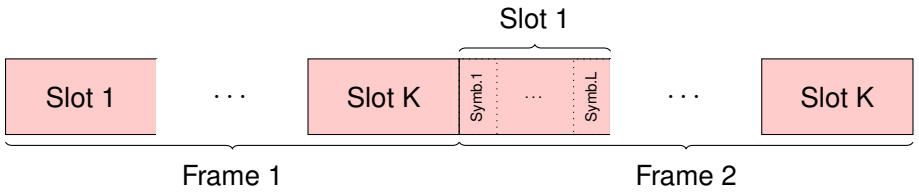


Figure 2.3: Example of two frames over time, each slot containing exactly L symbols.

data on one or several PRBGs.

Collision Channel

Setting aside the technicalities of the technology used, we now consider a generic frame structure that consists of K slots per frames, without subframes ($N_{\text{slot}} = 1$), as shown in 2.3, and a single subcarrier. We further assume that packets fit exactly in a slot and that, because devices are all perfectly synchronized, no packets overlaps two different slots.

Let $\boldsymbol{\delta}_n^t = \begin{bmatrix} \delta_{n,1}^t & \dots & \delta_{n,K}^t \end{bmatrix} \in \{0,1\}^K$ be the vector indicating for device n which slots are used for the transmission of the packet. Note that $\boldsymbol{\delta_n^t}$ can contain several 1, meaning that the same packet might be sent several times in a frame. The channel in slot k of frame t becomes:

$$\mathbf{y}_k^t = \sum_{n=1} \delta_{n,k}^t \mathbf{d}_n^t + \mathbf{w}_k^t. \tag{2.4}$$

We denote $\boldsymbol{\delta}^t \in \{0,1\}^{N \times K}$ the matrix with elements $\delta_{n,k}$ characterizing which resources are utilized in frame t.

Example 2.2.1. Suppose the network consists of $N = 3$ devices, $K = 3$ resources (time slots), the

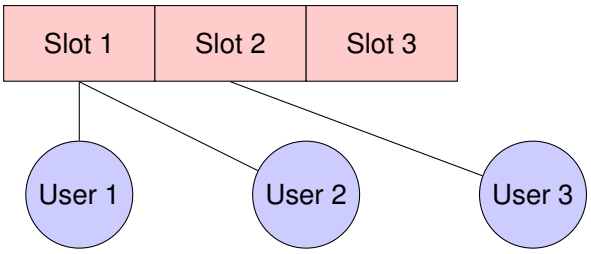


Figure 2.4: Example of a collision: in a frame of $K = 3$ slots, packets of device 1 and 2 collide in the first slot: neither of them can be decoded whereas packet of device 3 can be decoded.

matrix of transmit slots is $\boldsymbol{\delta} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ which is also represented in Fig. 2.4. No collision would have occurred if either device 1 or device 2 had used slot 3 instead of slot 1.

As seen in Example 2.2.1, the choice of the communication resource is important to avoid collisions between packets. In the next subsections, we review different methods to select one or several slots, under the assumption that no collision can be decoded. Note that in the following, to ease the notations, the superscript t, denoting the frame index, is dropped whenever the frame index is unnecessarily.

2.2.4 From RACH-based to GFRA

The channel model of (2.4) is exhaustive enough to delve into how the selection of the transmit resources $\delta_{n,k}$ can be done as well as what the data packets $\mathbf{d}_n$ might contain.

In telecommunications, we differentiate two types of data:

- the payload data, actual useful data that a device wishes to send
- the overhead, used for: control, signaling, synchronization, identification of devices, orchestration of the network, request of resources...

The symbols within a packet $\mathbf{d}_n$ might contain either payload or overhead data or both.

Whenever a device has data to send, it will seek to transmit on a channel. As the exact moment (which frame number t) a device will seek to access the channel is unknown to the base station, the term random access is used. We distinguish two types of random access procedure:

- random access channel (RACH)-based, using a large amount of overhead, particularly suited for MBB
- grant-free random access (GFRA), designed for limiting as much as possible the overhead, particularly suited for MTC and uRLLC applications.

RACH-based

The RACH procedure is based on a four-way handshake and is used to identify the active devices and to obtain transmission parameters including:

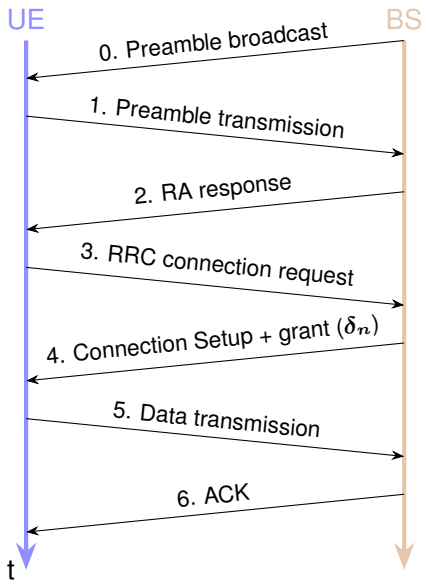
- a grant for exclusive use of one or several resources
- cell-wide parameters of the radio ressource control (RRC) protocol (including open loop power control α and P_0, as will be discussed in Chapter 5.5)

The RACH procedure is initiated by devices whenever they join the network or if they have data to send after being inactive for a while. The procedure is similar in both 4G and 5G and is represented in steps 0-5 of Fig. 2.5a. The procedure can be described as follows [94]:

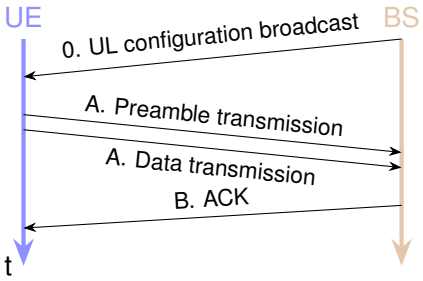
- Step 0: the BS periodically broadcasts information about a set of preamble sequences.
- Msg 1: The devices wishing to announce themselves choose a random preamble in the preamble set and transmit it.
- Msg 2: The BS broadcasts a message comprising the list of all the preambles that have been detected as well as, for all detected preambles, a resource grant for the transmission of the next message.
- Msg 3: Devices send their own unique identifier on the transmit resource corresponding to the preamble it sent in Step 1. If the message of Step 2 does not contain a device's preamble, then the device waits and reinitiates a RACH procedure later.
- Msg 4: the BS acknowledges the presence of the devices on the network, allowing devices to know if they have been properly detected. The message also contains information about cell-wide parameters (like α and P_0, described in Chapter5.5) and a grant of resource (slots and/or PRBs) that the device can use to transmit its data.

Usage of the RACH procedure brings, to the BS, the knowledge of the active devices and because the BS decides which device uses which resource, collisions can be avoided. If the payload to be transmitted is big and does not have strict temporal constraints (as is the case for most eMBB applications) then performing a RACH procedure to obtain a grant is interesting.

However, it is inadequate for MTC: devices shouldn't send 3 packets of overhead if they have a single payload packet to transmit (as is common in MTC networks). Furthermore, it is also inappropriate for uRLLC as the delay induced is not negligible. In 4G, the RACH procedure induces a delay of 10 ms (1 frame) and the grant request an additional delay of 5ms making a total delay of 15 ms before sending the payload [94].



(a) Grant based transmission to ensure collision free data transmissions.



(b) 2-step random access (GFRA) without using a RACH.

Figure 2.5: Comparison between the Grant-based and Grant-free random access.

RACH-Free (GFRA)

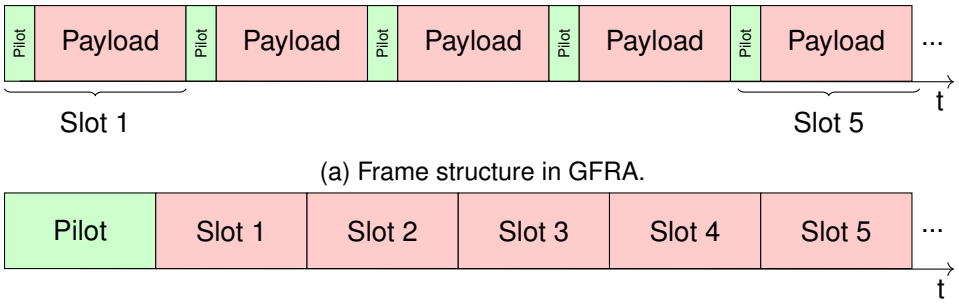
Another possibility is to perform RACH-free and grant-free transmissions using what is known as GFRA [2, Section 19.1.2], [20, 62, 83]. This scheme, depicted in Fig. 2.5b, has the following steps:

- Step 0: BS broadcasts periodically uplink configuration information common to all devices.
- Msg A: the devices randomly select a preamble, transmit it and right after transmit their payload. The preamble selected defines a resource on which the preamble and the data should be transmitted. Similarly to RACH-based access, the preamble is drawn from a set common to all devices. The selected preamble defines a specific slot in which the payload should be transmitted [83].
- Msg B: The BS acknowledges the identity of detected devices and the successful decoding of the data.

For both RACH-based and GFRA transmissions, the design of the preamble sequences is important: they should be orthogonal to ensure that the BS can distinguish the different sequences, the amount of preamble sequences should be large enough to obtain a low probability that two device choose the same sequence, and they should be small enough to avoid wasting too much power in the transmission of the preamble. In 4G and 5G, Zadoff-Chu sequences [16] of different length are used. The BS can use algorithms based on generalized approximate message passing (GAMP) [18] to identify directly which device is active and estimate the channel parameters [2].

The preamble is transmitted using first few symbols of the slot. As seen in Fig. 2.6a, where for

[2] Such algorithms are the subject of the thesis [17] of Lélio Chetot, former member of Inria MARACAS team.



(a) Frame structure in GFRA.

(b) Equivalent frame structure that is considered in the thesis.

Figure 2.6: Example of frame structures used in GFRA and the equivalent model we consider.

simplicity we show only a single subcarrier, the slots are thus split into two subslots, one for the pilot (preamble) and one for the payload. In the remaining of this chapter, Chapter 3 and Chapter 4, we consider a conceptually equivalent model for GFRA model where devices first transmit their preamble in a slot reserved for pilots and then transmit their payload in a subsequent slot of their choice. To preserve the same device identification properties, the preamble of the devices in the second frame structure should be longer, as depicted in Fig. 2.6b.

In comparison with RACH-based transmissions, GFRA allows devices to send their payload faster and with less overhead, making this scheme particularly useful for MTC (to save transmit power) and uRLLC (to transmit faster). The number of preamble sequences being fixed, if the number of active devices in a frame is high, then it is highly probable that devices identification errors will occur. As we will discuss in Chapter 3, we can distinguish different types of errors:

- thinking a payload packet belongs to the wrong device,
- receiving a payload packet without knowing the owner,
- detecting a device but not receiving its payload.

These errors will affect the estimated probability of activity of each device by the station and can propagate in the optimization of the resource allocation.

Furthermore, in GFRA, collisions of payload data packet are likely to occur due to the BS not being able to perform resource scheduling. The choice of the transmit resource(s) $\delta_{n,k}$ made by the device is thus crucial. In the following, we review different state of the art coordination-free methods to select the transmit resources.

2.2.5 Selection of Resources in GFRA

Frame slotted ALOHA

The most popular and most basic, scheme is slotted ALOHA (SA) [90], with its framed version frame slotted ALOHA (FSA) [109]. Inspired by the ALOHA protocol, where devices send their payload at any time, this last variant works by selecting a single transmit resource uniformly at random among the K available resources independently of the other devices. The elements of $\boldsymbol{\delta}_n$ of (2.4) are thus:

$$\delta_{n,k} = \begin{cases} 1 & k = r_n \\ 0 & k \neq r_n \end{cases}, \; r_n \sim \mathcal{U}(K), \; \forall n, \tag{2.5}$$

$$\tag{2.6}$$

where $\mathcal{U}$ is the discrete uniform law over K elements. Let $\mathbf{r} = \begin{bmatrix} r_1 \dots r_N \end{bmatrix}$ the vector of the selected transmit slot of the devices. The GFRA protocol using FSA is described in Alg. 1 If FSA is used

Algorithm 1: GFRA using FSA.

1 (Downlink) Sync signal sent by the BS to indicate beginning of the first frame.
2 **while** *True* **do**
3 (Uplink) Each active device n sends their preamble $\mathbf{b}_n$ on a common control channel.
4 (Local at devices) Each active device n selects their transmit slot $r_n \sim \mathcal{U}(K)$ and construct vector $\boldsymbol{\delta}_n$ according to 2.5.
5 (Uplink) Each active device n sends their data on the drawn slot r_n.
6 (Local at BS) BS decodes data present in each slot k.
7 (Local at devices) Wait for beginning of next frame.
8 **end**

then the probability that at least two devices collide is:

$$\begin{aligned} \Pr(\text{at least one collision}) &= 1 - \Pr(\text{no collision}) \\ &= 1 - \frac{K!\binom{K}{N}}{K^N}, \end{aligned} \tag{2.7}$$

where $K!\binom{K}{N}$ is the (ordered) number of way of choosing N different slots among K slots. Note that $\Pr(\text{at least one collision}) = 1$ if $N > K$.

Coming back to the example of Fig. 2.4, a collision occurs with probability $1 - \frac{3!\binom{3}{3}}{3^3} = 1 - \frac{6}{27} = 0.77$.

Such a scheme is not very efficient, even if $N \ll K$ as the probability that at least two devices choose the same slot is rarely negligible.

Coded random access

In the case where the number of available slots is much higher than the number of devices in the network, then sending several repetitions of the same packet can drastically reduce the probability of having collisions. The vector $\boldsymbol{\delta}_n$ contains several 1 and thus can be viewed as a codeword [79]. Viewing the uncoordinated MAC problem as a coding problem is known as coded random access or unsourced MAC [5, 69] and is expected to be a key enabler to massive multiple access for 6G. This is the idea behind collision resolution diversity slotted ALOHA (CRDSA)[14], irregular repetition slotted ALOHA (IRSA)[68], and coded slotted ALOHA (CSA)[78] protocols. In IRSA, the packet of device n is repeated l_n times, where l_n is known as the device degree of device n. In CSA, it is not repetitions of the same packet that are transmitted but different encoded versions. CRDSA is a special case of IRSA with $l_n = 2,\ \forall n$. In these three protocol, each copy contains, in addition to the data, the index of the slots that contain the other copies of the packet. For CRDSA and IRSA, the decoding follows the principle of a successive interference cancellation (SIC) decoder: if one of the packets is decoded then, the receiver can remove the corresponding copy from the other slots, possibly allowing for another packet to be decoded. In CSA, a SIC decoder is also used in conjunction with the decoder associated to the encoder of the packets. The decoding process stops whenever all packets have been decoded or if there are cycles in the decoding graph.

Focusing now on IRSA, the number of repetition l_n sent by device n is drawn randomly from a common probability distribution $l_n \sim \boldsymbol{\Lambda}$, $\Pr(l_n = d) = \Lambda_d$, $\sum_d \Lambda_d = 1$.

The selection of the set of transmit slots $\mathcal{I}$ is performed uniformly at random without replacement. Defining $\mathcal{K}_l = \{\mathcal{J} \subseteq \{1, \dots, K\} : |\mathcal{J}| = l, \mathcal{J}_i \neq \mathcal{J}_j,\ \forall i \neq j\}$ the set of all subset of size l, $\delta_{n,k}$ is then:

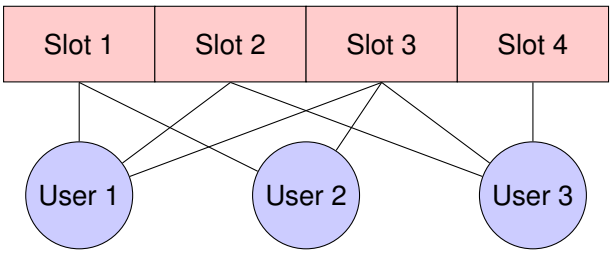
$$\delta_{n,k} = \begin{cases} 1 & i \in \mathcal{I}_n \\ 0 & i \notin \mathcal{I}_n \end{cases}, \ \mathcal{I}_n \sim \mathcal{U}\left(\mathcal{K}_{l_n}\right), \ l_n \sim \boldsymbol{\Lambda}. \tag{2.8}$$

FSA is a special case of IRSA, with $l_n = 1\ \forall n$, and $\mathcal{I}_n \sim \mathcal{U}(\{1 \dots K\})\ \forall n$.

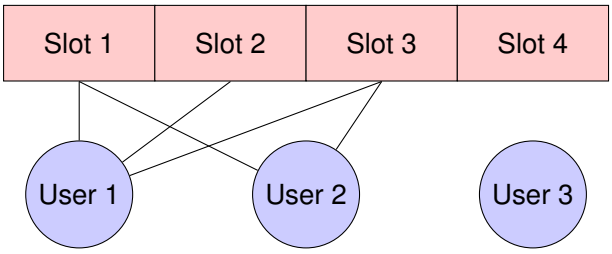
The GFRA protocol using IRSA is described in Alg. 2, and an example of the SIC decoding process can be found in Fig. 2.7.

Algorithm 2: GFRA using IRSA

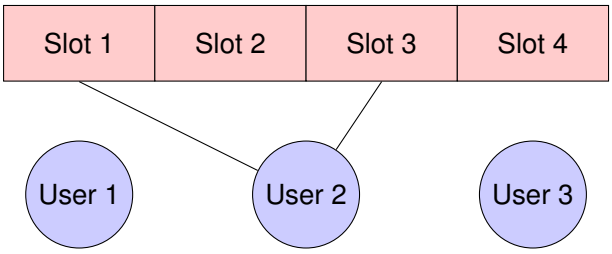
(Downlink) Sync signal sent by the BS to indicate beginning of the first frame.
while *True* **do**
    (Uplink) Each active device n sends their preamble $\mathbf{b}_n$ on a common control channel.
    (Local at devices) Each active device n draws its number of replicas $l_n \sim \Lambda$.
    (Local at devices) Each active device n construct its vector $\boldsymbol{\delta}_n$ using 2.8 (by drawing uniformly at random and without replacement l_n transmit slots).
    (Uplink) Each active device n send their data according to $\boldsymbol{\delta}_n$.
    (Local at BS) BS tries to decode data present in each slot k, using SIC if needed.
    (Local at devices) Wait for beginning of next frame.
end



(a) Step 1: the BS can decode the packet of user 3 in slot 4.



(b) Step 2: user 3 packet being decoded, the BS can now subtract it from slot 1, 2 and 3 decode user 1 in slot 2.



(c) Step 3: user 2 can now be decoded.

Figure 2.7: Example transmission in an IRSA frame with 3 users and 4 slots with device degree distribution $\mathbf{l} = \begin{bmatrix} 3 & 2 & 3 \end{bmatrix}$. In this example all collisions can be recovered.

The main difficulty in IRSA is choosing the right device degree probability distribution $\mathbf{\Lambda}$ for the devices. As described in IRSA's founding paper [68], a good probability distribution of the degree is $\Lambda = \begin{bmatrix} 0 & 0 & 0.5 & 0.28 & 0 & 0 & 0 & 0 & 0.22 \end{bmatrix}$, found by density evolution in the asymptotic case where $K \to \infty$ and $N \to \infty$ and the ratio $\frac{N}{K}$ is kept constant. However, using this distribution means that with a non-negligible probability, the devices will send 8 replicas of their packet. As a result, one of the main drawback of IRSA is the energy consumption induced by the transmission of several replicas per packet. The method is thus particularly suited for uRLLC communication (where latency is important), however for MTC communications, where transmit power is important, it is preferable to favor distributions with a lower average number of replicas.

Decentralized/Distributed Access

Driven by progress on decentralized and federated learning, there is a significant research effort to develop decentralized access protocol. An example is Learn2MAC [24], in which the devices select a code $\boldsymbol{\delta}_n$ defining on which slots to transmit according to a probability distribution. Then, after each frame, the BS informs the devices of the status of each slot (collision, idle, decoded). Based on this information and the knowledge of their respective transmit slots, the devices can run a step of an exponentiated gradient algorithm to update their probability distribution. Nonetheless, Learn2MAC has some drawbacks: the search space is big as the total number of possible codes is 2^K; and devices should have sufficient computation capabilities to perform the gradient computation and update.

Another example is in [89], in which, the authors propose to formalize the random access problem with strict delivery deadlines as a decentralized partially observable Markov decision process (Dec-POMDP). The states of the agents (devices) are whether they have a packet to send, and they take actions (on which slot to send their packet) according to a local policy. They then received a reward if the transmission was successful and finally update their local policy accordingly, with the help of a Deep-Q Network. The objective of the agents is to find the local policy that maximizes the expected reward starting from a given state and taking the actions according to the policy.

Because of the required computation capabilities of the devices to perform the learning mechanisms, decentralized random access protocols are particularly well suited for uRLLC applications, as they allow a faster access than performing a full RACH procedure. They are not particularly relevant for eMBB as usually a latency of several milliseconds is tolerable and not realistic for MTC devices which can not be expected to apply computationally expensive learning algorithms as they require to have the lowest power consumption possible.

2.2.6 Multi-User AWGN with Sporadic Activity

The multiple access methods presented above assume that all devices are active in all frames. This assumption is often unrealistic for IoT networks and MTC as the amount of packet that devices in these network need to send is often small. This leads to the concept of sporadic communications, and can be integrated easily in the model of 2.4:

$$\mathbf{y}_k^t = \sum_{n=1}^{N} \delta_{n,k}^t x_n^t \mathbf{d}_n^t + \mathbf{w}_k^t, \tag{2.9}$$

where $x_n^t \in \{0,1\}$ represent whether device n is active or not in frame t. The above model can be written more concisely by defining the set of active device in slot k of frame t: $\mathcal{A}_k^t = \{n : \delta_{n,k}^t x_n^t = 1\}$.

$$\mathbf{y}_k^t = \sum_{n \in \mathcal{A}_k^t} \mathbf{d}_n^t + \mathbf{w}_k^t. \tag{2.10}$$

The two previous multiple access method FSA and IRSA work similarly with sporadic activity [42] by replacing the number of devices N by the average number of active devices, under the assumption that all devices have the same a priori probability of being active and are independent of each other. In other words, a ubiquitous assumption in the definition and design of multiple access methods, is that the activity of the devices in the network is independent and identically distributed (i.i.d.), and stationary. Defining the activity vector $\mathbf{x}^t = [x_1^t, \dots x_N^t] \in \{0,1\}^N$, $\mathbf{X}$ its associated random vector, and $N_a^t = \sum_{n=1}^{N} x_n^t$ the number of active devices in frame t, this means:

$$\begin{aligned}
\Pr(\mathbf{X}^t = \mathbf{x}^t) &= \prod_{n=1}^{N} \Pr(X_n^t = x_n^t) && \text{(independence)} \\
&= \Pr(X_1^t = 1)^{N_a^t} \Pr(X_1^t = 0)^{N - N_a^t} && \text{(identically distributed)} \\
&= \Pr(X_1 = 1)^{N_a^t} \Pr(X_1 = 0)^{N - N_a^t} && \text{(stationary).}
\end{aligned}$$

However, in practice, these three assumptions may not hold. Hence, we can consider, $\mathbf{X}^t \sim p_{\mathbf{X}^t}$, where $p_{\mathbf{X}^t}$ is a generic probability distribution that might contain correlated and heterogeneous random variable. Implication of having heterogeneous activity pattern in the resource allocation is largely unexplored in the state of the art, except for few papers (reviewed below) [57, 85] that consider only pairwise correlation, or higher order correlation [117, 118]. Chapters 3 and 4 pursue the exploration of the line of work.

FSA with Soft Scheduling

If the i.i.d. assumptions are not met, it is interesting to individualize the slot access probabilities and to diverge from the uniform selection. The key idea is that correlation can be viewed as a form of coordination that can be exploited in the design of the random access strategy. In the first article proposing soft scheduling, [57], the main application was to allocate different slots to highly correlated devices, ensuring they never collide. In FSA with soft scheduling, the single transmit slot of device n is not selected uniformly at random but according to a probability distribution $\mathbf{A}_n$. Whenever the device is active ($x_n = 1$), $\delta_{n,k}$ becomes,

$$\delta_{n,k} = \begin{cases} 1 & k = r_k \\ 0 & k \neq r_k \end{cases}, \; r_k \sim \mathbf{A}_n, \; \forall n. \tag{2.11}$$

Like for FSA (2.5), let $\mathbf{r} = \begin{bmatrix} r_1 & \dots r_N \end{bmatrix} \sim \mathbf{A}$ the vector of the selected transmit slot of the devices. The distributions of all devices form a matrix, called the allocation matrix, $\mathbf{A} \in \mathbb{R}^{N \times K}$, whose each of its elements represent the probability that a device will use a given slot to transmit its data: $A_{n,k} = \Pr(\delta_{n,k} = 1 | X_n = 1)$.

Example 2.2.2 (Allocation matrix)**.** Similar to Example 2.2.1, consider a network with $N = 3$ devices and $K = 3$ slots. The allocation matrix that has been used is the ALOHA allocation matrix:

$$\mathbf{A}_{\text{ALOHA}} = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix}.$$

Assuming that all devices are active $\mathbf{X}^t = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}, \forall t$, as seen in (2.7) the matrix $\mathbf{A}_{\text{ALOHA}}$ leads to a fully decodable frame with probability $\frac{6}{27}$. However, in this example, as the number of slots is equal to the number devices using the orthogonal allocation matrix:

$$\mathbf{A}_{\perp} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

or any matrix that is identical up to a permutation of the columns leads to a decodable frame with probability 1.

The transmission protocol using FSA with soft scheduling is found in Alg. 3

Usage of FSA with soft scheduling raises two questions:

1. How should the allocation matrix $\mathbf{A}$ should be chosen?
2. How is information about the devices' activity distribution $p_{\mathbf{X}^t}$ can be obtained?

Algorithm 3: GFRA using FSA with soft scheduling.

(Downlink) Sync signal sent by the BS to indicate beginning of the first frame and inform devices about the allocation matrix $\mathbf{A}$.
while *True* **do**
    (Uplink) Each active device n sends its preamble $\mathbf{b}_n$ on a common control channel.
    (Local at devices) Each active device n randomly selects its transmit slot r_n according to the probability distribution $\mathbf{A}_n$.
    (Uplink) Each active device n sends its data on the slot r_n.
    (Local at BS) BS decodes data present in each slot k.
    (Local at devices) Wait for beginning of next frame.
end

Design of the Allocation Matrix

In these type of multiple access protocols, we are interested in the expected performance of the algorithm, not the performance of an individual frame. The allocation matrix $\mathbf{A}$ can be optimized offline using common optimization methods, based on some knowledge of the activity statistics. In the first paper to investigate soft scheduling, [57], the marginals of the distribution $\Pr(\mathbf{X} = \mathbf{x})$ as well as its pairwise correlations are assumed to be known. The inclusion/exclusion principle is then used to compute upper and lower bounds on the expected throughput a given allocation matrix $\mathbf{A}$ provides. Heuristics are then developed to find an allocation matrix that maximizes the throughput. In [85], the throughput maximization problem, considering only pairwise correlation between devices, is transformed into a quadratic program, and it is shown that the matrices maximizing the throughput are all binary.

In [117, 118], the optimization of the allocation matrix $\mathbf{A}$ is performed in an online manner: in each frame the activity vector $\mathbf{X}^t$ is assumed to be known, allowing to compute a stochastic gradient of the throughput. As a result, algorithms similar to SGD (described in Section A.2) can be used to provably converge to a stationary point of the throughput. This method thus does not require a priori information about the distribution $p_{\mathbf{X}^t}$ but needs error-free observations of the realizations $\mathbf{X}^t \sim p_{\mathbf{X}^t}$ to guarantee convergence to an allocation matrix having good expected performance.

Nevertheless, device identification occurs over wireless channels, with a variable reliability, we might wonder what would be the implication on the network of either not detecting active devices or miss-identifying them.

Q1: What is the impact of having device identification errors when optimizing the allocation matrix $\mathbf{A}$?

Q2: What can be done to mitigate the eventual performance loss?

2.2.7 Channels with Multiplicative Noise

Multiple user single input single output (MU-SISO)

The multi-user AWGN is powerful to abstract the PHY layer in order to focus on the design on MAC layer, however it is interesting to consider a more advanced PHY layer as it can also provide possibilities to recover the collisions that might appear. Starting from 2.9, we can now explicitly introduce the channel multiplicative noise:

$$\mathbf{y}_k = \sum_{n=1}^{N} \delta_{n,k} x_n h_n \sqrt{P}_n \mathbf{d}_n + \mathbf{w}_k \tag{2.12}$$

$$\mathbf{y}_k = \sum_{\mathcal{A}_k} h_n \sqrt{P}_n \mathbf{d}_n + \mathbf{w}_k, \tag{2.13}$$

where $P_n \in \mathbb{R}$ is the transmit power of device n, and $h_n \in \mathbb{C}$ is known as the channel coefficient, which represent the effect of the propagation (distance, reflections, obstacles) between the transmitted signal and the received signal. The coefficient h_n is random and consists of three components:

- Path loss, accounting for the reduction of the signal power as it propagates through space. It depends on the distance between the transmitter and the receiver, and is expressed in dB:

  $$PL(\rho_n) = -10\eta \log_{10}(\rho_n),$$

  with ρ_n is the distance between device n and the BS, and $\eta > 1$ is the pathloss exponent that depends on the environment and the technology used. Different values for η and more elaborated pathloss models can be found in [76]. Some example are: $\eta = 2$ for transmissions in free space, $\eta \leq 2$ for transmission in "waveguided" environments like corridors, $\eta = 2.6$ in an office and $\eta = 3$ in a house.

- Shadowing, representing possible physical obstacles between the transmitter and the receiver and brings an additional attenuation to the signal. The shadowing is strongly dependent on the environment considered, and allows to model spatially correlated channel coefficient. A commonly used model is the log-normal shadowing [32, 41]. The shadowing term for device n, in dB, is $Sh_n \sim \mathcal{N}(0, \sigma^2_{Sh})$.

- Fading, modeling the reflection of the signal on different surfaces. Different fading models exist, the most famous and most used being the Rayleigh channel when $f_n \sim \mathcal{CN}(0, \sigma^2_f I)$. The Rayleigh fading stands for the effect of multiple reflections of the signal on different surfaces without a dominating line of sight, as such the amplitude $|f_n|$ follows a Rayleigh distribution $f_R(x) = \frac{x}{\sigma^2_f} e^{-\frac{x}{2\sigma^2_f}}$. Another popular propagation model is the Ricean fading [88], used when a dominating line of sight between the device and the BS is present.

The overall coefficient between the antenna of the device and the antenna of the BS taking into account these three components is:

$$h_n = f_n \sqrt{10^{\frac{PL(\rho_n)+Sh_n}{10}}} \tag{2.14}$$

A channel model like (2.14) will be used in Chapter 4.

Multiplexing in the Power Domain: Capture Effect and SIC

Suppose a single active device is transmitting data with power P over a channel like (2.13). The receive power of this packet is $P_r = |h|^2 P$ and its associated signal to noise ratio (SNR) is :

$$\text{SNR} = \frac{P_r}{\sigma^2} \tag{2.15}$$

$$= \frac{|h|^2 P}{\sigma^2}, \tag{2.16}$$

where σ^2 is the variance of the Gaussian noise $\mathbf{w}$. Assuming, the PHY layer is working optimally we can assume that the transmission of the packet will be successful if $\text{SNR} > \gamma,\ \gamma > 0$.

Considering now several active devices, the signal to interference plus noise ratio (SINR) of device n can be expressed as:

$$\text{SINR}_n = \frac{|h_n|^2 P_n}{\sum_{m \in \mathcal{A}_k \setminus \{n\}} |h_m|^2 P_m + \sigma^2}. \tag{2.17}$$

Under what conditions the data of device n can still be decoded? If the SINR of device n is still higher than the threshold γ then we can assume that the packet will be correctly decoded. This phenomenon is known as capture effect and is well studied in MAC protocols like FSA [90, 109] or more recently IRSA [22].

If a packet was decoded then, instead of considering it as noise for the other devices, we can seek to remove it from the received signal allowing to increase the SINR of the other devices. Repeating this process iteratively until either all packets are decoded or no packets are transmitted with enough power gives the SIC algorithm at the basis of PD-NOMA. Then a question to consider is which device should be decoded first? A commonly used heuristic is, for the BS to order the devices in terms of receive power and to try to decode first the device with the highest receive power, then if decoding is successful, proceed to the decoding of the device with the second highest receive power, *etc.* Other heuristics can also be considered and finding a good ordering is the topic of many paper [25, 26, 27]

Example 2.2.3 (Power levels in AWGN)**.** Suppose a multi-user AWGN channel with $\sigma^2 = 1$ is used and the base station implements a SIC receiver. The SINR threshold above which the transmission is

assumed to be decodable is $\gamma = 1$. Devices transmit at different power. The channel is thus:

$$\mathbf{y} = \sum_{i=1}^{N} \sqrt{P_i}\mathbf{d}_i + \mathbf{n}. \tag{2.18}$$

The construction of the transmit power vector ensuring that all devices have a sufficient SINR to be decoded is as follows:

1. The first device should be able to transmit only in the presence of noise: $\mathrm{SINR}_1 = \frac{P_1}{1} > 1 \Rightarrow P_1 = 1 + \epsilon_1,\ \epsilon_1 > 0$,
2. The second device should be able to transmit in the presence of the first device and the noise: $\mathrm{SINR}_2 = \frac{P_2}{P_1+1} > 1 \Rightarrow P_2 = 2 + \epsilon_2,\ \epsilon_2 > \epsilon_1$,
3. The third device should be able to transmit in the presence of the two first devices and the noise: $\mathrm{SINR}_3 = \frac{P_3}{P_1+P_2+1} > 1 \Rightarrow P_3 = 4 + \epsilon_3,\ \epsilon_3 > \epsilon_2$,
4. ...
5. The last device should transmit at power $P_N = 2^{N-1} + \epsilon_N,\ \epsilon_N > \epsilon_{N-1}$.

In practice, the number of power levels is strictly limited by the maximal transmit power P_{MAX}, defined in the standard of the technology used or constrained by some regulations. Furthermore, the definition of power levels becomes unclear whenever more than one antenna is used at the BS, as it depends on how the different observations on each antenna are combined [111].

Example 2.2.3 showcases that a set of devices can interfere but still be fully decodable by using a SIC decoder and properly selected transmit powers. In the case of FSA with soft scheduling, the set devices choosing the same slot is indirectly controllable through the allocation matrix $\mathbf{A}$. As such, it would be preferable to ensure that devices that are likely to transmit on the same slot, transmit at a power that allows their decoding.

Q3: In models similar to (2.12), can the selection of the resource $\boldsymbol{\delta}$ be optimized jointly with the transmit power of the devices ?

Multiple user multiple input multiple output (MU-MIMO)

To further increase the reliability of the channel, it is common to assume that the BS and the devices possess several antennas. The main advantage is to reduce the effect of bad channel condition or of high noise variance. The antennas of the transmitter can be used in different ways:

- sending several symbols in parallel,
- sending a function of the symbols sent on some antennas on the other antennas (repetition, linear combination) to give the receiver extra information on the symbols sent,

The number of independent symbols sent in parallel is referred as the rank, the number of layers, or the number of streams. We will interchangeably use these three terminologies.

The channel coefficient described in (2.14) becomes a matrix $\mathbf{H}_n \in \mathbb{C}^{N_r \times N_t}$ with elements representing the channel conditions between receive antenna i and transmit antenna j :

$$h_{n,i,j} = f_{n,i,j}\sqrt{10^{\frac{PL(\rho_n)+Sh_n}{10}}}, \tag{2.19}$$

where the pathloss and shadowing term are constant across the different antennas because they only depend on the position of the device n, only the random fading factor depends on the (i, j) pair.

In the uplink channel, we can define N_t the number of transmit antennas of the devices, which we assume to be the same for all devices, and N_r the number of the receive antennas of the BS. We can define the MU-MIMO system:

$$\mathbf{Y}_k = \sum_{n=1}^{N} \delta_{n,k} x_n \mathbf{H}_n^{\mathsf{H}} \mathbf{Q}_n \sqrt{\operatorname{diag}(\mathbf{P}_{n,k})}\mathbf{D}_n + \mathbf{W}_k \tag{2.20}$$

$$= \sum_{n \in \mathcal{A}_k} \mathbf{H}_n^{\mathsf{H}} \mathbf{Q}_n \sqrt{\operatorname{diag}(\mathbf{P}_{n,k})}\mathbf{D}_n + \mathbf{W}_k, . \tag{2.21}$$

As there are several transmit antennas, more data symbols can be sent in parallel, hence we have $\nu_n \leq N_t$ the number of parallel data streams used by device n, $\mathbf{D}_n \in \mathbb{C}^{\nu_n \times L}$, $\mathbf{Y}_k^t \in \mathbb{C}^{N_r \times L}$, $\mathbf{W}_n \in \mathbb{C}^{N_r \times L}$, $\mathbf{Q}_n \in \mathbb{C}^{N_t \times \nu_n}$, the precoder spreading the data symbols on the transmit antennas, and $\mathbf{P}_n \in \mathbb{R}^{\nu_n}$ the vector of transmit power on each data stream of device n. The system can be further expressed as:

$$\mathbf{Y}_k = \tilde{\mathbf{H}}_k^{\mathsf{H}} \mathbf{D}_k + \mathbf{W}_k. \tag{2.22}$$

$\tilde{\mathbf{H}}^{\mathsf{H}}$ is the effective channel obtained by stacking the matrices $\{\mathbf{H}_n^{\mathsf{H}}\mathbf{Q}_n \operatorname{diag}(\mathbf{P}_{n,k}),\ \forall n \in \mathcal{A}_k\}$ column wise and $\mathbf{D}_k$ is obtained by stacking the matrices $\{\mathbf{D}_n,\ \forall n \in \mathcal{A}_k\}$ row wise.

Example 2.2.4 (Increase in diversity)**.** Suppose that $N_t = 4$, it is possible to send the symbols on the two antennas simultaneously to increase diversity in the observations of the BS, increasing the probability of correctly decoding the symbols. In this case $\nu = 1$ and $\mathbf{Q} = \frac{1}{2}\begin{bmatrix}1 & 1 & 1 & 1\end{bmatrix}^T$.

Example 2.2.5 (Precoder in 5G)**.** If the quality of the channels are good, then sending 4 repetitions might be considered as a waste of power, it is then possible to double the number of sent symbols by creating two parallel data stream. For example a possible precoder used in 5G, to send two parallel streams over four antennas is the following [37, Table 5.3.3A.2-3]:

$$\mathbf{Q} = \frac{1}{2}\begin{bmatrix}1 & 0\\ 1 & 0\\ 0 & 1\\ 0 & -j\end{bmatrix}.$$

The symbols of the first stream are sent using antennas 1 and 2 and of the second stream using antennas 3 and 4.

The precoders of Examples 2.2.4 and 2.2.5 work by being agnostic to the channel (they do not depend on $\mathbf{H}$). It is also worth noting that some other precoder spread the symbols over the antennas and over several channel uses, giving rise to space-time bloc codes (STBCs), like Alamouti coding [6]. Such codes are not considered in this thesis.

The usage of a MU-MIMO channel as described in Eq(2.20) can bring additional opportunities to enable the decoding of devices sharing the same resources. In particular if the channels $\mathbf{H}_n$ are known, either at the transmitter (channel state information at transmitter (CSIT)), the receiver channel state information at receiver (CSIR), or both (channel state information at transmitter and receiver (CSITR)), they can be exploited to provide spatial multiplexing.

Spatial Multiplexing

If the channel is known at the transmitter (CSIT), specific precoders depending on the channel can be applied to (2.20) to limit as much as possible the interference between the devices. However, computing these precoders can be computationally intensive as they often rely on matrix inversion and thus might not be suited for MTC and IoT networks. The BS, having less computational constraints is better suited perform such tasks. Assuming a CSIR scenario, the devices send their data being agnostic to the channel and the BS implements a decoder to recover the transmitted packets in presence of interference. We review here three main linear decoders: maximum ratio combining (MRC), zero forcing (ZF), minimum mean squared error (MMSE).

- MRC is a decoder that seeks to maximize the receiver power of the signal of a given device, by reapplying the channel of the devices we seek to decode to the received signal $\mathbf{Y}_k$ It is expressed as:

$$M_{\mathrm{MRC},n} = \tilde{\mathbf{H}}_n. \tag{2.23}$$

  The resulting best approximation (in terms of received power) $\tilde{\mathbf{D}}_n$ of $\mathbf{D}_n$ sent over resource k is thus:

$$\begin{aligned}
\tilde{\mathbf{D}}_n &= M_{\mathrm{MRC},n}\mathbf{Y}_k && (2.24)\\
&= \tilde{\mathbf{H}}_n \left(\sum_{n\in\mathcal{A}_k} \tilde{\mathbf{H}}_n^{\mathsf{H}}\mathbf{D}_n + \mathbf{W}_k \right) && (2.25)\\
&= \tilde{\mathbf{H}}_n\tilde{\mathbf{H}}_n^{\mathsf{H}}\mathbf{D}_n + \left(\sum_{m\in\mathcal{A}_k\setminus n} \tilde{\mathbf{H}}_n\tilde{\mathbf{H}}_m^{\mathsf{H}}\mathbf{D}_n + \tilde{\mathbf{H}}_n\mathbf{W}_k \right). && (2.26)
\end{aligned}$$

  Conceptually, a MRC receiver is trying to focalize the signal is seeks to decode. As a result, the received power of the signal from the device is maximized, however, the interference from the other transmitting devices is not taken into account. If the product $\mathbf{H}_n\mathbf{H}_m^{\mathsf{H}}$ is close to the null matrix $\mathbf{0}$ then device m will not interference with device n, but no guarantees are provided.

- ZF is another decoder that seeks to reduce as much as possible the interference from the other devices. Starting from the effective channel model (2.22), we can express the ZF matrix as the Moore Penrose pseudo inverse of the channel $\tilde{\mathbf{H}}_n^{\mathsf{H}}$:

$$\mathbf{M}_{\mathrm{ZF},n} = \left(\tilde{\mathbf{H}}_n \tilde{\mathbf{H}}_n^{\mathsf{H}}\right)^{-1} \tilde{\mathbf{H}}_n. \tag{2.27}$$

  Which applied to $\mathbf{Y}_k$, allows to cancel the channel of the device:

$$\begin{aligned}
\tilde{\mathbf{D}}_n &= M_{\mathrm{ZF},n} \mathbf{Y}_k && (2.28)\\
&= \left(\tilde{\mathbf{H}}_n \tilde{\mathbf{H}}_n^{\mathsf{H}}\right)^{-1} \tilde{\mathbf{H}}_n \left(\sum_{n \in \mathcal{A}_k} \tilde{\mathbf{H}}_n^{\mathsf{H}} \mathbf{D}_n + \mathbf{W}_k \right) && (2.29)\\
&= \mathbf{D}_n + \left(\sum_{m \in \mathcal{A}_k \setminus n} \left(\tilde{\mathbf{H}}_n \tilde{\mathbf{H}}_n^{\mathsf{H}}\right)^{-1} \tilde{\mathbf{H}}_n \tilde{\mathbf{H}}_m^{\mathsf{H}} \mathbf{D}_n + \left(\tilde{\mathbf{H}}_n \tilde{\mathbf{H}}_n^{\mathsf{H}}\right)^{-1} \tilde{\mathbf{H}}_n \mathbf{W}_k \right). && (2.30)
\end{aligned}$$

  If some channels are almost colinear, the Moore Penrose pseudo-inverse will likely contain diverging coefficient, leading to poor performances.

- MMSE is a decoder similar to ZF but that seeks to minimize the mean square error. It is also more robust than ZF as it does not tend to diverge.

$$\mathbf{M}_{\mathrm{MMSE},n} = \left(\tilde{\mathbf{H}}_n \tilde{\mathbf{H}}_n^{\mathsf{H}} + \sigma^2 I\right)^{-1} \tilde{\mathbf{H}}_n. \tag{2.31}$$

  Adding the term $\sigma^2 I$, before the matrix inversion, allows avoiding having matrices that are almost singular and thus results in more stable results.

In a network whose channel model is like (2.20), three parameters are easily tunable: the resource selection $\boldsymbol{\delta}$ over which the base station might or might not have direct control (depending on whether the network is RACH-based or GFRA), the transmit power $\mathbf{P}$ and the number of streams used by each device ν_n.

Q4: In a similar spirit as **Q3**, can $\boldsymbol{\delta}$, $\mathbf{P}$, and ν_n be jointly optimized?

The uplink channel model is now completely defined. The following chapters will use different variations of the channel model (2.20) to present algorithms that to appropriately select the transmit resources $\boldsymbol{\delta}$, the transmit power $\mathbf{P}$, and/or the precoder $\mathbf{Q}$ under various constraint but also as a function of the activity distribution $p_{\mathbf{X}}$ of the devices.

2.3 Measuring the Performance of an Allocation

2.3.1 Throughput

The key performance metric we will use throughout this thesis is the throughput.

In a framed network (2.4) we can define the throughput of each device as the rate of packets that are received during a period of time (between frame index t_1 and frame index t_2) $(t_1, t_2) \in \mathbb{N}^2,\ t_1 < t_2$:

$$T_n = \frac{\text{Number of packets received from device } n \text{ between } t_1 \text{ and } t_2}{t_2 - t_1}, \tag{2.32}$$

and is expressed in packets per channel use. The throughput can be affected by the different PHY layer parameter like the variance of the additive Gaussian noise, or the presence of channel multiplicative noise between the device and the BS. If we make the assumption that the PHY layer is properly optimized, no packet loss occurs meaning that the throughput of a P2P network with a device sending a packet per frame is always 1.

In a multi-user network (2.4), because of the limited amount of resources, collisions might occur and reduce the throughput of some devices. We define by $\mathbf{T} = \left[T_1, \dots T_N\right]$ the vector of throughputs.

The multiple access strategy (selection of transmit resources $\boldsymbol{\delta}$) used will lead to different individual throughput values and different properties on the vector $\mathbf{T}$. Examples of different commonly desired properties are:

- $T_n = T_m,\ \forall n, m$ (egalitarian rule),
- $\max \sum_{i=n}^{N} T_n$, (max sum throughput),
- $\max \sum_{n=1}^{N} u_n$, with u_n a utility function (utilitarian rule), measuring the utility that a device makes of the resources,
- $\max \min_n T_n$ (maximin rule),
- proportionally fair: an allocation $\mathbf{T}$ is proportionally fair if for any other allocation $\mathbf{T}'$: $\sum_{n=1}^{N} \frac{T'_n - T_n}{T_n} \leq 0$, meaning that any change in allocation $\mathbf{T}$ will induce an average negative change,
- α-fair [74]: an allocation $\mathbf{T}$ is α-fair if for any other allocation $\mathbf{T}'$: $\sum_{n=1}^{N} \frac{T'_n - T_n}{T_n^{\alpha}} \leq 0$, where α is a hyperparameter controlling the trade-off between the throughput and fairness. Interesting values are $\alpha = 0$ leading to max sum throughput, $\alpha = 1$ giving proportional fairness, $\alpha = \infty$ giving the maximin rule.

The definition and assignation (indirectly via the multiple access strategy) of throughput to devices is thus a resource allocation problem. Generally, finding an allocation satisfying desired properties implies solving difficult optimization problems [15, 48, 74], in particular, if the objective or the constraints contain non-convex or non-differentiable functions.

2.3.2 Success Rate, Collision Rate

The throughput only gives insight on the number of packets received by the BS but doesn't give information on whether all packets of a particular device have been received. A device with a low throughput could be either a device with a high communication need and a high packet loss (*e.g.* due to collisions) or a device with a low communication need that doesn't suffer from collisions.

Similarly to the throughput, the success rate of device n between frames t_1 and t_2 is defined as:

$$S_n = \frac{\text{Number of packets received from device } n \text{ between } t_1 \text{ and } t_2}{\text{Number of packets transmitted by device } n \text{ between } t_1 \text{ and } t_2}. \tag{2.33}$$

In a network where only collisions are source of packet loss like in (2.9), the collision rate of device n is:

$$C_n = 1 - S_n. \tag{2.34}$$

Both the success rate and the collision rate can be thought as a utility function of the throughput. If a device has a success rate of 1, then allocating more resources to that particular device, will not increase further its throughput as the current allocation already satisfies its need.

2.3.3 Jain's Index: a Fairness Measure

For a given vector of utility $\mathbf{u} = \begin{bmatrix} u_1 & \dots & u_N \end{bmatrix}$ (throughput, success or collision rate...), one might wonder whether the allocation that is made is fair. Situation where the utility function is high for some devices and close to 0 for others might be perceived as unfair while situations where all devices have an equal utility can be seen as maximally fair. The Jain's index [51] can help get insight on whether an allocation is fair:

$$J(\mathbf{U}) = \frac{\left(\sum_{n=1}^{N} U_n\right)^2}{N \sum_{n=1}^{N} U_n^2}. \tag{2.35}$$

The index ranges between $\frac{1}{N}$ and 1, 1 being attained when the allocation induces an equal utility for each device and $J(\mathbf{U}) = \frac{n}{N}$ when n devices have an equal utility and $N - n$ devices have a utility of 0.

2.3.4 Case Study: Throughput and Fairness Trade-off

In the case of FSA with soft scheduling (described in Alg. 3), the allocation matrix $\mathbf{A}$ is key to decide which device transmit in which slot and to control how likely some devices will collide. Thus, different allocation matrices will lead to different throughput vectors. Furthermore, the sporadicity of the network model (2.9) also affects the utility that a device makes out of a resource.

Considering that the distribution of the activity vector $\mathbf{X} \sim p_{\mathbf{X}}$ is stationary, the throughput (2.32) of any device n can be defined over a single frame ($t_2 = t_1 + 1$) regardless of the frame index

t_1. As a result, the average throughput per frame of device n, is the probability that no other active device will collide with device n whenever it seeks to transmit. It can be expressed as:

$$\begin{aligned} T(\mathbf{A}, p_{\mathbf{X}})_n &= \mathbb{E}_{\mathbf{X}\sim p_{\mathbf{X}}}\left[\mathbb{E}_{\mathbf{r}\sim\mathbf{A}}\left[\sum_{k=1}^{K} X_n \mathbb{1}\{r_i = k\} \prod_{m\neq n} (X_m \mathbb{1}\{r_m \neq k\})\right]\right] \\ &= \mathbb{E}_{\mathbf{X}\sim p_{\mathbf{X}}}\left[\mathbb{E}_{\mathbf{r}\sim\mathbf{A}}\left[\sum_{k=1}^{K} X_n \mathbb{1}\{r_i = k\} \prod_{m\neq n} (1 - X_m \mathbb{1}\{r_m = k\})\right]\right] \\ &= \mathbb{E}_{\mathbf{X}\sim p_{\mathbf{X}}}\left[\sum_{k=1}^{K} X_n A_{n,k} \prod_{m\neq n} (1 - X_m A_{m,k})\right]. \end{aligned} \tag{2.36}$$

Moreover, the collision rate of device n can be expressed as a function of its throughput:

$$C(\mathbf{A}, \mathbf{p})_n = 1 - \frac{T(\mathbf{A}, \mathbf{P})_n}{\mathbb{E}_{X_n\sim p_{X_n}}[X_n]}, \tag{2.37}$$

where p_{X_n} is the marginal of $p_{\mathbf{X}}$ for device n.

Example 2.3.1 (Independent but heterogeneous network)**.** Consider a network of $N = 4$ devices and $K = 3$ slots where devices are independent but not equally likely to be active in each frame, $\mathbf{X} \sim \mathrm{Ber}(\mathbf{p})$ with $\mathbf{p} = \begin{bmatrix} 0.2 & 0.3 & 0.4 & 0.6 \end{bmatrix}$. The throughput of devices in the network can be expressed using (2.36):

$$T(\mathbf{A}, \mathbf{p})_n = \sum_{k=1}^{K} p_n A_{n,k} \prod_{m\neq n} (1 - p_m A_{m,k}).$$

The normalized sum-throughput is then computed as:

$$T^N(\mathbf{A}, \mathbf{p}) = \frac{1}{\mathbb{E}_{\mathbf{X}\sim p_{\mathbf{X}}}[\sum_{n=1}^{N} X_n]} \sum_{n=1}^{N} T(\mathbf{A}, \mathbf{p})_n \tag{2.38}$$

$$= \frac{1}{\sum_{n=i}^{N} p_n} \sum_{n=1}^{N} T(\mathbf{A}, \mathbf{p})_n. \tag{2.39}$$

It represents the average transmission success rate, with $T^N(\mathbf{A}, \mathbf{p}) = 1$ whenever all packet of all devices in the network are successfully decoded. However, in the case of networks with heterogeneous activity, maximizing the sum-throughput often leads to sacrificing the devices least active in order to favor devices having the biggest impact on this metric.

Table 2.1 shows the throughput vector, the normalized sum-throughput and the Jain's index for different matrices. The ALOHA matrix achieve a poor normalized sum-throughput but because all devices are equally likely to use any slot, the fairness index in terms of collision rate is higher, even though device 4 as a collision rate that is 0.1 lower than device 1. $\mathbf{A}_{\mathrm{thr}}$ achieves the highest normalized sum-throughput by assigning a collision free slot to the two most active devices, providing an increase of $\approx 36\%$, but this comes at the price of a low Jain's index $J(\mathbf{T})$ as the individual throughput values are quite disparate and a highly unfair distribution of the collisions because only the two first devices have a

	$\mathbf{A}_{\text{ALOHA}}$	$\mathbf{A}_{\text{thr}}$	$\mathbf{A}_{\text{fair}}$	$\mathbf{A}_{\text{td}}$
$\mathbf{A}$	$\begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & 0.83 & 0.17 \\ 0.93 & 0.07 & 0 \\ 0 & 0.51 & 0.49 \\ 0.44 & 0.01 & 0.55 \end{bmatrix}$	$\begin{bmatrix} 0.26 & 0.65 & 0.09 \\ 0.13 & 0.79 & 0.08 \\ 0.1 & 0.06 & 0.84 \\ 0.95 & 0.03 & 0.02 \end{bmatrix}$
$\mathbf{T}$	[0.12 0.19 0.27 0.44]	[0.14 0.24 0.4 0.6]	[0.14 0.21 0.29 0.45]	[0.13 0.23 0.35 0.52]
$T^N(\mathbf{A}, \mathbf{p})$	0.68	0.92	0.74	0.81
$J(\mathbf{T})$	0.83	0.80	0.85	0.81
$\mathbf{C}$	[0.38 0.35 0.33 0.27]	[0.3 0.2 0 0]	[0.26 0.27 0.27 0.25]	[0.36 0.24 0.13 0.14]
$J(\mathbf{C})$	0.986	0.48	0.999	0.83

Table 2.1: Individual throughput $\mathbf{T}$, normalized sum-throughput $T^N(\cdot)$ and Jain's Index $J(\cdot)$ for different allocation matrices in a network of independent devices with activity probabilities $\mathbf{p} = \begin{bmatrix} 0.2 & 0.3 & 0.4 & 0.6 \end{bmatrix}$.

risk of having collisions. On the contrary, the matrix $\mathbf{A}_{\text{fair}}$ achieves the highest collision fairness, with a collision rate that is almost equal for all devices, and outperforms $\mathbf{A}_{\text{ALOHA}}$ in terms of normalized sum-throughput and collision fairness, indicating $\mathbf{A}_{\text{ALOHA}}$ is not on the Pareto front of the trade-off between $T^N(\cdot)$ and $J(\mathbf{C})$. Note that component-wise, the throughput vector of $\mathbf{A}_{\text{fair}}$ is higher than the one of $\mathbf{A}_{\text{ALOHA}}$ and the collision rate vector of $\mathbf{A}_{\text{fair}}$ is lower than the one of $\mathbf{A}_{\text{ALOHA}}$. Finally, we can try to balance the three metrics and use matrix $\mathbf{A}_{\text{td}}$.

Example 2.3.1 shows the interest of tuning the allocation matrix $\mathbf{A}$ to the statistics of the network. By doing so, a fine control over of the performances and the properties of the network can be operated.

2.4 Conclusion

In this chapter, we introduced the different models that we use in this thesis as well as the relevant literature. In the following chapters, we will with develop different resource allocation methods in the context of MTC and eMBB. Table 2.2 present which models are used in the different Chapters.

We first consider, in Chapter 3, an IoT network with an access policy based on FSA with soft scheduling (Alg. 3). The allocation matrix $\mathbf{A}$ is optimized iteratively, frame after frame, based on the activity vector $\mathbf{X}$ that might contain errors. Chapter 3 answers to **Q1** and **Q2**; it studies the impact of uncertainty in the optimization of the resource allocation and present ways of mitigating the performance loss.

Then, Chapter 4, still using GFRA, but under the assumption of error-free activity vector $\mathbf{X}$, answers to **Q3**; we add resources from the power domain and propose to jointly optimize the slot selection matrix $\mathbf{A}$ and the transmit power of the devices. Furthermore, we exploit the correlations of the underlying process driving the activity of the devices to improve the throughput of the network.

	Model (PHY)	Selection of resources (MAC)
Chapter 3	(2.9)	Alg.3
Chapter 4	(2.20) (with $N_t = 1$)	Alg.3
Chapter 5	(2.20)	RACH (Fig.2.5a)

Table 2.2: PHY/MAC models used in the different chapters of the thesis.

Finally, in Chapter 5, in the context of an eMBB network, we explore the possibility and the interest of jointly optimizing several transmission parameters: the power, the PRBs and the rank of each device of the network, allowing to answer to **Q4**.

CHAPTER

3 Slot Allocation with Imperfect Device Detection

Contents

3.0 Summary

Driven by the need of reliable GFRA protocols, there has been an interest in designing FSA algorithms for networks with heterogeneous activity probabilities, in particular FSA with soft scheduling, as described in Chapter 2.2.6. These works have established that the throughput can be significantly improved over a standard FSA with uniform slot allocation. However, the algorithms for optimizing the probability a device accesses each slot require perfect knowledge of the active devices within each frame. In practice, this assumption is limiting as device identification algorithms in GFRA rarely provide activity estimates with zero error. In this chapter, we seek to answer to question **Q**1 and **Q**2. To do so, we study the impact of having only limited knowledge of the activity statistics and propose a new algorithm to optimize the access policy in the presence of activity estimation errors.

This Chapter is based on the work published in J3, C3 and C4 (resp. [52, 54, 55]).

3.1 Introduction

A challenge in large-scale multiple access is ensuring reliable transmissions while efficiently utilizing resources and limiting latency. One approach to this challenge in cellular systems is GFRA, described in Chapter 2.2.4 [20, 28, 67, 94], where active devices transmit a preamble immediately followed by data transmission. In contrast with the random access channel (RACH) procedure, GFRA does not require a response from a base station before transmitting a data packet. As a consequence, the access delay is reduced.

While the RACH procedure provides devices with reserved resources, this is not the case for GFRA. As such, devices must select their own resources, such as the time slot they will use for data transmission. Contention resolution is therefore a critical problem in GFRA, requiring a careful selection of slots by devices in order to reduce re-transmissions.

Due to the lack of coordination in GFRA and limited information about the statistics of the network, slot selection policies are often based on variants of FSA [90]. In the original form of FSA [109], active devices randomly select a single slot within a frame in a uniform fashion (Alg. 1).

Other variants of FSA have also been proposed where devices may utilize more than one slot in a single frame including IRSA [68] (Alg. 2) and CSA [78]. Nevertheless, slotted ALOHA remains the *de facto* MAC protocol used IoT applications [112, 113], in particular when the average number of active devices is close to the number of available resources.

3.1.1 Related Work

A ubiquitous assumption in the design of random access protocols is that devices are homogeneous; namely, each device is active independently within a frame with a common probability. This assumption applies to the classical FSA protocols [109], as well as recent work focusing on stability [112] and age of information (AoI) minimization [49, 113].

The homogeneity assumption is also the basis of variants of FSA. In the CRDSA scheme [14], two identical copies of a data packet are transmitted in two different slots. Due to the repetitions, the reception of a single packet is sufficient to decode the data. The performance can be further improved by utilizing SIC [106], where successfully decoded packets can be subtracted from other slots allowing for additional packets from other devices to be decoded. The IRSA scheme [68] improves upon CRDSA by allowing for additional repetitions, with the quantity selected from a probability distribution known to the base station. Decoding of data packets is achieved via message passing algorithms analogous to decoding of low-density parity check (LDPC) [38]. In recent work, the impact of SIC has been investigated [97] and tailored to device identification algorithms [100].

CSA [78] generalizes IRSA by splitting a data block into several packets and applying a packet-level linear block code. This yields coded packets instead of repetitions as in CRDSA and IRSA. The code for each device is selected from a predefined set according to a code probability distribution. As for IRSA, decoding at the base station is achieved via message passing algorithms. Recent work has investigated code distribution design for erasure channels [116], the impact of interference cancellation errors [31], the optimization of the power distribution [101], and the impact of time-dependence in packet arrivals for individual devices [98].

In practice, the homogeneity assumption may not hold, either due to heterogeneous activity probabilities or statistical dependence in the activity of multiple devices. For example, devices may be sensors that observe different phenomena and have heterogeneous activity probabilities. Sensors may also observe a common phenomenon, which induces correlation in their activity.

As the homogeneity assumption is ubiquitous, a key question is whether any heterogeneity in device activities can be exploited in order to improve the performance of slotted ALOHA schemes. As FSA is the basis of modern multiple access systems in the context of the IoT [112], it is natural to first relax the homogeneity assumption for this family of protocols.

To this end, in [57], a new variant of FSA has been proposed in order to account for heterogeneity in the activity of the devices. In this scheme, the probability that a given active device accesses each slot is optimized based on the joint probability distribution of the device activities. The main benefit of this approach is that devices with high activity probabilities or correlation can be allocated in different slots. As a consequence, the probability of contention can be significantly reduced, leading to an improved throughput.

Two key difficulties arising in the approach proposed in [57] are: (i) obtaining knowledge of the activity distribution; and (ii) optimizing the slot allocation probabilities for each device. In [57],

these difficulties were addressed by assuming known pairwise correlations between device activities and utilizing a heuristic method to optimize the allocation probabilities based on upper and lower bounds on the expected throughput.

In [118], the two key difficulties (i) and (ii) were addressed via a data-driven approach. In each frame, the activity of each device, rather than the probability distribution, was assumed to be known. The slot allocation probabilities were then updated frame after frame via SGD. In [117, 118], contention was further managed by exploiting SIC at the physical layer. The slot allocation probabilities were then chosen to maximize the expected sum-rate or the expected number of devices with a SINR exceeding a desired threshold.

3.1.2 Main Contributions

The schemes in [57, 117, 118] were shown to significantly improve the performance in comparison with the standard FSA scheme by exploiting heterogeneity in the distribution of the device activities. However, it was assumed that either the probability distribution of device activities or the device activity in each frame is perfectly known.

In practice, preamble detection and packet decoding is rarely error-free in large-scale multiple access systems [59, 119], even when device heterogeneity is accounted for [17, 18]. Similarly, advanced NOMA transmissions based on blind signature classification [77] also have classification errors. Moreover, the impact of imperfect knowledge of device activities in each frame affects system performance is poorly understood, as well as how to mitigate the impact of activity estimation errors on the optimization of slot allocation probabilities.

In this chapter, we propose a throughput maximization algorithm to optimize slot allocation probabilities in FSA accounting for imperfect knowledge of device activities. Our main contributions are as follows:

(i) We demonstrate that the algorithms recently proposed in [117, 118] can be highly sensitive to activity estimation errors, leading to a suboptimal throughput. This is significant as, in the absence of activity estimation errors, these algorithms achieve a high performance. Moreover, we show that the suboptimality arises from bias in gradient estimates due to estimation errors.

(ii) To mitigate the impact of the bias arising from activity estimation errors, we exploit importance weighting in a manner analogous to sample bias correction [114] and bias reduction in private synthetic data [40]. We prove that by weighting the gradient estimates in the stochastic optimization algorithm in [117, 118], the slot allocation probabilities converge to a stationary point with probability one.

(iii) In practice, the weight on the gradient estimates requires the evaluation of the true activity distribution and the imperfect activity distribution induced by erroneous activity estimation. As these distributions may be difficult to obtain, we propose heuristic weights that are readily available in practical systems, leading to new stochastic optimization algorithms for the slot

allocation probabilities.

(iv) The proposed algorithms are validated by an extensive numerical study under different types of errors. Several models of activity estimation errors are considered, including independent and symmetric or asymmetric errors and a realistic error model based on generalized approximate message passing (GAMP) [18] arising in pilot-based device identification. Under each model for activity estimation errors, numerical results show that our algorithm is robust to activity estimation errors. Moreover, our algorithm leads to performance improvements over the algorithms in [117, 118], which do not account for activity estimation errors.

This chapter is organized as follows: in Section 3.2 we detail the system model. In Section 3.3, we recall the stochastic resource optimization algorithm in [117, 118], which does not account for activity estimation errors. In Section 3.4, we study the impact of activity estimation errors and develop a new stochastic resource optimization algorithm to mitigate the errors. In Section 3.5, we present a numerical study comparing our new algorithm with existing methods that do not account for activity estimation errors. In Section 3.6, we conclude.

3.2 System Model

Consider a network consisting of a BS equipped with N_r antennas and N devices equipped with a single antenna. Transmissions by active devices occur in frames over a single subcarrier using a protocol similar to NB-IoT each frame containing K slots.

3.2.1 Device Activity and Resource Selection

Within a given frame, each of the N devices are either active or inactive. The activity of device n in frame t is represented by the binary Bernoulli distributed random variable $X_n^t \sim \mathrm{Ber}(p_n)$, where $p_n \in [0,1]$ is the probability that device n is active. That is, $X_n^t \in \{0,1\}$ with $X_n^t = 1$ if device n is active and $X_n^t = 0$ otherwise. The activity vector in frame t is then denoted by $\mathbf{X}^t \in \{0,1\}^N$ and we denote the vector of device activity probabilities by $\mathbf{p} = [p_1, \ldots, p_N] \in [0,1]^N$. To simplify notations, we write in the remaining of this chapter $\mathbf{X}^t \sim \mathbf{p}$ instead of $\mathbf{X}^t \sim \mathrm{Ber}(\mathbf{p})$. We make the following assumptions for $\mathbf{X}^t,\ t = 1, 2, \ldots$:

(i) X_n^t is independent of $X_m^t,\ m \neq n$.

(ii) $\mathbf{X}^t$ is independent of $\mathbf{X}^{t'},\ t' \neq t,\ t \in \mathbb{N}$.

(iii) $\mathbf{p}$ is *not* perfectly known to the access point. In the following sections the imperfect knowledge of $\mathbf{p}$ is denoted by $\tilde{\mathbf{p}}$.

In GFRA, each active device selects the time-frequency resources utilized for data transmission without coordinating with the access point nor any other devices. In the absence of any coordination, a common policy for resource selection is the classical FSA protocol [109]. As described in Chapter 2.2.6, a more general policy allows devices to select resources with different probabilities. In this policy,

the resource allocation matrix $\mathbf{A} \in \mathbb{R}_{+}^{N \times K}$ describes the probability that a given device chooses a particular slot.

$$A_{nk} = \Pr(\text{device } n \text{ selects resource } k | X_n = 1) \tag{3.1}$$

3.2.2 GFRA Protocol and Device Identification

A generic resource selection and data transmission protocol using a pilot-based identification is given in Alg. 4. Specific protocols vary in how device identification information is communicated to the access point. In the sequel, device identification is critical for optimization of FSA with heterogeneous device activity distributions. We consider two types of protocols where device identification information is included within the preamble or in the data packet.

Algorithm 4: Grant-Free Random Access Protocol.

(Downlink) Sync signal sent by the BS to indicate beginning of the first frame and inform devices about the allocation matrix $\mathbf{A}$.
while *True* **do**
    (Uplink) Each active device n sends their preamble $\mathbf{b}_n$ on a common control channel.
    (Local at devices) Each active device n randomly selects the slot r_n with probability A_{nr_n}.
    (Uplink) Each active device n sends their data on the selected slot r_n.
    (Local at BS) BS decodes data present in each slot k.
    (Local at devices) Wait for beginning of next frame.
end

Identification at the PHY Layer: Device Identification in the Preamble

In GFRA, each time a device seeks to transmit, it randomly selects a preamble from a set of preambles and transmits it for identification purposes. In this chapter, we will neglect the probability that two devices use the same preamble. We can thus conceptually assume that each device n is assigned with a unique preamble of length L, $\mathbf{b}_n \in \mathbb{C}^L$ that remains the same regardless of the frame index. In each frame, the preamble of each active device is then transmitted over a block fading control channel consisting of L symbols, represented by the Pilot slot of Fig. 2.6b. The output of the control channel in frame t, $\mathbf{Y}^t \in \mathbb{C}^{L \times N_r}$, is given by

$$\mathbf{Y}^t = \mathbf{B}\mathbf{H}^t + \mathbf{W}^t, \tag{3.2}$$

where $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_N]$ is the matrix of preambles, $\mathbf{H}^t \in \mathbb{C}^{N \times N_r}$ is the matrix of channel coefficients for all devices and BS antennas, and $\mathbf{W}^t \in \mathbb{C}^{L \times N_r}$ is the additive white Gaussian noise.

Given the channel output $\mathbf{Y}_{\text{Pilot}}^t$ of the pilot slots, an estimate of the channel coefficients $\hat{\mathbf{H}}^t$ and of the activity vector $\hat{\mathbf{X}}^t$ can be obtained. This is typically achieved via approximate message passing algorithms such as the GAMP algorithm in [18]. Active devices then transmit their data by randomly selecting a data slot based on the allocation matrix $\mathbf{A}$. Data decoding is carried out by exploiting the estimate of the channels $\hat{\mathbf{H}}^t$ obtained from the GAMP algorithm.

Identification at the MAC Layer: Device Identification in the Data

An alternative approach is to utilize the preambles only for the purpose of channel estimation. In this case, the protocol proceeds in the same fashion as in Sec. 8; however, the output of the pilot slot $\mathbf{Y}^t_{\text{Pilot}}$ is only used to estimate the channel coefficients, $\hat{\mathbf{H}}^t$, of the active devices. Active devices then transmit their data as well as a unique identifier in a slot randomly selected based on the allocation matrix $\mathbf{A}$. The key difference is that data decoding reveals both the data of the active devices and their identity. As a consequence, the activity estimate $\hat{\mathbf{X}}^t$ is only obtained after data decoding.

3.2.3 Feedback of the Allocation Matrix $\mathbf{A}$

The GFRA protocol in Alg. 4 does not require a multistep handshake procedure as in the RACH procedure. However, it is necessary to inform devices of the allocation matrix $\mathbf{A}$ in order to select the resource to be used in each frame. We assume that the activity statistics of the devices typically do not change significantly over a large number of frames. As a consequence, it is only necessary to communicate the matrix $\mathbf{A}$ to the devices every F frames, where the choice F accounts for downlink usage constraints.

On the other hand, the BS obtains new information about the activity of each device at each frame. This information can be utilized to update the matrix $\mathbf{A}$ at each frame at the BS, as is discussed in the following section. We emphasize that the BS *does not* need to communicate the updated allocation matrix at each frame.

3.3 Stochastic Resource Allocation Problem

The stochastic resource allocation protocol described in Alg. 4 is common in FSA systems and their variants. For example, in the simplest form of ALOHA, an equal probability is assigned to each time slot for all devices. In IRSA and CSA, the slot or code allocation is also randomized [68, 78]. However, these protocols have been designed under the assumption of an equal activity probability for all devices.

In this section, we formalize the problem of stochastic resource allocation in FSA for devices with *heterogeneous activity probabilities*. We present an algorithm based on [118] to optimize the allocation *under the assumption of perfect knowledge of the activities* $\mathbf{X}^t$ for the throughput objective. A new algorithm to mitigate the impact of imperfect knowledge of $\mathbf{X}^t$ due to preamble detection or data decoding errors is developed in Sec. 3.4.

3.3.1 Objective

A key performance metric of a network is the throughput that can be characterized, in the scenario we consider in this chapter, as the expected number of collision-free transmissions per frame. As seen in Chapter 2.3.4, given an allocation matrix $\mathbf{A}$ and activity vector $\mathbf{X}$, the instantaneous throughput for device n is given by

$$T_n(\mathbf{A};\mathbf{X}) = \sum_{k=1}^{K} X_n A_{nk} \prod_{\substack{m=1 \\ m\neq n}}^{N} (1 - X_m A_{mk}), \tag{3.3}$$

and the total instantaneous throughput by:

$$T^{\mathbf{X}}(\mathbf{A}) = \sum_{n=1}^{N} T_n(\mathbf{A};\mathbf{X}). \tag{3.4}$$

After averaging over the activity vector $\mathbf{X}$, the throughput is then

$$T(\mathbf{A};\mathbf{p}) = \mathbb{E}_{\mathbf{X}\sim\mathbf{p}}[T^{\mathbf{X}}(\mathbf{A})]. \tag{3.5}$$

In the case were the activity of each device is independent, the throughput is simplified as

$$T(\mathbf{A};\mathbf{p}) = \sum_{n=1}^{N}\sum_{k=1}^{K} p_n A_{nk} \prod_{\substack{m=1 \\ m\neq i}}^{N} (1 - p_m A_{mk}). \tag{3.6}$$

In order to compare the performance of the algorithms in scenarios with different activity probability, we define the normalized throughput as

$$T^N(\mathbf{A};\mathbf{p}) = \frac{1}{\sum_{n=1}^{N} p_n} T(\mathbf{A};\mathbf{p}).$$

3.3.2 Stochastic Optimization Problem

An optimal allocation $\mathbf{A}^*$ is a solution to the stochastic optimization problem

$$\mathbf{A}^* \in \arg \min_{\substack{\mathbf{A}\in\mathbb{R}_+^{N\times K}: \\ \sum_{k=1}^{K} A_{nk}=1,\ n=1,\ldots,N}} -T(\mathbf{A};\mathbf{p}). \tag{3.7}$$

In general, the objective $T(\mathbf{A};\mathbf{p})$ is non-convex in $\mathbf{A}$.

If the distribution $\mathbf{p}$ is known to the BS, it is, in principle, possible to obtain solutions to (3.7) via gradient descent (GD). However, this approach has a very high complexity due to the fact that

the throughput has 2^N terms. Indeed,

$$T(\mathbf{A};\mathbf{p}) = \sum_{\mathbf{x}\in\{0,1\}^N} \Pr(\mathbf{X}=\mathbf{x}) T^{\mathbf{X}}(\mathbf{A}). \tag{3.8}$$

In practice, only limited knowledge of $\mathbf{p}$ is available as it must be estimated based on the activity estimates $\hat{\mathbf{X}}^t$ obtained from the preamble or data decoding. As the activity estimates are error-prone, $\hat{\mathbf{X}}^t$ can be considered as being drawn from another distribution $\hat{\mathbf{p}}$, different from $\mathbf{p}$. An alternative approach is to directly utilize the activity estimates $\hat{\mathbf{X}}^t$ to solve the allocation problem in (3.7). As observed in [118], this can be achieved via projected SGD.

Suppose, for the moment, that the activity vectors $\mathbf{X}^1, \mathbf{X}^2, \ldots$ for each frame are perfectly known to the BS. The projected SGD algorithm (summarized in Alg. 5) updates the allocation matrix $\mathbf{A}$ each frame via the recursion

$$\mathbf{A}^{t+1} = \Pi_{\mathcal{H}} \left\{ \mathbf{A}^t + \gamma^{t+1} g(\mathbf{A}^t; \mathbf{X}^{t+1}) \right\}, \tag{3.9}$$

where (γ^t) is a positive step size sequence, $\Pi_{\mathcal{H}}$ denotes the Euclidean projection on the constraint set

$$\mathcal{H} = \{\mathbf{A} \in \mathbb{R}_+^{N\times K} : \sum_{k=1}^{K} A_{ik} = 1,\ i = 1, \ldots, N\}, \tag{3.10}$$

and $g(\mathbf{A}^t; \mathbf{X}^{t+1})$ is a stochastic gradient estimate based on the activity vector $\mathbf{X}^{t+1}$. In particular, the gradient estimate for the throughput objective is given by (the superscript t is dropped for the sake of clarity).

$$g(\mathbf{A};\mathbf{X}) = \sum_{n=1}^{N} g_n(\mathbf{A};\mathbf{X}), \tag{3.11}$$

$$\begin{aligned} g(\mathbf{A};\mathbf{X})_{ql} &= \sum_{n=1}^{N} g_n(\mathbf{A};\mathbf{X})_{ql} \\ &= X_q \prod_{\substack{m=1 \\ m\neq q}}^{N} (1 - X_m A_{ml}) - \sum_{\substack{n=1 \\ n\neq q}}^{N} X_q X_n A_{nl} \prod_{\substack{m=1 \\ m\neq n \\ m\neq q}}^{N} (1 - X_m A_{ml}) \end{aligned} \tag{3.12}$$

where

$$g_n(\mathbf{A};\mathbf{X})_{ql} = \begin{cases} X_q \prod_{\substack{m=1 \\ m\neq q}}^{N} (1 - X_m A_{ml}) & \text{if } q = n \\ -X_q X_n A_{nl} \prod_{\substack{m=1 \\ m\neq n \\ m\neq q}}^{N} (1 - X_m A_{ml}) & \text{if } q \neq n \end{cases}$$

Note that in this case, $g(\mathbf{A};\mathbf{X}) \in \mathbb{R}^{N\times K}$ is an unbiased estimate of $\nabla_{\mathbf{A}} T(\mathbf{A};\mathbf{p}) = \mathbb{E}_{\mathbf{X}\sim\mathbf{p}}[g(\mathbf{A};\mathbf{X})]$ due to the absence of device identification errors. As such, under appropriate conditions on the step

size sequence $\{\gamma^t\}$ and perfect knowledge of $\mathbf{X}^1, \mathbf{X}^2, \ldots$, the iterates $\{\mathbf{A}^t\}$ converge almost surely to a stationary point of $T(\mathbf{A}; \mathbf{p})$ [118].

Algorithm 5: Stochastic optimization algorithm when device identification is error-free.

Choose initial allocation matrix $\mathbf{A}^1 \in \mathbb{R}^{N \times K}$ such that $\sum_{k=1}^{K} A^1_{n,k} = 1,\ n = 1, \ldots, N$, and step-size sequence $\{\gamma^t\}$ with $\gamma^t > 0,\ t = 1, 2, \ldots$
$t \leftarrow 1$
while *not converged* **do**
  Based on $\mathbf{X}^{t+1} \sim \mathbf{p}$, compute an unbiased estimate $g(\mathbf{A}^t; \mathbf{X}^{t+1})$ of $\nabla_{\mathbf{A}^t} T(\mathbf{A}; \mathbf{p})$.
  $\mathbf{A}^{t+1} \leftarrow \Pi_{\mathcal{H}}[\mathbf{A}^t + \gamma^{t+1} g(\mathbf{A}^t; \mathbf{X}^{t+1})]$
  $t \leftarrow t + 1$
end

In practice, however, perfect knowledge of $\mathbf{X}^1, \mathbf{X}^2, \ldots$ is not available to the BS due to errors in device identification. As we discuss in the following section, this can lead to a significant degradation in performance.

3.4 Mitigating Errors in Activity Estimation

3.4.1 Impact of Activity Estimation Error

As noted in the previous section, perfect knowledge of device activities $\mathbf{X}^1, \mathbf{X}^2, \ldots$ is not available to the BS. A key question is then the impact on the performance of Alg. 5. In fact, errors in the activity estimates can lead to a significant performance degradation, even in small networks as illustrated in the following example.

Example 3.4.1. Consider a network consisting of three devices sharing two time slots. The true activity probabilities of the devices are $\mathbf{p} = \begin{bmatrix} 0.3 & 0.4 & 0.9 \end{bmatrix}$. Suppose that the device identification algorithm is not always able to distinguish between the first and last devices. In particular, with probability ϵ, the estimated device activity vector available to the BS is in fact governed by $\hat{\mathbf{p}} = \begin{bmatrix} 0.9 & 0.4 & 0.3 \end{bmatrix}$. The observed distribution is thus $\mathbf{p}' = (1 - \epsilon)\mathbf{p} + \epsilon\hat{\mathbf{p}}$.

Fig. 3.1 shows the impact of ϵ on the network throughput $T(\mathbf{A})$ after optimizing $\mathbf{A}$ with Alg. 5 for 500 frames. The throughput of standard FSA is also plotted, where every element of the allocation matrix $\mathbf{A}$ is $\frac{1}{2}$. The case $\epsilon = 0$ corresponds to the case of perfect device identification and $\epsilon = 1$ to the case where the activity vector is always drawn from $\hat{\mathbf{p}}$.

Observe that as ϵ increases (corresponding to a higher probability of errors in the estimation of $\mathbf{X}$), the throughput significantly decreases compared with the baseline where $\mathbf{X}$ has been perfectly estimated. If the average number of errors is too large, then the resulting throughput is worse than ALOHA.

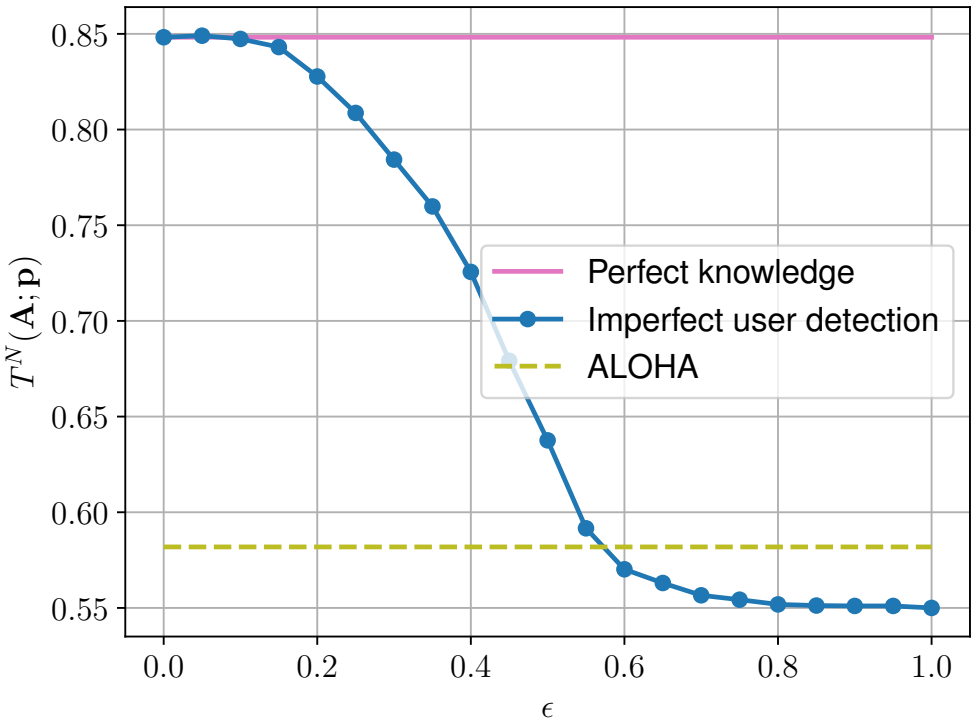


Figure 3.1: Expected throughput with device identification errors for a network of three devices sharing two slots. Baseline, in pink, corresponds to $\epsilon = 0$. Drawing samples $\mathbf{X}_t$ with probability ϵ from $\hat{\mathbf{p}}$ instead of $\mathbf{p}$ (in blue) can reduce the throughput by up to 35%.

3.4.2 Unbiasing Gradient Estimates

In order to develop an algorithm to mitigate the impact of device identification errors, we first examine the impact of the errors on the SGA algorithm in Alg. 5. To this end, consider the update rule

$$\begin{aligned}\mathbf{A}^{t+1} &= \Pi\left\{\mathbf{A}^t + \gamma^{t+1} g(\mathbf{A}^t; \mathbf{X}^{t+1})\right\} \\ &= \mathbf{A}^t + \gamma^{t+1} g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1}) + \gamma^{t+1}\mathbf{Z}^{t+1},\end{aligned} \tag{3.13}$$

where $\gamma^{t+1}\mathbf{Z}^{t+1}$ is the smallest vector (w.r.t the Euclidean norm) needed to project $\mathbf{A}^t + \gamma^{t+1} g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1})$ into the constraint set $\mathcal{H}$. We then have

$$\begin{aligned}\mathbf{A}^{t+1} = \mathbf{A}^t + \gamma^{t+1}\left(\mathbb{E}_{\mathbf{X}\sim\mathbf{p}}[g(\mathbf{A}^t; \mathbf{X}^{t+1})] + \beta^{t+1} + \partial\mathbf{M}^{t+1}\right) \\ + \gamma^{t+1}\mathbf{Z}^{t+1},\end{aligned} \tag{3.14}$$

where

$$\begin{aligned}\beta^{t+1} &= \mathbb{E}_{\hat{\mathbf{X}}\sim\hat{\mathbf{p}}}[g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1})] - \mathbb{E}_{\mathbf{X}\sim\mathbf{p}}[g(\mathbf{A}^t; \mathbf{X}^{t+1})] \\ \partial\mathbf{M}^{t+1} &= g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1}) - \beta^{t+1} - \mathbb{E}_{\mathbf{X}\sim\mathbf{p}}[g(\mathbf{A}^t; \mathbf{X}^{t+1})].\end{aligned} \tag{3.15}$$

Note that $\mathbb{E}_{t+1}[\partial\tilde{\mathbf{M}}^{t+1}] = 0$, where $\mathbb{E}_{t+1}[\cdot]$ is the conditional expectation with respect to the σ-algebra generated by the iterates up to time t.

In the case that there are no device identification errors, $\beta^{t+1} = 0$, $g(\mathbf{A}^t; \mathbf{X}^{t+1})$ is an unbiased estimate of the gradient. Moreover, Alg. 5 is guaranteed to converge to a stationary point under the hypotheses in Theorem 5.2.10 of [65]. However, when $\beta^{t+1} \neq 0$, the gradient estimate is biased. As a consequence, there is no guarantee of convergence, even to a stationary point. The resulting performance reduction is illustrated in Fig. 3.1.

3.4.3 Proposed Algorithm

Because the performance reduction is due to biased gradient estimates, it is necessary to utilize a bias reduction method. Bias reduction algorithms have been exploited in various machine learning problems; e.g., sample selection bias correction [114] and privacy [40]. We now adapt these methods to our problem.

Consider the weighting function $w : \mathbf{x} \mapsto w(\mathbf{x})$ for $\mathbf{x} \in \{0,1\}^N$ consisting of the ratio between a target distribution and a proposal distribution, it is defined by

$$w(\mathbf{x}) = \frac{\Pr(\mathbf{X} = \mathbf{x})}{\Pr(\hat{\mathbf{X}} = \mathbf{x})}, \tag{3.16}$$

where $\mathbf{X}$ is the true activity vector and $\hat{\mathbf{X}}$ is the estimated activity vector. Suppose that $w(\mathbf{x}) < \infty,\ \mathbf{x} \in \{0,1\}^N$ where we adopt the convention that $\frac{0}{0} = 0$. For the purposes of bias reduction, a key property of $w(\mathbf{x})$ is then that for all $\mathbf{A} \in \mathcal{H}$,

$$\begin{aligned}
\mathbb{E}_{\mathbf{X}\sim\mathbf{p}}[g(\mathbf{A}; \mathbf{X})] &= \sum_{\mathbf{x}\in\{0,1\}^N} g(\mathbf{A}; \mathbf{x}) \Pr(\mathbf{X} = \mathbf{x}) \\
&= \sum_{\mathbf{x}\in\{0,1\}^N} g(\mathbf{A}; \mathbf{x}) w(\mathbf{x}) \Pr(\hat{\mathbf{X}} = \mathbf{x}) \\
&= \mathbb{E}_{\hat{\mathbf{X}}\sim\hat{\mathbf{p}}}[w(\hat{\mathbf{X}}) g(\mathbf{A}; \hat{\mathbf{X}})].
\end{aligned} \tag{3.17}$$

In other words, $w(\hat{\mathbf{X}})g(\mathbf{A}; \hat{\mathbf{X}})$ is an unbiased estimate of $\mathbb{E}_{\mathbf{X}\sim\mathbf{p}}[g(\mathbf{A}; \mathbf{X})]$ as long as $w(\mathbf{x}) < \infty$ for all $\mathbf{x}$ such that $\Pr(\mathbf{X} = \mathbf{x}) > 0$. As such, as we rigorously establish in Theorem 1, this choice of weight overcomes the key problem preventing convergence of Alg. 5.

Algorithm 6: Stochastic optimization algorithm with device identification errors.

Choose initial allocation matrix $\mathbf{A}^1 \in \mathbb{R}^{N\times K}$ such that $\sum_{k=1}^{K} A^1_{n,k} = 1,\ n = 1, \ldots, N$, and step-size sequence $\{\gamma^t\}$ with $\gamma^t > 0,\ t = 1, 2, \ldots$
$t \leftarrow 1$
while *not converged* **do**
  Based on the estimate $\hat{\mathbf{X}}^{t+1} \sim \hat{\mathbf{p}}$, compute a biased estimate $g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1})$ of $\nabla_{\mathbf{A}^t} T(\mathbf{A}; \mathbf{p})$.
  $\mathbf{A}^{t+1} = \Pi_{\mathcal{H}}[\mathbf{A}^t + \gamma^{t+1} w(\hat{\mathbf{X}}^{t+1}) g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1})]$
  $t \leftarrow t + 1$
end

Incorporating the weight $w(\mathbf{x})$ in (3.16) into Alg. 5 leads to Alg. 6. This algorithm has the following convergence guarantee.

Theorem 1. The iterates $\mathbf{A}^t$ of Alg. 6 converge almost surely as $t \to \infty$ to a stationary point provided that the step size sequence $\{\gamma^t\}$ satisfies

(i) $\gamma^t > 0,\ t > 0$;

(ii) $\sum_{t=1}^{\infty} \gamma^t = \infty$;

(iii) $\sum_{t=1}^{\infty} (\gamma^t)^2 < \infty$,

and $w(\mathbf{x}) < \infty$ for all $\mathbf{x} \in \{0,1\}^N$.

Proof. Observe that the update rule can be written as

$$\begin{aligned}
\mathbf{A}^{t+1} &= \mathbf{A}^t + \gamma^{t+1} \Big(\mathbb{E}_{\hat{\mathbf{X}}^{t+1} \sim \hat{\mathbf{p}}}[w(\hat{\mathbf{X}}^{t+1}) g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1})] \\
&\quad w(\hat{\mathbf{X}}^{t+1}) g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1}) - \mathbb{E}_{\hat{\mathbf{X}}^{t+1} \sim \hat{\mathbf{p}}}[w(\hat{\mathbf{X}}^{t+1}) g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1})] \Big) \\
&\quad + \gamma^{t+1} \mathbf{Z}^{t+1} \\
&= \mathbf{A}^t + \gamma^{t+1} \Big(\mathbb{E}_{\mathbf{X}^{t+1} \sim \mathbf{p}}[g(\mathbf{A}^t; \mathbf{X}^{t+1})] + \partial \tilde{\mathbf{M}}^{t+1} \Big) \\
&\quad + \gamma^{t+1} \mathbf{Z}^{t+1} \\
&= \mathbf{A}^{t+1} + \gamma^{t+1} \Big(\nabla_{\mathbf{A}^t} T(\mathbf{A}^t) + \partial \tilde{\mathbf{M}}^{t+1} \Big) + \gamma^{t+1} \mathbf{Z}^{t+1},
\end{aligned} \tag{3.18}$$

where

$$\begin{aligned}
&\partial \tilde{\mathbf{M}}^{t+1} \\
&= w(\hat{\mathbf{X}}^{t+1}) g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1}) - \mathbb{E}_{\hat{\mathbf{X}}^{t+1} \sim \hat{\mathbf{p}}}[w(\hat{\mathbf{X}}^{t+1}) g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1})]
\end{aligned} \tag{3.19}$$

satisfies $\mathbb{E}_{t+1}[\partial \tilde{\mathbf{M}}^{t+1}] = 0$, where $\mathbb{E}_{t+1}[\cdot]$ is the conditional expectation with respect to the σ-algebra generated by the iterates up to time t. The last equality then follows from the identity in (3.17).

To establish convergence, we apply Theorem 5.2.1 in [65]. To do so, we verify the following conditions:

- A.5.2.1:

$$\sup_t \mathbb{E}_{\hat{\mathbf{X}}^{t+1} \sim \hat{\mathbf{p}}}[w(\hat{\mathbf{X}}^{t+1}) g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1})] < \infty, \tag{3.20}$$

  which holds under the assumption that $w(\mathbf{x}) < \infty$ for all $\mathbf{x} \in \{0,1\}^N$.

- A.5.2.2: There exists $\overline{g} : \mathbb{R}^{N \times M} \to \mathbb{R}$ such that

$$\mathbb{E}_{\hat{\mathbf{X}}^{t+1} \sim \hat{\mathbf{p}}}[w(\hat{\mathbf{X}}^{t+1}) g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1})] = \overline{g}(\mathbf{A}^t), \tag{3.21}$$

  which holds by setting $\overline{g}(\mathbf{A}^t) = \nabla_{\mathbf{A}^t} T(\mathbf{A}^t; \mathbf{p})$.

- A.5.2.3: $\nabla_{\mathbf{A}} T(\mathbf{A}; \mathbf{p})$ is continuous, which immediately follows from the definition of $T(\mathbf{A}; \mathbf{p})$.
- A.5.2.5: The gradient estimate $w(\hat{\mathbf{X}}^{t+1}) g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1})$ is unbiased, which follows from (3.17).

A further condition on the constraint must also hold. As the same constraint is present in [118] for the case without errors, the same argument can be applied to complete the proof. □

3.4.4 Computing the Importance Weight

Theorem 1 shows that even when there are errors in $\hat{\mathbf{X}}$, Alg. 6 converges to a stationary point for the problems in (3.7). This is in contrast to the standard SGA algorithm in Alg. 5, for which these guarantees hold only when there are no errors.

However, the implementation of Alg. 6 requires knowledge of both $\mathbf{p}$ and $\hat{\mathbf{p}}$ in order to compute the weight $w(\mathbf{x})$ in (3.16). In practice, the BS does not have perfect knowledge of $\mathbf{p}$ nor $\hat{\mathbf{p}}$. Nevertheless, estimates can be obtained as follows:

(i) Estimation of the target distribution $\mathbf{p}$: The target distribution corresponds to the prior distribution required by the GAMP algorithm. Knowledge of the prior is obtained via expectation-maximization algorithms (see [36] for details on GAMP and estimation of the prior). Another possibility is that the devices can periodically feedback estimates of activity probabilities. We model the resulting estimated distribution, accounting for errors, by adding a Gaussian perturbation to the true distribution, $\tilde{\mathbf{p}} = \mathbf{p} + \boldsymbol{\eta}$, with $\boldsymbol{\eta} \sim \mathcal{N}(0, \sigma^2 I)$, where $\tilde{\mathbf{p}}$ is clipped, if needed, to be between 0 and 1. We denote $\tilde{\mathbf{X}} \sim \text{Ber}(\tilde{\mathbf{p}})$ and $\tilde{w}(\mathbf{x}) = \frac{\Pr(\tilde{\mathbf{X}}=\mathbf{x})}{\Pr(\hat{\mathbf{X}}=\mathbf{x})}$ the estimated importance weight.

(ii) Estimation of the proposal distribution $\hat{\mathbf{p}}$: the estimation of $\hat{\mathbf{p}}$ can be achieved by empirical distribution estimation via the outputs of the device identification algorithm (e.g., GAMP).

As a consequence, the weight $w(\mathbf{x})$ in (3.16) is not perfectly known in practical systems. Nevertheless, as we show in the following section, an imperfect estimate of the weight $w(\mathbf{x})$ still yields improved performance over Alg. 5, which ignored the impact of device identification errors.
Moreover, if $\Pr(\hat{\mathbf{X}} = \mathbf{x})$ is close to 0, $w(\mathbf{x})$ will be very large. This mean that the optimization algorithm can take very large steps, pushing the optimization towards undesired local maxima. To increase the stability of the algorithm, we propose Alg. 7, in which we clip the weight of iteration t to a maximal value of $\kappa^t > 0$.

3.5 Numerical Results

3.5.1 Parameters and Baseline Methods

To support our theoretical analysis, we provide simulation results for different types of errors introduced by the device detection algorithms. We compare the throughput given by several methods:

Algorithm 7: Stochastic optimization algorithm with device identification errors, approximate target distribution and weight clipping.

Choose initial allocation matrix $\mathbf{A}^1 \in \mathbb{R}^{N\times K}$ such that $\sum_{k=1}^{K} A^1_{n,k} = 1,\ n = 1,\dots,N$, clipping parameter sequence $\kappa^t > 0$, and step-size sequence $\{\gamma^t\}$ with $\gamma^t > 0,\ t = 1, 2, \dots$
$t \leftarrow 1$
while *not converged* **do**
  Based on the estimate $\hat{\mathbf{X}}^{t+1} \sim \hat{\mathbf{p}}$, compute a biased estimate $g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1})$ of $\nabla_{\mathbf{A}^t} T(\mathbf{A}; \mathbf{p})$
  $\tilde{w}(\hat{\mathbf{X}}^{t+1}) = \min\left\{\kappa^t, \frac{\Pr(\tilde{\mathbf{X}}=\mathbf{x})}{\Pr(\hat{\mathbf{X}}=\mathbf{x})}\right\}$
  $\mathbf{A}^{t+1} = \Pi_{\mathcal{H}}[\mathbf{A}^t + \gamma^{t+1}\tilde{w}(\hat{\mathbf{X}}^{t+1}) g(\mathbf{A}^t; \hat{\mathbf{X}}^{t+1})]$
  $t \leftarrow t + 1$
end

- Optimization of $\mathbf{A}$ using Alg. 5 with perfect detection of devices, as baseline, where the true activity vector $\mathbf{X}$ is known,
- Optimization of $\mathbf{A}$ using Alg. 5 with imperfect device detection representing the optimization obtained when using the error-prone activity vector $\hat{\mathbf{X}}$,
- Optimization of $\mathbf{A}$ using Alg. 7 with true weight w for Fig. 3.2 and Fig. 3.4 (with true target $\mathbf{p}$ empirically estimated proposal $\hat{\mathbf{p}}$ for Fig. 3.5)
- Optimization of $\mathbf{A}$ using Alg. 7 with an imperfect weight due to a target distribution $\tilde{\mathbf{p}}$ that is known up to a Gaussian perturbation of different standard deviations σ.
- A greedy allocation $\mathbf{A}_h$ obtained via Alg. 8: the devices that are the most likely to transmit are allocated their own slot, while all the others devices share a single slot.

Algorithm 8: Construction of Greedy Allocation Matrix $\mathbf{A}_h$.

Given a sorted vector of activity probabilities $\mathbf{p}$ in the network and K slots
$k \leftarrow 1$
$\mathbf{A} \leftarrow \mathbf{0}_{N\times K}$
for $i \leftarrow 1;\ i \leq N - K;\ i \leftarrow i + 1$ **do**
  $A_{i,1} \leftarrow 1$
end
for $k \leftarrow 2;\ k \leq K;\ k \leftarrow k + 1$ **do**
  $A_{N-k+1,k} \leftarrow 1$
end

- The Frame Slotted ALOHA allocation, a constant matrix having $A_{ik} = \frac{1}{K}$ as elements,
- $\mathbf{A}^0$ represents the initial random matrix that was used.

In the following figures, we plot the normalized throughput, corresponding to the expected throughput per frame of the optimized matrix $\mathbf{A}$ for a given network of N devices with activity probability p_i. It should be noted that if this policy is implemented for a large number of frames, a small difference in the per-frame throughput can lead to significant differences in the number of successfully decoded packets over long time periods.

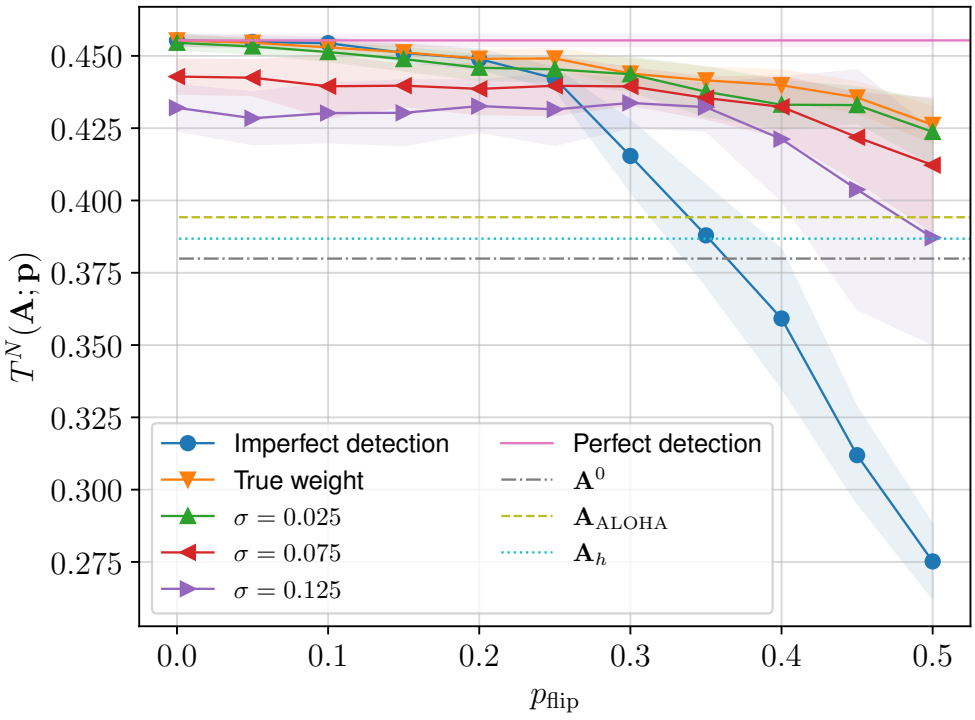


Figure 3.2: Resulting throughput after 10000 frames for different values of p_{flip}.

The simulated network consists in $N = 20$ devices and $K = 5$ slots, 5 being the number of slots of an NB-IoT frame in the 3.75 kHz subcarrier spacing [73]. We run the different methods for 10 000 frames. We should note that in this scenario of NB-IoT, the duration of a slot is 2 ms and a frame 10 ms [73], thus 10 000 frames represent 100 s. Each method is simulated using the same random sequence of activity vectors, and we repeat the simulations 20 times to isolate the effect of the detection errors on the resulting allocation. For the weight $\tilde{w}$ in Alg. 7, a new target distribution $\tilde{\mathbf{p}}$ is drawn for each run according to the description in Section 3.4.4.

The step size and the maximum weight in the different algorithms are set constant across all simulations, frames and type of channels with value $\gamma^t = \frac{1}{100}$ and $\kappa^t = 5$. In our following figures, the shaded area represent $\pm$ the standard deviation.

3.5.2 Symmetric Errors

Consider a detection algorithm where the false alarm and miss-detection probabilities for each device are equal and given to $p_{\text{flip}} \in [0, 0.5]$. In other words, if device i is active, the probability it is not detected is p_{flip}. Similarly, if device i is not active, it is detected with probability p_{flip}. In this case,

$$\hat{p}_i = p_i + p_{\text{flip}} - 2p_{\text{flip}}p_i, \;\; i = 1, \ldots, N. \tag{3.22}$$

Fig. 3.2 shows the impact of the error probability p_{flip} on the throughput achieved by each algorithm. The activity probability of each device i, p_i, is drawn once independently from the uniform distribution Unif$[0, 0.45]$ and kept constant across the different runs.

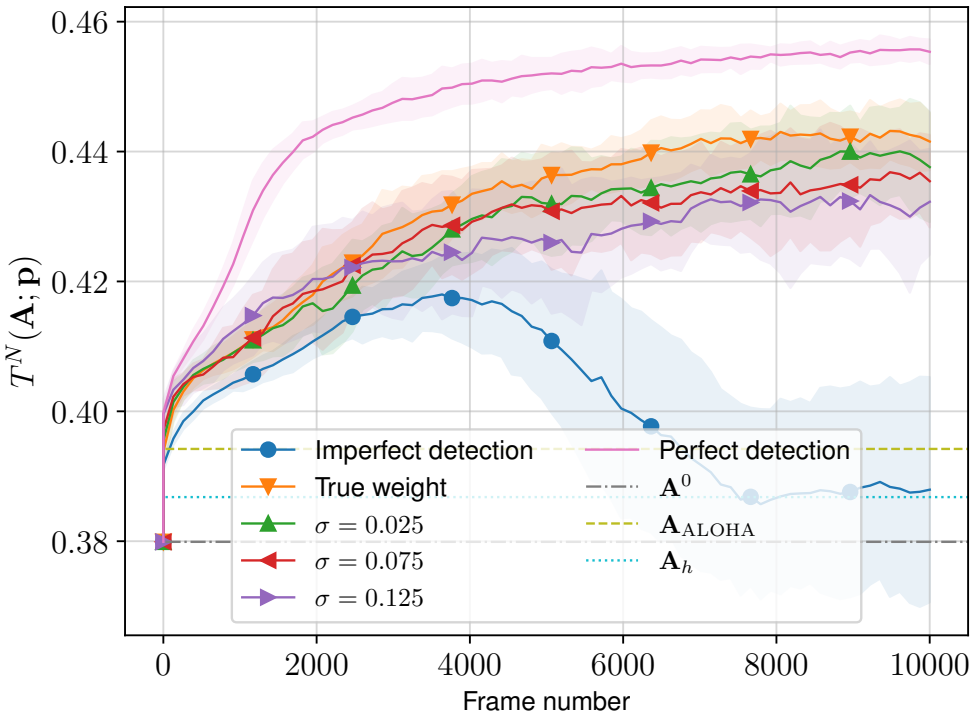


Figure 3.3: Trajectories of the different method presented in Fig. 3.2 with $p_{\text{flip}} = 0.35$.

Observe in Fig. 3.2 that $\mathbf{A}_h$ (in dashed light blue) has the worst performance. This is due to the inefficiency of allocating a single device to a slot when the activity probabilities are low. The random initial allocation $\mathbf{A}^1$ and the ALOHA allocation $\mathbf{A}_{\text{ALOHA}}$ are poorly performing. This is due to the fact that neither account for the throughput objective in (3.3). For small values of p_{flip} (< 0.25) the algorithms have similar performance; however for higher error probability, applying Alg. 5 while ignoring device identification errors (blue curve) leads to significant degradation in the throughput. The proposed Alg. 6 with perfect knowledge of the weight in (3.16) (orange curve) and with imperfect weight (green, red, purple) always outperforms Alg. 5 with identification errors. This highlights the utility of our proposed algorithm as it mitigates the effect of device identification errors. Fig. 3.3 shows the trajectories for each method for $p_{\text{flip}} = 0.35$. We can see that the blue curve (imperfect detection) starts by increasing slightly before falling into a bad local maxima whilst the proposed algorithms converge to a better one. We should note that the curves for methods different than the perfect detection are not monotonically increasing because they are evaluated based on the objective $T^N(\mathbf{A};\mathbf{p})$ for the true $\mathbf{p}$ whilst the activity vector $\hat{\mathbf{X}}$ used in the computation of the gradient is drawn from $\hat{\mathbf{p}}$. The green, red and purple curves correspond to the proposed Alg. 6 with imperfect knowledge of the weight, obtained by perturbing the true weight by Gaussian noise. As the standard deviation of the noise increases, the throughput performance degrades. This is due to the fact that the noise introduces bias in the gradient estimates. Nevertheless, Alg. 6 with weights corrupted by noise still outperforms Alg. 5 (blue curve), which does not account for device identification errors.

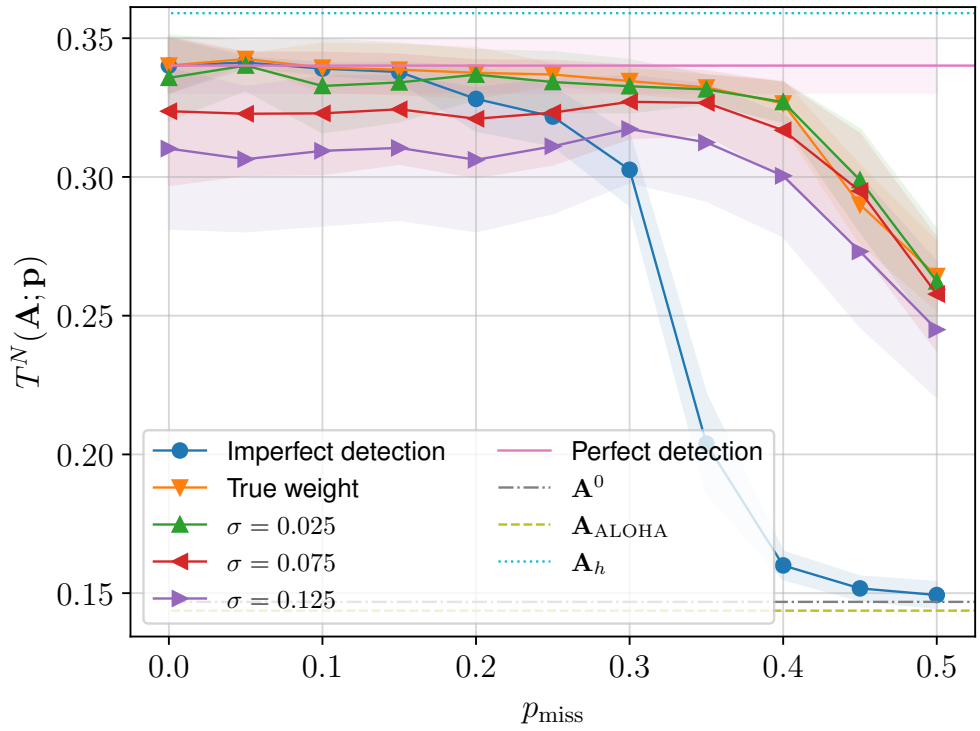


Figure 3.4: Resulting throughput after 10000 frames for different values of p_{miss}.

3.5.3 Asymmetric Errors

We now consider a detection algorithm that has Pr(false alarm) $= 0$ and Pr(miss detection) $= p_{\text{miss}}$ for $p_{\text{miss}} \in [0, 0.5]$ common to each device. In other words, if device i is active, the probability that it is not detected is p_{miss}. In contrast to the scenario of Fig. 3.2, if device i is not active, then the probability of it being detected is zero.

$$\hat{p}_i = (1 - p_{\text{miss}})p_i, \ \ i = 1, \ldots, N. \tag{3.23}$$

Fig. 3.4 shows the impact of the error probability p_{miss} on the throughput achieved by each algorithm. The activity probability of each device i, p_i, is drawn once independently from the uniform distribution $\mathcal{U}[0, 0.9]$ and kept constant across the runs. That is, there is a high probability that many devices will have an activity probability $p_i > \frac{1}{2}$.

Observe in Fig. 3.4, the initial random allocation $\mathbf{A}^1$ and the ALOHA allocation $\mathbf{A}_{\text{ALOHA}}$ have the worst performance. Due to the high heterogeneity of the network, the gap between $\mathbf{A}_{\text{ALOHA}}$ and the best performing method is larger than in the previous scenario. In this case, as some devices have a probability of activity $p_i > \frac{1}{2}$, the greedy heuristic (in dashed blue) performs the best. The pink curve corresponds to Alg. 5 with perfect device activity estimation. For small and large values of p_{miss}, the pink curve is higher than any of the curves corresponding to proposed Alg. 6 and Alg. 5 with device identification errors. In particular, observe that, as for the symmetric errors, applying Alg. 5 while ignoring device identification errors (blue curve) leads to significant degradation in the throughput. Alg. 6 with perfect knowledge of the weight in (3.16) to mitigate device identification

errors (orange curve) always outperforms Alg. 5.

Again, the green, red and purple curves show that the throughput degrades when the standard deviation of the prior increases, but they still outperform Alg. 5 (blue curve), which does not account for device identification errors, when the number of errors is high.

3.5.4 Errors Arising from GAMP-Based Detection

Finally, we consider the use of an active device detection algorithm based on GAMP [86]. As described in Section 8, each device n is given a unique complex preamble (or pilot) $\mathbf{b}_i$ of length $L = 15$ with $b_{n,l} \sim \mathcal{CN}(0, 1)$, for $l = 1 \dots 15$, in each frame a new set of pilots is drawn. As $L < N$, it is impossible to find a set of pilots that are all mutually orthogonal. Pilots are sent with a unit transmit power and are received by a single antenna (*i.e.*, $N_A = 1$ in this scenario). Devices are active according to the following distribution:

$$\mathbf{p} = [0.01\ 0.03\ 0.09\ 0.14\ 0.21\ 0.21\ 0.23\ 0.27\ 0.32\ 0.33\ 0.34\ 0.42\ 0.43\ 0.47\ 0.52\ 0.56\ 0.58\ 0.61\ 0.65\ 0.8].$$

Pilots are multiplied by a complex channel coefficient and corrupted by some Gaussian noise. The modulus of the channel coefficients are

$$|\mathbf{H}| = [1.6\ 0.8\ 0.5\ 0.5\ 1.2\ 1\ 2.4\ 0.3\ 1.0\ 0.1\ 0.5\ 1.2\ 1.7\ 0.2\ 2.5\ 1.6\ 2.1\ 1.4\ 0.5\ 0.2]$$

(the matrix $\mathbf{H}$ is a vector as there is a single receive antenna).

As the posterior distribution $\hat{\mathbf{p}}$ of GAMP is difficult to compute, it is estimated using a Monte Carlo approximation by running the network for 10000 frames, and this estimation is used as the denominator of the importance weight.

Fig. 3.5 shows the effect of the additive Gaussian noise on the throughput of the resulting allocation given by each algorithm. The figure is shown as a function of the transmit SNR, *i.e.* the ratio of the transmit power, 1, over the variance of the noise without taking into account the channel coefficient. Note that the received SNR of the device with the worst channel ranges between $[0, -10.6]$dB. The algorithms are run for 10000 frames.

In Fig. 3.5, similarly to the asymmetric errors, the random initial allocation $\mathbf{A}^0$ and the ALOHA allocation $\mathbf{A}_{\text{ALOHA}}$ have the worst performance. In this case, the greedy heuristic (dashed blue) performs the best as some devices have a probability of activity $p_i > 0.5$. For small and large values of SNR, the orange curve, representing the weight consisting of the true target and the empirical distribution estimation, is above the curves of the algorithm Alg. 6. For high values of SNR, almost no errors are introduced by the GAMP algorithm. In this regard, all algorithms achieve a similar throughput. For smaller values of SNR ($\leq$ 6dB), the Alg. 5 with imperfect detection (blue curve) shows a degraded performance. This is partly due to the fact that the most likely device has a bad channel gain, meaning that the GAMP algorithm often fails to decode it. Therefore, the resulting resource allocation does not manage to take it sufficiently into account, leading to a decrease in

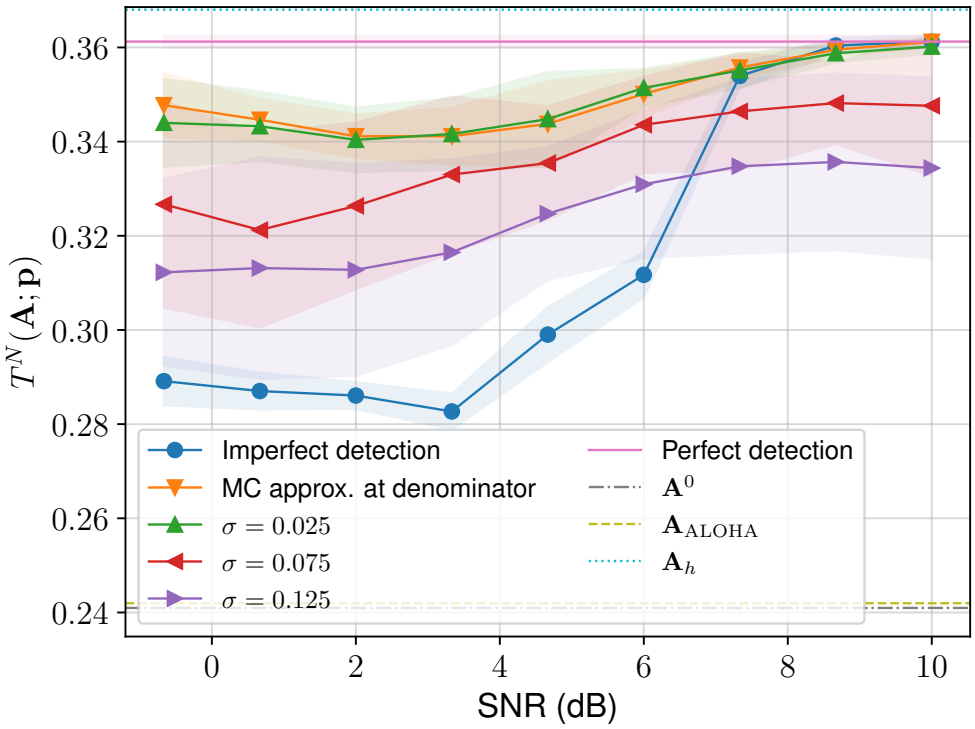


Figure 3.5: Resulting throughput after 10000 frames for different values of SNR.

performance. Note that the proposed Alg. 7 (green, red, and purple) outperforms Alg. 5 (blue curve).

3.6 Conclusions

A problem arising the design of FSA algorithms with heterogeneous devices is the need for perfect activity estimates in each frame. In this chapter, we have studied the impact of imperfect activity estimation and observed a significant throughput performance degradation in existing algorithms. To overcome this problem, we have proposed an importance weighted bias mitigation strategy. This algorithm is shown to be capable of guaranteeing almost sure convergence of stochastic gradient methods to a stationary point of the throughput maximization problem. A numerical study shows that our method outperforms existing algorithms which do not account for imperfect activity estimation. Moreover, our approach is robust to uncertainty in the importance weights.

CHAPTER

4 Joint Slot Allocation and Power Control

Contents

4.0 Summary

The frames of protocols using GFRA can be conceptually seen as split into two parts: device identification; and data transmission, which can be viewed as a form of FSA. A common assumption in FSA is device homogeneity; that is, the probability that a device seeks to transmit data in a particular frame is the same for all devices and is independent of the other devices. Recent work has investigated the possibility of tuning the FSA protocol as a function of the statistics of the network by changing the probability for a particular device to access a particular slot. However, as, highlighted in **Q3**, power control with a SIC receiver has not yet been considered to further increase the performance of a FSA with soft scheduling protocol. In this chapter, we propose an algorithm to jointly optimize both the slot selection and the transmit power of the devices to minimize the probability that devices are outed of the network. The algorithm works sequentially, optimizing frame after frame the resource allocation based on the devices that are active in each frame. As a result, the possible correlations in the devices' activity are observed and can be used to further increase the objective function. We show via a simulation study that our algorithms can outperform the baseline based on FSA.

This chapter is partially based on the work published in C2 ([53]).

4.1 Introduction

The selection of transmit resources with limited coordination between devices and with the base station is a key problem in wireless multiple access networks which led to a family of protocols known as ALOHA. As seen in Chapter 2.2.5, Slotted ALOHA [90] protocols, and most of its variants such as FSA [109], CSA [78] and IRSA [68], assume that the network is homogeneous, meaning that devices are independent and equally likely to transmit data. However, within the context of modern communication networks like IoT or event-driven communication, the probability that a device is active in a specific frame is likely to vary from device to device. Furthermore, it is also probable that the devices will exhibit correlations in their activity patterns, if they monitor the same event for example.

A key question is whether exploiting knowledge of devices' heterogeneity and correlations can improve the resource utilization through the individualization of the resource allocation policy. To this end, new variants of Frame Slotted ALOHA have been proposed, as described in Alg. 3, in Chapter 2.2.6, and in Chapter 3, where the probability that a particular device chooses a particular slot is not uniform but is optimized as a function of the statistics of the network. In [57, 85], stochastic resource selection is optimized based on the joint probability of activity of the devices. Optimization is based on heuristics in [57] and by solving a quadratic program in [85]. However, both approaches require the knowledge, by the base station, of the activity probability and the correlation coefficient of the devices in the network. To relax these two assumptions, the work in [117, 118] proposed to sequentially optimize, via SGD the stochastic resource allocation based only on the devices active in each frame.

Collisions arising from multiple devices utilizing the same resource is a limiting factor when seeking to increase the density of the network. For this reason, it is common to assume that the base station is able to exploit SIC. In SIC, the data of each device is sequentially decoded and subtracted from the received signal in order to increase the SINR of the devices that are yet to be decoded. SIC was used in [117] to take advantage of the different channel conditions to further reduce the contention in the network. However, the power is assumed to be the same for all devices. Adding power control, in addition to the selection of the transmit slot, is desirable to better exploit the SIC decoder. As such, power control can be exploited similarly as PD-NOMA [75].

In this chapter, we propose an algorithm in the context of NB-IoT networks [47] to jointly optimize two different types of resources: the temporal slot (orthogonal) and the transmit power (non-orthogonal). The contributions of this chapter are as follows:

(i) we propose an algorithm to jointly optimize the slot selection and the transmit power of the devices to minimize their outage probability, with a SIC receiver at the base station. In contrast, existing work [117] using SIC assumes devices all transmit at the same power and thus only exploits the channel characteristics of each device.

(ii) We show via a simulation study that our algorithm can exploit the heterogeneity and the correlations in the activity of the devices, leading to improvements in throughput performance and in energy efficiency.

This chapter is organized as follows: in Section 4.2 we present our system model, in Section 4.3 we define the objective function, in Section 4.4 we introduce the optimization method and some implementation subtleties, and finally we show simulation results in Section 4.5.

4.2 System Model

Consider a single input multiple output (SIMO) network consisting of a single base station with N_r antennas, and N devices equipped with a single antenna. Similarly to the NB-IoT protocol, we consider that uplink communications occur over a single carrier that is shared by the devices. Devices are assumed to be synchronized, and the time is split into frames. Each frame consists of two parts: device identification and data transmission, as shown in Fig. 2.6b. During the device identification part, all devices that seek to transmit data send their unique identifier (pilot) at the same time. K slots are available for data transmission, and active devices can transmit data on only one slot per frame. Each slot is further divided into L symbols. The considered communication protocol is described in Alg. 9.

Algorithm 9: Communication Protocol.

(Downlink) Sync signal sent by the BS to indicate beginning of the first frame and inform devices about the allocation matrix $\mathbf{A}^1$ and the power matrix $\mathbf{P}^1$.
$t \leftarrow 1$
while *True* **do**
  (Uplink) Each active device n sends their preamble $\mathbf{b}_n$ on a common control channel.
  (Local at BS) The BS decodes the preambles and obtain the activity vector $\mathbf{X}^t$ without errors, and the CSI.
  (Local at devices) Each active device n randomly selects its transmit slot $r_n^t \sim \mathbf{A}_n^t$.
  (Uplink) Each active device n sends its data at power $P_{nr_n^t}^t$ on the selected slot r_n^t.
  (Local at BS) BS decodes data present in each slot k using a SIC decoder.
  (Local at devices) Wait for beginning of next frame.
  (Local at BS) Compute the next transmission parameters $\mathbf{A}^{t+1}$, $\mathbf{P}^{t+1}$.
  (Downlink) BS broadcasts $\mathbf{A}^{t+1}$, $\mathbf{P}^{t+1}$.
  $t \leftarrow t+1$
end

4.2.1 Activity Model

In each frame, only a subset of the N devices is active. Let $X_n^t \in \{0,1\}$ be the state of device n in frame t (respectively inactive or active), and $\mathbf{X}^t \in \{0,1\}^N$ the vector of active devices in frame t. Let $\mathcal{A}^t$ and N_a^t be the set and the number (respectively) of devices that are active in frame t, $\mathcal{A}^t = \{n : \ X_n^t = 1\}$, and $N_a^t = |\mathcal{A}^t| = \sum_{n=1}^N X_n^t$. In each frame, the vector $\mathbf{X}^t$ is drawn from the same probability distribution $p_{\mathbf{X}}$, thus the activity vector $\mathbf{X}^t$ in frame t is independent of the other frames; *i.e.*, $\mathbf{X}^t$ is independent of $\mathbf{X}^{t'}$ $\forall t \neq t'$. For clarity, we drop the superscript t whenever the frame index is not relevant. No assumption is made on the distribution $p_{\mathbf{X}}$, and, in general, $p_{\mathbf{X}}(\mathbf{x}) \neq \prod_{n=1}^N p_{X_n}(x_n)$. We consider the following assumptions:

(i) the base station does not have knowledge of the distribution $p_{\mathbf{X}}$.

(ii) the base station does not have knowledge of the marginals, the Pearson correlation coefficients nor any moments of the distribution $p_{\mathbf{X}}$.

(iii) we assume that, during the pilot phase, the base station detects without error the active devices (*e.g.* using receivers like [18]), and thus has access to the samples $\mathbf{X}^t$, $\forall\ t = 1, 2, \ldots$.

4.2.2 Slot Selection

Like in Chapter 3 and as described in Chapter 2.2.6 and Alg. 3, the choice of the slots is made randomly by the devices by following individual probability distributions, defined in the allocation matrix $\mathbf{A}^t$ which is computed and optimized centrally by the BS. Let $\mathcal{A}_k^t$ and $N_{a,k}^t$ be the set and the number of devices that are active in frame t and which selected the slot k to perform their transmission. Using the $N \times K$ binary matrix $\boldsymbol{\delta}^t$ defined in Chapter 2.2.6, we have $\mathcal{A}_k^t = \{n : \ \delta_{n,k}^t X_n^t = 1\}$ and $N_{a,k}^t = |\mathcal{A}_k^t| = \sum_{n=1}^N \delta_{n,k}^t X_n^t$.

4.2.3 Transmit Power

Contrarily to the slot selection, the transmit power is entirely determined by the matrix $\mathbf{P}^t \in \mathbb{R}_+^{N\times K}$, computed and optimized centrally. The elements of $\mathbf{P}^t$, $P_{n,k}^t$ are the transmit power of device n in slot k of frame t, the transmit power is thus dependent on the slot. The transmit power of each individual device is constrained by $P_{\min,i} \leq P_{n,k}^t \leq P_{\max}$, where $0 < P_{\min,i} \leq P_{\max}$. $P_{\min,n}$ has to be large enough to ensure reliable reception, if no other device are transmitting, and is defined later in (4.4). $P_{\max}$ is a value that depends on the standard or other regulations.

4.2.4 Channel Model and Receiver

The channel between the device n and the base station is modeled by a channel $\mathbf{h}_n \in \mathbb{C}^{N_r}$, described in Section 4.5.1. During a slot, devices send a fixed number of symbols L. In the slot k of frame t, assuming there are $N_{a,k}^t$ active devices, the received signal $\mathbf{Y}_k^t \in \mathbb{C}^{L\times N_r}$ is as seen in (2.22):

$$\mathbf{Y}_k^t = \sqrt{\text{diag}(\mathbf{P}_k^t)}\mathbf{D}^t(\mathbf{H}^t)^{\mathsf{H}} + \mathbf{W}^t, \tag{4.1}$$

where $\text{diag}(\mathbf{P}_k^t)$ represents the $N_{a,k}^t \times N_{a,k}^t$ diagonal matrix of the devices' transmit power in slot k, $\mathbf{D}^t = \left[\mathbf{d}_1^{t\,\mathsf{H}} \quad \dots \quad \mathbf{d}_{N_{a,k}^t}^{t\,\mathsf{H}}\right] \in \mathbb{C}^{L\times N_{a,k}^t}$, with $\mathbb{E}[\|\mathbf{d}_n\|^2] = 1\ \forall n$, represents the data symbols sent by the active devices, $\mathbf{H}^t = \left[\mathbf{h}_1^{t\,\mathsf{H}} \quad \dots \quad \mathbf{h}_{N_{a,k}^t}^{t\,\mathsf{H}}\right] \in \mathbb{C}^{N_r\times N_{a,k}^t}$ is the channel matrix, assuming the channel is constant throughout the frame t, and $\mathbf{W}^t \in \mathbb{C}^{L\times N_r}$ is circularly Gaussian $\sim \mathcal{CN}(0, \sigma^2 I)$.

In this work, perfect CSIR is assumed (which can be obtained jointly with the vector of active devices in the preamble phase using GAMP-based algorithm [18, 19]) and a MRC spatial filter is utilized, as described in Chapter 2.2.7. The signal associated to device n is (superscript t is dropped to ease the notation):

$$\hat{\mathbf{y}}_n = \mathbf{Y}_k \frac{\mathbf{h}_n}{\|\mathbf{h}_n\|^2} = \sqrt{P_{n,k}}\mathbf{d}_n + \sum_{m\in\mathcal{A}_k\setminus\{n\}} \sqrt{P_{m,k}}\mathbf{d}_m \frac{\mathbf{h}_m^\dagger \mathbf{h}_n}{\|\mathbf{h}_n\|^2} + \tilde{\mathbf{w}}_n, \tag{4.2}$$

where $\hat{\mathbf{y}}_n \in \mathbb{C}^L$ is the received approximation of $\mathbf{d}_n$, and $\tilde{\mathbf{w}}_n \sim \mathcal{CN}\left(0, \sigma^2 \text{diag}\left(\frac{1}{\|\mathbf{h_n}\|^2}\right)\sigma^2\right)$.

Let $\mathcal{S} \subset \mathcal{P}(N)$, with $\mathcal{P}(N)$ the power set of $\{1,\dots,N\}$, be a set of devices ordered in decreasing received power in slot k: $\|\mathbf{h}_{\mathcal{S}(i)}\|^2 P_{\mathcal{S}(i),k} \geq \|\mathbf{h}_{\mathcal{S}(j)}\|^2 P_{\mathcal{S}(j),k},\ \forall\ i < j$. For a given power matrix $\mathbf{P}$, we can express the SINR in slot k of the first device $S(1)$ in the set $\mathcal{S}$ under the noise and the interference of the other devices of $\mathcal{S}$:

$$\text{SINR}_k(\mathcal{S}, \mathbf{P}) = \begin{cases} \dfrac{P_{\mathcal{S}(1),k}}{\frac{\sigma^2}{\|\mathbf{h}_{\mathcal{S}(1)}\|^2} + \sum_{m=2}^{|\mathcal{S}|}\left(\frac{|\mathbf{h}_{\mathcal{S}(m)}^\dagger \mathbf{h}_{\mathcal{S}(1)}|}{\|\mathbf{h}_{\mathcal{S}(1)}\|^2}\right)^2 P_{\mathcal{S}(m),k}} & |\mathcal{S}| > 1 \\ \dfrac{\|\mathbf{h}_{\mathcal{S}(1)}\|^2 P_{\mathcal{S}(1),k}}{\sigma^2} & |\mathcal{S}| = 1 \end{cases} \tag{4.3}$$

We assume that the data of $\mathcal{S}(1)$ can be decoded if $\text{SINR}_k(\mathcal{S}, \mathbf{P}) > \gamma;\ \gamma > 1$. An outage arises when $\text{SINR}_k(\mathcal{S}, \mathbf{P}) \leq \gamma$. Under this model, the minimum transmit power of device n is irrespective of the

slot and is expressed as:

$$P_{\min,n} = \frac{\gamma\sigma^2}{\|\mathbf{h}_n\|^2} + \epsilon, \tag{4.4}$$

with $\epsilon > 0$, obtained when no interfering devices are present.

We additionnaly assume that the receiver performs SIC, as described in Chapter 2.2.7: if $\mathrm{SINR}_k(\mathcal{S}, \mathbf{P}) > \gamma$, the receiver can proceed to compute $\mathrm{SINR}_k(\mathcal{S} \setminus \mathcal{S}(1), \mathbf{P})$, otherwise all undecoded devices are considered outed of the network.

Remark. Because the channel model we use in the simulations exhibits spatially correlated channel coefficients as described in Section 4.5.1, making use of a ZF receiver is unlikely to perform well. Furthermore, due to our choice to sort the devices in terms of decreasing received power for the SIC process, the MRC receiver is favored over a MMSE receiver as it maximizes the received power of each signal. We emphasize that optimal ordering for SIC with multiple antennas is still an active research area [25, 26, 27] and sorting by received power is only a heuristic. Using other heuristics like ordering by SINR could provide different results and could lead to utilizing other receiving strategies.

4.3 Objective

In this chapter, we are interested in maximizing the expected number of devices that transmit without an outage, as described in [117]. A device is considered as outed of the network whenever the MRC filter followed by the SIC decoding process cannot decode the data of the device, either because its SINR is not greater than the threshold, or because a device with a higher received power was not decoded. This objective is the extension of the throughput maximization used in Chapter 3 to multi-antenna systems.

In a similar spirit as the objective of Chapter 3, the objective is defined as an expectation over the activity distribution of the device $p_{\mathbf{X}}$:

$$T(\mathbf{A}, \mathbf{P}) = \mathbb{E}_{\mathbf{X}\sim p_{\mathbf{X}}}[T(\mathbf{A}, \mathbf{P}; \mathbf{X})] \tag{4.5}$$

$$T(\mathbf{A}, \mathbf{P}; \mathbf{X}) = \sum_{k=1}^{K} \sum_{\mathcal{S}\in\mathcal{P}(\mathcal{A}_k)} Q_k(\mathbf{A}, \mathcal{S})\mathrm{SIC}_k(\mathcal{S}, \mathbf{P}), \tag{4.6}$$

where $\mathcal{P}(\mathcal{A})$ represents the power set of the active devices (the indices of $\mathbf{X}$ where $X_n = 1$), and

$$Q_k(\mathbf{A}, \mathcal{S}) = \prod_{i\in\mathcal{S}} A_{ik} \prod_{j\in\mathcal{S}^c} (1 - A_{jk}), \tag{4.7}$$

represents the probability that all devices in a subset of the active devices choose a particular slot while the other devices choose another slot. Note the similarity with (3.3). SIC_k represents the number of devices within a set of devices $\mathcal{S}$ that have a sufficient SINR to be decoded using a SIC

decoder, and is defined as:

$$\begin{aligned}\mathrm{SIC}_k(\mathcal{S},\mathbf{P}) &= \mathbb{1}\left\{\mathrm{SINR}_k(\mathcal{S},\mathbf{P})>\gamma\right\}\\ &\quad+\mathbb{1}\left\{\mathrm{SINR}_k(\mathcal{S},\mathbf{P})>\gamma\right\}\mathbb{1}\left\{\mathrm{SINR}_k(\mathcal{S}\setminus\{\mathcal{S}(1)\},\mathbf{P})>\gamma\right\}\\ &\quad\dots\\ &\quad+\mathbb{1}\left\{\mathrm{SINR}_k(\mathcal{S},\mathbf{P})>\gamma\right\}\dots\{\mathrm{SINR}_k(\mathcal{S}\setminus\{\mathcal{S}(1)\dots\mathcal{S}(|\mathcal{S}|-1)\},\mathbf{P})>\gamma\}\\ &=\sum_{l=0}^{|\mathcal{S}|-1}\mathbb{1}\left\{\mathrm{SINR}_k(\mathcal{S},\mathbf{P})>\gamma\right\}\prod_{m=1}^{l}\mathbb{1}\left\{\mathrm{SINR}_k(\mathcal{S}\setminus\{\mathcal{S}(1),\dots,\mathcal{S}(m)\},\mathbf{P})>\gamma\right\}. \end{aligned}\tag{4.8}$$

All sets $\mathcal{S}\in\mathcal{P}(\mathcal{A})$ are sorted in terms of decreasing received power. Note that if a device $\mathcal{S}(n)$ cannot be decoded, then all subsequent devices $\mathcal{S}(m)$, with $n<m$, also cannot be decoded.

To allow for a fair comparison of the performance under different network statistics, we define the normalized expected number of devices that transmit without outage:

$$T^N(\mathbf{A},\mathbf{P})=\frac{\mathbb{E}_{\mathbf{X}\sim p_{\mathbf{X}}}[T_1(\mathbf{A},\mathbf{P};\mathbf{X})]}{\sum_{n=1}^{N}\mathbb{E}_{X_n\sim p_{X_n}}[X_n]}.\tag{4.9}$$

4.4 Proposed Algorithm

To maximize the objective defined in (4.5), two types of resources are utilized:

- the slot selection policy is employed to schedule devices on different time slot,
- the transmit power of the devices is used to perform power multiplexing, taking advantage of the SIC decoder.

In addition, two types of physical properties are exploited:

- the correlations in the activity distribution of the devices $p_{\mathbf{X}}$, with the goal of ensuring that highly correlated devices do not use the same resources,
- the channel coefficients, allowing, for example, to put on the same slot and power resource devices that have orthogonal channels.

Note that only the slot selection policy $\mathbf{A}$ and the transmit power can be used to maximize (4.5), the channel coefficients and the correlations being dependent on the scenario considered. The optimization problem is written as follows:

$$\begin{aligned}\min_{\mathbf{A}\in\mathbb{R}_+^{N\times K},\ \mathbf{P}\in\mathbb{R}_+^{N\times K}}\quad & -T(\mathbf{A},\mathbf{P})\\ \text{subject to:}\quad & P_{\min,n}\le P_{n,k}\le P_{\max},\ \forall n\in\{1\dots N\},\ k\in\{1\dots K\}\\ & \sum_{k=1}^{K}A_{n,k}=1,\ \forall n\in\{1\dots N\}.\end{aligned}\tag{4.10}$$

Note that problem (4.10) is non-convex due to the nature of (4.7). Note also that there is no guarantee there exists a unique solution, and that the objective in (4.10) is not continuous and thus not differentiable due to the indicator function in (4.8). To circumvent the latter issue, we propose to approximate the indicator with a properly scaled sigmoid function centered on the threshold γ:

$$\widetilde{\mathrm{SIC}}_k(\mathcal{S},\mathbf{P}) = \sum_{l=0}^{|\mathcal{S}|-1} \sigma_f\left(b\mathrm{SINR}_k(\mathcal{S},\mathbf{P})-\gamma\right) \cdot \prod_{m=1}^{l} \sigma_f\left(b\mathrm{SINR}_k\left(\mathcal{S}\setminus\{\mathcal{S}(1),\ldots,\mathcal{S}(m)\},\mathbf{P}\right)-\gamma\right), \tag{4.11}$$

where $\sigma_f(x)=\frac{1}{1-e^{-x}}$ is the sigmoid function and b is a sharpness parameter. We thus can define:

$$\tilde{T}(\mathbf{A},\mathbf{P};\mathbf{X}) = \sum_{k=1}^{K}\sum_{\mathcal{S}\in\mathcal{P}(\mathcal{A})} Q_k(\mathbf{A},\mathcal{S}\mid\mathbf{X})\widetilde{\mathrm{SIC}}_k(\mathcal{S},\mathbf{P}). \tag{4.12}$$

Because of the complex correlation structure that $p_{\mathbf{X}}$ might exhibit, we propose to optimize $\mathbf{A}$ and $\mathbf{P}$ in (4.5) in an iterative manner using SGD, similarly to Chapter 3, based on the vector of active devices $\mathbf{X}^t$ of each frame t. In contrast to Chapter 3, such vector is assumed to be error-free during the pilot phase of GFRA that takes place before the data transmission phase. The sample $\mathbf{X}^t \sim p_{\mathbf{X}}$ is utilized to yield $\tilde{T}(\mathbf{A}^t,\mathbf{P}^t;\mathbf{X}^t)$. Considering that the two optimization variables $\mathbf{A}$ and $\mathbf{P}$ have different impacts and that the activity of the devices is heterogeneous, we use ADAGRAD [29], as described in Appendix A.3, to properly scale the learning rate of the different variables. The optimization algorithm is detailed in Alg. 10, where ADAGRAD is as implemented in the Pytorch package [81]. The operator $\Pi_{\mathcal{H}}\{\cdot\}$ represents, like in Chapter 3, the projection of $\mathbf{A}$ to the closest point in norm ℓ_1 in the constraint set $\mathcal{H}=\{\mathbf{A}\in\mathbb{R}_+^{N\times K} : \sum_{k=1}^{K} A_{n,k}=1,\ \forall n\}$, and the operator $\Pi_{[P_{\min,n},P_{\max}]}$ represents the clipping of $P_{n,k}$ between $P_{\min,n}$ and $P_{\max}$.

Algorithm 10: Optimization of $\mathbf{A}$ and $\mathbf{P}$ used to solve (4.10).

Choose initial allocation matrix $\mathbf{A}^1$ initial power allocation $\mathbf{P}^1$, and step-size sequence $\{\mu^t\}$.
$t \leftarrow 1$
while *not converged* **do**
  Detect active devices to obtain $\mathbf{X}^t$.
  $(\mathbf{A}^{t+1},\mathbf{P}^{t+1}) \leftarrow \texttt{ADAGRAD}(\mu^t,(\mathbf{A}^t,\mathbf{P}^t),\tilde{T}(\mathbf{A}^t,\mathbf{P}^t;\mathbf{X}^t))$
  $\mathbf{A}^{t+1} \leftarrow \Pi_{\mathcal{H}}[\mathbf{A}^{t+1}]$
  $\mathbf{P}^{t+1} \leftarrow \Pi_{[P_{\min,n},P_{\max}]}[\mathbf{P}^{t+1}]$
end

4.4.1 Practical Consideration: Sampling the Power Set

A major drawback of 4.12 is the presence of the power set of all active devices, which has a cardinality of $|\mathcal{P}(\mathcal{A})| = 2^{N_a}$, leading to an exponential running time in the number of active devices, which can be problematic if N_a is big. To reduce the complexity, we propose to sample uniformly at random without replacement $N_s \leq |\mathcal{P}(\mathcal{A})|$ sets from the power set and only use these to compute the objective.

$$\begin{aligned}
T(\mathbf{A},\mathbf{P};\mathbf{X}) &= \sum_{k=1}^{K} \sum_{\mathcal{S}\in\mathcal{P}(\mathcal{A})} Q_k(\mathbf{A},\mathcal{S} \mid \mathbf{X})\mathrm{SIC}_k(\mathcal{S},\mathbf{P}) \\
&= \sum_{k=1}^{K} |\mathcal{P}(\mathcal{A})| \sum_{\mathcal{S}\in\mathcal{P}(N)} \frac{1}{|\mathcal{P}(\mathcal{A})|} Q_k(\mathbf{A},\mathcal{S} \mid \mathbf{X})\mathrm{SIC}(\mathcal{S},\mathbf{P}) \\
&= |\mathcal{P}(\mathcal{A})| \sum_{k=1}^{K} \mathbb{E}_{S\sim\mathcal{U}(\mathcal{P}(\mathcal{A}))} \left[Q_k(\mathbf{A},\mathcal{S} \mid \mathbf{X})\mathrm{SIC}(\mathcal{S},\mathbf{P})\right] \\
&= |\mathcal{P}(\mathcal{A})| \left(\sum_{k=1}^{K} \sum_{i=1}^{N_s} \frac{1}{N_s} Q_k(\mathbf{A},\mathcal{S}_i \mid \mathbf{X})\mathrm{SIC}(\mathcal{S}_i,\mathbf{P}) + \epsilon(\mathbf{A},\mathbf{P};\mathbf{X},\mathcal{S}_i) \right) \\
&\approx |\mathcal{P}(\mathcal{A})| \sum_{k=1}^{K} \sum_{i=1}^{N_s} \frac{1}{N_s} Q_k(\mathbf{A},\mathcal{S}_i \mid \mathbf{X})\mathrm{SIC}(\mathcal{S}_i,\mathbf{P})
\end{aligned}$$

where $S \sim \mathcal{U}(\mathcal{P}(\mathcal{A}))$ means that S is drawn uniformly at random in the power set of the active devices and $\mathcal{S}_i$ is the i-th sample out of the N_s samples drawn uniformly at random and without replacement from $\mathcal{P}(\mathcal{A})$, and where $\epsilon\left(\mathbf{A},\mathbf{P},\mathbf{X},\mathcal{S}_i\right) \in \mathbb{R}$ is the error term that vanishes when $N_s \to 2^{N_a}$.

The factor $|\mathcal{P}(\mathcal{A})|$ is necessary to maintain the property:

$$\lim_{N_s\to|\mathcal{P}(\mathcal{A})|} |\mathcal{P}(\mathcal{A})| \sum_{k=1}^{K} \sum_{i=1}^{N_s} \frac{1}{N_s} Q_k(\mathbf{A},\mathcal{S}_i \mid \mathbf{X})\mathrm{SIC}(\mathcal{S}_i,\mathbf{P}) = T(\mathbf{A},\mathbf{P};\mathbf{X}) \tag{4.13}$$

4.5 Numerical Results

4.5.1 Simulation Scenario

Channel Model

We consider a network consisting of a single cell, with a BS equipped with $N_r = 2$ antennas and $N = 15$ devices with a single transmit antenna. The cell is represented as a $D \times D$ square grid of size $D = 100$, with the BS lying at the center $(x_{\mathrm{BS}}, y_{\mathrm{BS}}) = (\frac{D}{2}, \frac{D}{2})$. The width of the grid is 10 units and thus each unit is made of 10 samples. The cell is represented in Fig. 4.2.

The frames consist of $K = 3$ slots. The channels are assumed to be fixed during all the simulations

and are the same for all the slots. They consist of pathloss, shadowing and fading.

The channel between antenna i of the BS and device n is characterized as follows, as seen in Chapter 2.2.7:

$$h_{n,i} = f_{n,i}\sqrt{10^{\frac{PL(x_n,y_n)+Sh(x_n,y_n)}{10}}} \tag{4.14}$$

where:

- $f_{n,i} \sim \mathcal{CN}(0,1)$ is the Rayleigh fading coefficient,
- $PL(x_n, y_n) = -10\eta \log_{10}(\rho_n)$ is the pathloss at distance $\rho_n = \sqrt{x_n^2 + y_n^2}$, with x_n and y_n the Cartesian coordinates of device n, and $\eta = 2.6$, representing an office-like environment.
- $Sh(x_n, y_n)$ is the shadowing term representing the possible obstacles in the line of sight between the BS and the device. In Chapter 2.2.7 shadowing was introduced as a log-normal distribution independently affecting each device. To introduce spatial correlation in the channel model, the shadowing term at each position of the 100×100 grid is computed by creating a 100×100 grid of random samples from $\mathcal{N}(0, 20)$, and then applying a Gaussian filter with a Gaussian kernel of standard deviation 2.

The channel map used for the simulations can be seen in Fig. 4.2. Based on the channels of the devices and regardless of the transmit power, we can have insights on whether the MRC receiver will increase or reduce the interference of co-transmitting devices by computing the MRC interference coefficient $M_{n,m} = \dfrac{\mathbf{h}_m^\dagger \mathbf{h}_n}{\|\mathbf{h}_n\|^2}$. Note that this coefficient is not symmetric: $M_{n,m} \neq M_{m,n}$. If $M_{n,m} < 1$ then the MRC reduces the interference of device m, otherwise it is increased. If $M_{n,m} = 0$, then the two channels are orthogonal, meaning that the devices will not interfere regardless of the transmit power of the interfering device. Fig. 4.1 shows the MRC interference coefficient associated to the channel map in Fig. 4.2.

Activity Model

The activity distribution $p_{\mathbf{X}}$ of the devices is generic and available to the access point only trough samples $\mathbf{X} \sim p_{\mathbf{X}}$. To simulate an activity distribution that is non-trivial, with correlations and heterogeneous marginal activity, we use the following setup inspired by sensor networks:

- Devices are placed in the grid at fixed coordinates x_n and y_n.
- Each device has a detection radius $r_n \sim \mathcal{U}[1, 3]$.
- In each frame, events are generated on the grid according to an homogeneous Poisson point process (HPPP): the position of the events are drawn uniformly at random on the grid and the number of events follows a Poisson distribution of intensity λ. Events are generated independently of the frame index.
- Devices that have at least one event within their detection radius will be active in the frame.

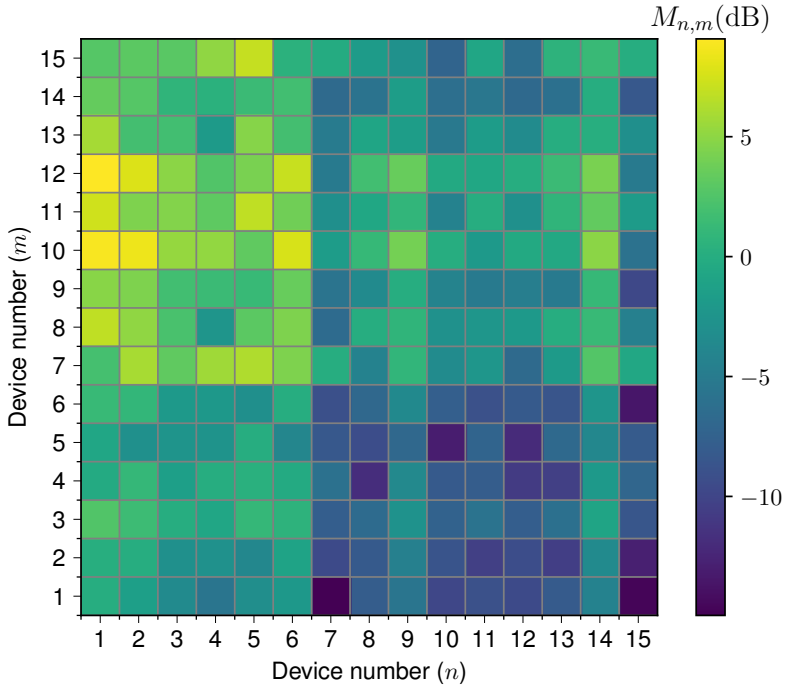


Figure 4.1: MRC interference coefficient matrix $\mathbf{M}$ in dB.

The resulting simulation scenario (device positions, detection radius and channel gains) is found in Fig. 4.2. Devices have different detection radius, thus they have different marginal activity probability. Because they are placed in clusters, several devices are likely to see the same event leading to correlations in the vectors $\mathbf{X}^t$. The color represents the gain of the channel at different (x, y) positions.

The properties (marginals, correlations...) of the activity distribution $p_{\mathbf{X}}$ are strongly dependent on the intensity λ, they cannot be explicitly expressed but can be estimated via simulation.

Example 4.5.1 (Properties of $p_{\mathbf{X}}$ for $\lambda = 0.01$)**.** The marginals $\mathbf{p} = \begin{bmatrix} p_{X_1} & \dots & p_{X_N} \end{bmatrix}$ of $p_{\mathbf{X}}$ are computed by sampling 100 000 realization of $\mathbf{X}$ giving (in sorted order):

$$\mathbf{p} = [0.03\ 0.04\ 0.06\ 0.06\ 0.07\ 0.10\ 0.10\ 0.11\ 0.12\ 0.13\ 0.15\ 0.15\ 0.16\ 0.17]$$

The average number of active devices per frame is 1.6. Based on the 100 000 samples, we can also obtain insights on the correlation structure of $p_{\mathbf{X}}$ by computing the Pearson correlation coefficients $R_{n,m} = \frac{\mathrm{Cov}(X_n, X_m)}{\sqrt{\mathrm{Var}(X_n)\mathrm{Var}(X_m)}}$. The Pearson matrix $\mathbf{R}$ is represented in Fig. 4.3.

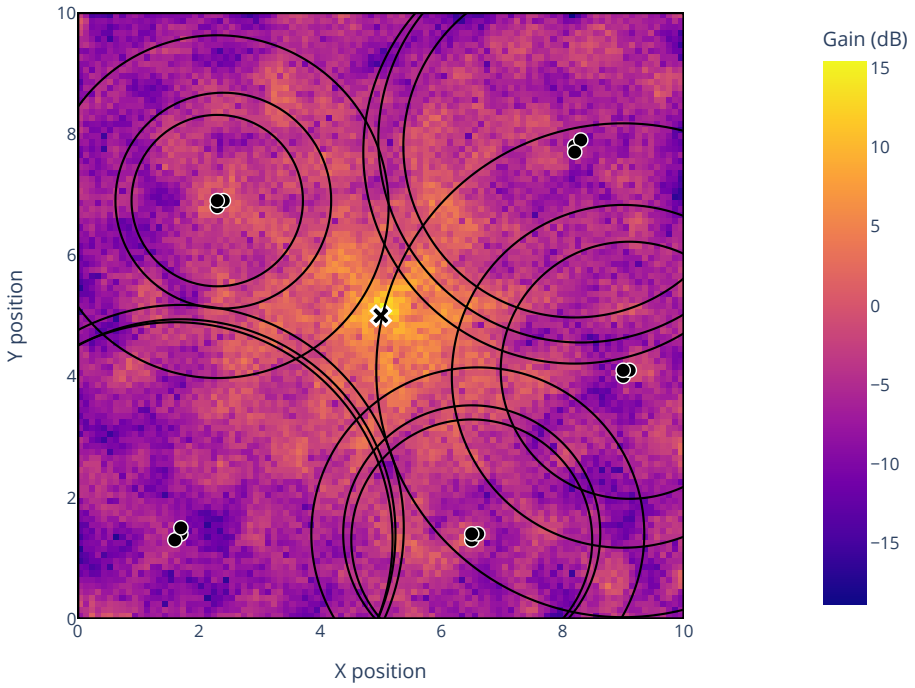


Figure 4.2: Position of the 15 devices, represented by a black dot with their detection radius and the channel gain in the cell. The BS is at the center, represented by a cross.

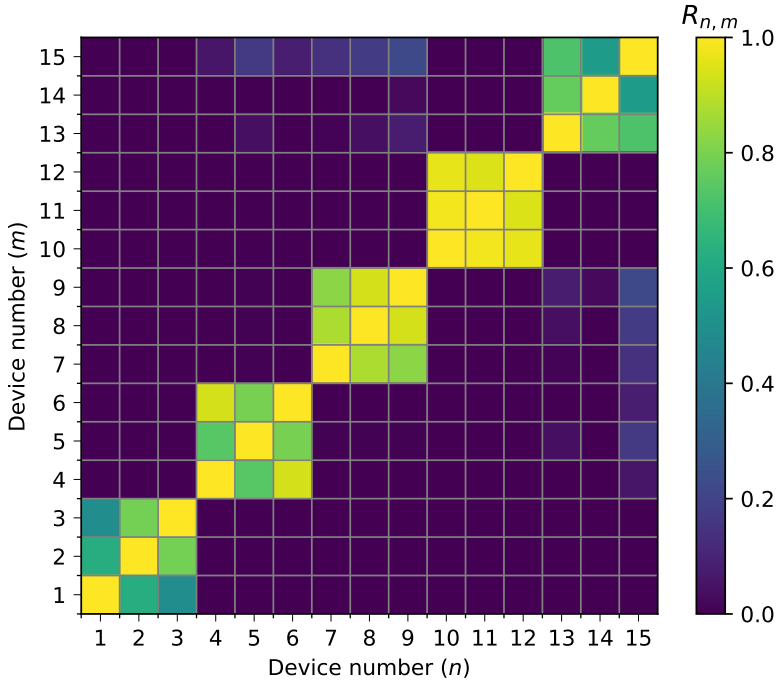


Figure 4.3: Pearson coefficient matrix $\mathbf{R}$, for an intensity of the HPPP of $\lambda = 0.01$. The clusters of 3 devices present in Fig. 4.2 clearly appear as highly correlated with a coefficient close to 1.

4.5.2 Simulation Parameters and Methods Considered

We evaluate the performance of a slot and a power allocation optimized with Alg. 10 in two different scenarios:

- a highly constrained scenario, with $P_{\max} = 6$,
- a less constrained scenario, with $P_{\max} = 12$.

We compare the performance of four different $(\mathbf{A}, \mathbf{P})$ pairs:

(i) Initialization points $\mathbf{A}^1$ and $\mathbf{P}^1$ of Alg. 10

(ii) Optimization with Alg. 10 with $X_n^t \sim p_{X_n}$, $\forall n$ and X_n independent of X_m, $\forall m \neq n$, *i.e.* same marginals as $p_{\mathbf{X}}$ but with independent devices.

(iii) Optimization with Alg. 10 with $\mathbf{X}^t \sim p_{\mathbf{X}}$

(iv) Allocation matrix $\mathbf{A}_{\text{ALOHA}}$ with transmission at a random power in each frame $P_{n,k} \sim \mathcal{U}([P_{\min,n}, P_{\max}])$.

The comparison between ((ii)) and ((iii)) allows to see the advantage of taking into account the correlation of the devices while optimizing the access policy.

In all scenarios, the initial power matrix used corresponds to $P^1_{n,k} = P_{\min,n}$, $\forall k$ as defined in (4.4):

$$\mathbf{P}^1 = \begin{bmatrix} 0.642 & 0.642 & 0.642 \\ 0.7622 & 0.7622 & 0.7622 \\ 1.2744 & 1.2744 & 1.2744 \\ 1.182 & 1.182 & 1.182 \\ 0.8459 & 0.8459 & 0.8459 \\ 0.9511 & 0.9511 & 0.9511 \\ 4.5208 & 4.5208 & 4.5208 \\ 3.4823 & 3.4823 & 3.4823 \\ 2.1266 & 2.1266 & 2.1266 \\ 5.5045 & 5.5045 & 5.5045 \\ 4.2074 & 4.2074 & 4.2074 \\ 5.4802 & 5.4802 & 5.4802 \\ 3.166 & 3.166 & 3.166 \\ 1.5519 & 1.5519 & 1.5519 \\ 4.7563 & 4.7563 & 4.7563 \end{bmatrix} \tag{4.15}$$

We evaluate the performance of the different methods as a function of the intensity λ of the HPPP. For a given λ, the process to evaluate ((ii)) and ((iii)) is performed according to the following steps:

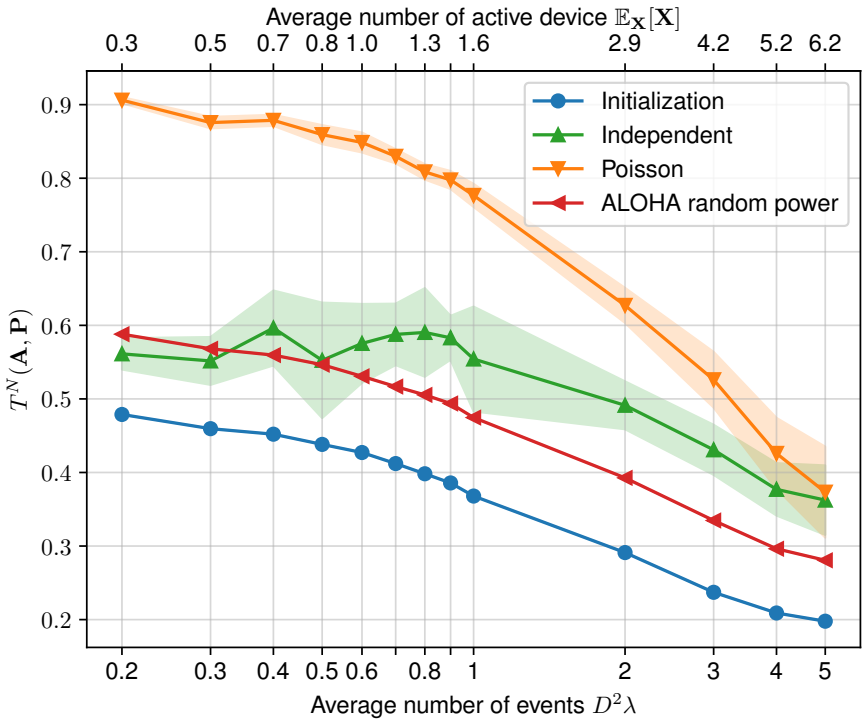


Figure 4.4: Resulting objective $T^N(\mathbf{A}, \mathbf{P})$ after optimizing for 30 000 frames. The shaded area represents $\pm$ the standard deviation.

1. For a given $\mathbf{A}^1$, Alg. 10 is run for 30 000 frames after which (4.5) is evaluated. Due to the difficulty of evaluating the probability of having a particular activity vector ($\Pr(\mathbf{X} = \mathbf{x})$ is unknown), the objective is obtained by computing (4.6) for 4 500 activity vectors $\mathbf{X} \sim p_{\mathbf{X}}$ and averaging them, giving a Monte Carlo approximation of the objective.

2. Because the sequence of activity vectors $\mathbf{X}^t \sim p_{\mathbf{X}},\ t = 1 \dots 30\,000$ affects the directions taken by Alg. 10, Step 1 is repeated for 10 different sequences of activity vector. The resulting objectives are averaged to give an expected performance of the $(\mathbf{A}^1, \mathbf{P}^1)$ pair.

3. Furthermore, because Alg. 10 only provides a guarantee of convergence to a local stationary point (due to the non convexity of (4.5) in $\mathbf{A}$), Step 1 and Step 2 are repeated with 5 different $\mathbf{A}^1$, each drawn uniformly at random within the constraint space $\mathcal{H}$, and with the same initial power matrix $\mathbf{P}^1$ defined in (4.15). This allows to explore the optimization space.

4. Out of the 5 initializations of Step 3, we keep the three initializations that lead (respectively) to the best objective for method ((i)), ((ii)) and ((iii)). We then plot the average resulting objective of the 10 different trajectories associated to the best initialization for method ((ii)) and for method ((iii)).

4.5.3 Simulation Results

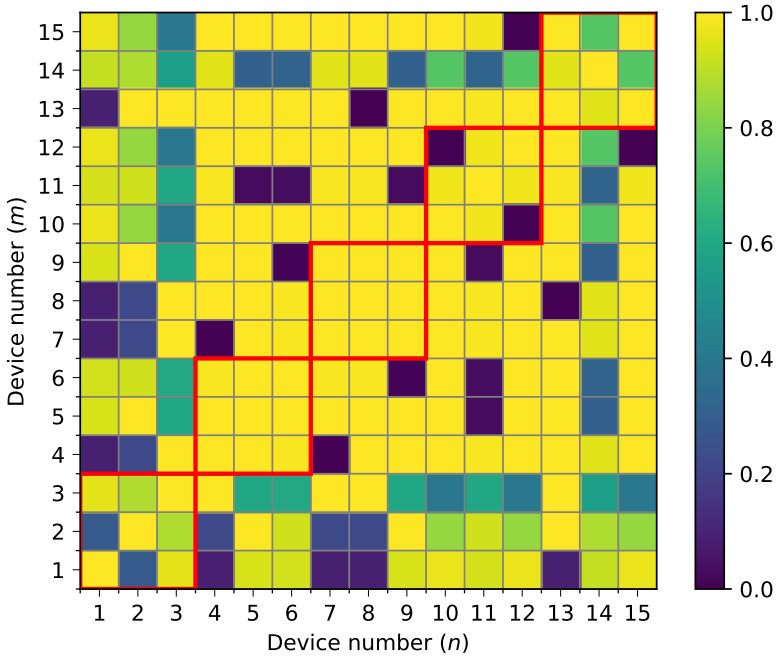
Few Power Resources

Fig. 4.4 shows the normalized objective (4.9) for λ ranging between 0.002 and 0.05, corresponding to an average of $D^2\lambda = 0.2$ events per frames until $D^2\lambda = 5$ events per frame. Note that an event might spark either zero, one, or several devices; thus 5 events per frame does not mean an average of 5 active devices. The expected number of devices is shown as a second x-axis on top of the plot. For the low activity regime ($D^2\lambda \leq 0.1$), the optimization with the independent network only provides a small increase (a maximum of 18% at $D^2\lambda = 0.8$) compared to the ALOHA baseline. In this regime, if the devices are all independent, the number of active devices is likely to be either 0 or 1. As a result no changes are made in the allocation of resources by ADAGRAD. For the optimization that takes into account the correlation, the optimization algorithm saw with higher probability several active devices even if the average number of active devices is the same. With this advantage, the optimization algorithm is able to orthogonalize devices that are in the same cluster. As a result, significant improvement over the ALOHA baseline and the independent case is obtained. The number of decoded devices reaches 90% at the evaluation when the average number of event per frame is 0.2 and provides an improvement of 40% over the independent case and of 65% over the ALOHA baseline.

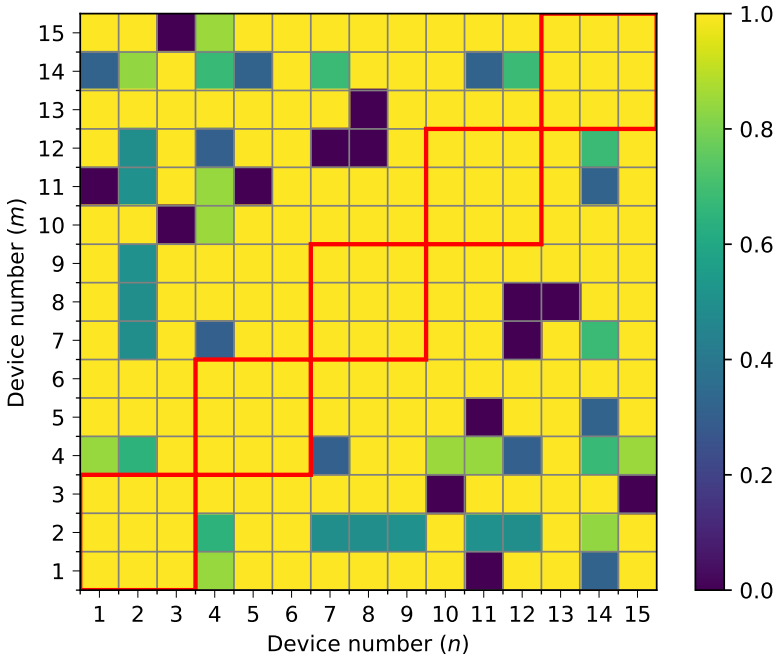
In the high activity regime ($D^2\lambda > 0.1$, more than 1 event per frame in average), we can observe that the performance of all methods drops because the resources available are too limited. The optimization taking into account the Poisson correlations remains the method with the best performance.

The difference of performance in the resulting resource allocation is due to the correlation structure of the HPPP. To gain some insight, after optimizing the allocation with an intensity of $D^2\lambda = 0.1$, Fig. 4.5 shows the value of $T^N(\mathbf{A}, \mathbf{P}; \mathbf{X})$ for all possible pair of active devices, *i.e.* for all $(n, m) : X_n = 1, X_m = 1, X_l = 0,\ \forall l \neq n, m$. As it can be seen in Fig. 4.5a, when optimizing with the assumption that the devices are independent, some devices that often transmit together (like 10 and 13) can never be decoded whenever both of them are active. On the other hand, all devices in the clusters of the optmization that follows the HPPP (Fig. 4.5b) are, at least, pairwise decodable.

Regarding the transmit power, Fig. 4.6 shows $\mathbb{E}_{\mathbf{X}}\left[\sum_{n=1}^{N} \mathbb{E}_{\mathbf{A}_n}[X_n \mathbf{P}_n]\right]$, the expected total transmit power of all devices in the network. We can see that the total transmit power increases with the average number of active devices per frame. The initialization (blue, circle markers) is the lowest because the transmit power of all the devices is set to $P_{\min,n}, \forall n$. Interestingly, the ALOHA baseline, the optimization assuming independence and the optimization taking into account the correlations all have similar total transmit power. This suggests that the optimization algorithm manages to assign various transmit power to the devices.



(a) Probability that data of n and m can be decoded, whenever n and m are the only two active devices and when the optimization is carried with independent activity.



(b) Probability that data of n and m can be decoded, whenever n and m are the only two active devices and when the optimization is carried with activity following the HPPP.

Figure 4.5: $T^N(\mathbf{A}, \mathbf{P}; \mathbf{X})$ with $X_n = 1, X_m = 1, X_l = 0 \; \forall l \neq n, m$ for all possible n, m for $D^2\lambda = 0.01$. The red squares represent clusters of devices correlated present in Fig. 4.3.

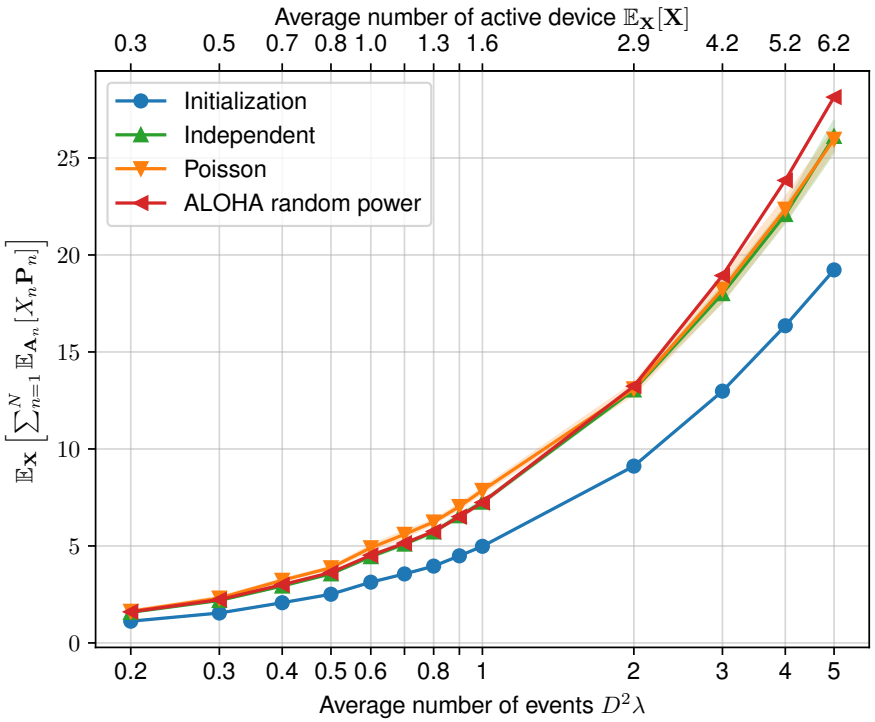


Figure 4.6: Resulting expected total transmit power $\mathbb{E}_{\mathbf{X}}\left[\sum_{n=1}^{N}\mathbb{E}_{\mathbf{A}_n}[\mathbf{P}_n]\right]$ after optimizing for 30 000 frames.

High $P_{\max}$

We evaluate the performances in the same scenario as previously, except that the maximal transmit power $P_{\max}$ is set to 12 instead of 6. Fig. 4.7 and Fig. 4.8 show the expected outage and the power for different values of $D^2\lambda$. The plots are similar to the ones of the previous scenario, however it is interesting to notice that the gap between the optimization with the Poisson correlation and the independent model is now considerably smaller for $D^2\lambda > 0.4$. This highlights that the interest of taking into account the correlation of the device is particularly noticeable when the available communication resource as highly constrained. Overall, the higher transmit power (Fig. 4.8) allows to increase the performance of all methods, (except the initialization that stays the same).

For small values of $D^2\lambda$, the ALOHA baseline outperforms the optimization with the independent assumption because the algorithms rarely observes more than one active device in a frame.

4.6 Conclusion

To fully exploit the capability of a SIC-enabled base station, controlling the transmit power of the devices as well as their transmit slot is important. In this chapter, we proposed algorithms for jointly optimizing both slot and power allocations for all devices of the network. Doing so increases the performance of the network, measured here by the expected number of devices that transmit without outage. Furthermore, the algorithm we propose is able to take into account the correlations in the activity of the devices, ensuring that devices that are likely to transmit together are decodable.

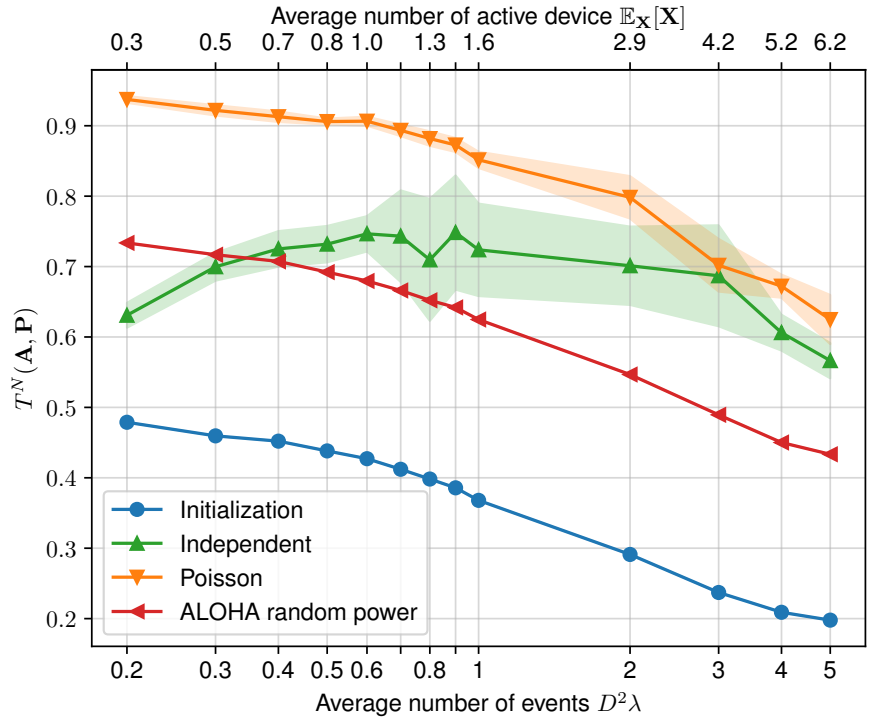


Figure 4.7: Resulting objective when $P_{\max} = 12$.

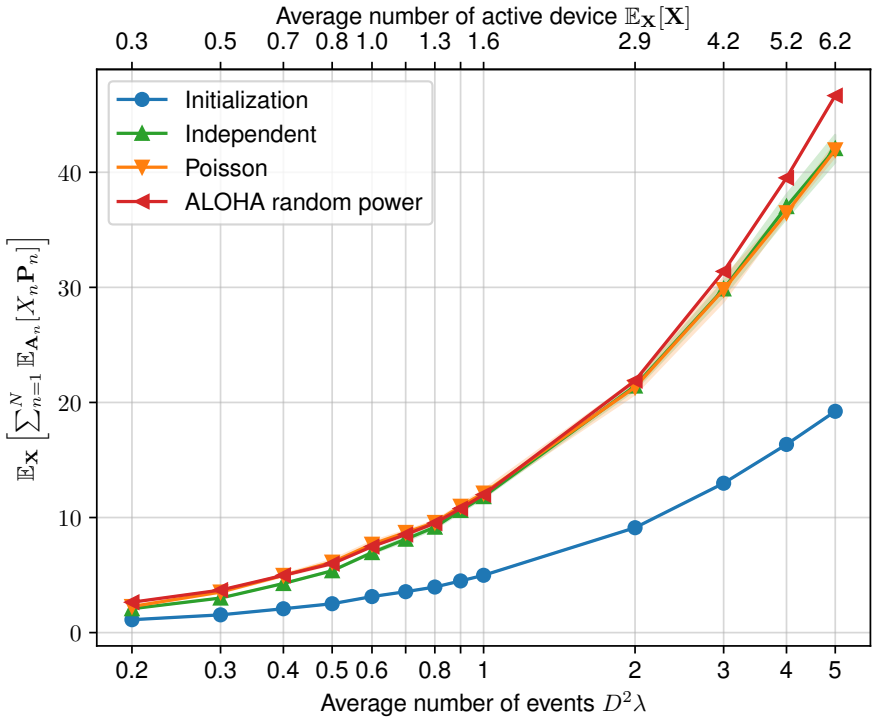


Figure 4.8: Resulting transmit power $P_{\max} = 12$.

Our simulation study shows that our algorithm, with its awareness of the correlations, can significantly outperform existing ALOHA baselines in terms of throughput using a similar power.

CHAPTER

5 Joint Power, PRB and Rank Allocation

Contents

5.0 Summary

For 6G to support a higher number of devices at higher rates, it is envisioned that networks will need to be more adaptive. To this end, transmission parameters, which are currently common to the devices in the cell (in 4G and 5G), can be individualized. Contrarily to the previous chapters, we focus on a eMBB type of network with the aim of maximizing the rate of the devices, assuming they are all always active. We consider three transmission parameters: the transmit power, the number of PRBs allocated to each device, and the number of layers (also known as spatial streams or rank[1]) used by the devices; we investigate the possible advantageis of using jointly optimized per-device parameters instead of cell-wide parameters. Our simulation results show that considering PRB allocation, power optimization, and layer selection as a joint problem allows to outform the baselines in terms of rates and power consumption.

This chapter is partially based on the works J2 ([99]) and P1.

5.1 Introduction

Leaving aside the MTC scenario, we now move to a eMBB-type of network, where the BS and the devices exploit all the frequencies of the OFDM RG instead of using a single subcarrier.

In the current 5G standard, the transmit power of the devices is designed so that all the devices in a cell are received with the same power spectral density (PSD) at the base station, easing the equalization step and ensuring a fair treatment of the devices regardless of their distance to the BS. The transmit power is subject to a hard constraint that indirectly affects the number of PRB used by the device. However, this way of defining the transmit power can make the devices close to the BS transmit at a power that is higher than necessary and devices close to the edge of the cell to transmit at a power that is not high enough to meet the required received power, which can strongly affect the quality of their transmission. Moreover, the function allocating the PRBs is the same for all devices and is independent of the channel quality of each PRB [35, Chapter 6.1.2.2]. Similarly, the number of layers used by the devices is defined by the rank indicator [64], a parameter computed independently of the PRBs allocation and the power.

Optimization of these parameters has been individually investigated: in [21] an adaptive layer selection method is reported, in [87] the optimization of the PRB allocation is performed using a graph neural network (GNN). Regarding power control, several work try to leverage ML solutions to learn a useful power allocation: [23, 61] present solutions based on reinforcement learning (respectively for 5G and 6G), and [70] describes a method based on Bayesian inference to leverage prior knowledge that could be obtained through already deployed networks. On the other hand, the joint optimization of several transmit parameters is less investigated, we can mention [9] in which a joint PRB and power allocation method has been described, but it does not include the layer optimization and focuses on the coexistence of uRLLC and eMBB for downlink traffic.

[1] We will interchangeably use these three terms

In this work, we investigate the possibility of jointly optimizing the power, the PRB and the layer allocation. The contributions of this chapter are as follows:

- Instead of optimizing the target received power, we propose to directly optimize the transmit power, without imposing identical received power, jointly with the number of PRB and number of layers of the devices.
- We formalize an optimization problem for jointly optimizing these three parameters with the goal of finding a proportional fair rate allocation to the devices in the cell. This problem is first expressed as a non-convex mixed integer non-linear program (MINLP) that can be relaxed as a convex non linear program (NLP) which can be solved using a mathematical solver (IPOPT) [107].
- To satisfy some practical constraints, (latency) and obtain a constant execution time, we develop a function approximator based on a neural network (NN) that learns the input-output relationship of the mathematical solver, allowing to considerably reduce the computational burden.
- We conduct a simulation analysis comparing our proposed method with different 5G baselines.

This chapter is organized as follows: in Section 5.2, we describe the resources available, the system model and the assumptions we make; in Section 5.3, we describe the ideal optimization problem we would like to solve; in Section 5.4, we define a relaxation of the problem and present the neural network we use to learn the solver; in Section 5.5, we present the baselines we compare to, inspired from the 5G standard; finally, in Section 5.6, we present simulation results showing the benefits of our method.

5.2 System Model

5.2.1 Resources

We consider a cellular network made of a single cell. Transmission resources are as described in Chapter 2.2.3 and Fig. 2.1. The subcarrier spacing is S_p =30 kHz. A PRBG consists of a single PRB ($N_{\text{block}} = 1$) and the total number of available PRBs is $N_{\text{PRB}} = 24$. The number of subcarriers per PRB is $N_{\text{SC}} = 12$ and the number of OFDM symbols per slot is $L = 14$. Frames last for 10 ms and consist of $N_{\text{SF}} = 10$ subframes of $N_{\text{slot}} = 2$ slots, making virtually 20 slots per frame and a total of $N_{\text{symb,tot}} = LN_{\text{slot}}N_{\text{SF}} = 280$ OFDM symbols per frame. REs are a time-frequency pair in the resource grid. Let $e = (\varphi, \tau)$, with $\varphi \in [1, N_{\text{PRB}}N_{\text{SC}}]$ the subcarrier index and $\tau \in [1, 2 \dots [$ the symbol index (regardless of the frame index). RE e is the $((\varphi \mod N_{\text{SC}}) + 1)$-th subcarrier of PRB $f = \lfloor \frac{\varphi}{N_{\text{SC}}} \rfloor$ and the $((\tau \mod L) + 1)$-th OFDM symbol of slot $\lfloor \frac{\tau}{L} \rfloor \mod N_{\text{slot}}N_{\text{SF}}$ of frame $t = \lfloor \frac{\tau}{N_{\text{symb,tot}}} \rfloor$.

We denote by $t = 1, 2, \dots$ the frame index and by $f = 1, \dots, N_{\text{PRB}}$ the PRB index.

Motivated by eMBB applications, in contrast to the previous chapters, we assume that devices

successfully performed a RACH procedure, as described in Fig. 2.5a. As a result, we focus on a set of $N = 8$ devices that are active for a long time and have been co-scheduled by the BS to transmit in the same slot for all frames ($X_n^t = 1,\ n = 1, \dots, N,\ t = 1, 2, \dots$), the other slots being used by other devices, or for other usage (control information, downlink data transmission...).

The PRBs used for transmission are, however, not necessary the same for all devices, and vary across the frames. The PRBs allocated in frame t to the devices are represented by the binary matrix $\boldsymbol{\delta}^t \in \{0,1\}^{N \times N_{\text{PRB}}}$, with $\delta_{n,f}^t = 1$ if device n is allocated to PRB f and 0 otherwise. Furthermore, we define by $\delta_n(e)$ the function indicating if RE e belongs to a PRB and a slot that have been allocated to device n.

We consider a MU-MIMO system where the base station is equipped with N_r receive antennas and the devices all have N_t transmit antennas.

5.2.2 Model at RE Resolution

For a given RE e, the channel matrix of device n is denoted by: $\mathbf{h}_{n,\varphi,\tau} = \mathbf{h}_{n,e} \in \mathbb{C}^{N_r \times N_t}$. Symbols of device n are drawn from a unit-energy constellation $\mathcal{D}_n$ and are transmitted on one or several layers. Let $\boldsymbol{\nu}_e = \begin{bmatrix} \nu_{1,e} & \dots & \nu_{N,e} \end{bmatrix}^{\mathsf{T}} \in \mathbb{N}^N\ \nu_{n,e} \leq N_t\ \forall n, e$, with $\nu_{n,e}$ the number of layers used by device n on e. Symbols are denoted by $\mathbf{d}_{n,e} = \begin{bmatrix} d_{n,e,1} & \dots & d_{n,e,\nu_{n,e}} \end{bmatrix}^{\mathsf{T}} \in \mathbb{C}^{\nu_{n,e}}$. The precoder of device n on RE e is denoted by $\mathbf{q}_{n,e} \in \mathbb{C}^{N_t \times \nu_{n,e}}$, it defines how symbols are spread on the antennas and describe which transmit antennas are used by each device. Symbol $d_{n,e,j}$ is sent at power $p_{n,e,j} = p_{n,\varphi,\tau,j} \in \mathbb{R}_+$, under constraint of a maximal total transmit power per frame:

$$\sum_{\varphi=1}^{N_{\text{SC}} N_{\text{PRB}}} \sum_{\tau=(t-1)N_{\text{symb,tot}}+1}^{tN_{\text{symb,tot}}} \sum_{j=1}^{\nu_{n,e}} \delta_n(e) p_{n,\varphi,\tau,j} \leq P_{\max},\ \forall e,\ \forall n. \tag{5.1}$$

Let $\mathbf{p}_{n,e} \in \mathbb{R}_+^{\nu_{n,e}}$ be the vector of transmit powers on the different layers of device n on RE e. The received signal at the BS on e $\mathbf{y}_e \in \mathbb{C}^{N_r}$ is:

$$\mathbf{y}_e = \sum_{n=1}^{N} \delta_n(e) \mathbf{h}_{n,e} \mathbf{q}_{n,e} \sqrt{\text{diag}(\mathbf{p}_{n,e})} \mathbf{d}_{n,e} + \mathbf{w}_e, \tag{5.2}$$

where $\mathbf{w}_e \sim \mathcal{CN}(0, \sigma^2 I_{N_r})$ is the additive Gaussian noise, which can be whitened if necessary. By defining $\bar{\mathbf{h}}_e^t \in \mathbb{C}^{N_r \times \nu_{\text{tot},e}}$, the effective channel obtained by stacking the matrices $\left\{ \mathbf{h}_{n,e} \mathbf{q}_{n,e},\ \forall n = 1, \dots, N \text{ such that : } \delta_n(e) = 1 \right\}$ rowise, and $\bar{\mathbf{d}}_e = \begin{bmatrix} \sqrt{\text{diag}(\mathbf{p}_{1,e})} \mathbf{d}_{1,e} \\ \vdots \\ \sqrt{\text{diag}(\mathbf{p}_{N,e})} \mathbf{d}_{N,e} \end{bmatrix} \in \mathbb{C}^{\nu_{\text{tot},e}}$, with $\nu_{\text{tot},e} = \sum_{n=1}^{N} \delta_n(e) \nu_{n,e}$, the vector of vertically stacked symbol vectors sent by all devices in RE e. We can redefine (5.2) as:

$$\mathbf{y}_e = \bar{\mathbf{h}}_e \bar{\mathbf{d}}_e + \mathbf{w}_e. \tag{5.3}$$

We consider that the channel is known at the receiver, and we assume the BS use a ZF receiver to recover data symbols $\bar{\mathbf{d}}_e$, as described in (2.27):

$$(\bar{\mathbf{h}}_e^{\mathsf{h}}\bar{\mathbf{h}}_e)^{-1}\mathbf{h}_e^{\mathsf{H}}\mathbf{y}_e = (\bar{\mathbf{h}}_e^{\mathsf{H}}\bar{\mathbf{h}}_e)^{-1}\mathbf{h}_e^{\mathsf{H}}\left(\bar{\mathbf{h}}_e\bar{\mathbf{d}}_e + \mathbf{w}_e\right) \tag{5.4}$$

$$= \bar{\mathbf{d}}_e + (\bar{\mathbf{h}}_e^{\mathsf{H}}\bar{\mathbf{h}}_e)^{-1}\mathbf{h}_e^{\mathsf{H}}\mathbf{w}_e \tag{5.5}$$

$$= \bar{\mathbf{d}}_e + \tilde{\mathbf{w}}_e, \tag{5.6}$$

where the covariance matrix of the noise $\tilde{\mathbf{w}}_e$ is now:

$$(\bar{\mathbf{h}}_e^{\mathsf{H}}\bar{\mathbf{h}}_e)^{-1}\mathbf{h}_e^{\mathsf{H}}\left(\left(\bar{\mathbf{h}}_e^{\mathsf{H}}\bar{\mathbf{h}}_e\right)^{-1}\mathbf{h}_e^{\mathsf{h}}\right)^{\mathsf{H}} = (\bar{\mathbf{h}}_e^{\mathsf{H}}\bar{\mathbf{h}}_e)^{-1}\mathbf{h}_e^{\mathsf{H}}\mathbf{h}_e(\bar{\mathbf{h}}_e^{\mathsf{H}}\bar{\mathbf{h}}_e)^{-1^{\mathsf{H}}}$$
$$= (\bar{\mathbf{h}}_e^{\mathsf{H}}\bar{\mathbf{h}}_e)^{-1}.$$

Remark (ZF receiver)**.** The choice of a ZF receiver might seems surprising considering that, as discussed in Chapter 2.2.7, the ZF receiver will likely yield poor performance results whenever the matrix $\bar{\mathbf{h}}_e$ is ill-conditioned. However, because the devices follows the transmission parameters dictated by the BS, we can safely assume that no devices with channels leading to an ill-conditioned matrix are co-scheduled together in the same slot.

Remark (Communication of transmission parameters)**.** The transmission parameters have to be sent to the devices over a downlink channel. The transmit slot that co-schedules the device is obtained through the RACH procedure. The other parameters could be transmitted with the feedback of the BS. To ensure the smallest downlink data transmission possible, the transmit power should be discretized and limited to certain predefined values. In this work, we choose to neglect the possible consequences of such a discretization and assume that the BS can use as much downlink as necessary.

We are particularly interested in computing, for a given PRB allocation $\boldsymbol{\delta}^t$, power allocation $\mathbf{p}_{n,e}$, $\forall n$, and layer allocation $\boldsymbol{\nu}^t$ the rate $r_{n,e}$ of each device in e. We can define first the SINR of device n for layer j on $e = (\varphi, \tau)$:

$$\mathrm{SINR}_{n,e,j}(\bar{\mathbf{h}}_e, p_{n,e,j}) = \begin{cases} \frac{p_{n,e,j}}{\left(\bar{\mathbf{h}}_e^{\mathsf{H}}\bar{\mathbf{h}}_e\right)^{-1}_{l,l}} & \delta_n(e) = 1 \\ 0 & \delta_n(e) = 0 \end{cases}, \tag{5.7}$$

where l is the index of the symbol transmitted by device n on layer j in $\bar{\mathbf{d}}_e$: $l = j + \sum_{m=1}^{n}\delta_m(e)\nu_m$. The transmission occurs at rate:

$$r_{n,e,j}(\bar{\mathbf{h}}_e, p_{n,e,j}) = \log_2(1 + \mathrm{SINR}_{n,e,j}(\bar{\mathbf{h}}_e)). \tag{5.8}$$

Giving a total rate per RE of:

$$r_{n,e}(\bar{\mathbf{h}}_e, \mathbf{p}_{n,e}) = \sum_{j=1}^{\nu_{n,e}} r_{n,e,j}(\bar{\mathbf{h}}_e). \tag{5.9}$$

5.2.3 Model at PRB Resolution

Typically, channels are considered to be constant across the subcarriers of a PRB and through the OFDM symbols of the frame. Furthermore, for practical reasons[2], it is preferable to define transmission parameters that are only dependent on the PRB and the frame used instead of the RE. To emphasize this point, we rewrite (5.2) as a function of the PRB and the frame indices[3], assuming that RE e belongs to PRB f and frame t:

$$\mathbf{y}_e = \sum_{n=1}^{N} \delta_{n,f}^t \mathbf{H}_{n,f}^t \mathbf{Q}_{n,f}^t \sqrt{\mathrm{diag}(\mathbf{P}_{n,f}^t)} \mathbf{d}_{n,e} + \mathbf{w}_e \tag{5.10}$$

$$= \bar{\mathbf{H}}_f^t \bar{\mathbf{d}}_e + \mathbf{w}_e, \tag{5.11}$$

where $\bar{\mathbf{H}}_f^t = \bar{\mathbf{h}}_e$, and $\bar{\mathbf{d}}_e = \begin{bmatrix} \sqrt{\mathrm{diag}(\mathbf{P}_{1,f}^t)}\mathbf{d}_{1,e} \\ \vdots \\ \sqrt{\mathrm{diag}(\mathbf{P}_{N,f}^t)}\mathbf{d}_{N,e} \end{bmatrix} \in \mathbb{C}^{\nu_{\mathrm{tot}}}$. We can now express the rate in a given PRB f and a given frame t as the sum-rate of all the $N_{\mathrm{SC}} \times L$ REs:

$$r_{n,f}^t(\bar{\mathbf{H}}_f^t, \mathbf{P}_{n,f}^t) = N_{\mathrm{SC}} L r_{n,e}(\bar{\mathbf{H}}_f^t, \mathbf{P}_{n,f}^t), \tag{5.12}$$

where we multiply by L as devices transmit once per frame, in the prescribed slot. Finally, the total rate of device n in frame t can be expressed as:

$$r_n^t(\bar{\mathbf{H}}_1^t, \ldots, \bar{\mathbf{H}}_{N_{\mathrm{PRB}}}^t, \mathbf{P}_{n,1}^t \ldots \mathbf{P}_{n,N_{\mathrm{PRB}}}^t) = \sum_{f=1}^{N_{\mathrm{PRB}}} \delta_{n,f}^t N_{\mathrm{SC}} L r_{n,e}(\bar{\mathbf{H}}_f^t, \mathbf{P}_{n,f}^t). \tag{5.13}$$

5.3 Joint Power, PRB and Layer Optimization

With equations (5.8-5.13) we can see the relevant parameters that the BS can tune in order to optimize the rates in a given frame t are:

- $P_{n,f}^t,\ \forall n, f$, the transmit power of each device in each PRB,
- $\delta_{n,f}^t,\ \forall n, f$, the PRBs in which each device are transmitting,
- $\nu_n^t,\ \forall n$, the number of layers used by each device.

The last two parameters appear indirectly in the definition of $\bar{\mathbf{H}}_f^t$. We thus propose to formalize an optimization problem to jointly optimize these three parameters with the aim of maximizing the total rates of the devices. To ensure fairness between devices and avoid favoring devices with good channel conditions, we seek to find the parameters that maximize the proportional fair rate

[2] Mainly, maintaining a low peak to power average ratio (PAPR).

[3] To maintain consistency with the previous chapters, the frame index t is written as a superscript.

(described in Chapter 2.3.1) under constraints of power and rate:

$$\max_{\substack{\mathbf{P}^t \in \mathbb{R}^{N \times N_{\mathrm{PRB}} \times N_t} \\ \boldsymbol{\delta}^t \in \{0,1\}^{N \times N_{\mathrm{PRB}}} \\ \boldsymbol{\nu}^t \in \mathbb{N}^N}} \sum_{n=1}^{N_u} \log\left(r_n^t(\bar{\mathbf{H}}_1^t, \ldots, \bar{\mathbf{H}}_{N_{\mathrm{PRB}}}^t, \mathbf{P}_{n,1}^t \ldots \mathbf{P}_{n,N_{\mathrm{PRB}}}^t)\right) \tag{5.14a}$$

$$\text{Subject to: } \sum_{f=1}^{N_{\mathrm{PRB}}} \delta_{n,f}^t \sum_{j=1}^{\nu_n^t} N_{\mathrm{SC}} L P_{n,f,j}^t \leq P_{\max}, \qquad \forall n \tag{5.14b}$$

$$r_{\min} \leq \frac{\sum_{j=1}^{\nu_n^t} \sum_{f=1}^{N_{\mathrm{PRB}}} r_{n,f,j}^t}{\nu_n^t \sum_{f=1}^{N_{\mathrm{PRB}}} \delta_{n,f}^t}, \qquad \forall n \tag{5.14c}$$

$$\delta_{n,f}^t r_{\min} \leq \delta_{n,f}^t r_{n,f,j}^t \leq r_{\max}, \qquad \forall n, f, j \tag{5.14d}$$

$$N_{\mathrm{PRB,min}} \leq \sum_{f=1}^{N_{\mathrm{PRB}}} \delta_{n,f}^t, \qquad \forall n \tag{5.14e}$$

$$\nu_n^t \leq N_t \qquad \forall n, \tag{5.14f}$$

where $\boldsymbol{\delta}^t$ and $\boldsymbol{\nu}^t$ appear in $r_n^t(\cdot)$ through $\bar{\mathbf{H}}_f^t$.

Remark. A meaningful additional constraint would be to impose the allocated PRBs to be contiguous as it would help the waveform to maintain a low PAPR. These types of practical details are out of the scope of this thesis and are thus not considered.

This problem in not convex due to the constraint (5.14c) and contains a mix of integer $(\boldsymbol{\delta}^t, \boldsymbol{\nu}^t)$ and continuous variables $(\mathbf{P}^t)$. For this type of problem, MINLP solvers (like Bonmine [11] or Baron [93]) can be used, but they are slow and/or unlikely to give the optimal solution.

5.4 Computing the Allocation: Practical Considerations

5.4.1 An Alternative Optimization Problem

An alternative is to fix $\delta_{n,f}^t = 1 \ \forall n, f$, $\nu_n^t = N_t, \ \forall n$ and solve a problem similar to (5.14) but only to optimize the power:

$$\max_{\mathbf{P}^t \in \mathbb{R}^{N \times N_{\mathrm{PRB}} \times N_t}} \sum_{i=1}^{N_u} \log\left(r_n^t(\bar{\mathbf{H}}_1^t, \ldots, \bar{\mathbf{H}}_{N_{\mathrm{PRB}}}^t, \mathbf{P}_{n,1}^t \ldots \mathbf{P}_{n,N_{\mathrm{PRB}}}^t)\right) \tag{5.15a}$$

$$\text{Subject to: } \sum_{f=1}^{N_{\mathrm{PRB}}} \delta_{n,f}^t \sum_{j=1}^{\nu_n^t} N_{\mathrm{SC}} L P_{n,f,j}^t \leq P_{\max}, \qquad \forall n \tag{5.15b}$$

$$0 \leq \delta_{n,f}^t r_{n,f,j}^t \leq r_{\max}, \qquad \forall n, f, j. \tag{5.15c}$$

Note that (5.15c) is a relaxed version of (5.14d) imposing no minimal rate, and thus no minimal power per PRB. The problem is now convex and can be solved using mathematical solvers like IPOPT [107]. Heuristics are then used to find the number of PRBs and the number of layers used. The optimization process is found in Alg. 11, where OPT$(\cdot)$ is the function solving (5.15) and works

as follows:

- Assuming that all devices are transmitting in all PRBs and all layers, a first power allocation is found (Steps 2-5).
- For each device, the PRBs having a rate below $r_{\min}$ are discarded (Step 8).
- The number of layers is found in Steps 11-16; starting from a single layer, if adding another layer improves the total rate and maintains an average rate per PRB and layers above $r_{\min}$, then the layer is used.
- If the resulting PRB allocation does not satisfy 5.14e, then the $N_{\text{PRB,min}}$ best PRBs are used.
- Finally, the solver is run once again with the PRB allocation and layer allocation found in the previous steps to find a new power.

Algorithm 11: Algorithm to find the power, PRB and layer allocation.

1 **Function** `parameter_optimization(`$\mathbf{H}^t_{1,f}$,...,$\mathbf{H}^t_{N,f}$`)`
2 $\quad\boldsymbol{\delta}^t_f \leftarrow \mathbf{1}$; `// Matrix of ones`
3 $\quad\boldsymbol{\nu}^t \leftarrow \begin{bmatrix} N_t & \dots & N_t \end{bmatrix}$; `// Constant vector of` N_t
4 $\quad\bar{\mathbf{H}}^t_f \leftarrow \begin{bmatrix} \mathbf{H}^t_{1,f}\mathbf{Q}^t_{\mathbf{1,f}} & \dots & \mathbf{H}^t_{N,f}\mathbf{Q}^t_{N,f} \end{bmatrix}$; `// stacking` $\{\mathbf{H}^t_{n,f}\mathbf{Q}^t_{n,f}\}$ `column-wise`
5 $\quad\mathbf{P}^t \leftarrow \text{OPT}(\bar{\mathbf{H}}^t_1, \dots, \bar{\mathbf{H}}^t_{N_{\text{PRB}}})$
6 $\quad$**for** $n \in N$ **do**
7 $\quad\quad$Compute $r^t_{n,f,j}\ \forall f,\ j$; `// According to (5.8)`
8 $\quad\quad\delta^t_{n,f,j} \leftarrow \mathbb{1}\{r^t_{n,f,j} > r_{\min}\}\ \forall f, j$
9 $\quad\quad r_b \leftarrow 0$
10 $\quad\quad$**for** $j \in 1, 2, \dots N_t$ **do**
11 $\quad\quad\quad\delta^t_{n,f} \leftarrow \prod_{i=1}^{j} \delta^t_{n,f,j}$
12 $\quad\quad\quad r \leftarrow \sum_{f=1}^{N_{\text{PRB}}} \delta^t_{n,f} \sum_{i=1}^{j} r^t_{n,f,j}$; `// Compute total rate`
13 $\quad\quad\quad r_m = \frac{r}{j\sum_{f=1}^{N_{\text{PRB}}} \delta^t_{n,f}}$; `// Compute mean rate`
14 $\quad\quad\quad$**if** $r > r_b$ and $> r_{\text{mean}} > r_{\min}$ **then**
15 $\quad\quad\quad\quad r_b \leftarrow r$
16 $\quad\quad\quad\quad\nu^t_n \leftarrow j$
17 $\quad\quad\quad$**end**
18 $\quad\quad$**end**
19 $\quad\quad$**if** $\sum_{f=1}^{N_{\text{PRB}}} \delta^t_{n,f} < N_{\text{PRB,min}}$ **then**
20 $\quad\quad\quad\nu^t_n \leftarrow 1$
21 $\quad\quad\quad\boldsymbol{\delta}^t_n \leftarrow$ Find the $N_{\text{PRB,min}}$ PRBs with the biggest rates $r^t_{n,f,1}$
22 $\quad\quad$**end**
23 $\quad\quad$Reconstruct $\bar{\mathbf{H}}^t_f$ with the new $\boldsymbol{\delta}^t$ and $\boldsymbol{\nu}^t\ \forall f$.
24 $\quad\quad\mathbf{P}^t \leftarrow \text{OPT}(\bar{\mathbf{H}}^t_1, \dots, \bar{\mathbf{H}}^t_{N_{\text{PRB}}})$
25 $\quad\quad$**return** $\mathbf{P}^t, \boldsymbol{\delta}^t, \boldsymbol{\nu}^t$
26 $\quad$**end**

5.4.2 Neural Network to Learn (5.15)

For a given CSI, the BS need to run Alg. 11 to obtain an allocation. In an ideal case, the BS receives a CSI at each frame, meaning that the BS has approximately 10 ms to compute the transmission parameters and send them to the devices, hoping the channels will not change too much in the subsequent frame. Even if Alg. 11 makes use of convex problems to find the allocation, its running time is likely to be longer than 10 ms. Furthermore, its running time is not constant for all inputs, due to the IPOPT solver that runs until a certain precision is met.

To circumvent these issues, we propose to use a neural network to learn the input/output relationship of Alg. 11 and act as a function approximator. This approach is interesting because, once trained, the running of the NN will consist in a series of matrix multiplication and activation layers providing approximately the same running time regardless of the input. Furthermore, with the help of the ZF receiver, the base station can neglect the interference between the devices. This means that for each devices, which PRBs are used, the transmit power on each used PRBs, and the number of layers, solely depends on the coefficients of the ZF receiver, $\lambda_{f,l} = \frac{1}{\left(\bar{H}_f^{t\mathsf{H}} \bar{H}_f^t\right)^{-\mathrm{T}}_{l,l}}$, l being the indices corresponding to the symbols associated to the device.

The input of the neural network should thus be the coefficients $\boldsymbol{\lambda} \in \mathbb{C}^{N_{\mathrm{PRB}} N_t}$ of a single device.

Because we seek to learn three different outputs of different type, we propose to learn three different neural networks:

$$
\begin{aligned}
v_1 &: \mathbb{R}^{N_{\mathrm{PRB}} N_t} \to \mathbb{R}^{N_{\mathrm{PRB}}}, \\
v_2 &: \mathbb{R}^{N_{\mathrm{PRB}} N_t} \to \{0,1\}^{N_{\mathrm{PRB}}}, \\
v_3 &: \mathbb{R}^{N_{\mathrm{PRB}} N_t} \to \{0,1\}^{N_t}.
\end{aligned}
$$

v_1 performs a regression task to find the optimal transmit power, while v_2 and v_3 perform classification tasks to find (respectively) the PRB allocation and the number of layers to use (by summing over the output of v_3).

Network Structures

The neural networks consist of 8 layers (with 6 hidden), the number of neurons at each layer of the neural network are obtained by trial and error and are:

- $v_1 : N_{\mathrm{PRB}} \times N_t, 512, 256, 256, 256, 256, 128, N_{\mathrm{PRB}}$ with ReLU activation functions and is evaluated with a mean square error (MSE) loss (regression task).
- $v_2 : N_{\mathrm{PRB}} \times N_t, 512, 256, 256, 256, 256, 128, N_{\mathrm{PRB}}$ with ReLU activation functions except for the last layer that uses a sigmoid activation to ensure the output is binary (or very close to binary), and evaluated with a binary cross-entropy (BCE) loss (classification task).
- $v_3 : N_{\mathrm{PRB}} \times N_t, 512, 256, 256, 256, 256, 128, N_t$ with ReLU activation functions except for the last layer that uses a sigmoid activation for the same reasons as v_2 and also uses a BCE loss

(classification task).

Training

To perform the training, we constitute a dataset of 80 000 $\boldsymbol{\lambda} \in \mathbb{C}^{N_{\text{PRB}}N_t}$. Each $\boldsymbol{\lambda}$ is obtained by simulating the channel matrices of the 8 devices on each PRB, then the matrices $\left(\bar{H}_f^{t\mathsf{H}}\bar{H}_f^t\right)^{-1}$ are computed for all PRBs. For each PRB, 8 different $\boldsymbol{\lambda}_f \in \mathbb{C}^{N_t}$ can be obtained by forming groups of N_t elements of the coefficients: $\left[\frac{1}{(\bar{H}_f^{t\mathsf{H}}\bar{H}_f^t)^{-1}_{i,i}} \quad \cdots \quad \frac{1}{(\bar{H}_f^{t\mathsf{H}}\bar{H}_f^t)^{-1}_{i+N_t,i+N_t}}\right]$ with $i = \{1, Nt+1, 2N_t+1, \ldots, 8N_t+1\}$. Afterwards, the 8 different groups of each PRB can be merged together to make 8 different $\boldsymbol{\lambda} = \left[\boldsymbol{\lambda}_1 \quad \ldots \quad \boldsymbol{\lambda}_{N_{\text{PRB}}]}\right] \in \mathbb{C}^{N_{\text{PRB}}N_t}$. In other words, because the ZF receiver makes all devices independent, the simulation of a network of 8 devices gives 8 samples to use for the training. The output consists of three vectors:

- a vector of size N_{PRB} corresponding to the power allocation,
- a binary vector of size N_{PRB} corresponding to the PRB allocation,
- and a scalar corresponding to the number of layers used by the device.

The dataset is split into a training dataset containing 80% of the full dataset and a test dataset containing 20%. The dataset is trained using mini-batches of size 80.

For the PRB allocation, the output dataset shows great imbalance. Indeed, almost $\approx 70\%$ of the devices transmit on all the PRBs. Such an imbalance encourages the neural network to learn to almost always predict an allocation to all PRBs. To avoid this effect and encourage the neural network to also predict different allocations, we augment the dataset in the following way:

1. Copy 3 times each input matrix that do not lead to allocation of 24 PRBs.
2. Perturb the input matrix of copy with some Gaussian noise $\mathcal{N}(0, 0.001)$.
3. Assign to each copy the same PRBs allocation as the original sample.

Doing so increases our dataset from 80 000 to 157 790 samples for v_2.

Similarly to the PRB allocation, the dataset of the layer selection is highly imbalanced, with more than 93% of the samples that allocate 4 layers. The same data augmentation trick is used again but with 10 copies of each sample that does not allocate 4 layers. This makes the dataset of v_3 of size 137 620. The optimizer used is Adam [63] with a learning rate of 0.0001.

5.5 Baselines

5.5.1 Power Control in Cellular Network

In 4G and 5G standards, the transmit power of any device n is computed according to the following power control formula [34, Section 7]:

$$P_{\mathrm{tr},n} = \min\{P_{\max}, P_0 + 10\log(N_{\mathrm{PRB},n}) + \alpha PL + CL\}, \tag{5.16}$$

where:

- $P_{\max}$ is the maximum transmit power, expressed in dBm,
- $N_{\mathrm{PRB},n}$ is the number of OFDM resource blocks assigned to device n,
- P_0 is the target received power at the BS, expressed in dBm,
- PL is the pathloss compensation term
- $\alpha \in [0, 1]$ is a coefficient controlling the inter-cell interference, if $\alpha = 1$ then the pathloss is fully compensated but transmission by devices at the edge of the cell will likely generate interference in the neighboring cells,
- CL is the closed loop power control (CPLC) parameter, allowing to refine the transmit power. This parameter can be sent by the BS during its feedback to the device, meaning that power can be adapted at short time intervals. However, to limit the overhead induced by sending this piece of information, CL can take only four values: $CL \in \{-1, 0, 1, 3\}$ dBm.

P_0 and α are known as the open loop power control (OLPC) parameters, they can be configured either statically or dynamically [70] but have to be the same for all devices in the cell. Several issues are associated to these two parameters. Firstly, as was investigated in [43], some sets of parameters might yield very poor throughput, in particular when considering a network of several cells. Secondly, these parameters can be changed only using the network layer RRC protocol, introducing a non-negligible latency. Indeed, they are initially received by the device during the "Connection Setup" of the RACH procedure, as shown in Step 4 of Fig. 2.5a. If they change after the initial setup of a device, a RRC ConnectionReconfiguration procedure can be triggered by the BS, but this induces at least communication of two messages as described in [36, Section 5.3.5.3].

Parameter	Value
N_{simul}	2 000
N	8
L	14
N_{PRB}	24
N_{SC}	12
N_r	64
$P_{\max}$	10 dBm
α	1 (no intra cell interference)
P_0	$\{-75, -85, -90, -100, -110, -120\}$dBm
CL	0 dBm
$N_{\text{PRB,min}}$	1
$r_{\min}$	0.2
$r_{\max}$	8

Table 5.1: Different simulation parameters.

5.5.2 PRB Scheduling

The network has a maximum of N_{PRB} PRBs available for usage, the number of allocated PRBs to each device is computed according to the formula:

$$N_{\text{PRB},n} = \min\left\{N_{\text{PRB}}, \max\left\{N_{\text{PRB,min}}, \lfloor 10^{\frac{P_{\max}-P_0-\alpha PL_n}{10}} \rfloor\right\}\right\}. \tag{5.17}$$

In other words, devices transmit on all PRBs if their transmit power, computed with (5.16), is lower than $P_{\max}$. Otherwise, the number of PRBs is the maximum number of PRBs that ensures transmission at a power lower than $P_{\max}$. Finally, to avoid having devices without allocated PRBs, there is a minimum number of PRBs, $N_{\text{PRB,min}}$.

Whenever the number of PRBs allocated to a device is less that N_{PRB}, the selected PRBs are the ones with the best channel gain.

5.5.3 Layer Selection

The number of layers used by each device depends on the ratio between the biggest and smallest eigenvalues of each expected channel matrices. Let $\mu_1 \dots \mu_{N_t}$ be the ordered eigenvalues of $\mathbb{E}_{f,t}[{\mathbf{H}^t_{n,f}}^{\mathsf{H}}\mathbf{H}^t_{n,f}] \in \mathbb{C}^{N_t \times N_t}$. To ensure the relative quality of each used layers, we compute the number of layers used by device n as : $\nu_n = \max\{i : \frac{\mu_i}{\mu_1} \geq 0.5\}$.

5.6 Simulation Results

In this section we show plots for the resulting rates, power, number of PRBs and layers used by the different methods.

Table 5.1 summarizes the main parameters of our simulations.

We perform comparison between seven different optimization methods:

- Naive scheme: all the devices transmit at 10 dBm of total power, using all PRBs, and on all layers.
- Optimization of the transmit power, PRBs and layers (named "Solver" in Tab. 5.2) using Alg. 11.
- Optimization using the neural network described in Section 5.4.2 trained to learn Alg. 11.
- Baselines optimization using the baseline described in Section 5.5 for $P_0 = \{-75, -85, -90, -100\}$ dBm, $\alpha = 1$, and $CL = 0$ dBm

For all methods, the minimum number of used PRB is $N_{\text{PRB,min}} = 1$, and the minimum and maximum rates per RE are respectively $r_{\min} = 0.2$ and $r_{\max}$ =8 bit/RE.

We compute the different metrics for networks consisting of $N = 8$ devices each equipped with $N_t = 4$ antennas, and a single BS with $N_r = 64$ antennas. The channels consist in pathloss and fading: the pathloss of device n on PRB f is defined, in dB, as the sum of a pathloss per device and a pathloss per PRB $PL_n + PL_f$. The coefficient i, j of the channel matrix on PRB f of device n are:

$$H_{n,f,i,j} = f_{n,f,i,j} \sqrt{10^{\frac{PL_n + PL_f}{10}}} \tag{5.18}$$

with $f_{n,f,i,j} \sim \mathcal{CN}(0, 1)$, the Rayleigh fading, and $PL_n \sim \mathcal{N}(-105, 20^2)$ and $PL_f \sim \mathcal{N}(-10, 10^2)$. The AWGN noise is the thermal noise drawn form $\mathcal{N}(0, Tk_BB)$ with $T = 290$ K, the ambient temperature, $k_B = 1.3810^{-23}$, the Boltzmann constant, and $B = N_{\text{SC}}S_p = 12 \times 30\,000 = 360\,000$ Hz, the bandwidth of a PRB in the subcarrier spacing of $S_p = 30$ kHz. Evaluation of the different methods is performed over $N_{\text{simul}} = 2\,000$ different channel realizations. In Fig. 5.1-5.5, we present the cumulative distribution functions (CDFs) of the $N \times N_{\text{simul}} = 16\,000$ values of the different quantities of interest and in Tab. 5.2 their (geometric or arithmetic) mean.

The first metric we are interested in is the rates, represented in Fig. 5.1 by the average rate per RE, defined as $\frac{r_n}{N_t L N_{\text{SC}} N_{\text{PRB}}}$. Regarding the baselines, we can observe that decreasing the target received power from -75 dBm to -100 dBm generally increases the performance. Further reducing to -110 dBm and -120 dBm reduces the number of devices that have a low rate but cannot offer a rate higher than 5.3 and 2.4 bit/RE (respectively). As a result, the probability that a device does not the reach the minimum rate (Fig. 5.2, and Fig. 5.1, zoom) decreases. The outage probability is defined as the number of samples below $r_{\min}$ normalized by the number of devices: $\frac{\Pr(r<r_{\min})}{N}$.

The rates obtained by the solver-based methods (green and orange) and the naive scheme provide rates that are significantly higher than the baselines. This is reflected in the geometric mean (Tab. 5.2, first row), where the geometric mean of the solver-based method is 2 bit/RE higher than the best performing baseline (-110 dBm). Furthermore, by using the solver-based method or the naive scheme, approximately 30% of the devices have a rate at (or close to) $r_{\max} = 8$ contrarily to less than 2% for the baselines at $P_0 \leq -90$dBm. The naive scheme, consisting in transmitting at full power on all

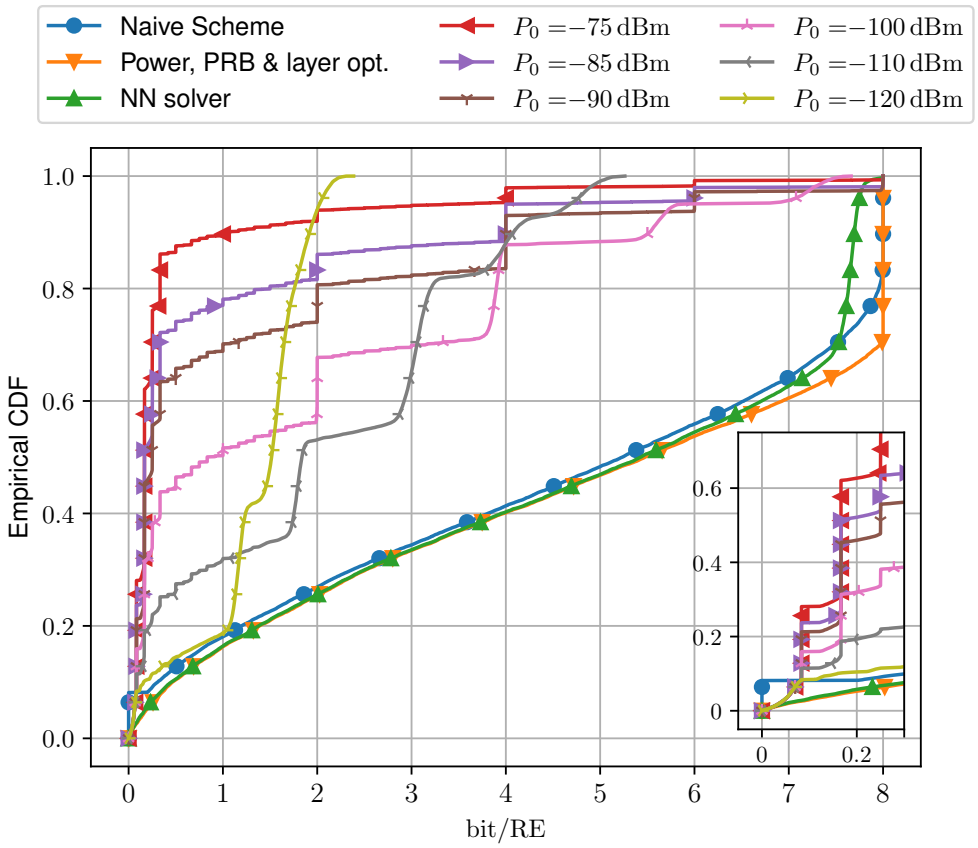


Figure 5.1: CDF of the average rate per RE.

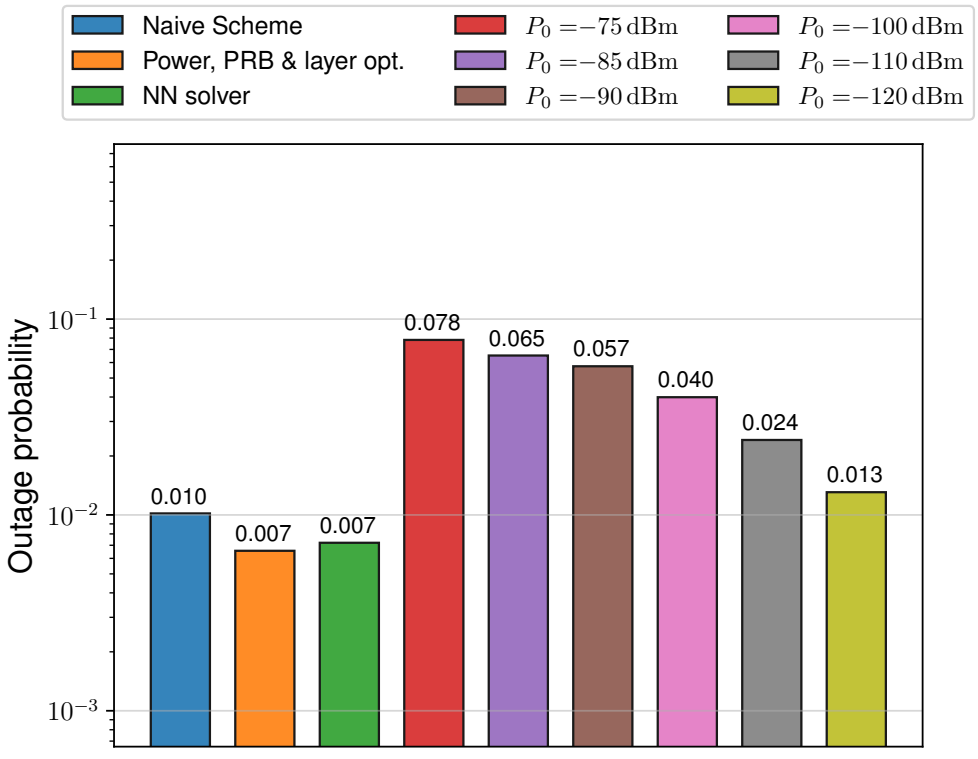


Figure 5.2: Probability that a device is outed of the network.

	Naive	Solver	NN solver	Baselines: P_0 (dBm)					
				-75	-85	-90	-100	-110	-120
Rates (bit/RE)	2.05	3.37	3.25	0.22	0.35	0.46	0.83	1.24	1.04
Power (dBm)	10	6.3	6.4	9.4	8.3	7.4	4.4	-0.4	-7.1
PRBs	24	19.2	19	3	5.6	7.3	11.5	15.9	19.7

Table 5.2: Mean values of the different metrics. The geometric mean is used for the rates and the arithmetic mean is used for the power and the number of PRBs.

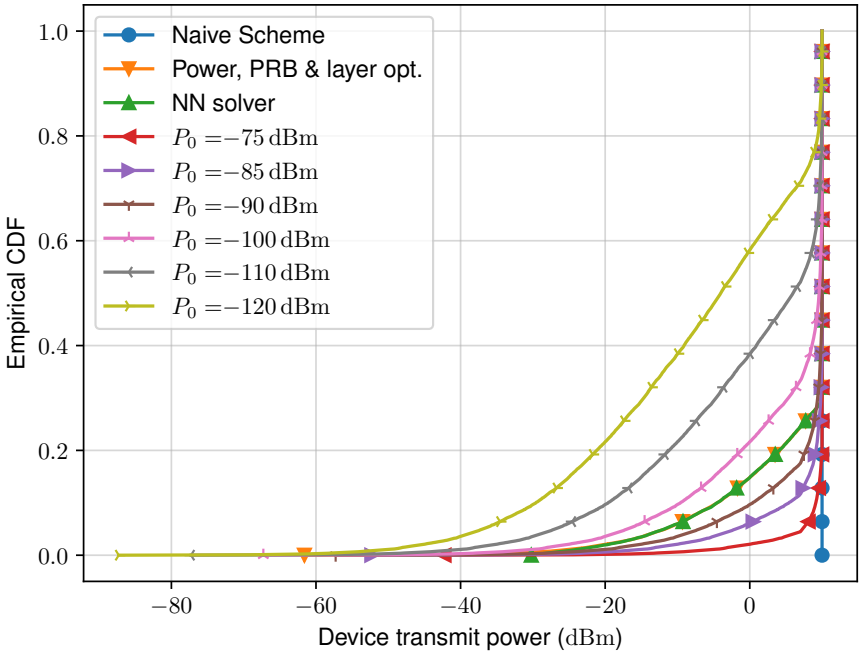


Figure 5.3: CDF of the total transmit power per device.

PRBs and all layers, closely follows the solver-based method and seems interesting in terms of rates, but has a higher number of outed devices. As a result, its geometric average (first row of Tab. 5.2) is lower than the one of the solver-based method. Moreover, using the naive scheme in practice will lead to significant inter-cell interference, thus a study considering several cells will strongly disfavor the naive method.

Regarding the transmit power shown in Fig. 5.3, we can observe that solvers' transmit power lies between the baselines at target received power of -100 dBm and -90 dBm. Baselines at -85 dBm and -75 dBm transmit at a higher power than the solver-based methods, but does not induce an increase in rates (purple and red curves of Fig. 5.1).

Due to the compensation of the pathloss and because the transmit power has to be the same on all PRBs, devices easily reach the maximal transmit power, when using the baselines. Thus, they can not transmit on all the PRBs at their disposition (Fig. 5.4). For baselines higher than -100 dBm, devices transmit at high power on few PRBs (the ones with the best channels) whereas baselines below -100 dBm transmit on more PRBs. The solver-based methods, because they can

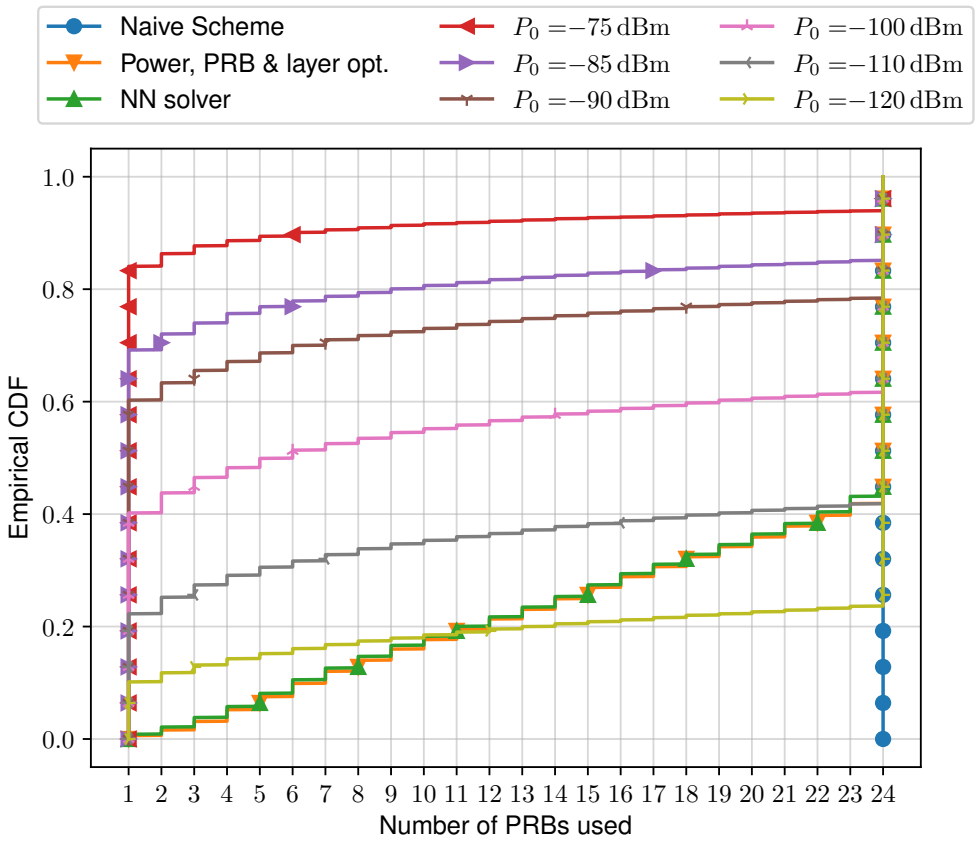


Figure 5.4: CDF of the number of PRBs used.

adapt the transmit power on each PRB depending on their quality, are able to make a better usage of the different PRBs, explaining how they can reach higher data rates with a lower transmit power than some of the baselines.

Regarding the layers used (Fig. 5.5 and Tab. 5.2 third row), we can observe that the solver-based allocations use more layers compared to the baselines.

Finally, because the curve of the NN solver are following closely the curves of the joint optimizer, we can infer that the NN solver successfully learned the input/output relationship of the optimization.

5.7 Conclusion

In this chapter, we proposed a method to increase the data rate (eMBB) of devices in an network. Our key contribution is to jointly consider the optimization of the transmit power, the PRBs and the layers used by the devices, allowing to have parameters adapted to the channel conditions of each device. We first expressed an ideal non-convex MINLP problem that we relaxed to a convex NLP problem (to obtain the power optimization) used in conjunction with an algorithm based on heuristics to compute the PRBs and the layers. Our simulation study shows the benefits of our method compared to the state of the art used in 5G through different quantities of interest. As a result, this work could be an interesting feature to enable higher data rates in 6G networks. Because

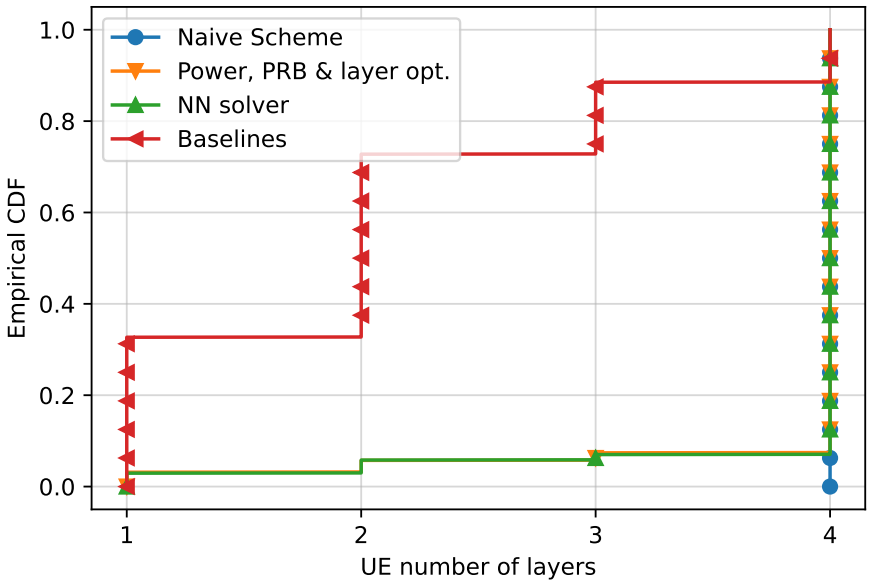


Figure 5.5: CDF of the number of layers used.

our method is based on CSI, the transmission parameters should be recomputed each time the channels change, leading to a high computational overload by the BS and a delay before sending the allocation. To circumvent these drawbacks, we propose to use a neural network to learn the input/output of the method we propose, allowing for a faster and constant time computation of the solution.

In contrast with the state of the art, our method does not impose a target received power common to all devices of the network. As future work, our method could be used in conjunction with a SIC receiver in the power domain at the BS to further increase the data rate.

CHAPTER

6 Conclusion and Future Work

Contents

6.1 Conclusion

In this thesis, we presented different methods to perform uplink resource allocation in a MTC and an eMBB context.

First, with applications to MTC networks, we studied of allocation policies that are functions of the statistics driving the devices' activity. Largely unexplored in the state of the art, such allocation policies relax the ubiquitous assumption that all devices are independent and equally likely to be active, and open the possibility of further improving the throughput of the network. The optimization can be performed in two different ways: off-line, if the underlying activity statistics are known, or online using the knowledge the active devices in each frame. Chapter 3 and Chapter 4 focus on the second case.

In Chapter 3, we proposed a resource allocation algorithm that is optimized via a variant of SGD using importance sampling. The algorithm we propose is particularly useful whenever the user detection is imperfect. Motivated by some practical consideration, we studied different causes of imperfect user detection: being unable to decode a payload packet containing the device identifier, leading to missing some active devices, and a low SNR in the reception of the pilots, leading to confusion between devices. We showed that applying a gradient based algorithm to perform the optimization of the slots without taking into account the possible errors lead to a decrease in the performance of the resulting allocation. The modified version of SGD we proposed has a step size that depends on the likelihood of the received data instead of being constant (or function of the iteration index). If the received activity vector is likely to be error-free (*e.g.* based on side knowledge, or prior information) the step size taken is big, and if the activity vector seems unlikely, then a small step is taken. We provided simulation results showing that our method is able to partially compensate the effects errors can induce in different scenarios inspired from NB-IoT protocol. The main limitation of this work is the reliance on some prior information to determine the step size.

In Chapter 4, we presented, under the assumption of a perfect device identification, a method to address the joint optimization of slot and of power. The primary motivation is to further increase the throughput of a frame slotted network by exploiting the power domain and the channels of the devices. The algorithm we proposed is able to take the correlations of the devices into account to ensure that highly correlated devices are utilizing resources that allow their decoding. In the presented simulations, the devices' correlations can help design an allocation policy that has a high throughput, highlighting what could be called *Statistical Multiple Access*. The effect is particularly noticeable when the communication resources are limited compared to the communication needs of the devices in the network. The algorithm we developed suffers from an exponential number of terms in the computation of the objective (and of the gradient), making it challenging to increase the number of devices without loss of precision and performance.

Then, in Chapter 5, the MTC scenarios are put aside and replaced by an eMBB objective. The throughput maximization is replaced by a proportional fair criterion, optimized through the allocation of PRBs, power, and layers. In the state of the art of 5G networks, the allocation of each

of these resources to the devices is not jointly considered and is independent of the allocation of resources to the other devices. We proposed an ideal optimization problem to optimize the resource allocation based on the channel conditions, and relaxed the assumption that the transmit power of a device should be uniform across its allocated PRBs. To satisfy practical constraints (mainly latency, and near constant execution time regardless of the input), we first proposed a relaxation of the optimization problem, followed by a function approximator based on a neural network to learn the input/output relationship of the relaxed problem. In our simulation study, in terms of rate per RE, our methods (solver-based or neural network-based) significantly outperforms the different baselines at a comparable transmit power, and generally are able to use more PRBs and layers.

The drawbacks of this work are twofold: we are relying on perfect CSI and use a single base station in our analysis. Firstly, in practice, it is highly likely that the base station will have only imperfect CSI, either because the measurements might contain errors, or because they can't be performed on every RE. As a result, it is uncertain how our algorithms will behave whenever the true CSI differs from the ones measured. Secondly, eMBB networks are rarely unicellular, hence we should also investigate how the algorithms we developed could be adapted to take into account the particularities of multicellular networks, in particular how the communication of one cell might interfere with the communication of the other. One possibility is to replace the objective function by another that additionally seek to minimize the interference on the other cells. This could be achieved by using for example, a regularizer term that is a function of the received power at the BSs of the other cells. Another possibility could be to perform the optimization at the macro-cell level and thus compute the transmit parameters of all devices in all neighboring cells at the same time.

We propose hereafter possible extensions of our works.

6.2 Perspectives

6.2.1 Time-dependent Activity Patterns

In Chapter 3 and Chapter 4, the process driving the underlying activity distribution is assumed to be constant throughout the frames. In practice, however, the networks are likely to be more dynamic, mainly for two main possible reasons:

- devices possess a queue where the packets that have not been received yet by the BS are stored, either because they collided and should be retransmitted or because they have not been sent yet,
- the underlying process changes,

The first point has been addressed in [118] where it is assumed that the process driving the activity follows an irreducible Markov chain with a stationary distribution. Indeed, it can be shown, using [65, Chapter 6], that the convergence theorem can be generalized to more complete stochastic processes than the ones present in Appendix A.2. Hence, the convergence guarantees can be extended to devices with a queue, as long as the queuing process follows a Markov process.

The second point however is trickier, given that no stationary distribution exists, the online stochastic approximation algorithms Alg. 5 and Alg. 16 will fail to converge to a local maxima. If the process remains piecewise stationary, then a solution could be to restart the learning process as soon as the process changes and tune the hyperparameters in order to obtain a new allocation policy as fast as possible. The trigger for starting a new optimization could be based on statistical tests. In the case where the process is not piecewise stationary, gradient-based algorithms are not suited and other types of optimization algorithm should be considered. As proposed in [72], approaches inspired from reinforcement learning and multi-arm bandit problems could provide gradient-free resource allocation algorithms for dynamic networks.

6.2.2 Security Concerns: Denial of Sleep Attacks

A key assumption of Chapter 3 and Chapter 4 is that devices will follow the allocation matrix $\mathbf{A}$ and are not malicious. Denial of sleep [12, 13, 39] is a type of attack in an IoT network where some malicious devices are trying to obtain as many resources as possible from the BS to prevent other devices from transmitting. In a GFRA network using FSA with a soft scheduling policy and optimized with algorithms like Alg. 5 or Alg. 10, the allocation attributed to a device is directly linked to how often its pilot is detected in the preamble slot (how often its bit $X = 1$). Thus, malicious devices could pretend to be highly active by sending their pilot in almost all the frames to obtain more resources, but not sending data in the data transmission part (or sending noise). The performance of the network is thus affected. The throughput (3.6) metric fails to capture the presence of malicious devices, and should be replaced by what can be called the "goodput", defined as the sum throughput of the legitimate devices under the interference of the legitimate devices and the malicious ones. Let $\mathcal{B}$ be the set of legitimate devices and $\mathcal{E}$ the set of malicious devices:

$$G(\mathbf{A}; \mathbf{p}) = \sum_{n \in \mathcal{B}} \sum_{k=1}^{K} p_n A_{nk} \prod_{m \in \{\mathcal{B} \cup \mathcal{E}\} \setminus \{n\}} (1 - p_m A_{mk}).$$

Example 6.2.1. Consider the following scenario: a network made of $K = 5$ slots and $N = 20$ devices, the 19 first being legitimate and the 20^{th} being malicious. The activity of the legitimate devices is $\mathbf{X} \sim \text{Ber}(\mathbf{p})$ with $\mathbf{p} \sim \mathcal{U}(0, 0.5)$ fixed. Supposing the activity of the adversarial device is $X_{20} \sim \text{Ber}(0.7)$, the "goodput", is thus the throughput of the 19 useful devices, taking into account the collisions generated by the 20^{th} device (note the sum going to 19 and the product going to 20).

$$G(\mathbf{A}; \mathbf{p}) = \sum_{n=1}^{19} \sum_{k=1}^{5} p_n A_{nk} \prod_{\substack{m=1 \\ m \neq i}}^{20} (1 - p_m A_{mk}). \tag{6.1}$$

In Fig. 6.1, we can see the impact of the adversarial device on the goodput of the network. We compare three scenarios:

1. The throughput of the network of the 19 legitimate devices without the adversarial.

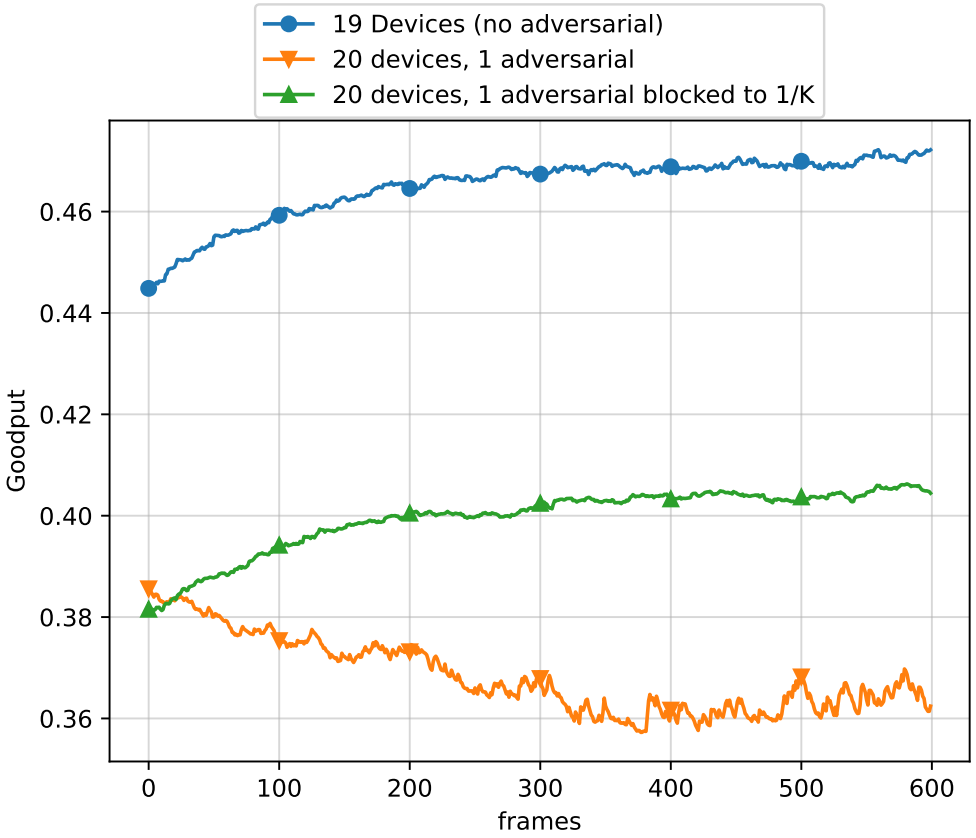


Figure 6.1: Trajectories of the different scenarios over 600 frames.

2. The BS naively uses $T(\mathbf{A};\mathbf{p})$ to optimize the network, using Alg. 5, unknowing of the presence of a malicious device.

3. The BS knows that the 20^{th} device is malicious and tries to limit its impact by forcing it to use a uniform slot selection: the 19 legitimate devices are optimized using Alg. 5 but last row of $\mathbf{A}$ is always $\mathbf{A}_{20} = \begin{bmatrix} \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \end{bmatrix}$.

Because the adversarial device is highly active, in scenario 2, the BS will tend to allocate a slot only for the malicious device, the throughput objective (3.6) will be maximized, but not the goodput (6.1). The resulting goodput is thus significantly affected. In contrast, in scenario 3, if the BS is able to identify the malicious device, it can try to limit its impact by assigning it a uniform slot selection thus avoiding to waste a slot.

Example 6.2.1 highlights the necessity of knowing how to deal with adversarial devices. A uniform allocation has been used to reduce the impact of the adversarial device but is not necessarily the best solution. A better way could be to use the goodput $G(\mathbf{A},\mathbf{X})$ directly in Alg. 5.

Additionally, more extensive scenarios where either the BS or the malicious devices have more flexibility could be interesting. What would be the strategies used by the malicious devices if they know that the BS might detect suspicious devices? What could happen if the network contains selfish devices that do not follow the allocation matrix $\mathbf{A}$ provided by the BS and try to minimize

their own collision probability?

6.2.3 Heterogeneous IRSA

As discussed in Chapter 2.2.5, similarly to FSA, the parameters of IRSA are common to all devices: the degree selection distribution Λ is the same for all devices and, once the degree is selected, the distribution for the selection of the transmit slots is uniform. In Chapter 3 and Chapter 4, we sought to individualize the probability distribution of selecting a particular slot. This idea can be extended to IRSA.

If the number of slots per frames K is fixed, the two-step process used to form the vector $\boldsymbol{\delta}_n$ (described in Alg. 2) can be conceptually replaced by a single-step process. Let $\mathcal{C} = \{0,1\}^K$ be the set of all binary vectors of size K, where a 1 at position k represents transmission in slot k. For a given Λ, we can compute the probability of all the possible codewords $\boldsymbol{\delta} \in \mathcal{C}$ as:

$$\Pr(\boldsymbol{\delta}) = \Lambda_l \frac{1}{\binom{K}{l}} \tag{6.2}$$

$$l = \sum_{k=1}^{K} \delta_k \tag{6.3}$$

Thus, the selection of the codeword $\boldsymbol{\delta}$ can be viewed as immediately drawing a codeword from $\mathcal{C}$, where the probabilities of each of the codewords are specified by $\Pr(\boldsymbol{\delta})$.

This point of view of IRSA, known as random codeword (RC)-IRSA [46] [45, Chapter 8.3] is similar to the setup used in Learn2Mac [24], also described in Chapter 2.2.5. It opens the possibilities to use more generic probability distributions for the codewords, where two codewords with the same number of replicas but different slots might have different probabilities. The main drawback is an optimization space that is much bigger, 2^K instead of K. In [45], the optimization is done with the help of deep-reinforcement learning (RL) algorithms. However, the new probability distributions are still the same for each device, suggesting that individualizing them to each device could potentially lead to further performance improvements, in particular if the activity of the devices are correlated. The optimization would thus become 2^{NK}, raising questions on how to find interesting probability distributions in such big spaces, especially when N and K become large.

6.2.4 Downlink Constellation Learning

The idea of individualizing the transmit parameters to the devices can be transposed to downlink. In contrast with methods like linear precoders (Chapter 2.2.7), superposition coding associated with a SIC decoder [7], or rate splitting multiple access (RSMA) [56], a possibility is to design non-linear constellations for the broadcast channel. In [103], we have shown that online gradient descent algorithms, similar to the ones used in Chapter 4, can be applied to such a task.

Under full knowledge of the CSIT, the BS can generate data symbols, simulate what each device receives, and then compute the mutual information between the BS and each device. The different

mutual information can be aggregated with an aggregation function to obtain a gradient, with respect to the constellation points, can be obtained. Repeating this process until convergence allows to find constellations $\mathcal{D}_n$ tailored to the channels that achieve higher mutual information than traditional methods. The resulting joint constellation (constellations of all devices represented in the space of transmit antennas) is not necessary a linear combination of all the individual constellations.

CHAPITRE

7 Résumé en Français

Sommaire

7.1 Allocation de Ressources sous Incertitude

7.1.1 Introduction

L'un des défis de l'accès multiple à grande échelle est d'assurer des transmissions fiables tout en utilisant efficacement les ressources et en limitant les temps de latence. Une possible solution pour les réseaux cellulaires est GFRA, décrite dans le Chapitre 2.2.4 [20, 28, 67, 94], où les dispositifs actifs transmettent un préambule immédiatement suivi de la transmission de leurs données. Contrairement à la procédure classique utilisant un canal à accès aléatoire (RACH), GFRA n'exige pas de réponse de la part d'une BS avant de transmettre un paquet de données. Par conséquent, le délai d'accès est réduit puisque que seul l'échange de deux paquets est nécessaire, au lieu de quatre dans la procédure RACH classique.

Alors que la procédure RACH permet de fournir aux dispositifs des ressources qui leur sont réservées, ce n'est pas le cas pour GFRA. Les dispositifs doivent donc sélectionner leurs propres ressources, tel que le créneau temporel qu'ils utiliseront pour la transmission des données. La résolution des conflits est donc un problème critique dans GFRA, nécessitant une sélection minutieuse des créneaux temporels par les dispositifs afin d'éviter les retransmissions.

En raison du manque de coordination dans GFRA et des informations limitées sur les statistiques du réseau, les politiques de sélection des créneaux sont souvent basées sur des variantes de slotted ALOHA [90] tel que FSA [109] dans laquelle les dispositifs actifs sélectionnent uniformément aléatoirement un seul créneau dans une trame (Alg. 1).

D'autres variantes de FSA ont également été proposées dans lesquelles les dispositifs peuvent utiliser plus d'un créneau dans une même trame. On peut notamment citer IRSA [68] (Alg. 2) et CSA [78]. Néanmoins, le protocole slotted ALOHA reste le protocole MAC *de facto* utilisé dans les applications IoT [112, 113], en particulier lorsque le nombre moyen de dispositifs actifs est proche ou supérieur au nombre de ressources disponibles. Dans ce dernier cas, il semble intéressant de chercher à exploiter les statistiques générant l'activité des dispositifs pour réduire la probabilité de collisions. En effet, dans les réseaux IoT il est probable que certains dispositifs aient des activations corrélées ou différentes.

Dans [117, 118], les auteurs proposent de modifier les distributions de probabilité de choix de créneaux des dispositifs en fonction de leurs statistiques d'activation. Pour ce faire, dans chaque trame, les dispositifs actifs sont détectés, puis l'espérance du nombre de dispositifs décodés est calculé. Enfin, cet objectif peut être dérivé afin d'appliquer un algorithme de gradient stochastique sur la base des dispositifs détectés comme étant actifs.

Cependant, compte tenu du nombre de dispositifs transmettant leur séquence pilote en même temps, il est peu probable que, dans chaque trame, tous les dispositifs soient détectés sans erreurs. Dans cette première contribution, nous nous intéressons donc à l'impact que pourrait avoir une détection imparfaite des dispositifs.

7.1.2 Modèle

Considérons un réseau à créneaux temporels composé d'une BS et de N dispositifs équipés d'une antenne unique et partageant une sous porteuse commune (comme cela serait le cas en utilisant la technologie NB-IoT). Les transmissions s'effectuent sur des trames de tailles fixes dans lesquelles chaque dispositif n est actif avec une probabilité p_n. L'activité de chaque dispositif est représenté par la variable aléatoire $X_n \sim \text{Ber}(p_n)$ et la distribution de probabilité jointe par $p_{\mathbf{X}}$. Nous considérons que les dispositifs sont mutuellement indépendants, donc :

$$p_{\mathbf{X}}(\mathbf{x}) = \Pi_{n=1}^{N} p_n^{x_n}(1-p_n)^{1-x_n},\ \mathbf{x} \in \{0,1\}^N. \tag{7.1}$$

Dans la trame t, l'activité de tous les dispositifs est noté par $\mathbf{X}^t = [X_1^t, \ldots, X_N^t]^T$. Les vecteurs d'activités $\mathbf{X}^t$ sont mutuellement indépendants dans le temps t.

Protocole de Transmission

Le protocole de transmission, une variante de FSA [109], est décrit dans Alg. 12. Dans chaque trame t, les dispositifs actifs choisissent un créneau sur lequel transmettre en fonction d'une matrice d'allocation de ressources $\mathbf{A}^t$. Les éléments de cette matrice, $A_{ij}^t,\ i \in \{1,\ldots,N\},\ j \in \{1,\ldots,K\}$ représentent la probabilité que le dispositif i sélectionne le créneau j, sous contrainte que $\sum_j A_{ij}^t = 1,\ i = 1,\ldots,N$ (puisque chaque dispositif actif transmet une seule fois par trame). Contrairement aux protocoles ALOHA classiques, la sélection du créneau de transmission ne suit pas forcément une loi uniforme. Cette matrice peut donc être optimisée trame après trame dans le but de maximiser un objectif.

Algorithme 12 : Protocole de transmission dans la trame t en fonction de $\mathbf{A}^t$.

1 (Liaison descendante) Signal de synchronisation marquant le début de la trame.
2 **tant que** *Vrai* **faire**
3 Chaque dispositif actif n tire aléatoirement un créneau r_n selon la distribution de probabilité définie par la ligne n de $\mathbf{A}^t$.
4 Chaque dispositif actif n envoie son pilote puis ses données dans le créneau choisi r_n.
5 La BS décode les données et estime $\mathbf{X}^t$ (les dispositifs actifs dans la trame.
6 La BS calcule $\mathbf{A}^{t+1}$ sur la base de l'estimation de $\mathbf{X}^t$.
7 La BS envoi $\mathbf{A}^{t+1}$ aux dispositifs.
8 **fin**

Objectif d'Optimisation

L'objectif principal est l'optimisation de la matrice d'allocation de ressources $\mathbf{A}$ (l'exposant t est enlevé pour alléger la notation lorsque l'indice de la trame n'est pas important). Ce problème a été récemment adressé dans [117, 118] lorsque le vecteur d'activité est connu parfaitement dans chaque trame. On cherche à maximiser le débit du réseau :

$$T(\mathbf{A};\mathbf{p}) = \mathbb{E}_{\mathbf{X}\sim\mathbf{p}}[T^{\mathbf{X}}(\mathbf{A})], \tag{7.2}$$

où $T^{\mathbf{X}}(\mathbf{A})$ est défini par :

$$T_n(\mathbf{A};\mathbf{X}) = \sum_{k=1}^{K} X_n A_{nk} \prod_{\substack{m=1 \\ m\neq n}}^{N} (1 - X_m A_{mk})$$

$$T^{\mathbf{X}}(\mathbf{A}) = \sum_{n=1}^{N} T_n(\mathbf{A};\mathbf{X})$$

Cet objectif peut être interprété comme étant la proportion de créneaux dans lesquels un seul dispositif transmet ; nous cherchons donc à éviter au maximum les collisions. L'objectif peut être normalisé afin de comparer différentes statistiques d'activation :

$$T(\mathbf{A};\mathbf{p})^N = \frac{1}{\sum_{n=1}^{N} p_n} \mathbb{E}_{\mathbf{X}\sim\mathbf{p}}[T^{\mathbf{X}}(\mathbf{A})]. \tag{7.3}$$

Le problème de l'allocation de ressources est ainsi de trouver une matrice d'allocation qui maximise $T(\mathbf{A})$:

$$\mathbf{A}^* = \underset{\mathbf{A}\in\mathbb{R}_+^{N\times K};\sum_k A_{n,k}=1}{\arg\min} -T(\mathbf{A}).$$

L'objectif $T(\mathbf{A})$ n'est pas convexe en $\mathbf{A}$. Nous pouvons aussi souligner que les procédés présentés dans cet article ne se limitent pas à la maximisation du débit ; en changeant d'objectif, il devient possible de, par exemple, maximiser l'équité proportionnelle.

7.1.3 Optimisation par Descente de Gradient en Ligne

Tel qu'observé dans [117, 118], le problème (7.2) peut être vu comme un problème d'optimisation stochastique. Si l'on connaît parfaitement les vecteurs d'activités $\mathbf{X}^1, \mathbf{X}^2, \ldots$, alors il est possible de converger presque surement vers un point stationnaire en utilisant une descente de gradient en ligne. Soit :

$$\mathcal{H} = \{\mathbf{A} \in \mathbb{R}_+^{N\times K} : \sum_{k=1}^{K} A_{n,k} = 1,\ n = 1,\ldots,N\} \tag{7.4}$$

l'espace de contrainte de la matrice $\mathbf{A}$, et $\Pi_{\mathcal{H}}[\cdot]$ l'opérateur de projection vers le point de $\mathcal{H}$ le plus proche en norme Euclidienne. Pour un $\mathbf{X}$ donné, il est possible de calculer un gradient stochastique $g(\mathbf{A};\mathbf{X})$ ayant pour éléments $g(\mathbf{A};\mathbf{X})_{ql}$, $q \in \{1,\ldots,N\}, l \in \{1,\ldots,K\}$ [118] :

$$\begin{aligned} g(\mathbf{A};\mathbf{X})_{ql} = X_q \prod_{\substack{m=1 \\ m\neq q}}^{N} (1 - X_m A_{ml}) \\ - \sum_{\substack{n=1 \\ n\neq q}}^{N} X_q X_n A_{nl} \prod_{\substack{m=1 \\ m\neq n \\ m\neq q}}^{N} (1 - X_m A_{ml}). \end{aligned} \tag{7.5}$$

Dans ce cas, $g(\mathbf{A};\mathbf{X})$ est une estimation non biaisée de $\nabla T(\mathbf{A}) = \mathbb{E}_{\mathbf{X}}[\nabla_{\mathbf{A}} T^{\mathbf{X}}(\mathbf{A})]$.

Algorithme 13 : Algorithme d'optimisation stochastique avec identification sans erreurs.

Choisir une matrice d'allocation initiale $\mathbf{A}^1 \in \mathbb{R}^{N\times K}$ telle que $\sum_{k=1}^{K} A^1_{n,k} = 1,\ n = 1,\dots,N$, et une séquence de pas $\{\alpha^t\}$ avec $\alpha^t > 0$.
$t \leftarrow 1$
tant que *non convergé* **faire**
  Sur la base de $\mathbf{X}^t$, calculer une estimation non biaisée $g(\mathbf{A}^t;\mathbf{X}^t)$ de $\nabla_{\mathbf{A}^t} T(\mathbf{A}^t)$.
  $\mathbf{A}^{t+1} \leftarrow \Pi_{\mathcal{H}}[\mathbf{A}^t + \alpha^t g(\mathbf{A}^t;\mathbf{X}^t)]$
  $t \leftarrow t+1$
fin

Cependant, si les vecteurs d'activités $\mathbf{X}^t$ contiennent des erreurs, alors cette estimation de gradient est biaisée, ce qui entraîne la convergence vers une solution sous-optimale et induit une perte de performance.

Procédé de Réduction du Biais

La cause majeure de perte de performance est due à l'introduction de biais dans les étapes de descente de gradient à cause de la détection imparfaite des dispositifs. Afin de compenser ce biais, il est possible d'utiliser l'échantillonnage préférentiel [114, 40]. Considérons la fonction de poids $w : \mathbf{x} \mapsto w(\mathbf{x})$ avec $\mathbf{x} \in \{0,1\}^N$ définie par :

$$w(\mathbf{x}) = \frac{\Pr(\mathbf{X}=\mathbf{x})}{\Pr(\hat{\mathbf{X}}=\mathbf{x})}, \tag{7.6}$$

où $\mathbf{X}$ est le vecteur d'activités sans erreurs et $\hat{\mathbf{X}}$ est son estimation ; obtenu par exemple avec un algorithme de détection de dispositifs tel que GAMP[18, 19].

Une propriété importante du poids $w(\mathbf{x})$ est que pour tout $\mathbf{A} \in \mathbb{R}^{N\times K}$,

$$\begin{aligned}\mathbb{E}_{\mathbf{X}}[g(\mathbf{A};\mathbf{X})] &= \sum_{\mathbf{x}} g(\mathbf{A};\mathbf{x})\Pr(\mathbf{X}=\mathbf{x}) \\ &= \sum_{\mathbf{x}} g(\mathbf{A};\mathbf{x})\frac{\Pr(\mathbf{X}=\mathbf{x})}{\Pr(\hat{\mathbf{X}}=\mathbf{x})}\Pr(\hat{\mathbf{X}}=\mathbf{x}) \\ &= \mathbb{E}_{\hat{\mathbf{X}}}[w(\hat{\mathbf{X}})g(\mathbf{A};\hat{\mathbf{X}})].\end{aligned} \tag{7.7}$$

Autrement dit, $w(\hat{\mathbf{X}})g(\mathbf{A};\hat{\mathbf{X}})$ est une estimation non biaisée de $\nabla T(\mathbf{A})$, si $\mathbf{w}(\mathbf{x}) < \infty,\ \forall \mathbf{x}$ tel que $\Pr(\mathbf{X}=\mathbf{x}) > 0$. De ce fait, il est possible de compenser le biais introduit par les erreurs et de maintenir la convergence de Alg. 13 en utilisant ce poids dans l'étape de mise à jour. Cette dernière devient :

$$\mathbf{A}^{t+1} \leftarrow \Pi_{\mathcal{H}}[\mathbf{A}^t + \alpha^t w(\mathbf{x}) g(\mathbf{A}^t;\hat{\mathbf{X}}^t)] \tag{7.8}$$

Ce nouvel algorithme possède des garanties de convergence sous certaines conditions habituelles

sur la séquence de pas α^t.

Étape de Mise à Jour Proposée

Si le poids (7.6) permet de garantir la convergence, il est cependant difficile le calculer :

(i) le numérateur $\Pr(\mathbf{X} = \mathbf{x})$ se base sur une distribution qui n'est pas parfaitement connue.

(ii) le dénominateur $\Pr(\hat{\mathbf{X}} = \mathbf{x})$ se base sur une distribution postérieure qui peut être difficile à calculer suivant la complexité de l'algorithme utilisé pour la détection de dispositifs.

La distribution utilisée au dénominateur peut être estimée en observant la sortie de l'algorithme de détection de dispositifs et en calculant une approximation de Monte-Carlo sur de nombreuses trames. Nous supposerons donc cette distribution comme connue. La distribution de probabilité utilisée au numérateur peut typiquement être estimé par un algorithme espérance-maximisation. Si les conditions pour que cet algorithme fonctionne sont satisfaites, il nous fournira un estimateur de la distribution avec une certaine variance. Il est donc raisonnable de supposer que l'on peut avoir accès à une distribution $\tilde{\mathbf{X}} \sim \mathrm{Ber}(\tilde{\mathbf{p}})$, où $\tilde{p}_n = p_n + \boldsymbol{\eta},\ \eta_n \sim \mathcal{N}(0, \sigma^2)$, $\tilde{p}_i$ étant forcé à être dans $[0, 1]$, et donc à un poids $\tilde{w}$ de la forme :

$$\tilde{w}(\mathbf{x}) = \frac{\Pr(\tilde{\mathbf{X}} = \mathbf{x})}{\Pr(\hat{\mathbf{X}} = \mathbf{x})}. \tag{7.9}$$

Nous proposons donc d'utiliser la mise à jour suivante :

$$\mathbf{A}^{t+1} \leftarrow \Pi_{\mathcal{H}}[\mathbf{A}^t + \alpha^t \tilde{w}(\mathbf{x}) g(\mathbf{A}^t; \hat{\mathbf{X}}^t)] \tag{7.10}$$

Le biais introduit par cette nouvelle mise à jour n'est pas 0 puisque le poids $\tilde{w}$ ne permet pas de débiaiser complètement le gradient. Cependant, comme présenté dans la section 7.1.4, des gains intéressants peuvent tout de même être obtenus, suggérant que le biais introduit par $\tilde{w}$ est plus faible que celui introduit par la détection imparfaite.

7.1.4 Simulations

Nous évaluons le débit résultant dans un scénario où un algorithme de détection de dispositifs se base sur un algorithme GAMP tel que décrit dans [18]. La détection se base sur des pilotes uniques associés à chaque dispositif qui transmet ce dernier lors d'une phase d'annonce avant de transmettre ses données. Cela permet d'effectuer à la fois l'estimation du canal et la détection des dispositifs actifs. En fonction du SNR du pilote, l'algorithme de détection de dispositifs fera plus ou moins d'erreurs.

Nous considérons un réseau composé de $N = 20$ dispositifs, chaque trame étant composée de $K = 5$ créneaux. Chaque dispositif suit une probabilité d'activité $p_n \sim \mathcal{U}[0, 0.9],\ \forall n$. Les dispositifs sont mutuellement indépendants. L'optimisation se déroule sur 10 000 trames et les résultats sont

moyennés sur 20 exécutions. L'espace $\mathcal{H}$ étant relativement grand et la fonction $T(\mathbf{A})$ non-convexe, nous supposons que la BS peut effectuer l'optimisation pour plusieurs (10) matrices en parallèle et envoyer celle donnant les meilleures performances aux dispositifs. Les matrices $\mathbf{A}^1$ candidates sont tirées aléatoirement dans $\mathcal{H}$. Nous présentons le débit résultant de diverses méthodes :

- l'utilisation de Alg.13 avec une détection parfaite des dispositifs,
- l'utilisation de Alg.13 avec une détection imparfaite des dispositifs,
- l'utilisation de Alg.13 avec une détection imparfaite des dispositifs, mais avec la mise à jour utilisant le poids de correction exact 7.8,
- l'utilisation de Alg.13 avec une détection imparfaite des dispositifs, mais avec la mise à jour utilisant le poids de correction inexact pour différentes valeurs d'écart type,
- la matrice que donne le protocole ALOHA $\mathbf{A}_{\text{ALOHA}}$: la matrice constante $A_{ij} = \frac{1}{K}\ \forall i, j$,
- une allocation gloutonne $\mathbf{A}_h$ ou les dispositifs les plus actifs possèdent un créneau pour eux seuls.

Dans ce scénario, nous considérons que chacun des 20 dispositifs possède un pilote unique qui est une séquence de $L = 15$ échantillons tirés aléatoirement dans $\mathcal{CN}(0, \sqrt{P})$ ou P est la puissance d'émission commune à tous les dispositifs. Au début de la trame, un créneau spécial d'annonce est utilisé par tous les dispositifs actifs dans lequel ils transmettent leurs pilotes. En fonction du SNR à la réception, l'algorithme de détection de dispositifs introduit plus ou moins d'erreurs dans les vecteurs de dispositifs actifs $\hat{\mathbf{X}}^t$. Dans la Fig. 7.1, nous pouvons observer que si le SNR est faible, de nombreuses erreurs sont introduites, ce qui réduit les performances de l'allocation résultante si aucune compensation n'est effectuée (courbe bleu). Cependant, elles restent supérieures au débit d'une matrice ALOHA classique. Nos méthodes (orange, bleu, rouge, violet) permettent ainsi de limiter la baisse de performance. Il est intéressant de voir que dans ce cas, l'allocation gloutonne $\mathbf{A}_h$ donne les meilleures performances.

7.1.5 Conclusion

L'allocation stochastique de ressources permet d'obtenir des gains de performance notables comparé à un protocole ALOHA classique. En revanche, pour fonctionner de manière adéquate, il est nécessaire de correctement détecter les dispositifs actifs dans chaque trame. Dans cette contribution, nous avons proposé un moyen de compenser ces erreurs en utilisant l'échantillonnage préférentiel. Les résultats numériques montrent que les algorithmes de descente de gradient avec échantillonnage préférentiel peuvent se montrer résistants à la variance dans l'estimation de la probabilité d'activité des dispositifs.

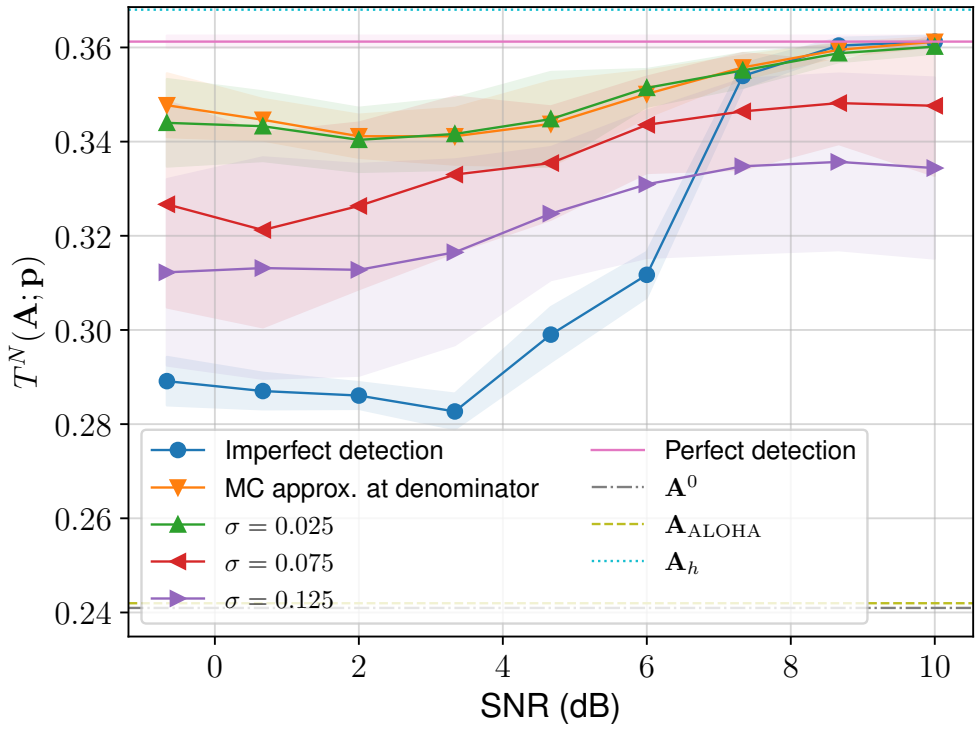


FIGURE 7.1 : Performance des différentes méthodes lorsque $\mathbf{X}$ est estimé par GAMP.

7.2 Allocation Conjointe de Créneaux et de Puissance

7.2.1 Introduction

Comme vue dans la section précédente, en optimisant la matrice d'allocation de ressources $\mathbf{A}$, il est possible d'augmenter le débit du réseau comparativement à la méthode de référence (ALOHA). Néanmoins, même en supposant que tous les dispositifs sont parfaitement détectés dans chaque trame, des collisions de paquets seront toujours présentes, surtout si le nombre de ressources disponibles est inférieur au nombre de dispositifs. Dès lors, les collisions résultant de l'utilisation de la même ressource par plusieurs dispositifs constituent un facteur limitant lorsque l'on cherche à augmenter la densité du réseau. De ce fait, il est courant de supposer que la station de base est en mesure d'exploiter un algorithme SIC. Avec un tel algorithme, les données de chaque dispositif sont décodées séquentiellement et soustraites du signal reçu afin d'augmenter le SINR des dispositifs qui doivent encore être décodés. SIC a été utilisé dans [117] pour tirer parti des différentes conditions du canal afin de réduire davantage la contention dans le réseau. Cependant, la puissance est supposée être la même pour tous les dispositifs. L'ajout d'un contrôle de puissance, en plus de la sélection du créneau d'émission, est souhaitable pour mieux exploiter le décodeur SIC. Ainsi, le contrôle de la puissance peut être exploité de la même manière que PD-NOMA [75].

Dans cette section, nous proposons un algorithme dans le contexte des réseaux NB-IoT [47] pour optimiser conjointement deux types de ressources différentes : le créneau temporel (orthogonal) et la puissance d'émission (non orthogonale).

7.2.2 Modèle

Nous supposons un modèle SIMO dans lequel les dispositifs sont équipés d'une seule antenne à l'émission et où la station de base est équipée de N_r antennes en réception. Comme dans la section précédente, dans chaque trame, un sous-ensemble de dispositifs est actif, selon une distribution $\mathbf{X} \sim p_{\mathbf{X}}$, et chacun du sous-ensemble actif choisit un créneau de transmission selon la matrice d'allocation de ressources $\mathbf{A}$, que nous cherchons à optimiser.

Le canal entre le dispositif n et la station de base est modélisé par un canal avec atténuation, affaiblissement et zones d'ombre $\mathbf{h}_n \in \mathbb{C}^{N_r}$. Durant un créneau, les dispositifs envoient un nombre fixe de symboles L. Dans le créneau k de la trame t, en supposant qu'il y ait $N_{a,k}^t$ dispositifs actifs, le signal reçu $\mathbf{Y}_k^t \in \mathbb{C}^{L \times N_r}$ est, comme on le voit dans (2.22) :

$$\mathbf{Y}_k^t = \sqrt{\mathrm{diag}(\mathbf{P}_k^t)}\mathbf{D}^t(\mathbf{H}^t)^{\mathsf{H}} + \mathbf{W}^t, \tag{7.11}$$

où $\mathrm{diag}(\mathbf{P}_k^t)$ représente la matrice de puissance de transmission des dispositifs dans le créneau k, $\mathbf{D}^t$ la matrice des symboles des données des dispositifs, $\mathbf{H}^t$ est la matrice de canal, et $\mathbf{W}^t$ est le bruit Gaussien.

Dans cette section, nous supposons que le canal est connu au niveau du récepteur (CSIR), ainsi il est possible d'utiliser un filtre spatial MRC. Le signal après filtrage est donc :

$$\hat{\mathbf{y}}_n = \mathbf{Y}_k \frac{\mathbf{h}_n}{\|\mathbf{h}_n\|^2} = \sqrt{P_{n,k}}\mathbf{d}_n + \sum_{m \in \mathcal{A}_k \setminus \{n\}} \sqrt{P_m}\mathbf{d}_m \frac{\mathbf{h}_m^\dagger \mathbf{h}_n}{\|\mathbf{h}_n\|^2} + \tilde{\mathbf{w}}_n, \tag{7.12}$$

avec $\hat{\mathbf{y}}_n \in \mathbb{C}^L$ l'approximation de $\mathbf{d}_n$.

7.2.3 Optimisation Conjointe de Puissance et de Créneau

L'objectif considéré dans cette section est le suivant :

$$T(\mathbf{A}, \mathbf{P}) = \mathbb{E}_{\mathbf{X} \sim p_{\mathbf{X}}}[T(\mathbf{A}, \mathbf{P}; \mathbf{X})] \tag{7.13}$$

$$T(\mathbf{A}, \mathbf{P}; \mathbf{X}) = \sum_{k=1}^{K} \sum_{\mathcal{S} \in \mathcal{P}(\mathcal{A}_k)} Q_k(\mathbf{A}, \mathcal{S}) \mathrm{SIC}_k(\mathcal{S}, \mathbf{P}), \tag{7.14}$$

où $\mathcal{P}(\mathcal{A})$ représente l'ensemble des parties de l'ensemble des dispositifs actifs, et

$$Q_k(\mathbf{A}, \mathcal{S}) = \prod_{i \in \mathcal{S}} A_{ik} \prod_{j \in \mathcal{S}^c} (1 - A_{jk}), \tag{7.15}$$

définit la probabilité qu'un ensemble de dispositifs actifs choisissent un créneau tandis que les autres choisissent n'importe quel autre créneau. Enfin, $\mathrm{SIC}_k(\cdot)$ désigne le nombre de dispositifs pouvant être décodés (le nombre de dispositifs ayant un SINR suffisant) en utilisant un décodeur SIC.

L'objectif 7.13 peut être optimisé, trame après trame, en utilisant l'algorithme ADAGRAD décrit

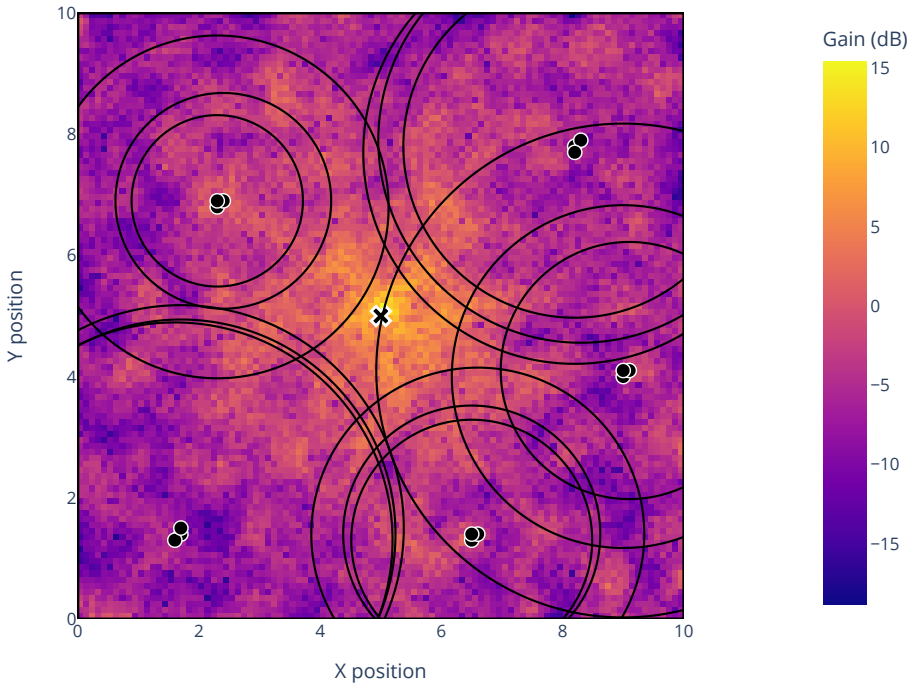


FIGURE 7.2 : Gain du canal dans la cellule et position des dispositifs.

dans l'Annexe A.3. La fonction $\mathrm{SIC}_k(\cdot)$ n'est pas dérivable, cependant elle peut être approximée par une fonction sigmoïde.

7.2.4 Simulations

Nous présentons des simulations inspirées d'un réseau de capteurs représenté dans la Fig. 7.2. Le réseau de capteurs comprend une cellule consistant en une BS et $N = 15$ dispositifs. Chaque dispositif possède une aire de détection dont le rayon varie. Dans chaque trame, des évènements sont générés uniformément aléatoirement dans la cellule, le nombre d'évènements correspondant à une loi de Poisson d'intensité λ. Les dispositifs actifs dans la trame sont ceux qui possèdent un évènement dans leur rayon de détection.

Nous lançons l'optimisation des matrices $\mathbf{A}$ et $\mathbf{P}$ pendant 30 000 trames puis nous évaluons la fonction objectif (7.13) sur 4 500 trames. Dans la Fig. 7.3, nous comparons l'objectif résultant de quatre méthodes différentes :

- les matrices d'initialisation $\mathbf{A}$ et $\mathbf{P}$,
- l'utilisation avec de la matrice $\mathbf{A}_{\mathrm{ALOHA}}$ avec une puissance aléatoire,
- l'optimisation avec les activités tirés du scénario basé sur un processus de Poisson,
- l'optimisation avec des dispositifs indépendants, dont les marginales sont identiques à celle du processus de Poisson.

Nous comparons l'objectif résultant en fonction de l'intensité du processus de Poisson. Il est possible

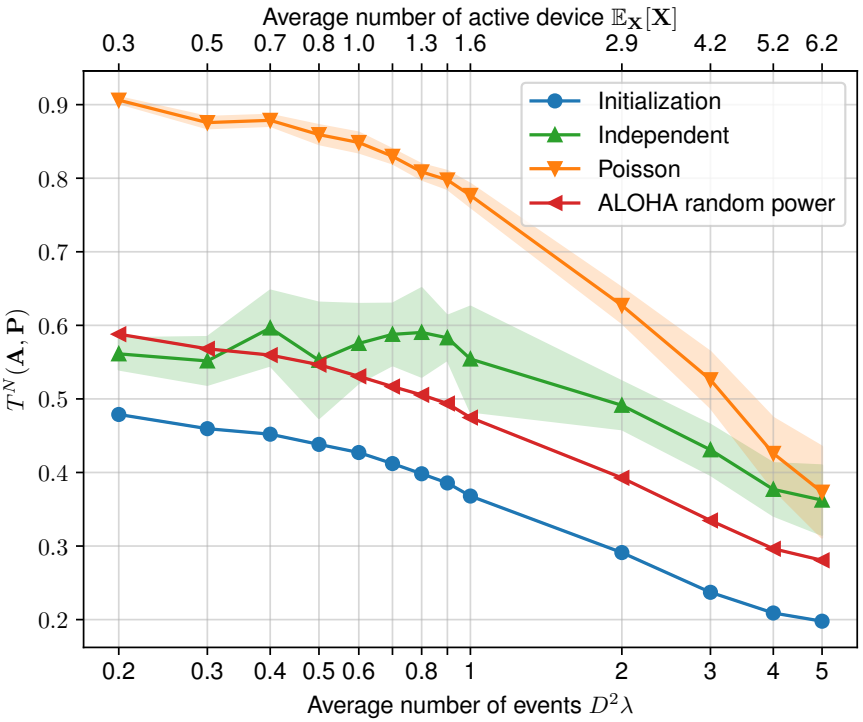


FIGURE 7.3 : Débit normalisé du réseau en fonction du nombre moyen d'évènements.

d'observer que les deux premières méthodes ont les moins bonnes performances. L'optimisation avec les dispositifs indépendants n'apporte que peu d'intérêt (voir pas du tout dans les intensités faibles) tandis que l'optimisation qui tient en compte les corrélations permet de augmenter significativement les performances du réseau, soulignant ainsi l'intérêt de notre méthode.

7.2.5 Conclusion

Dans cette section, nous avons présenté un moyen d'optimiser de manière conjointe la puissance d'émission des dispositifs ainsi que la sélection de leur créneau de transmission. La méthode que nous proposons améliore nettement les méthodes de l'état de l'art, en particulier lorsque les dispositifs du réseau sont susceptibles d'avoir des activités corrélées.

7.3 Allocation Jointe de PRB, de Puissance et de Couches

7.3.1 Introduction

Laissant de côté le scénario MTC, nous nous intéressons maintenant à un réseau de type eMBB, où la BS et les dispositifs exploitent toutes les fréquences de la RG OFDM, au lieu d'utiliser une seule sous-porteuse.

Actuellement, dans la 5G, la puissance d'émission des dispositifs est conçue afin que tous les dispositifs d'une cellule soient reçus avec la même PSD à la station de base, ce qui facilite l'étape d'égalisation et garantit un traitement équitable des dispositifs, quelle que soit leur distance par

rapport à la BS. La puissance d'émission est soumise à une contrainte stricte qui affecte indirectement le nombre de PRB utilisés par le dispositif.

Toutefois, cette façon de définir la puissance d'émission peut amener les dispositifs proches de la BS à émettre à une puissance plus élevée que nécessaire et les dispositifs proches du bord de la cellule à émettre à une puissance qui n'est pas assez élevée pour atteindre la puissance de réception requise, ce qui affecte fortement la qualité de leur transmission. D'autre part, la fonction d'attribution des PRB est la même pour tous les dispositifs et est indépendante de la qualité du canal de chaque PRB [35, Chapter 6.1.2.2]. De même, le nombre de couches utilisées par les dispositifs est défini par l'indicateur de rang [64], un paramètre calculé indépendamment de l'allocation de PRB et de la puissance.

7.3.2 Modèle et Problème

Nous considérons un réseau composé d'une unique cellule équipé de $N_r = 64$ antennes et d'un nombre fixe (N=8) de dispositifs, chacun équipé de $N_t = 4$, antennes qui transmettent tous dans le même créneau temporel. $N_{\text{PRB}} = 24$ PRB sont disponibles.

Sur un RE du PRB f, le signal reçu par la BS est :

$$\mathbf{y}_e = \sum_{n=1}^{N} \delta_{n,f} \mathbf{H}^t_{n,f} \mathbf{Q}^t_{n,f} \sqrt{\text{diag}(\mathbf{P}^t_{n,f})} \mathbf{d}_{n,e} + \mathbf{w}_e \tag{7.16}$$

$$= \bar{\mathbf{H}}^t_f \bar{\mathbf{d}}_{n,e} + \mathbf{w}_e, \tag{7.17}$$

où $\delta_{n,f}$ est un bit indiquant si le PRB f est alloué au dispositif n et $\mathbf{Q}$ est le précodeur définissant le nombre de couches (flux de données parallèle) utilisées par le dispositif n.

En utilisant la formule $r_{n,e,j} = \log_2(1 + \text{SINR}_{\text{n,e,j}})$ pour calculer le débit d'un dispositif dans un RE sur une couche j, et en utilisant le fait qu'il y ait L symboles dans un créneau et N_{SC} dans un PRB, on peut calculer le débit de chaque dispositif dans un PRB sur une couche particulière :

$$r_{n,f,j} = L N_{\text{sc}} r_{n,e,j}. \tag{7.18}$$

Dès lors, il est possible d'obtenir le débit r_n en sommant $r_{n,f,j}$ sur toutes les couches j et tous les PRB f que le dispositif utilise.

On peut ainsi exprimer le problème d'optimisation conjointe de la puissance d'émission, des PRB

et des couches utilisés :

$$\max_{\substack{\mathbf{P}\in\mathbb{R}^{N\times N_{\mathrm{PRB}}\times N_t}\\ \boldsymbol{\delta}\in\{0,1\}^{N\times N_{\mathrm{PRB}}}\\ \boldsymbol{\nu}\in\mathbb{N}^N}} \sum_{i=1}^{N_u} \log\left(r_n(\bar{\mathbf{H}}^t_1,\ldots,\bar{\mathbf{H}}^t_{N_{\mathrm{PRB}}},\mathbf{P}^t_{n,1}\ldots\mathbf{P}^t_{n,N_{\mathrm{PRB}}})\right) \tag{7.19a}$$

$$\text{Tel que : } \sum_{f=1}^{N_{\mathrm{PRB}}} \delta_{n,f} \sum_{j=1}^{\nu_n} N_{\mathrm{SC}} L P^t_{n,f,j} \leq P_{\max}, \qquad \forall t,n \tag{7.19b}$$

$$r_{\min} \leq \frac{\sum_{j=1}^{\nu_n}\sum_{f=1}^{N_{\mathrm{PRB}}} r^t_{n,f,j}}{\nu_n \sum_{f=1^{N_{\mathrm{PRB}}}} \delta^t_{n,f}} \qquad \forall n \tag{7.19c}$$

$$\delta_{n,f} r_{\min} \leq \delta_{n,f} r_{n,f,j} \leq r_{\max}, \qquad \forall n,f,j \tag{7.19d}$$

$$N_{\mathrm{prb,min}} \leq \sum_{f=1}^{N_{\mathrm{prb}}} \delta_{n,f} \qquad \forall n \tag{7.19e}$$

$$\nu_n \leq N_t \qquad \forall n. \tag{7.19f}$$

Ce problème présente quelques difficultés : il est composé de variables entières et réelles et n'est pas convexe. Ces deux particularités peuvent être contournées en relaxant le problème et en le combinant avec un algorithme heuristique pour trouver l'allocation de PRB et de couches.

Un dernier point bloquant est que l'utilisation d'un solveur mathématique pour résoudre le problème relaxé peut s'avérer longue et la durée de résolution peut varier selon l'instance traitée. Ainsi, afin de satisfaire à des contraintes temporelles (l'allocation doit être trouvée en moins d'une trame radio, 10 ms), nous proposons de créer un jeu de données d'entrée/sortie et d'apprendre le fonctionnement du solveur combiné aux heuristiques par 3 réseaux de neurones (un pour la puissance, les PRB et les couches).

7.3.3 Simulations

État de l'Art

Dans les normes 4G et 5G, la puissance d'émission de chaque dispositif n est calculée selon la formule de contrôle de puissance suivante [34, Section 7] :

$$P_{\mathrm{tr},n} = \min\{P_{\max}, P_0 + 10\log(N_{\mathrm{PRB},n}) + \alpha PL + CL\}, \tag{7.20}$$

où :

- $P_{\max}$ représente la puissance d'émission maximale, exprimée en dBm,
- $N_{\mathrm{PRB},n}$ est le nombre de OFDM blocs de ressources attribués au dispositif n,
- P_0 est la puissance cible reçue à la BS, exprimée en dBm,
- PL est le terme de compensation de l'affaiblissement,

- $\alpha \in [0, 1]$ est un coefficient contrôlant l'interférence intercellulaire, si $\alpha = 1$ l'affaiblissement est entièrement compensé, cependant la transmission par des dispositifs au bord de la cellule génèrera probablement des interférences dans les cellules voisines,
- l'élément CL est le paramètre CPLC qui permet d'affiner la puissance d'émission. Ce paramètre peut être envoyé par la BS lors de sa rétroaction vers le dispositif, ce qui signifie que la puissance peut être adaptée à court terme. Cependant, pour limiter le surcoût induit par l'envoi de cette information, CL ne peut prendre que quatre valeurs : $CL \in \{-1, 0, 1, 3\}$ dBm.

P_0 et α sont connus sous le nom de paramètres OLPC, ils peuvent être configurés de manière statique ou dynamique [70], mais doivent être les mêmes pour tous les dispositifs de la cellule. Ces deux paramètres posent plusieurs problèmes. Premièrement, comme cela a été étudié dans [43], certains ensembles de paramètres peuvent produire un débit très faible, en particulier si l'on considère un réseau composé de plusieurs cellules. Deuxièmement, ces paramètres ne peuvent être modifiés qu'à l'aide du protocole de couche réseau RRC, ce qui introduit un temps de latence non négligeable. En effet, ils sont initialement reçus par le dispositif au cours de l'établissement de la connexion de la procédure RACH, comme le montre l'étape 4 de la Fig. 2.5a. S'ils changent après la configuration initiale d'un dispositif, une procédure RRC ConnectionReconfiguration peut être déclenchée par la BS, mais cela induit au moins la communication de deux messages comme décrit dans [36, Section 5.3.5.3].

Nos simulations comparent le débit résultant de l'allocation obtenue par différentes valeurs de l'état de l'art, notre méthode ainsi que son approximation par les réseaux de neurones, et une méthode dite « naïve » qui consiste à transmettre à la puissance maximale sur tous les PRB et toutes les couches.

Fig. 7.4, présente la CDF du débit moyen par RE des différentes méthodes considérées. Nous pouvons observer que trois méthodes se distinguent particulièrement, les deux méthodes que nous proposons (vert et orange) et la méthode naïve. Ces trois méthodes permettent d'obtenir des débits nettement plus élevés que l'état de l'art et atteignent dans environ 30% des cas le débit maximal $r_{\max} = 8$.

Dans la Fig. 7.5, nous pouvons observer la CDF de la puissance d'émission des dispositifs. La méthode naïve y est clairement désavantagée puisqu'elle utilise toujours une puissance maximale à 10 dBm. Nous pouvons remarquer que nos méthodes possèdent une puissance d'émission comparable à celles des références à $P_0 - 100$ dBm et -90 dBm.

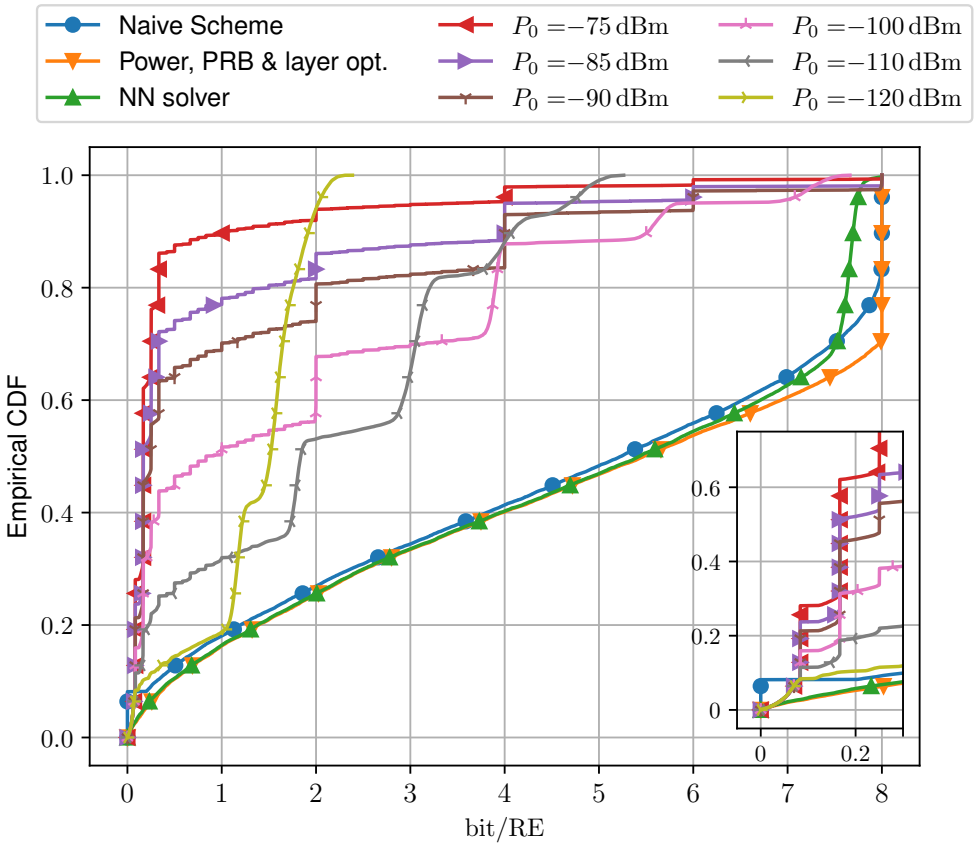


FIGURE 7.4 : CDF du débit moyen par RE.

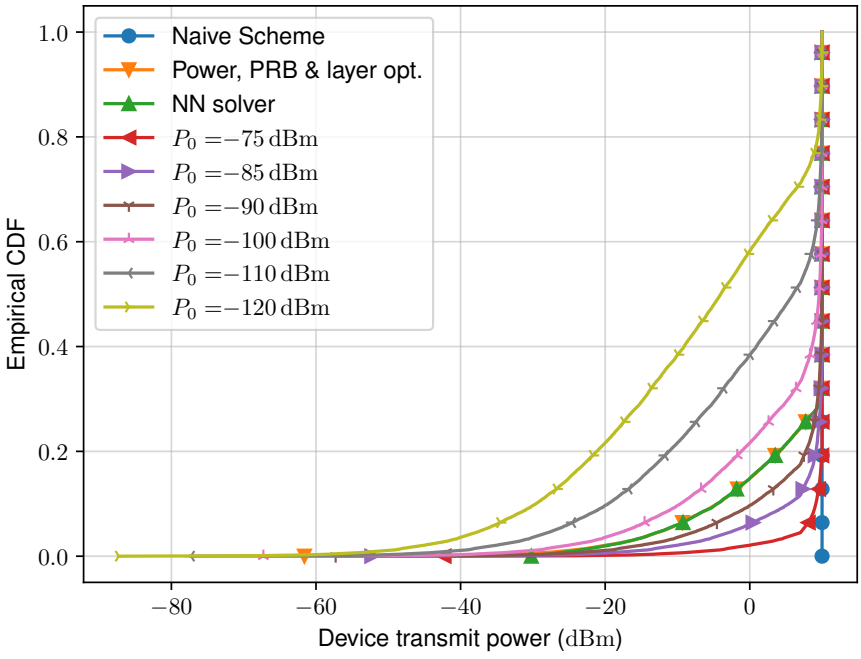


FIGURE 7.5 : CDF de la puissance d'émission totale.

7.3.4 Conclusion

Dans cette section, nous avons proposé une méthode pour augmenter le débit de données des dispositifs dans un réseau cellulaire axé sur l'eMBB. Notre principale contribution consiste à envisager conjointement l'optimisation de la puissance d'émission, des PRB et des couches utilisées par les dispositifs, ce qui permet d'avoir des paramètres adaptés aux conditions du canal de chaque dispositif. Nous avons d'abord exprimé un problème idéal non convexe que nous avons relaxé en un problème convexe (pour obtenir l'optimisation de la puissance) utilisé en conjonction avec un algorithme basé sur des heuristiques pour calculer les PRB et les couches. Notre étude par simulation montre l'avantage de notre méthode par rapport à l'état de l'art utilisé dans la 5G à travers différentes métriques. Par conséquent, ce travail pourrait constituer une caractéristique intéressante pour permettre des débits de données plus élevés dans les réseaux 6G. Comme notre méthode est basée sur les CSI, les paramètres de transmission doivent être recalculés à chaque changement de canal, ce qui entraîne une forte surcharge de calcul pour la BS et un délai avant l'envoi de l'allocation. Pour contourner ces inconvénients, nous proposons d'utiliser un réseau neuronal pour apprendre les entrées/sorties de la méthode que nous proposons, ce qui permet un calcul plus rapide et en temps constant de la solution.

Contrairement à l'état de l'art, notre méthode n'impose pas une puissance reçue cible commune à tous les dispositifs du réseau. Dans le cadre de travaux futurs, notre méthode pourrait être utilisée conjointement avec un récepteur SIC dans le domaine de la puissance à la BS afin d'augmenter davantage le débit de données.

Bibliography

[1] 3GPP. *NR; NR and NG-RAN Overall Description; Stage 2*. Technical Specification (TS) 38.300. 3rd Generation Partnership Project (3GPP), Mar. 2024.

[2] 3GPP. *Release 16 Description*. Technical Specification (TS) TR 21.916 V16.2.0. 3rd Generation Partnership Project (3GPP), June 2022.

[3] Daniel Abode, Ramoni Adeogun, and Gilberto Berardinelli. "Power Control for 6G Industrial Wireless Subnetworks: A Graph Neural Network Approach". In: *2023 IEEE Wireless Communications and Networking Conference (WCNC)*. Mar. 2023, pp. 1–6. DOI: 10.1109/WCNC55385.2023.10118984.

[4] Bhagawat Adhikari, Muhammad Jaseemuddin, and Alagan Anpalagan. "Resource Allocation for Co-Existence of eMBB and URLLC Services in 6G Wireless Networks: A Survey". In: *IEEE Access* 12 (2024), pp. 552–581. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2023.3343250.

[5] Patrick Agostini et al. *Evolution of the 5G New Radio Two-Step Random Access towards 6G Unsourced MAC*. May 2024. DOI: 10.48550/arXiv.2405.03348. arXiv: 2405.03348 [cs, math].

[6] S.M. Alamouti. "A Simple Transmit Diversity Technique for Wireless Communications". In: *IEEE Journal on Selected Areas in Communications* 16.8 (Oct. 1998), pp. 1451–1458. ISSN: 1558-0008. DOI: 10.1109/49.730453.

[7] F. Alberge. "Constellation Design with Deep Learning for Downlink Non-Orthogonal Multiple Access". In: *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. Sept. 2018, pp. 1–5. DOI: 10.1109/PIMRC.2018.8580937.

[8] Samad Ali et al. *6G White Paper on Machine Learning in Wireless Communication Networks*. Apr. 2020. DOI: 10.48550/arXiv.2004.13875. arXiv: 2004.13875 [cs, eess, math].

[9] Mohammed Almekhlafi et al. "Joint Resource and Power Allocation for URLLC-eMBB Traffics Multiplexing in 6G Wireless Networks". In: *ICC 2021 - IEEE International Conference on Communications*. June 2021, pp. 1–6. DOI: 10.1109/ICC42927.2021.9500443.

[10] Annu and P. Rajalakshmi. "Towards 6G V2X Sidelink: Survey of Resource Allocation—Mathematical Formulations, Challenges, and Proposed Solutions". In: *IEEE Open Journal of Vehicular Technology* 5 (2024), pp. 344–383. ISSN: 2644-1330. DOI: 10.1109/OJVT.2024.3368240.

[11] Pierre Bonami et al. "An Algorithmic Framework for Convex Mixed Integer Nonlinear Programs". In: *Discrete Optimization*. In Memory of George B. Dantzig 5.2 (May 2008), pp. 186–204. ISSN: 1572-5286. DOI: 10.1016/j.disopt.2006.10.011.

[12] Emilie Bout, Valeria Loscri, and Antoine Gallais. "How Machine Learning Changes the Nature of Cyberattacks on IoT Networks: A Survey". In: *IEEE Communications Surveys & Tutorials* 24.1 (2022), pp. 248–279. ISSN: 1553-877X. DOI: 10.1109/COMST.2021.3127267.

[13] M. Brownfield, Yatharth Gupta, and N. Davis. “Wireless Sensor Network Denial of Sleep Attack”. In: *Proceedings from the Sixth Annual IEEE SMC Information Assurance Workshop*. June 2005, pp. 356–364. DOI: 10.1109/IAW.2005.1495974.

[14] Enrico Casini, Riccardo De Gaudenzi, and Oscar Del Rio Herrero. “Contention Resolution Diversity Slotted ALOHA (CRDSA): An Enhanced Random Access Schemefor Satellite Access Packet Networks”. In: *IEEE Transactions on Wireless Communications* 6.4 (Apr. 2007), pp. 1408–1419. ISSN: 1558-2248. DOI: 10.1109/TWC.2007.348337.

[15] Violet Xinying Chen and J. N. Hooker. “Combining Leximax Fairness and Efficiency in a Mathematical Programming Model”. In: *European Journal of Operational Research* 299.1 (May 2022), pp. 235–248. ISSN: 0377-2217. DOI: 10.1016/j.ejor.2021.08.036.

[16] Xiaoming Chen et al. “Massive Access for 5G and Beyond”. In: *IEEE Journal on Selected Areas in Communications* 39.3 (Mar. 2021), pp. 615–637. ISSN: 1558-0008. DOI: 10.1109/JSAC.2020.3019724.

[17] Lélio Chetot. “Activity Models and Bayesian Estimation Algorithms for Wireless Grant-Free Random Access”. These de Doctorat. Lyon, July 2022.

[18] Lelio Chetot, Malcolm Egan, and Jean-Marie Gorce. “Joint Identification and Channel Estimation for Fault Detection in Industrial IoT With Correlated Sensors”. In: *IEEE Access* 9 (2021), pp. 116692–116701. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2021.3106736.

[19] Lélio Chetot, Malcolm Egan, and Jean-Marie Gorce. “Hybrid Generalized Approximate Message Passing for Active User Detection and Channel Estimation With Correlated Group-Heterogeneous Activity”. In: *IEEE Transactions on Communications* 72.7 (July 2024), pp. 3919–3933. ISSN: 1558-0857. DOI: 10.1109/TCOMM.2024.3367760.

[20] Jinho Choi et al. “Grant-Free Random Access in Machine-Type Communication: Approaches and Challenges”. In: *IEEE Wireless Communications* 29.1 (Feb. 2022), pp. 151–158. ISSN: 1536-1284, 1558-0687. DOI: 10.1109/MWC.121.2100135.

[21] Wei-Hung Chou et al. “An Adaptive Rank Selection Method in 3GPP 5G NR Systems”. In: *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. Dec. 2021, pp. 1912–1916.

[22] Federico Clazzer et al. “Irregular Repetition Slotted ALOHA over the Rayleigh Block Fading Channel with Capture”. In: *2017 IEEE International Conference on Communications (ICC)*. Paris, France: IEEE, May 2017, pp. 1–6. ISBN: 978-1-4673-8999-0. DOI: 10.1109/ICC.2017.7996796.

[23] Francisco Hugo Costa Neto et al. “Uplink Power Control Framework Based on Reinforcement Learning for 5G Networks”. In: *IEEE Transactions on Vehicular Technology* 70.6 (June 2021), pp. 5734–5748. ISSN: 1939-9359. DOI: 10.1109/TVT.2021.3074892.

[24] Apostolos Destounis et al. “Learn2MAC: Online Learning Multiple Access for URLLC Applications”. In: *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. Apr. 2019, pp. 1–6.

[25] Zhiguo Ding, Robert Schober, and H. Vincent Poor. “A New QoS-Guarantee Strategy for NOMA Assisted Semi-Grant-Free Transmission”. In: *IEEE Transactions on Communications* 69.11 (Nov. 2021), pp. 7489–7503. ISSN: 1558-0857. DOI: 10.1109/TCOMM.2021.3100598.

[26] Zhiguo Ding, Robert Schober, and H. Vincent Poor. “Unveiling the Importance of SIC in NOMA Systems—Part 1: State of the Art and Recent Findings”. In: *IEEE Communications Letters* 24.11 (Nov. 2020), pp. 2373–2377. ISSN: 1558-2558. DOI: 10.1109/LCOMM.2020.3012604.

[27] Zhiguo Ding, Robert Schober, and H. Vincent Poor. “Unveiling the Importance of SIC in NOMA Systems—Part II: New Results and Future Directions”. In: *IEEE Communications Letters* 24.11 (Nov. 2020), pp. 2378–2382. ISSN: 1558-2558. DOI: 10.1109/LCOMM.2020.3012601.

[28] Zhiguo Ding et al. “A Survey on Non-Orthogonal Multiple Access for 5G Networks: Research Challenges and Future Trends”. In: *IEEE Journal on Selected Areas in Communications* 35.10 (Oct. 2017), pp. 2181–2195. ISSN: 1558-0008. DOI: 10.1109/JSAC.2017.2725519.

[29] John Duchi, Elad Hazan, and Yoram Singer. “Adaptive Subgradient Methods for Online Learning and Stochastic Optimization”. In: *Journal of Machine Learning Research* (2011), p. 39.

[30] John Duchi et al. “Efficient Projections onto the L1-Ball for Learning in High Dimensions”. In: *Proceedings of the 25th International Conference on Machine Learning*. ICML ’08. New York, NY, USA: Association for Computing Machinery, July 2008, pp. 272–279. ISBN: 978-1-60558-205-4. DOI: 10.1145/1390156.1390191.

[31] Charles Dumas et al. “Design of Coded Slotted ALOHA With Interference Cancellation Errors”. In: *IEEE Transactions on Vehicular Technology* 70.12 (Dec. 2021), pp. 12742–12757. ISSN: 1939-9359. DOI: 10.1109/TVT.2021.3120069.

[32] V. Erceg et al. “An Empirically Based Path Loss Model for Wireless Channels in Suburban Environments”. In: *IEEE Journal on Selected Areas in Communications* 17.7 (July 1999), pp. 1205–1211. ISSN: 1558-0008. DOI: 10.1109/49.778178.

[33] Ericsson. *Ericsson Mobility Report*. Tech. rep. June 2024.

[34] ETSI. *5G;NR;Physical Layer Procedures for Control(3GPP TS 38.213 Version 16.2.0 Release 16)*. July 2020.

[35] ETSI. *5G;NR;Physical Layer Procedures for Data (3GPP TS 38.214 Version 16.2.0 Release 16)*. Jan. 2020.

[36] ETSI. *5G;NR;Radio Resource Control (RRC);Protocol Specification (3GPP TS 38.331 Version 15.3.0 Release 15)*. Jan. 2020.

[37] ETSI. *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation 3GPP TS 36.211 Version 13.13.0 Release 13*. Jan. 2020.

[38] R. Gallager. “Low-Density Parity-Check Codes”. In: *IRE Transactions on Information Theory* 8.1 (Jan. 1962), pp. 21–28. ISSN: 2168-2712. DOI: 10.1109/TIT.1962.1057683.

[39] Antoine Gallais et al. “Denial-of-Sleep Attacks against IoT Networks”. In: *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*. Apr. 2019, pp. 1025–1030. DOI: 10.1109/CoDIT.2019.8820402.

[40] Sahra Ghalebikesabi et al. *Mitigating Statistical Bias within Differentially Private Synthetic Data*. May 2022. arXiv: 2108.10934 [cs, stat].

[41] Andrea Goldsmith. *Wireless Communications*. Cambridge: Cambridge University Press, 2005. ISBN: 978-0-521-83716-3. DOI: 10.1017/CBO9780511841224.

[42] Alexandre Graell i Amat and Gianluigi Liva. “Finite-Length Analysis of Irregular Repetition Slotted ALOHA in the Waterfall Region”. In: *IEEE Communications Letters* 22.5 (May 2018), pp. 886–889. ISSN: 1558-2558. DOI: 10.1109/LCOMM.2018.2812845.

[43] Amir Haider, Rashmi Sharan Sinha, and Seung-Hoon Hwang. “Investigation of Open-Loop Transmit Power Control Parameters for Homogeneous and Heterogeneous Small-Cell Uplinks”. In: *ETRI Journal* 40.1 (2018), pp. 51–60. ISSN: 2233-7326. DOI: 10.4218/etrij.2017-0191.

[44] Md. Emdadul Haque et al. “A Survey of Scheduling in 5G URLLC and Outlook for Emerging 6G Systems”. In: *IEEE Access* 11 (2023), pp. 34372–34396. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2023.3264592.

[45] Iman Hmedoush. “Connectionless Transmission in Wireless Networks (IoT)”. PhD thesis. Sorbonne Université, May 2022.

[46] Iman Hmedoush, Cédric Adjih, and Paul Mühlethaler. “Deep Learning, Sensing-based IRSA (DS-IRSA): Learning a Sensing Protocol with Deep Reinforcement Learning”. Report. INRIA-SACLAY, June 2022.

[47] Andreas Hoglund et al. “Overview of 3GPP Release 14 Enhanced NB-IoT”. In: *IEEE Network* 31.6 (Nov. 2017), pp. 16–22. ISSN: 0890-8044. DOI: 10.1109/MNET.2017.1700082.

[48] J. N. Hooker and H. P. Williams. “Combining Equity and Utilitarianism in a Mathematical Programming Model”. In: *Management Science* 58.9 (Sept. 2012), pp. 1682–1693. ISSN: 0025-1909. DOI: 10.1287/mnsc.1120.1515.

[49] Yuhao Huang et al. “Age of Information Minimization for Frameless ALOHA in Grant-Free Massive Access”. In: *IEEE Transactions on Wireless Communications* 22.12 (Dec. 2023), pp. 9778–9792. ISSN: 1558-2248. DOI: 10.1109/TWC.2023.3273531.

[50] ITU. *Minimum Requirements Related to Technical Performance for IMT-2020 Radio Interface(s)*. Tech. rep. ITU-R M.2410-0. Nov. 2017.

[51] R. Jain, D. Chiu, and W. Hawe. *A Quantitative Measure Of Fairness And Discrimination For Resource Allocation In Shared Computer Systems*. Sept. 1998. DOI: 10.48550/arXiv.cs/9809099. arXiv: cs/9809099.

[52] Alix Jeannerot, Malcolm Egan, and Jean-Marie Gorce. “Exploiting Device Heterogeneity in Grant-Free Random Access: A Data-Driven Approach”. In: *IEEE Transactions on Vehicular Technology* (2024), pp. 1–11. ISSN: 1939-9359. DOI: 10.1109/TVT.2024.3396825.

[53] Alix Jeannerot, Malcolm Egan, and Jean-Marie Gorce. “Joint Slot and Power Optimization for Grant Free Random Access with Unknown and Heterogeneous Device Activity”. In: *2024 32nd European Signal Processing Conference (EUSIPCO)*. Aug. 2024, pp. 2077–2081. DOI: 10.23919/EUSIPCO63174.2024.10714984.

[54] Alix Jeannerot et al. “Compensation Des Erreurs d’Identification d’Utilisateurs Dans l’Allocation Stochastique de Ressources Pour Réseaux Aloha Hétérogènes”. In: *29° Colloque Sur Le Traitement Du Signal et Des Images*. 2023-1284. Grenoble: GRETSI - Groupe de Recherche en Traitement du Signal et des Images, Aout # 6 - Sept # 9 2023, p. 841–844.

[55] Alix Jeannerot et al. “Mitigating User Identification Errors in Resource Optimization for Grant-Free Random Access”. In: *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*. June 2023, pp. 1–6. DOI: 10.1109/VTC2023-Spring57618.2023.10200062.

[56] Hamdi Joudeh and Bruno Clerckx. “Robust Transmission in Downlink Multiuser MISO Systems: A Rate-Splitting Approach”. In: *IEEE Transactions on Signal Processing* 64.23 (2016), pp. 6227–6242.

[57] Anders E. Kalor, Osama A. Hanna, and Petar Popovski. “Random Access Schemes in Wireless Systems with Correlated User Activity”. In: *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. Kalamata: IEEE, June 2018, pp. 1–5. ISBN: 978-1-5386-3512-4. DOI: 10.1109/SPAWC.2018.8445866.

[58] Pradnya Kamble and Alam N. Shaikh. “Evolution Towards 6G Wireless Networks: A Resource Allocation Perspective with Deep Learning Approach - A Review”. In: *Advancements in Smart Computing and Information Security*. Ed. by Sridaran Rajagopal, Parvez Faruki, and Kalpesh Popat. Cham: Springer Nature Switzerland, 2022, pp. 117–132. ISBN: 978-3-031-23092-9. DOI: 10.1007/978-3-031-23092-9_10.

[59] Malong Ke et al. “Compressive Sensing-Based Adaptive Active User Detection and Channel Estimation: Massive Access Meets Massive MIMO”. In: *IEEE Transactions on Signal Processing* 68 (2020), pp. 764–779. ISSN: 1941-0476. DOI: 10.1109/TSP.2020.2967175.

[60] Malong Ke et al. “Next-Generation URLLC With Massive Devices: A Unified Semi-Blind Detection Framework for Sourced and Unsourced Random Access”. In: *IEEE Journal on Selected Areas in Communications* 41.7 (July 2023), pp. 2223–2244. ISSN: 1558-0008. DOI: 10.1109/JSAC.2023.3280981.

[61] Petteri Kela and Teemu Veijalainen. “Cooperative Action Branching Deep Reinforcement Learning for Uplink Power Control”. In: *2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. June 2023, pp. 484–489. DOI: 10.1109/EuCNC/6GSummit58263.2023.10188225.

[62] Junseok Kim et al. “Two-Step Random Access for 5G System: Latest Trends and Challenges”. In: *IEEE Network* 35.1 (Jan. 2021), pp. 273–279. ISSN: 1558-156X. DOI: 10.1109/MNET.011.2000317.

[63] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. Jan. 2017. DOI: 10.48550/arXiv.1412.6980. arXiv: 1412.6980 [cs].

[64] Lopamudra Kundu, Gang Xiong, and Joonyoung Cho. “Physical Uplink Control Channel Design for 5G New Radio”. In: *2018 IEEE 5G World Forum (5GWF)*. July 2018, pp. 233–238. DOI: 10.1109/5GWF.2018.8517042.

[65] Harold J. Kushner, George Yin, and Harold J. Kushner. *Stochastic Approximation and Recursive Algorithms and Applications*. Applications of Mathematics 35. New York: Springer, 2003. ISBN: 978-0-387-00894-3.

[66] Dong Li et al. “Power Allocation for 6G Sub-Networks in Industrial Wireless Control”. In: *2024 IEEE Wireless Communications and Networking Conference (WCNC)*. Apr. 2024, pp. 1–6. DOI: 10.1109/WCNC57260.2024.10571230.

[67] Liang Liu et al. “Sparse Signal Processing for Grant-Free Massive Connectivity: A Future Paradigm for Random Access Protocols in the Internet of Things”. In: *IEEE Signal Processing Magazine* 35.5 (Sept. 2018), pp. 88–99. ISSN: 1558-0792. DOI: 10.1109/MSP.2018.2844952.

[68] Gianluigi Liva. “Graph-Based Analysis and Optimization of Contention Resolution Diversity Slotted ALOHA”. In: *IEEE Transactions on Communications* 59.2 (Feb. 2011), pp. 477–487. ISSN: 0090-6778. DOI: 10.1109/TCOMM.2010.120710.100054.

[69] Gianluigi Liva and Yury Polyanskiy. “Unsourced Multiple Access: A Coding Paradigm for Massive Random Access”. In: *Proceedings of the IEEE* (2024), pp. 1–16. ISSN: 1558-2256. DOI: 10.1109/JPROC.2024.3437208.

[70] Lorenzo Maggi, Alvaro Valcarce, and Jakob Hoydis. “Bayesian Optimization for Radio Resource Management: Open Loop Power Control”. In: *IEEE Journal on Selected Areas in Communications* 39.7 (July 2021), pp. 1858–1871. ISSN: 1558-0008. DOI: 10.1109/JSAC.2021.3078490.

[71] Nurul Huda Mahmood et al. *White Paper on Critical and Massive Machine Type Communication Towards 6G*. May 2020. DOI: 10.48550/arXiv.2004.14146. arXiv: 2004.14146 [cs, eess].

[72] Alexandre Marcastel et al. “Gradient-Free Online Resource Allocation Algorithms for Dynamic Wireless Networks”. In: *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. July 2019, pp. 1–5. DOI: 10.1109/SPAWC.2019.8815409.

[73] Yiming Miao et al. “Narrowband Internet of Things: Simulation and Modeling”. In: *IEEE Internet of Things Journal* 5.4 (Aug. 2018), pp. 2304–2314. ISSN: 2327-4662. DOI: 10.1109/JIOT.2017.2739181.

[74] J. Mo and J. Walrand. “Fair End-to-End Window-Based Congestion Control”. In: *IEEE/ACM Transactions on Networking* 8.5 (Oct. 2000), pp. 556–567. ISSN: 1558-2566. DOI: 10.1109/90.879343.

[75] Mohamed Mounir, Mohamed B. El_Mashade, and Ashraf Mohamed Aboshosha. “On The Selection of Power Allocation Strategy in Power Domain Non-Orthogonal Multiple Access (PD-NOMA) for 6G and Beyond”. In: *Transactions on Emerging Telecommunications Technologies* 33.6 (2022), e4289. ISSN: 2161-3915. DOI: 10.1002/ett.4289.

[76] H. Obeidat et al. “Indoor Environment Propagation Review”. In: *Computer Science Review* 37 (Aug. 2020), p. 100272. ISSN: 1574-0137. DOI: 10.1016/j.cosrev.2020.100272.

[77] Jianxiong Pan et al. “AI-Driven Blind Signature Classification for IoT Connectivity: A Deep Learning Approach”. In: *IEEE Transactions on Wireless Communications* 21.8 (Aug. 2022), pp. 6033–6047. ISSN: 1558-2248. DOI: 10.1109/TWC.2022.3145399.

[78] Enrico Paolini, Gianluigi Liva, and Marco Chiani. “Coded Slotted ALOHA: A Graph-Based Method for Uncoordinated Multiple Access”. In: *IEEE Transactions on Information Theory* 61.12 (Dec. 2015), pp. 6815–6832. ISSN: 0018-9448, 1557-9654. DOI: 10.1109/TIT.2015.2492579.

[79] Enrico Paolini et al. “Coded Random Access: Applying Codes on Graphs to Design Random Access Protocols”. In: *IEEE Communications Magazine* 53.6 (June 2015), pp. 144–150. ISSN: 1558-1896. DOI: 10.1109/MCOM.2015.7120031.

[80] Jihong Park et al. “Extreme Ultra-Reliable and Low-Latency Communication”. In: *Nature Electronics* 5.3 (Mar. 2022), pp. 133–141. ISSN: 2520-1131. DOI: 10.1038/s41928-022-00728-8.

[81] Adam Paszke et al. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., pp. 8024–8035.

[82] Anita Patil, Sridhar Iyer, and Rahul Jashvantbhai Pandya. *A Survey of Machine Learning Algorithms for 6G Wireless Networks*. Mar. 2022. DOI: 10.48550/arXiv.2203.08429. arXiv: 2203.08429 [cs].

[83] Elena Peralta et al. “Two-Step Random Access in 5G New Radio: Channel Structure Design and Performance”. In: *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*. Apr. 2021, pp. 1–7. DOI: 10.1109/VTC2021-Spring51267.2021.9449057.

[84] Petar Popovski et al. “Wireless Access for Ultra-Reliable Low-Latency Communication: Principles and Building Blocks”. In: *IEEE Network* 32.2 (Mar. 2018), pp. 16–23. ISSN: 1558-156X. DOI: 10.1109/MNET.2018.1700258.

[85] Prasoon Raghuwanshi et al. “Channel Scheduling for IoT Access with Spatial Correlation”. In: *IEEE Communications Letters* (2024), pp. 1–1. ISSN: 1089-7798, 1558-2558, 2373-7891. DOI: 10.1109/LCOMM.2024.3369480.

[86] Sundeep Rangan. “Generalized Approximate Message Passing for Estimation with Random Linear Mixing”. In: *2011 IEEE International Symposium on Information Theory Proceedings*. July 2011, pp. 2168–2172. DOI: 10.1109/ISIT.2011.6033942.

[87] Farhad Rezazadeh et al. *Toward Explainable Reasoning in 6G: A Proof of Concept Study on Radio Resource Allocation*. July 2024. DOI: 10.48550/arXiv.2407.10186. arXiv: 2407.10186 [cs].

[88] S. O. Rice. “Mathematical Analysis of Random Noise”. In: *The Bell System Technical Journal* 23.3 (July 1944), pp. 282–332. ISSN: 0005-8580. DOI: 10.1002/j.1538-7305.1944.tb00874.x.

[89] Benoît-Marie Robaglia et al. “SeqDQN: Multi-Agent Deep Reinforcement Learning for Uplink URLLC with Strict Deadlines”. In: *2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. June 2023, pp. 623–628. DOI: 10.1109/EuCNC/6GSummit58263.2023.10188325.

[90] Lawrence G. Roberts. “ALOHA Packet System with and without Slots and Capture”. In: *ACM SIGCOMM Computer Communication Review* 5.2 (Apr. 1975), pp. 28–42. ISSN: 0146-4833. DOI: 10.1145/1024916.1024920.

[91] Mohammad Rowshan et al. “Channel Coding Toward 6G: Technical Overview and Outlook”. In: *IEEE Open Journal of the Communications Society* 5 (2024), pp. 2585–2685. ISSN: 2644-125X. DOI: 10.1109/OJCOMS.2024.3390000.

[92] Walid Saad, Mehdi Bennis, and Mingzhe Chen. “A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems”. In: *IEEE Network* 34.3 (May 2020), pp. 134–142. ISSN: 0890-8044, 1558-156X. DOI: 10.1109/MNET.001.1900287.

[93] Nikolaos V. Sahinidis. “BARON: A General Purpose Global Optimization Software Package”. In: *Journal of Global Optimization* 8.2 (Mar. 1996), pp. 201–205. ISSN: 1573-2916. DOI: 10.1007/BF00138693.

[94] Muhammad Basit Shahab et al. “Grant-Free Non-Orthogonal Multiple Access for IoT: A Survey”. In: *IEEE Communications Surveys & Tutorials* 22.3 (2020), pp. 1805–1838. ISSN: 1553-877X. DOI: 10.1109/COMST.2020.2996032.

[95] S. Shamai and S. Verdu. “The Impact of Frequency-Flat Fading on the Spectral Efficiency of CDMA”. In: *IEEE Transactions on Information Theory* 47.4 (May 2001), pp. 1302–1327. ISSN: 1557-9654. DOI: 10.1109/18.923717.

[96] Yandong Shi et al. “Machine Learning for Large-Scale Optimization in 6G Wireless Networks”. In: *IEEE Communications Surveys & Tutorials* 25.4 (2023), pp. 2088–2132. ISSN: 1553-877X. DOI: 10.1109/COMST.2023.3300664.

[97] Shin-Lin Shieh and Shih-Hung Yang. “Enhanced Irregular Repetition Slotted ALOHA Under SIC Limitation”. In: *IEEE Transactions on Communications* 70.4 (Apr. 2022), pp. 2268–2280. ISSN: 1558-0857. DOI: 10.1109/TCOMM.2022.3147505.

[98] María E. Sousa-Vieira and Manuel Fernández-Veiga. “Study of Coded ALOHA with Multi-User Detection under Heavy-Tailed and Correlated Arrivals”. In: *Future Internet* 15.4 (Apr. 2023), p. 132. ISSN: 1999-5903. DOI: 10.3390/fi15040132.

[99] K. Pavan Srinath, Alix Jeannerot, and Alvaro Valcarce Rial. *Joint Resource-Power Allocation and UE Rank Selection in Multi-User MIMO Systems with Linear Transceivers*. July 2024. DOI: 10.48550/arXiv.2407.16483. arXiv: 2407.16483 [cs, eess, math].

[100] Chirag Ramesh Srivatsa and Chandra R. Murthy. “User Activity Detection for Irregular Repetition Slotted Aloha Based MMTC”. In: *IEEE Transactions on Signal Processing* 70 (2022), pp. 3616–3631. ISSN: 1941-0476. DOI: 10.1109/TSP.2022.3185891.

[101] Jingrui Su, Guangliang Ren, and Bo Zhao. “NOMA-Based Coded Slotted ALOHA for Machine-Type Communications”. In: *IEEE Communications Letters* 25.7 (July 2021), pp. 2435–2439. ISSN: 1558-2558. DOI: 10.1109/LCOMM.2021.3067932.

[102] Yuanyuan Sun et al. “When Machine Learning Meets Privacy in 6G: A Survey”. In: *IEEE Communications Surveys & Tutorials* 22.4 (2020), pp. 2694–2724. ISSN: 1553-877X. DOI: 10.1109/COMST.2020.3011561.

[103] Maxime Vaillant, Alix Jeannerot, and Jean-Marie Gorce. *Joint Constellation Shaping Using Gradient Descent Approach for MU-MIMO Broadcast Channel*. Aug. 2024.

[104] Alvaro Valcarce et al. “The Role of AI in 6G MAC”. In: *2024 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. June 2024, pp. 723–728. DOI: 10.1109/EuCNC/6GSummit60053.2024.10597082.

[105] S. Verdu and S. Shamai. “Spectral Efficiency of CDMA with Random Spreading”. In: *IEEE Transactions on Information Theory* 45.2 (Mar. 1999), pp. 622–640. ISSN: 1557-9654. DOI: 10.1109/18.749007.

[106] Sergio Verdu. *Multiuser Detection*. 1st. USA: Cambridge University Press, Aug. 1998. ISBN: 978-0-521-59373-1.

[107] Andreas Wächter and Lorenz T. Biegler. “On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming”. In: *Mathematical Programming* 106.1 (Mar. 2006), pp. 25–57. ISSN: 1436-4646. DOI: 10.1007/s10107-004-0559-y.

[108] Zhe Wang et al. “A Tutorial on Extremely Large-Scale MIMO for 6G: Fundamentals, Signal Processing, and Applications”. In: *IEEE Communications Surveys & Tutorials* (2024), pp. 1–1. ISSN: 1553-877X. DOI: 10.1109/COMST.2023.3349276.

[109] J.E. Wieselthier, A. Ephremides, and L.A. Michaels. “An Exact Analysis and Performance Evaluation of Framed ALOHA with Capture”. In: *IEEE Transactions on Communications* 37.2 (Feb. 1989), pp. 125–137. ISSN: 1558-0857. DOI: 10.1109/26.20080.

[110] Nian Xia, Hsiao-Hwa Chen, and Chu-Sing Yang. “Emerging Technologies for Machine-Type Communication Networks”. In: *IEEE Network* 34.1 (Jan. 2020), pp. 214–222. ISSN: 1558-156X. DOI: 10.1109/MNET.001.1900132.

[111] Animesh Yadav et al. “On Performance Comparison of Multi-Antenna HD-NOMA, SCMA, and PD-NOMA Schemes”. In: *IEEE Wireless Communications Letters* 10.4 (Apr. 2021), pp. 715–719. ISSN: 2162-2345. DOI: 10.1109/LWC.2020.3041601.

[112] Jihong Yu et al. “Stabilizing Frame Slotted Aloha-Based IoT Systems: A Geometric Ergodicity Perspective”. In: *IEEE Journal on Selected Areas in Communications* 39.3 (Mar. 2021), pp. 714–725. ISSN: 0733-8716, 1558-0008. DOI: 10.1109/JSAC.2020.3018795.

[113] Zhiling Yue et al. “Age of Information Under Frame Slotted ALOHA-Based Status Updating Protocol”. In: *IEEE Journal on Selected Areas in Communications* 41.7 (July 2023), pp. 2071–2089. ISSN: 1558-0008. DOI: 10.1109/JSAC.2023.3280990.

[114] Bianca Zadrozny. “Learning and Evaluating Classifiers under Sample Selection Bias”. In: *Twenty-First International Conference on Machine Learning - ICML ’04*. Banff, Alberta, Canada: ACM Press, 2004, p. 114. DOI: 10.1145/1015330.1015425.

[115] Jie Zeng et al. “Achieving Energy-Efficient Massive URLLC Over Cell-Free Massive MIMO”. In: *IEEE Internet of Things Journal* 11.2 (Jan. 2024), pp. 2198–2210. ISSN: 2327-4662. DOI: 10.1109/JIOT.2023.3293008.

[116] Zhijun Zhang, Kai Niu, and Jincheng Dai. “Performance Bounds of Coded Slotted ALOHA Over Erasure Channels”. In: *IEEE Transactions on Vehicular Technology* 71.11 (Nov. 2022), pp. 12338–12343. ISSN: 1939-9359. DOI: 10.1109/TVT.2022.3192092.

[117] Ce Zheng et al. “Stochastic Resource Allocation for Outage Minimization in Random Access with Correlated Activation”. In: *2022 IEEE Wireless Communications and Networking Conference (WCNC)*. Austin, TX, USA: IEEE, Apr. 2022, pp. 1635–1640. ISBN: 978-1-66544-266-4. DOI: 10.1109/WCNC51071.2022.9771709.

[118] Ce Zheng et al. “Stochastic Resource Optimization of Random Access for Transmitters With Correlated Activation”. In: *IEEE Communications Letters* 25.9 (Sept. 2021), pp. 3055–3059. ISSN: 1089-7798, 1558-2558, 2373-7891. DOI: 10.1109/LCOMM.2021.3090110.

[119] Qiuyun Zou et al. “Message Passing Based Joint Channel and User Activity Estimation for Uplink Grant-Free Massive MIMO Systems With Low-Precision ADCs”. In: *IEEE Signal Processing Letters* 27 (2020), pp. 506–510. ISSN: 1558-2361. DOI: 10.1109/LSP.2020.2979534.

APPENDIX

A Gradient Methods for Optimization

Contents

Stochastic gradient descent is a widely used method to minimize a function. As this tool will be used extensively throughout the thesis, this section provides an overview of SGD as well as summary of its convergence guarantees. The motivation, intuition, as well as sketches of proof given here largely follows Chapter 4 and 5 of [65].

A summary of a popular variant of SGD, ADAGRAD is also provided.

A.1 Gradient Descent

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a continuous and differentiable function and $\nabla_{\mathbf{w}} f(\mathbf{w}) = \begin{bmatrix} \frac{\partial f(\mathbf{w})}{\partial w_1} & \dots & \frac{\partial f(\mathbf{w})}{\partial w_d} \end{bmatrix}$ its gradient. We are interested in finding its minimums $\mathbf{w}^\star$ which are obtained at the points $\mathbf{w}^\star$ where $\nabla_{\mathbf{w}} f(\mathbf{w}^\star) = \begin{bmatrix} 0 & \dots & 0 \end{bmatrix}$. The minimization of $f(\mathbf{w})$ thus amounts to finding the roots of $\nabla_{\mathbf{w}} f(\mathbf{w})$. Starting with an initial point $\mathbf{w}^1$, gradient descent is an iterative algorithm described in Alg. 14 that can be used to find a local stationary point, under conditions on the step size γ^t: $\sum_{t=1}^{\infty} \gamma^t = \infty;\ \gamma^t \geq 0\ \forall t, \gamma^t \to 0$. If the function f is convex, then the stationary point is also the global minimum. If the function f is non-convex, then other local stationary points can be obtained by running Alg. 14 with different initial point $\mathbf{w}^1$ and step size sequence $\{\gamma^t\}$.

The stopping condition of the while loop can be based on several criteria:

- a fixed number of iteration T,
- the reduction of the function being smaller than a threshold $|f(\mathbf{w}^{t+1}) - f(\mathbf{w}^t)| < \epsilon$, $\epsilon > 0$,
- a sequence of length l not providing significant improvement in the minimization of the function: $\sum_{l=1}^{L} |f(\mathbf{w}^{t+1}) - f(\mathbf{w}^{t+1-l})| < \epsilon$, $\epsilon > 0$.

Algorithm 14: Gradient Descent.

1 **Function** `gd(f)`
2 Choose initial $\mathbf{w}^1 \in \mathbb{R}^d$ and step-size sequence $\{\gamma^t\}$ with $\gamma^t > 0,\ t = 1, 2, \ldots$.
3 $t \leftarrow 1$
4 **while** *not converged* **do**
5 $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \gamma^t \nabla_{\mathbf{w}^t} f(\mathbf{w}^t)$
6 $t \leftarrow t + 1$
7 **end**
8 **return** $\mathbf{w}^t$

Usually, the motivation of GD is done through the Taylor expansion of order one of $f(\mathbf{w})$ around the point $\mathbf{w}^t$. We describe here another motivation, based on ordinary differential equation (ODE), appropriate for a better generalization to the stochastic case discussed in Section. A.2.

Consider Step. 5 of Alg. 14:

$$\mathbf{w}^{t+1} = \mathbf{w}^t + \gamma^t \nabla_{\mathbf{w}^t} f(\mathbf{w}^t). \tag{A.1}$$

Define the piece wise constant function $\mathbf{w}^t(\tau) = \sum_{l=t}^{\infty} \mathbf{w}^l \mathbb{1}\{\tau \in [\sum_{i=1}^{l} \gamma^i, \sum_{i=1}^{l} \gamma^i + \gamma^{l+1}]\}$, (A.1) can be rewritten as:

$$\mathbf{w}^{t+1}(\tau) = \mathbf{w}^t(\tau) + \gamma^t \nabla_{\mathbf{w}^t} f(\mathbf{w}^t) \tag{A.2}$$

$$\frac{\mathbf{w}^{t+1}(\tau) - \mathbf{w}^t(\tau)}{\gamma^t} = \nabla_{\mathbf{w}^t} f(\mathbf{w}^t). \tag{A.3}$$

Remark that the left-hand side of (A.3) looks like a derivative when $\gamma^t \to 0$. The sequence of function $\mathbf{w}^t(\tau)$ can be shown to converge[1] to a asymptotic function $\mathbf{w}(\tau)$ that is a solution of the mean ODE defined as:

$$\frac{\partial \mathbf{w}(\tau)}{\partial \tau} = \nabla_{\mathbf{w}^t} f(\mathbf{w}^t). \tag{A.4}$$

We are thus interested in the points were the function $\mathbf{w}(\tau)$ is flat, *i.e.* where $\frac{\partial \mathbf{w}(\tau)}{\partial \tau} = 0$ which will represent the roots of $\nabla_{\mathbf{w}^t} f(\mathbf{w}^t)$.

An important extension of GD is when the parameters $\mathbf{w}$ are constrained to lie within a certain constraint space $\mathcal{H}$. Sparing some mathematical technicalities, that can be found in [65, Chapter 4.3], it can be shown that convergence guarantees hold if:

- the projection back to the space of feasible $\mathbf{w} \in \mathcal{H}$ is done at after each step,
- the normal of the surface of the constraint space $\mathcal{H}$ is continuously differentiable.

Two topological spaces particularly interest us:

[1] The proof is out of the scope of this thesis and can be found in [65, Chap 4 and 5], it consists in essentially showing that the sequence of function $\mathbf{w}^t(\tau)$ is equicontinuous then applying the Arzelà-Ascoli theorem to show that there exists a converging subsequence.

- the hyperrectangle $\mathcal{H} = \{\mathbf{w} :\ l_i \leq w_i \leq u_i\},\ (l_i, u_i) \in \mathbb{R}^2,\ l_i \leq u_i,\ i = 1, \ldots, d$,
- the simplex $\mathcal{H} = \{\mathbf{w} :\ \sum_{i=1}^{d} w_i = 1\}$.

Hence, by defining $\Pi_{\mathcal{H}}[\mathbf{w}]$ the projection of $\mathbf{w}$ to its closest point (in terms of ℓ_2-norm) that lie in $\mathcal{H}$, the step of projected gradient descent can be written as:

$$\mathbf{w}^{t+1} = \Pi_{\mathcal{H}}[\mathbf{w}^t + \gamma^t \nabla_{\mathbf{w}^t} f(\mathbf{w}^t)] \tag{A.5}$$
$$= \mathbf{w}^t + \gamma^t \nabla_{\mathbf{w}^t} f(\mathbf{w}^t) + \gamma^t \mathbf{z}^t, \tag{A.6}$$

where $\mathbf{z}^t$ is known as the reflection term: it is the vector of smallest ℓ_2-norm that brings back $\mathbf{w}^t + \gamma^t \nabla_{\mathbf{w}^t} f(\mathbf{w}^t)$ into the constraint set $\mathcal{H}$. For the hyperrectangle the projection is obtained by clipping coordinate-wise the coordinates that are outside of $\mathcal{H}$. For projection onto the simplex, the projection can be obtained by using algorithms like the one described in [30]

A.2 Projected Stochastic Gradient Descent

The main issue of gradient descent is that computing the exact gradient can be computationally intensive, either because the number of parameters is big or because the function is difficult.

As a result we can choose to use a different function $g(\mathbf{w}^t)$ to perform the updates instead of relying on $\nabla^t_{\mathbf{w}} \left(f(\mathbf{w}^t)\right)$. When $g(\mathbf{w}^t)$ has some randomness, it is known as the stochastic approximation of the gradient.

Example A.2.1 (Randomly sampling $\nabla^t_{\mathbf{w}} f(\mathbf{w}^t)$)**.** A commonly used method to construct $g(\mathbf{w}^t)$ is to compute the gradient only for a random subset of the indices of the parameters. Defining $\mathcal{P}(\mathcal{X})$ the powerset of $\mathcal{X}$, the gradient estimate is:

$$g(\mathbf{w}^t)_i = \begin{cases} 0 & i \notin \mathcal{I}^t \\ \nabla_{\mathbf{w}^t} f(\mathbf{w}^t)_i & i \in \mathcal{I}^t \end{cases}, \mathcal{I}^t \sim \mathcal{U}\left(\mathcal{P}\left(\{1, \ldots, d\}\right) \setminus \emptyset\right).$$

Example A.2.2 (When objective function is an expectation)**.** Often, in machine learning application and in the following chapters of this thesis, the objective function is the expectation of a function of a random variable, thus $f(\mathbf{w}) = \mathbb{E}_X[h(\mathbf{w}; X)]$, with X a random variable. Depending on X computing the expectation can be difficult or even impossible. As a result, if T samples $\{x_i\},\ i = 1, \ldots T$ of X are available, a Monte Carlo approximation $\hat{f}(\mathbf{w}) = \frac{1}{T} \sum_{i=1}^{T} h(\mathbf{w}, x_i)$ of the expectation can be obtained. The stochastic approximation of the gradient is thus:

$$g(\mathbf{w}^t) = \nabla_{\mathbf{w}^t} \hat{f}(\mathbf{w}^t).$$

Alg. 15 describe the SGD algorithm, the main difference with Alg. 14 is the usage of the function $g(\mathbf{w}^t)$ in Step. 5 instead of $\nabla_{\mathbf{w}^t} f(\mathbf{w}^t)$. We might thus wonder what properties should $g(\cdot)$ posses in order to maintain the convergence guarantees of Al. 14.

Algorithm 15: Projected stochastic gradient descent.

1 **Function** `psgd(f)`
2 Choose initial allocation matrix $\mathbf{w}^1 \in \mathbb{R}^d$ such that $\mathbf{w}^1 \in \mathcal{H}$, and step-size sequence $\{\gamma^t\}$ with $\gamma^t > 0,\ t = 1, 2, \ldots$.
3 $t \leftarrow 1$
4 **while** *not converged* **do**
5 $\mathbf{w}^{t+1} \leftarrow \Pi_{\mathcal{H}}[\mathbf{w}^t - \gamma^t g(\mathbf{w}^t)]$
6 $t \leftarrow t + 1$
7 **end**
8 **return** $\mathbf{w}^t$

The theorem 5.2.1 of [65] allows to give convergence guarantees of Alg. 15 based on the properties of the function $g(\cdot)$ and the same conditions on sequence $\{\gamma^t\}$ as for Alg. 14. We briefly sketch the proof here.

Step. 5 can be rewritten as:

$$\mathbf{w}^{t+1} = \Pi_{\mathcal{H}}[\mathbf{w}^t + \gamma^t g(\mathbf{w}^t)] \tag{A.7}$$

$$= \mathbf{w}^t + \gamma^t g(\mathbf{w}^t) + \gamma^t \mathbf{z}^t \tag{A.8}$$

$$= \mathbf{w}^t + \gamma^t \left(\nabla_{\mathbf{w}^t} f(\mathbf{w}^t) + \delta M^t + \beta^t\right) + \gamma^t \mathbf{z}^t. \tag{A.9}$$

Where $\Pi_{\mathcal{H}}\{\mathbf{w}\}$ represents the projection to the closest element (in terms of ℓ_2 norm) of $\mathbf{w}$ in the constraint set $\mathcal{H}$, and where $\delta M^t = g(\mathbf{w}^t) - \mathbb{E}[g(\mathbf{w}^t)|w^1, g(\mathbf{w}^i), i \leq t]$ is a martingale difference noise accounting for the dependence over time in the values of $g(\mathbf{w}^t)$, and β^t is bias term. In the case that interest us in this thesis, the stochastic approximation of the gradient at time t, $g(\mathbf{w}^t)$ is independent of the stochastic approximation of the gradient at any other time t', meaning that $\mathbb{E}_t[\delta M^t] = 0$.

In the same spirit as (A.1), the sequence $\{\mathbf{w}^t\}$ generated by (A.7) can be shown to converge to the mean ODE defined in (A.4) under the following assumptions, described in [65, A.5.2.1-A.5.2.5]:

- (A.5.2.1) $\sup_t \mathbb{E}[g(\mathbf{w})^t] < \infty$,
- (A.5.2.2) $\mathbb{E}[g(\mathbf{w}^t)] = \nabla_{\mathbf{w}^t} f(\mathbf{w}^t) + \beta^t$,
- (A.5.2.3) $\nabla_{\mathbf{w}^t} f(\mathbf{w}^t)$ is continuous,
- (A.5.2.4) $\sum_{t=1}^{\infty} (\gamma^t)^2 < \infty;\ \sum_{t=1}^{\infty} \gamma^t = \infty;\ \gamma^t \geq 0\ \forall t, \gamma^t \to 0$,
- (A.5.2.5) $\sum_{t=1}^{\infty} \gamma^t |\beta^t| < \infty$.

Assumption (A.5.2.5) ensures the bias term becomes negligible and that it decreases faster than the step size.

A.3 The ADAGRAD Variant

As can be seen in Alg. 14, because $\gamma^t \in \mathbb{R}$ all parameters have the same learning rate. However, in practical applications not all parameters have the same impact on the objective function. A desirable property of an optimization algorithm would be to assign high learning rates to parameters having a high impact on the objective and small learning to parameters having a small impact. By doing so, the algorithm would focus on the most the important parameters and converge faster to the local stationary point. This is the driving idea behind the ADAGRAD algorithm, described in Alg. 16.

Algorithm 16: ADAGRAD.

1 **Function** `ADAGRAD(`f`)`
2 Choose initial allocation matrix $\mathbf{w}^1 \in \mathbb{R}^d$, and step-size sequence $\{\gamma^t\}$ with $\gamma^t > 0,\ t = 1, 2, \ldots$.
3 $t \leftarrow 1$
4 **while** *not converged* **do**
5 Based on the observation $\mathbf{x}^t$, compute $g(\mathbf{w}^t; \mathbf{x}^t)$.
6 $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \frac{\gamma^t}{\sqrt{\epsilon I + G_t}} g(\mathbf{w}^t; \mathbf{x}^t)$
`// Projection on` $\Pi_{\mathcal{H}}$ `can be added if needed`
7 $t \leftarrow t + 1$
8 **end**
9 **return** $\mathbf{w}^t$

The step size of ADAGRAD in Step. 6 slightly differs from the one in gradient descent: $\epsilon > 0$ is a term to avoid division by 0, I is the identity matrix and G_t is the matrix of accumulated squared gradient, defined as:

$$G_t = \begin{bmatrix} \sum_{\tau=1}^{t} g(\mathbf{w}^{\tau+1}; \mathbf{x})_1^2 & 0 & \ldots & 0 \\ 0 & \sum_{\tau=1}^{t} g(\mathbf{w}^{\tau+1}; \mathbf{x})_2^2 & \ldots & 0 \\ 0 & \ldots & \ddots & \vdots \\ 0 & \ldots & 0 & \sum_{\tau=1}^{t} g(\mathbf{w}^{\tau+1}; \mathbf{x})_d^2 \end{bmatrix}.$$

FOLIO ADMINISTRATIF

THÈSE DE L'INSA LYON, MEMBRE DE L'UNIVERSITÉ DE LYON

NOM : **JEANNEROT** Date de soutenance : **16/12/2024**

Prénoms : **Alix Sébastien Thibaut**

TITRE : **Uplink Resource Allocation Methods for Next-Generation Wireless Networks**

Nature : **Doctorat** Numéro d'ordre : **2024ISAL0109**

École Doctorale : **Électronique, Électrotechnique et Automatique**

Spécialité : **Traitement du Signal**

Résumé :
Face à la diversité des besoins en communication des réseaux 5G et de la future 6G, l'allocation des ressources disponibles est considérée comme un élément clé pour augmenter la densité de dispositifs, leur débit ou la fiabilité de leurs communications. Dans les réseaux de communication de type machine, des travaux récents ont proposé d'adapter l'allocation des ressources temporelles en fonction du processus sous-jacent qui régit l'activité des dispositifs. Cette thèse se concentre tout d'abord sur l'étude de l'impact d'une connaissance imparfaite de ce processus, et propose des méthodes pour atténuer le biais induit par les connaissances erronées. En second lieu, un algorithme permettant d'optimiser conjointement l'allocation des ressources temporelles et la puissance de transmission des dispositifs est proposé. L'algorithme permet aux dispositifs ayant une forte probabilité de transmettre au même moment, de le faire sur des ressources (temporelles ou de puissance) assurant leur décodabilité. Enfin, dans les réseaux ayant un objectif de haut débit, nous proposons d'optimiser conjointement la puissance, les ressources fréquentielles ainsi que le nombre de flux de données parallèles utilisées par les dispositifs. Notre étude par simulations témoigne que notre optimisation conjointe est plus performante que les méthodes utilisées actuellement en 5G pour lesquelles ces paramètres sont calculés indépendamment les uns des autres.

Mots-clés : **wireless networks, resource allocation, stochastic gradient descent, machine type communication, enhanced mobile broadband**

Laboratoire de recherche : **Centre of Innovation in Telecommunications and Integration of Service (CITI)**

Directeur de thèse : **Jean-Marie GORCE**

Présidente du jury :

Composition du Jury :

Valeria LOSCRI, Gianluigi LIVA, Inbar FIJALKOW, Petar POPOVSKI, Alvaro VALCARCE, Cédric ADJHI, Jean-Marie GORCE, Malcolm EGAN