Resource allocation and optimization for the non-orthogonal multiple access

Lou Salaün

INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2020IPPAT012

Thèse de doctorat

Resource Allocation and Optimization for the Non-Orthogonal Multiple Access

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à Télécom Paris

École doctorale n°626 Institut Polytechnique de Paris (ED IP Paris)
Spécialité de doctorat : Computing, Data and Artificial Intelligence

Thèse présentée et soutenue à Paris, le 12/03/2020, par

LOU SALAÜN

Composition du Jury :

Joanna Tomasik Professeur, Centrale-Supélec	Président
Vincent Poor Professeur, Princeton University	Rapporteur
Pierre Fouilhoux Maître de conférences HDR, Sorbonne Université	Rapporteur
Philippe Ciblat Professeur, Télécom Paris	Examinateur
Joanna Tomasik Professeur, Centrale-Supélec	Examinateur
Marceau Coupechoux Professeur, Télécom Paris	Directeur de thèse
Chung Shue Chen Research Scientist, Nokia Bell Labs	Co-directeur de thèse

Contents

List of Figures

List of Tables

List of Algorithms

Abbreviations

3DM	3-Dimensional Matching
3GPP	3rd Generation Partnership Project
AWGN	Additive White Gaussian Noise
BS	Base Station
CSI	Channel State Information
DGA	Double Gradient Descent Algorithm
DIWA	Double Iterative Waterfilling
DL	Downlink
DPGA	Double Pseudo-Gradient Descent Algorithm
DRL	Deep Reinforcement Learning
EB	Exabyte
eMBB	Enhanced Mobile Broadband
FPTAS	Fully Polynomial-Time Approximation Scheme
FTPC	Fractional Transmit Power Control
GA	Gradient Descent Algorithm
IoT	Internet-of-Things
ITU	International Telecommunication Union
IWA	Iterative Waterfilling Algorithm

JSPA	Joint Subcarrier and Power Allocation
LDDP	Lagrangian Duality and Dynamic Programming
LTE	Long-Term Evolution
M2M	Machine-to-Machine
MCKP	Multiple Choice Knapsack Problem
MC-NOMA	Multi-Carrier Non-Orthogonal Multiple Access
mMTC	Massive Machine Type Communications
NOMA	Non-Orthogonal Multiple Access
NPO	NP Optimization Problem
OFDMA	Orthogonal Frequency-Division Multiple Access
OMA	Orthogonal Multiple Access
PGA	Pseudo-Gradient Descent Algorithm
PTAS	Polynomial-Time Approximation Scheme
RRM	Radio Resource Management
SCPC	Single-Carrier Power Control
SCUS	Single-Carrier User Selection
SIC	Successive Interference Cancellation
SINR	Signal-to-Noise-plus-Interference Ratio
SIWA	Synchronous Iterative Waterfilling Algorithm
SNR	Signal-to-Noise Ratio
SOS	Second Order Statistics
TDMA	Time Division Multiple Access
TSDP	Two-Stage Dynamic Programming

UE	User Equipment
UL	Uplink
URLLC	Ultra Reliable Low Latency Communications
WSR	Weighted Sum-Rate

Acknowledgments

It has been a great pleasure to work towards the PhD degree at *Telecom Paris*, the *Laboratory of Information, Networking and Communication Sciences* (LINCS) and *Nokia Bell Labs*. The last three years have made me grow as a researcher and as a person. Besides acquiring new scientific competences, I learned how to communicate efficiently, how to organize my work, and most importantly, how to solve problems.

First of all, I would like to thank my supervisors, Marceau Coupechoux and Chung Shue (Calvin) Chen. I am grateful for their scientific and technical guidance. They shared with me their valuable knowledge on wireless communications, resource allocation and applied mathematics. They taught me how to write a scientific paper, how to present my research to experts and also to a broader audience. They encouraged me to conduct impactful research, while giving me the freedom to pursue my theoretical research interests.

I would like to express my gratitude to Vincent Poor and Pierre Fouilhoux for reviewing this thesis and for their insightful feedback. I am also thankful to Philippe Ciblat and Joanna Tomasik for being part of my thesis committee.

I would like to thank the people I have collaborated with, Chi Wan Sung, Yaru Fu, Cédric Adjih, Iman Hmedoush, Philippe Jacquet, Amira Alloum, Pierre Bergé, as well as the interns I co-supervised, Shashwat Mishra, Charles Dumas and Mingshan Zhang.

Many thanks to my colleagues at Nokia Bell Labs, Quan, Dalia, Céline, Quentin, Achille, Lamine, Ludovic, Fabio, Gérard, Fabien, Dimitrios, Alonso, Marco, Anne, François, Elie, Nidhi, Lorenzo, for their support and the innumerable interesting discussions we had. My thanks also go to my good friends, Guillaume, Fabien, Christian, Aloïs, Léo, Guilhem, Alexandre, Antoine, and many others.

I would like to express my deepest gratitude to my fiancee, Hélène, my parents and my family for motivating me to strive for the best and always supporting me through good times and bad times.

Abstract

Non-orthogonal multiple access (NOMA) is a promising technology to increase the spectral efficiency and enable massive connectivity in future wireless networks. In contrast to orthogonal schemes, such as OFDMA, NOMA can serve multiple users on the same frequency and time resource by superposing their signal in the power domain. One of the key challenges for radio resource management (RRM) in NOMA systems is to solve the joint subcarrier and power allocation (JSPA) problem.

In this thesis, we present a novel optimization framework to study a general class of JSPA problems. This framework employs a generic objective function which can be used to represent the popular weighted sum-rate (WSR), proportional fairness, harmonic mean and max-min fairness utilities. Our work also integrates various realistic constraints. We prove under this framework that JSPA is NP-hard to solve in general. In addition, we study its computational complexity and approximability in various special cases, for different objective functions and constraints.

In this framework, we first consider the WSR maximization problem subject to cellular power constraint. We propose three new algorithms: Opt-JSPA computes an optimal solution with lower complexity than current optimal schemes in the literature. It can be used as an optimal benchmark in simulations. However, its pseudo-polynomial time complexity remains impractical for real-world systems with low latency requirements. To further reduce the complexity, we propose a fully polynomial-time approximation scheme called ε-JSPA, which allows tight trade-offs between performance guarantee and complexity. To the best of our knowledge, ε-JSPA is the first polynomial-time approximation scheme proposed for this problem. Finally, Grad-JSPA is a heuristic based on gradient descent. Numerical results show that it achieves near-optimal WSR with much lower complexity than existing optimal methods.

As a second application of our framework, we study individual power constraints. Power control is solved optimally by gradient descent methods. Then, we develop

three heuristics: DGA, DPGA and DIWA, which solve the JSPA problem for centralized and distributed settings. The advantage of the distributed approach compared to the centralized one is that it reduces the computational complexity and requires only local information. This can be used to reduce the control signaling overhead in uplink systems for example. The downside is that the distributed power control achieves suboptimal sum-rate performance. We prove that such a distributed problem is a concave game and it has a unique Nash equilibrium. The performance and computational complexity of DGA, DPGA and DIWA are compared through simulations.

Chapter 1

Introduction

1.1 Context

The fifth generation of cellular network technology (5G) is now becoming a reality. It is expected to support three usage scenarios, each with different requirements (see Figures 1.1 and 1.2). First, Enhanced Mobile Broadband (eMBB) is expected to improve the user experience, in particular for video streaming, virtual reality and augmented reality applications. It allows peak data rates of up to 20 Gbps, a 1000-fold increase in the system's capacity, as well as improved mobility management and energy/spectrum efficiency. Secondly, Ultra Reliable Low Latency Communi-

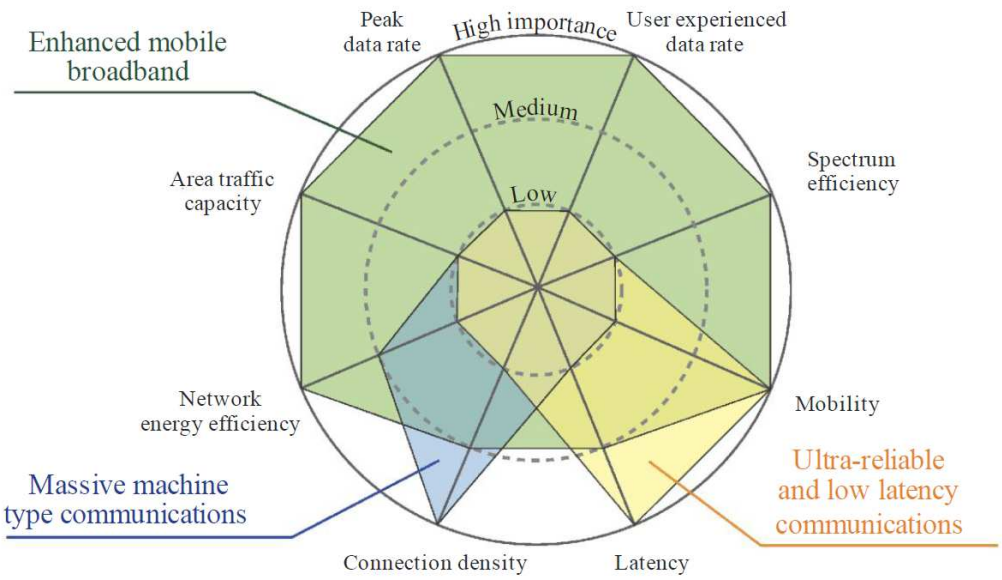

Figure 1.1: Key capabilities in different usage scenarios (source: [1, Figure 4])

cations (URLLC) support mission critical communications with strict requirements on latency and reliability. Some use cases are: remote surgery, autonomous vehicles and industrial automation [2]. Finally, Massive Machine Type Communications (mMTC) deal with connecting a massive number of devices in a dense environment. mMTC would enable Internet-of-Things (IoT) applications such as sensor networks, smart homes, connected wearables and connected car. eMBB will be the first use case implemented in 5G New Radio Phase 1 standards [3], while URLLC and mMTC are expected to be introduced in Phase 2 [4].

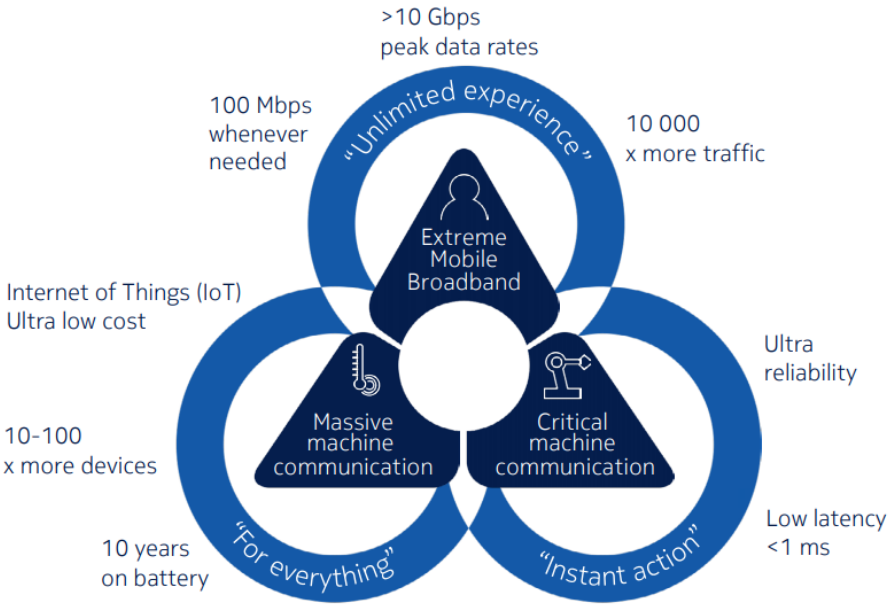

Figure 1.2: 5G enables new capabilities beyond mobile broadband (source: [5, Figure 1])

According to the International Telecommunication Union (ITU), the global mobile traffic is estimated to grow at a rate of around 50% per year between 2020 and 2030 [6]. As shown in Figure 1.3, the global mobile traffic is estimated to increase from 60 Exabytes (EB) in 2020 to 600 EB in 2025 and 5000 EB in 2030. We can see that the traffic for video services and machine-to-machine communications will both increase dramatically.

To meet the ever-increasing demand of wireless access and the constantly evolving requirements in terms of system capacity, data rates, massive connectivity and energy consumption, in 5G and beyond mobile networks, advance in different technologies is needed. Since radio spectrum is scarce, it is of fundamental importance to study how it can be used in the most efficient manner. As radio channel is broadcast

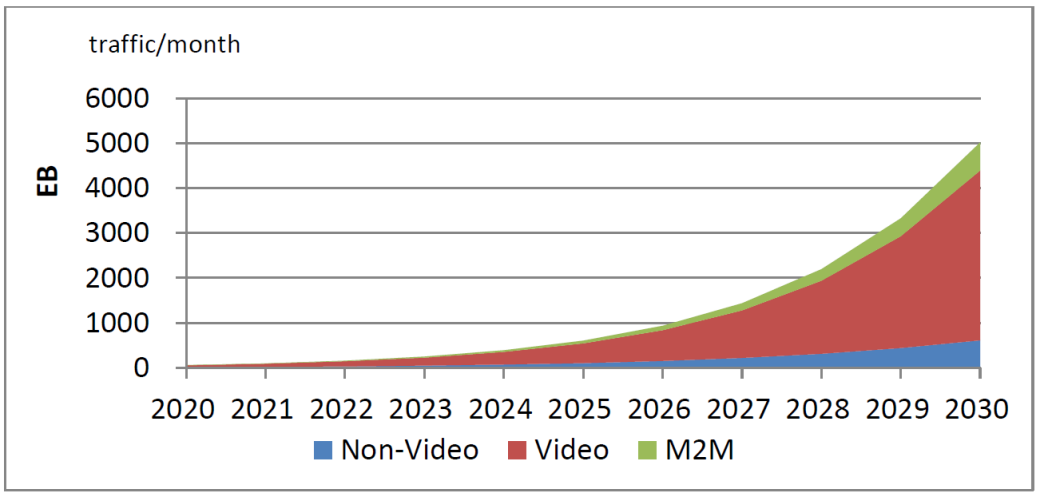

Figure 1.3: Estimations of global mobile traffic by service type from 2020 to 2030 (source: [6, Figure 9])

in nature, transmissions of different users inevitably interfere with one another. To coordinate their transmissions, multiple access schemes are needed and have been playing a major role in the designs of all the generations of mobile cellular networks.

Non-orthogonal multiple access (NOMA) is a promising technology to increase the spectral efficiency and enable massive connectivity in future wireless networks. In contrast to orthogonal schemes, such as orthogonal frequency-division multiple access (OFDMA) adopted in 3GPP-LTE and also 5G New Radio Phase 1 standards [3], NOMA can serve multiple users on the same frequency and time resource by superposing their signal in the power domain. At the receiver side, successive interference cancellation (SIC) is applied to decode the superposed signals.

1.2 Motivation

One of the key challenges for radio resource management (RRM) in NOMA systems is to jointly optimize the power control and subcarrier allocation. Indeed, careful optimization of the transmit powers is required to control the intra-carrier interference of superposed signals and maximize the achievable data rates. Besides, due to error propagation and decoding complexity concerns [7], subcarrier allocation for each transmission also needs to be optimized. This problem is called *joint subcarrier and power allocation* (JSPA) and has been extensively studied in the literature.

JSPA problems can be divided in three categories depending on their objective. First, utility maximization problems consist in maximizing a utility function of the

users' data rates given limited power budget. Secondly, power minimization problems aim at minimizing the system's power consumption subject to minimum data rate requirements (also called QoS constraints) [8]–[10]. Finally, energy efficiency problems combine the above two aspects and maximize the system's throughput to power consumption ratio [11]–[16]. In this thesis, we only consider utility maximization problems.

Several algorithms have been developed to perform power control in various systems and scenarios: Lagrange multiplier methods [13], [17]–[19], dynamic programming [20], monotonic optimization based on the branch-and-bound principle [21], [22], fractional transmit power control (FTPC) [23]–[26] and waterfilling algorithms [27]–[29]. Similarly, the subcarrier allocation has been optimized using matching theory [28], [30]–[32], dynamic programming [20], as well as heuristic user pairing strategies [13], [17], [26], [33]–[35]. What is lacking is a unified framework within which researchers can derive common properties between many problems, develop algorithms that work in general cases and study their computational complexity.

Furthermore, because of JSPA's complexity and intractability in general, optimal algorithms proposed in papers [20]–[22] cannot run in a reasonable amount of time in real systems. Hence, most solutions in the literature are heuristics with no theoretical performance guarantee. For example, a common approach adopted in [13], [17], [25], [26], [30], [33]–[36] and many other papers is to solve separately the power control and subcarrier allocation problems, therefore achieving sub-optimal results. In this context, sub-optimal algorithms with performance guarantees (e.g., polynomial-time approximations) can have a significant impact on the design of RRM schemes. Finding such algorithms remains an open problem.

1.3 Contributions and Structure of the Thesis

With the aforementioned considerations in mind, we propose in this thesis a novel optimization framework with the following aims:

- To provide a unified framework covering a large class of JSPA problems. Most data rates maximization JSPA problems in the literature are included in our framework. Indeed, we employ a generic objective function which covers most utilities considered in the field of resource allocation. Furthermore, we consider the basic power and SIC constraints arising from any NOMA problems.

- To improve the understanding of NOMA optimization problems. We study which properties can simplify the problem (separability, convexity, knapsack constraints, etc.) and which are the intractable structures (NP-hardness of combinatorial problems, multi-dimensional matching, etc.).

- To develop new tools facilitating the algorithm design. For example, although optimal solutions cannot be computed due to the problem's NP-hardness, we show that tight performance guarantees with controllable complexity can be obtained by the use of polynomial-time approximations. The techniques developed in this thesis will in turn allow to tackle efficiently more complex problems in the future.

- To be general and flexible enough to accommodate new constraints and scenarios. For example, QoS constraints are not considered in this work but can be added on top of the existing constraints.

This thesis is organized as follows. We define some mathematical notations, present the system model and explain the principle of NOMA in Chapter 2.

In Chapter 3, we present the general optimization framework, which will be used throughout this thesis to study various JSPA problems both in terms of computational complexity and algorithm design. It employs a generic objective function, namely the weighted generalized mean of the data rates, which can be used to represent the popular weighted sum-rate, proportional fairness, harmonic mean and max-min fairness utilities. Our framework also integrates realistic constraints: First, the cellular power constraint used in downlink systems to model the power budget available at the base station. Then, the individual power constraints used in uplink systems to model each user equipment's power budget. Finally, the combinatorial constraint which limits the number of superposed signals per subcarrier due to error propagation and decoding complexity issues [7]. We prove under this framework that JSPA is NP-hard to solve in general. In addition, we study its computational complexity and approximability in various special cases, for different objective functions and constraints.

In the large class of objective functions covered by our framework, the weighted sum-rate (WSR) has received much attention as it can achieve various tradeoffs between sum-rate performance and user fairness. We consider the WSR maximization problem subject to cellular power constraint in Chapter 4. We propose three new algorithms, namely Opt-JSPA, ε-JSPA and Grad-JSPA. Opt-JSPA computes

an optimal solution with lower complexity than current optimal schemes in the literature. It can be used as a benchmark for optimal WSR performance in simulations. However, its pseudo-polynomial time complexity remains impractical for real-world systems with low latency requirements. To further reduce the complexity, we propose a fully polynomial-time approximation scheme called ε-JSPA. It stands out by allowing to control a tight trade-off between performance guarantee and complexity. To the best of our knowledge, ε-JSPA is the first polynomial-time approximation scheme proposed for this problem. Finally, Grad-JSPA is a heuristic based on gradient descent. Numerical results show that it achieves near-optimal WSR with much lower complexity than existing optimal methods.

In Chapter 5, we study the sum-rate maximization problem with individual power constraints, as another application of our framework. We solve the power control sub-problem by an optimal gradient descent algorithm called GA. Furthermore, we propose a distributed game theoretic variant of this sub-problem in which each user optimizes its power allocation using local information only. The advantage of this approach compared to the centralized one is that it reduces the computational complexity and requires only local information. This can be used to reduce the control signaling overhead in uplink systems for example. The downside is that the distributed power control achieves suboptimal sum-rate performance. We prove that this problem is a concave game and it has a unique Nash equilibrium. We develop a pseudo-gradient descent method (PGA) which converges to the Nash equilibrium, and a heuristic iterative waterfilling algorithm (IWA). Then, we propose a three-step heuristic to perform joint subcarrier and power allocation. This three-step methodology requires to solve two power control sub-problems, which can be done by any of the aforementioned schemes. When either GA, PGA or SIWA is applied twice, we get respectively the double gradient algorithm (DGA), the double pseudo-gradient algorithm (DPGA) and the double iterative waterfilling algorithm (DIWA). We compare their performance and computational complexity through simulations.

We conclude in Chapter 6 and discuss about possible extensions of our framework, as well as open problems related to the design of efficient and practical JSPA schemes for NOMA.

1.4 Related Publications

[P1] L. Salaün, M. Coupechoux, and C. S. Chen, "Joint subcarrier and power allocation in NOMA: Optimal and approximate algorithms," under review, 2019. arXiv: 1910.00510 [math.OC].

[P2] ——, "Weighted sum-rate maximization in multi-carrier NOMA with cellular power constraint," in *IEEE Int. Conf. on Computer Commun. (Infocom)*, 2019, pp. 451–459.

[P3] L. Salaün, C. S. Chen, and M. Coupechoux, "Optimal joint subcarrier and power allocation in NOMA is strongly NP-hard," in *IEEE Int. Conf. Commun. (ICC)*, 2018.

[P4] Y. Fu, L. Salaün, C. W. Sung, and C. S. Chen, "Subcarrier and power allocation for the downlink of multicarrier NOMA systems," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 11 833–11 847, 2018.

[P5] Y. Fu, L. Salaün, C. W. Sung, C. S. Chen, and M. Coupechoux, "Double iterative waterfilling for sum rate maximization in multicarrier NOMA systems," in *IEEE Int. Conf. Commun. (ICC)*, 2017.

[P6] Y. Fu, L. Salaün, C. W. Sung, and C. S. Chen, "Distributed power allocation for the downlink of a two-cell MISO-NOMA system," in *IEEE 87th Veh. Technol. Conf. (VTC Spring)*, 2018.

Chapter 2

System Model and Notations

2.1 Mathematical Notations and Basic Definitions

2.1.1 Scalars, Vectors and Sets

We denote scalars by plain lower case letters, e.g., x, p, and vectors by bold lower case letters, e.g., $\boldsymbol{x}$, $\boldsymbol{p}$. In order to differentiate variables from constant scalars in the optimization framework, we may use plain upper case letters for constants when suitable, e.g., P, M and W. The i-th element of a vector $\boldsymbol{x}$ is denoted by x_i or x^i, depending on the circumstances. A vector $\boldsymbol{x}$ containing elements $x_1, \ldots, x_N$ can be represented by $\boldsymbol{x} = (x_1, \ldots, x_N)$ or $\boldsymbol{x} = (x_i)_{i\in\{1,\ldots,N\}}$. The dot product $\boldsymbol{x}\cdot\boldsymbol{y}$ between two d-dimensional vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined as:

$$\boldsymbol{x}\cdot\boldsymbol{y} \triangleq \sum_{i=1}^{d} x_i y_i.$$

In this work, we consider the l_1 norm defined as:

$$\|\boldsymbol{x}\|_1 \triangleq \sum_{i=1}^{d} |x_i|,$$

and the l_2 (Euclidean) norm:

$$\|\boldsymbol{x}\|_2 \triangleq \sqrt{\boldsymbol{x}\cdot\boldsymbol{x}} = \sqrt{\sum_{i=1}^{d} x_i^2}.$$

Finite sets are represented by upper case calligraphic letters, e.g., $\mathcal{K}$, $\mathcal{N}$, $\mathcal{U}$. The cardinality of a finite set $\mathcal{K}$ is denoted by $|\mathcal{K}|$. Let $\mathbb{R}$, $\mathbb{Q}$ and $\mathbb{N}$ be the set of real numbers, rational numbers and natural numbers (non-negative integers),

respectively. We define similarly $\mathbb{R}_+$, $\mathbb{Q}_+$ and $\mathbb{N}_+$ to be their restriction to positive numbers, and $\mathbb{R}_{0+} \triangleq \mathbb{R}_+ \cup \{0\}$ is the set of non-negative reals. The interval of real numbers between $a \in \mathbb{R}$ and $b \in \mathbb{R}$ is (a, b) if it is open, and $[a, b]$ if it is closed. In addition, the set of integers from 1 to $n \in \mathbb{N}$ is denoted by $\{1, \ldots, n\}$.

2.1.2 Functions and their Properties

Let $n \in \mathbb{N}$, any bijective function from $\{1, \ldots, n\}$ onto $\{1, \ldots, n\}$ is called a permutation of $\{1, \ldots, n\}$. We represent permutations by the lower case greek letter $\pi \colon \{1, \ldots, n\} \to \{1, \ldots, n\}$.

Besides permutations, we mainly consider real-valued functions $f \colon D \to \mathbb{R}$, where $D \subset \mathbb{R}^d$ and $d \in \mathbb{N}_+$. The gradient of a differentiable function f is denoted by ∇f, and the Hessian is $\nabla^2 f$. If its domain D lies in $\mathbb{R}$ (i.e., $d = 1$), then the notion of gradient reduces to the derivative, which is denoted by f'. Similarly, the Hessian becomes the second derivative, denoted by f''. In addition, notations ln, log, and $\log_a$ represent respectively the natural, base-10 and base-a logarithms defined on $D = \mathbb{R}_+$.

We present here some basic notions of convex analysis. A set S is convex if for all $\boldsymbol{x}, \boldsymbol{y} \in S$ and $t \in [0, 1]$, $t\boldsymbol{x} + (1 - t)\boldsymbol{y} \in S$. The convexity of a function is defined as follows.

Definition 1 (Convexity and Concavity of a function)**.**
A real-valued function $f \colon D \to \mathbb{R}$ is convex if for all $\boldsymbol{x}, \boldsymbol{y} \in D$ and $t \in [0, 1]$, we have:

$$f(t\boldsymbol{x} + (1 - t)\boldsymbol{y}) \leq tf(\boldsymbol{x}) + (1 - t)f(\boldsymbol{y})\,. \tag{2.1}$$

If f is differentiable, the convexity can be characterized by:

$$(\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})) \cdot (\boldsymbol{x} - \boldsymbol{y}) \geq 0. \tag{2.2}$$

Furthermore, f is strictly convex if the inequalities in Eqn. (2.1) and (2.2) are strict. We say that f is concave (resp. strictly concave) if $-f$ is convex (resp. strictly convex).

In convex optimization, the convergence rate of gradient descent algorithms depends on additional properties on f, namely the α-strong convexity and β-smoothness that we present in Definition 2. A detailed study of these properties and their implications can be found in [37, Section 9.1.2].

Definition 2 (Strong convexity and smoothness of a function)**.**
Let $\beta \geq \alpha > 0$, a real-valued function $f\colon D \to \mathbb{R}$ is α-strongly convex and β-smooth if for all $\boldsymbol{x}, \boldsymbol{y} \in D$ and $t \in [0,1]$, we have:

$$\frac{-\alpha t(1-t)}{2}\|\boldsymbol{x}-\boldsymbol{y}\|_2^2 \geq f(t\boldsymbol{x}+(1-t)\boldsymbol{y}) - tf(\boldsymbol{x}) - (1-t)f(\boldsymbol{y}) \geq \frac{-\beta t(1-t)}{2}\|\boldsymbol{x}-\boldsymbol{y}\|_2^2.$$

If f is differentiable, an equivalent definition is:

$$\beta\|\boldsymbol{x}-\boldsymbol{y}\|_2^2 \geq (\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})) \cdot (\boldsymbol{x}-\boldsymbol{y}) \geq \alpha\|\boldsymbol{x}-\boldsymbol{y}\|_2^2.$$

2.2 Principle of NOMA

In multiple access systems, the total frequency bandwidth and transmission duration are divided into resource blocks called subcarriers. The subcarriers are assigned to users, allowing them to transmit on given radio bandwidths and time slots. Orthogonal multiple access (OMA) refers to technologies in which radio resources allocated to different users are orthogonal (i.e., non-interfering) in time and frequency. As illustrated in Figure 2.1, only one user is served on each subcarrier and their signals do not interfere with other signals. Orthogonal frequency-division multiple access (OFDMA) adopted in 3GPP-LTE and also 5G New Radio Phase 1 standards [3] is an example of OMA scheme. In summary, OMA aims to avoid interference among users and have low-complexity signal decoding at the receiver side. However, such orthogonal schemes cannot always achieve the capacity of multi-user systems [7] and are suboptimal in terms of spectral efficiency [38].

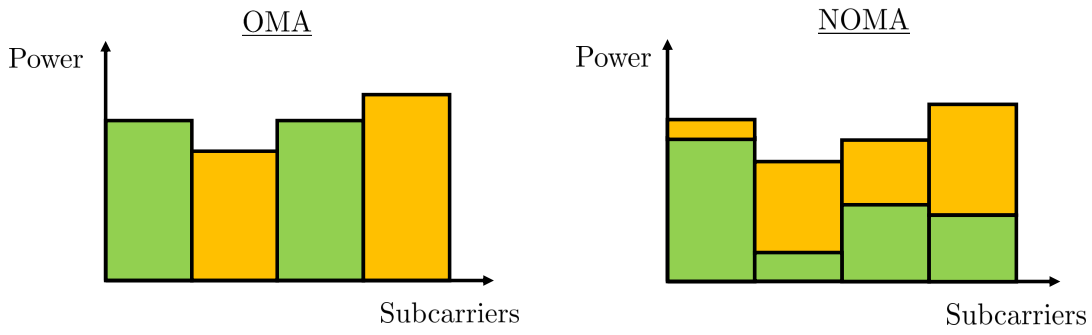

Figure 2.1: OMA vs NOMA. The two colors – and – represent the transmit power of two different users' signals

In contrast with OMA, NOMA allows to superpose several signals on the same subcarrier. In this work, we consider power domain NOMA in which the superposed signals are multiplexed in the power domain, as illustrated in Figure 2.1. The interference received due to superposed signals are mitigated by successive interference cancellation (SIC) [7], [24], [39]. To ensure successful decoding of the superposed signals and reduce the interference, the power of each signal should be thoroughly optimized. We explain in the next subsections the principle of NOMA – signal superposition and SIC – in both downlink and uplink scenarios. Moreover, we show that NOMA achieves higher data rates than OMA in a two users system.

2.2.1 Uplink transmission

We first describe how OMA works in a simple system with two users. Then, we explain the principle of NOMA in the uplink and compare its achievable rate region to OMA.

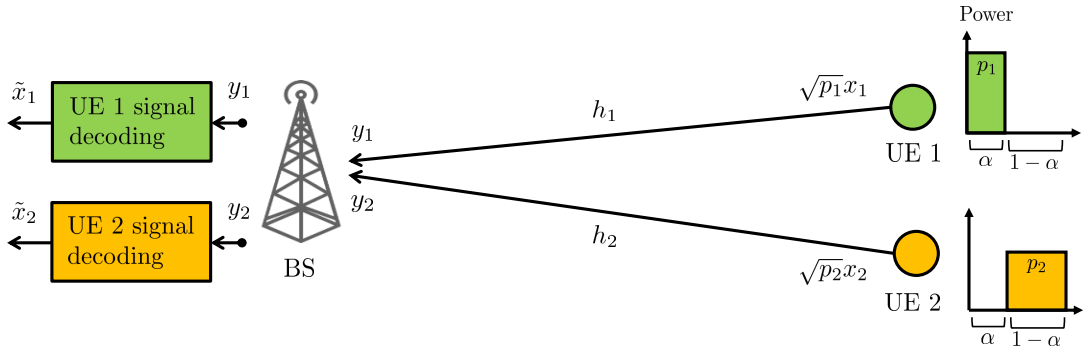

Figure 2.2: Two users OMA transmission in the uplink

We consider two users transmitting respectively signals x_1 and x_2 to the base station (BS). The power of signals x_1 and x_2 are p_1 and p_2, respectively. In the OMA system shown in Figure 2.2, user 1 is served on a fraction $\alpha \in [0, 1]$ of the total bandwidth (or equivalently, a fraction of the subcarriers). User 2 is served on a proportion $1 - \alpha$ of the total bandwidth. The received signal of user 1 at the BS is:

$$y_1 = h_1\sqrt{p_1}x_1 + n,$$

and the received signal of user 2 is:

$$y_2 = h_2\sqrt{p_2}x_2 + n,$$

where n is a Gaussian noise, h_1 and h_2 are the complex channel coefficients between the BS and users 1 and 2, respectively. In Figure 2.2, $\tilde{x}_1$ and $\tilde{x}_2$ are estimations of the input signals x_1 and x_2. They are obtained by decoding independently y_1 and y_2, as they are transmitted on orthogonal resources, i.e., non-interfering subcarriers. Given the total bandwidth W and the noise power spectral density N_0, the signal-to-noise ratio of user 1 and 2 are:

$$SNR_1 = \frac{g_1 p_1}{\alpha W N_0} \quad \text{and} \quad SNR_2 = \frac{g_2 p_2}{(1-\alpha) W N_0},$$

where $g_1 = |h_1|^2$ and $g_2 = |h_2|^2$ are the channel gains. The achievable data rates of each user, R_1 and R_2, can be obtained by Shannon capacity formula as follows:

$$R_1 = \alpha W \log_2\left(1 + \frac{g_1 p_1}{\alpha W N_0}\right), \tag{2.3}$$

$$R_2 = (1-\alpha)\, W \log_2\left(1 + \frac{g_2 p_2}{(1-\alpha)\, W N_0}\right). \tag{2.4}$$

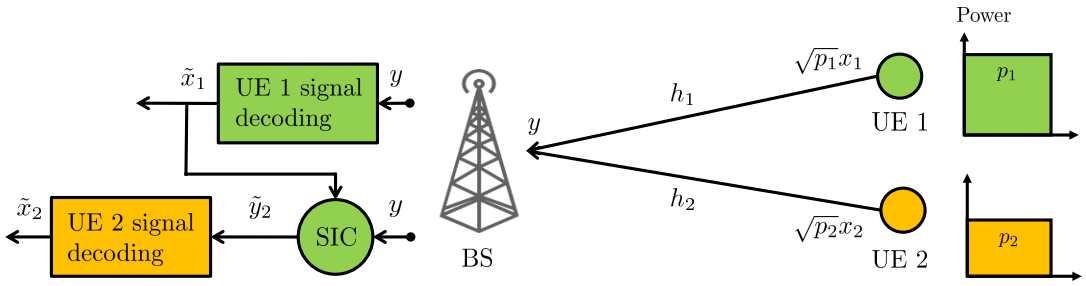

Figure 2.3: Two users NOMA transmission in the uplink

In the NOMA system presented in Figure 2.3, both users transmit on the same bandwidth W. The received signal at the BS is hence a superposition of x_1 and x_2:

$$y = h_1\sqrt{p_1}x_1 + h_2\sqrt{p_2}x_2 + n.$$

In this example, x_1 is decoded first by considering x_2 as noise. Its signal-to-interference-plus-noise ratio is:

$$SINR_1 = \frac{g_1 p_1}{W N_0 + g_2 p_2}.$$

Assuming that user 1's signal as been correctly decoded, $\tilde{x}_1$ is then subtracted from the superposed signal y (see block SIC in Fig. 2.3) to get signal $\tilde{y}_2 \approx h_2\sqrt{p_2}x_2 + n$, in which the interference caused by user 1 is canceled out. This is the principle of successive interference cancellation (SIC): when multiple signals are superposed, the first decoded signals are subtracted from y before decoding the next signal. The signal-to-interference-plus-noise ratio of x_2 after SIC is:

$$SINR_2 = \frac{g_2 p_2}{W N_0}.$$

The idea of power domain NOMA is to select signals to be multiplexed (based on their channel gains g_1 and g_2) and optimize their respective power so that all signals can be successfully decoded and to maximize each user's data rate. In this case, the NOMA data rates are:

$$R_1' = W \log_2\left(1 + \frac{g_1 p_1}{W N_0 + g_2 p_2}\right), \tag{2.5}$$

$$R_2' = W \log_2\left(1 + \frac{g_2 p_2}{W N_0}\right). \tag{2.6}$$

If x_2 is decoded first, the rates become:

$$R_1' = W \log_2\left(1 + \frac{g_1 p_1}{W N_0}\right), \tag{2.7}$$

$$R_2' = W \log_2\left(1 + \frac{g_2 p_2}{W N_0 + g_1 p_1}\right). \tag{2.8}$$

Figure 2.4 shows an example of the two users' achievable rates in the uplink. The curves correspond to the boundaries of each scheme's rate region (capacity region). That is, all points (i.e., rate pairs) below the curves are also achievable rates. The OMA curve is obtained by varying α from 0 to 1 in Eqn. (2.3) and (2.4). Point A corresponds to Eqn. (2.5) and (2.6), in which user 1 is decoded first. In the same way, point C corresponds to Eqn. (2.7) and (2.8), in which user 2 is decoded first. The rest of the NOMA curve is computed by sharing different fractions $\alpha \in [0, 1]$ of the transmission time between the two operating points A and C. We can see that NOMA achieves greater rates than OMA, except on point B where $\alpha = \frac{g_1 p_1}{g_1 p_1 + g_2 p_2}$. However, this operating point may be undesired as it gives more rate to the strong user 2 than to the weak user 1.

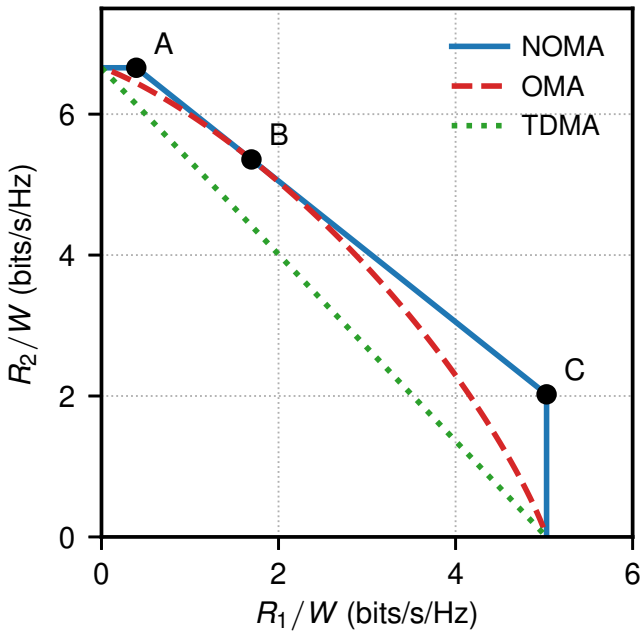

Figure 2.4: Capacity regions of two users accessing an uplink AWGN channel using NOMA, OMA or TDMA. Numerical values: $\frac{g_1 p_1}{W N_0} = 15$ dB and $\frac{g_2 p_2}{W N_0} = 20$ dB

In practical systems, allocating a fraction α of the total bandwidth is not always possible and depends on the granularity of the frequency resources. As a conse-

quence, the number of connected users in OMA is limited by the scheduling granularity and the subcarriers size. For example, if only one subcarrier were available at each time slot, then the rates achieved by sharing time would be greatly reduced (see TDMA curve in Fig. 2.4). In this regard, NOMA can serve significantly more users than OMA. Since NOMA allows signal superposition, it can also be applied to grant-free uplink transmissions for which scheduling is not possible due to low latency requirements [40], [41]. This shows that NOMA is a promising technology to achieve massive machine-type communications (mMTC) in beyond 5G wireless networks [42].

2.2.2 Downlink transmission

In the downlink, the BS serves users 1 and 2 with power p_1 and p_2, respectively. These power satisfy $p_1 + p_2 \leq P$, where P is the total power budget available at the BS. In the OMA system shown in Figure 2.5, the two users are served on orthogonal subcarriers, where user 1 gets $\alpha \in [0, 1]$ of the total bandwidth and user 2 gets $(1 - \alpha)$ of the total bandwidth. Their received signals are:

$$y_1 = h_1\sqrt{p_1}x_1 + n,$$
$$y_2 = h_2\sqrt{p_2}x_2 + n.$$

Therefore, their achievable data rates are:

$$R_1 = \alpha W \log_2\left(1 + \frac{g_1 p_1}{\alpha W N_0}\right), \tag{2.9}$$
$$R_2 = (1-\alpha)\, W \log_2\left(1 + \frac{g_2 p_2}{(1-\alpha)\, W N_0}\right), \tag{2.10}$$

which are similar to Eqn. (2.3) and (2.4) obtained in the uplink case.

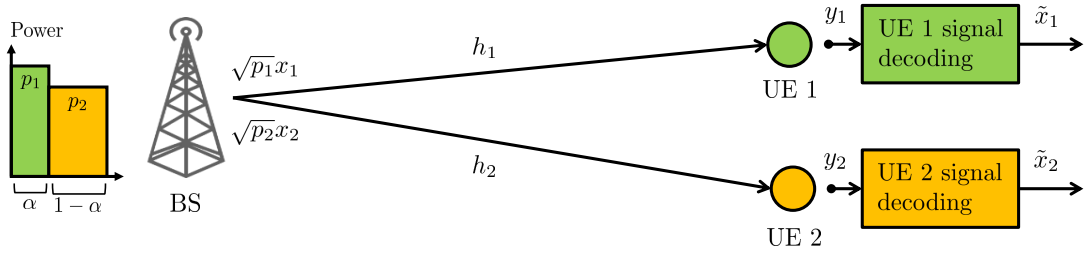

Figure 2.5: Two users OMA transmission in the downlink

Figure 2.3 illustrates the principle of NOMA in the downlink. The BS transmits on bandwidth W the superposed signal $\sqrt{p_1}x_1 + \sqrt{p_2}x_2$. The received signal y_1 and

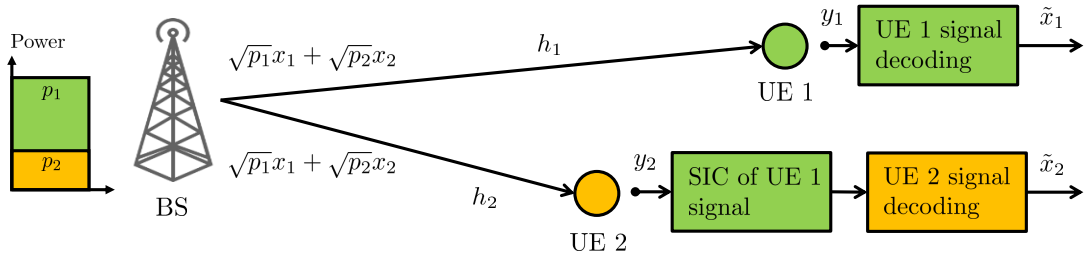

Figure 2.6: Two users NOMA transmission in the downlink

y_2 of each user is hence a superposition of x_1 and x_2:

$$y_1 = h_1 \left(\sqrt{p_1}x_1 + \sqrt{p_2}x_2\right) + n,$$
$$y_2 = h_2 \left(\sqrt{p_1}x_1 + \sqrt{p_2}x_2\right) + n.$$

In this example, user 1 is considered as the weak user, and user 2 as the strong user, i.e., $g_1 < g_2$. It has been proven in [7, Section 6.2] that the optimal rates can only be achieved by decoding the weak user 1 first, before performing SIC and decoding the strong user 2. This downlink optimal decoding order only depends on the channel conditions g_1 and g_2. In particular, it does not depend on the transmit power.

Following the optimal decoding order at user 1 receiver side, x_1 is decoded by considering x_2 as noise. Hence, its signal-to-interference-plus-noise ratio is:

$$SINR_1 = \frac{g_1 p_1}{WN_0 + g_1 p_2}.$$

Note that here the interference term is $g_1 p_2$, whereas it is $g_2 p_2$ in the uplink. At user 2 receiver side, x_1 is also decoded by considering x_2 as noise. Then user 2 subtract $\tilde{x}_1$ from the superposed signal y_2 before decoding its own signal x_2. The signal-to-interference-plus-noise ratio of x_2 after SIC is:

$$SINR_2 = \frac{g_2 p_2}{WN_0}.$$

The NOMA data rates are:

$$R_1' = W \log_2\left(1 + \frac{g_1 p_1}{WN_0 + g_1 p_2}\right), \tag{2.11}$$
$$R_2' = W \log_2\left(1 + \frac{g_2 p_2}{WN_0}\right). \tag{2.12}$$

The NOMA curve in Figure 2.7 is obtained by varying the power allocation p_1 and p_2 in Eqn. (2.11) and (2.12), subject to constraint $p_1 + p_2 = P$. We compute OMA similarly from Eqn. (2.9) and (2.10) by also varying α from 0 to 1. In NOMA

(1), the optimal decoding order is adopted to achieve rates (2.11) and (2.12), in which the weak user 1 is decoded first. NOMA (2) refers to decoding the strong user 2 first. NOMA (1) rate region boundary is strictly above that of OMA. This shows that NOMA can achieve greater spectral efficiency than OMA. Furthermore, the decoding order of SIC is particularly important as the data rates achieved by NOMA (2) are always suboptimal and even lower than that of OMA and TDMA. Unlike in the uplink where the decoding order may vary depending on the received power and the desired rate pairs in Fig. 2.4, the optimal decoding order in the downlink only depends on the channel conditions.

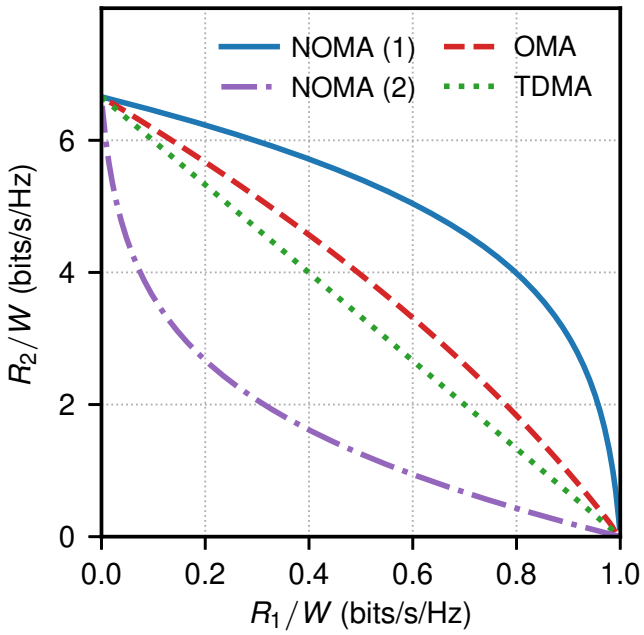

Figure 2.7: Two users downlink capacity regions. NOMA (1) corresponds to decoding the weak user 1 first, while NOMA (2) refers to decoding the strong user 2 first. Numerical values: $P = p_1 + p_2 = 1$ W, $\frac{g_1 P}{W N_0} = 0$ dB and $\frac{g_2 P}{W N_0} = 20$ dB

2.3 System Model

We define in this section the system model and notations used throughout this thesis. We consider a multi-carrier NOMA (MC-NOMA) system composed of one base station (BS) serving K users. Both downlink and uplink transmissions are considered. An example of downlink system is illustrated in Figure 2.8. We denote the index set of users by $\mathcal{K} \triangleq \{1, \ldots, K\}$, and the set of subcarriers by $\mathcal{N} \triangleq \{1, \ldots, N\}$.

The total system bandwidth W is divided into N subcarriers of bandwidth W_n, for each $n \in \mathcal{N}$, such that $\sum_{n\in\mathcal{N}} W_n = W$. We assume orthogonal frequency division, so that adjacent subcarriers do not interfere each other. Moreover, each subcarrier $n \in \mathcal{N}$ experiences frequency-flat block fading on its bandwidth W_n.

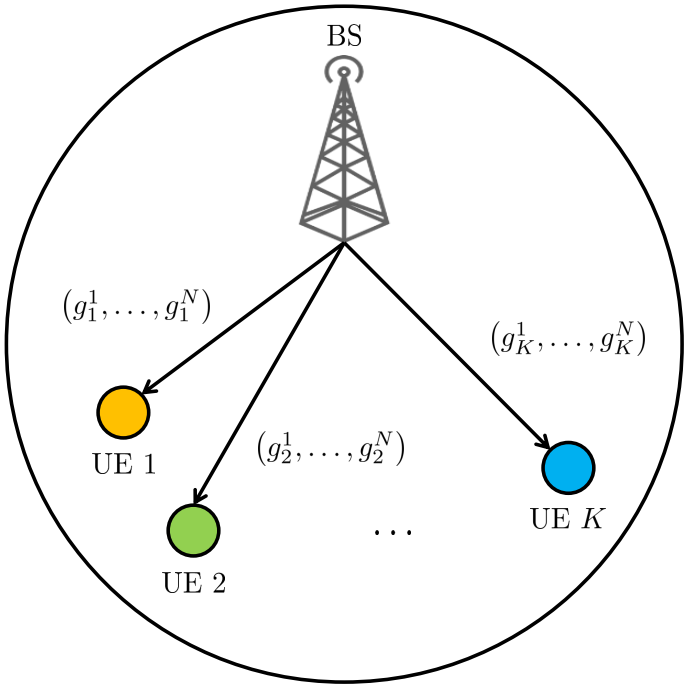

Figure 2.8: Downlink system model

For $k \in \mathcal{K}$ and $n \in \mathcal{N}$, let g_k^n and η_k^n be the channel gain and the received noise power of user k on subcarrier n. We assume that the channel can be accessed through either downlink or uplink transmissions. For downlink scenarios, the BS transmits a signal to each user k on subcarrier n with power p_k^n. For uplink scenarios, each user k transmits a signal to the BS on subcarrier n with power p_k^n. In both cases, we refer to p_k^n as "the allocated transmit power of user k" on subcarrier n. User k is said to be active on subcarrier n if $p_k^n > 0$, and inactive otherwise. For simplicity of notations, we define the normalized noise power as $\tilde{\eta}_k^n \triangleq \eta_k^n / g_k^n$. We denote by $\boldsymbol{p} \triangleq (p_k^n)_{k\in\mathcal{K},n\in\mathcal{N}}$ the vector of all transmit powers, $\boldsymbol{p}^n \triangleq (p_k^n)_{k\in\mathcal{K}}$ is the vector of all transmit powers on subcarrier n, and $\boldsymbol{p}_k \triangleq (p_k^n)_{n\in\mathcal{N}}$ is the vector of user k's transmit powers.

In power domain NOMA, several users are multiplexed on the same subcarrier using superposition coding. A common approach adopted in the literature is to limit the number of superposed signals on each subcarrier to be no more than M. The value of M is meant to characterize practical limitations of SIC due to decod-

ing complexity and error propagation [7]. We represent the set of active users on subcarrier n by $\mathcal{U}_n \triangleq \{k \in \mathcal{K} \colon p_k^n > 0\}$. The aforementioned constraint can then be formulated as $\forall n \in \mathcal{N},\ |\mathcal{U}_n| \leq M$. Each subcarrier is modeled as a multi-user Gaussian broadcast channel and SIC is applied at the receiver side to mitigate intra-band interference.

The SIC decoding order on subcarrier n is usually defined as a permutation function over the active users on n, i.e., $\pi_n \colon \{1, \ldots, |\mathcal{U}_n|\} \to \mathcal{U}_n$. However, for ease of reading, we choose to represent it by a permutation over all users $\mathcal{K}$, i.e., $\pi_n \colon \{1, \ldots, K\} \to \mathcal{K}$. These two definitions are equivalent in our model since the data rates (2.15) and (2.16) does not depend on the inactive users $k \in \mathcal{K} \setminus \mathcal{U}_n$, for which $p_k^n = 0$. For $i \in \{1, \ldots, K\}$, $\pi_n(i)$ returns the i-th decoded user's index. Conversely, user k's decoding order is given by $\pi_n^{-1}(k)$. Hence, the signals of users $\pi_n(1), \ldots, \pi_n(i-1)$ are decoded and subtracted from the superposed signal before decoding $\pi_n(i)$'s signal. Furthermore, user $\pi_n(i)$ is subject to interference from users $\pi_n(j)$, for $j > i$. In particular, $\pi_n(|\mathcal{U}_n|)$ is decoded last and is not subject to any intra-band interference if the previous $|\mathcal{U}_n| - 1$ users have been successfully decoded.

Note that the above discussion can be applied to both uplink and downlink cases. However, the decoding order should be chosen differently depending on which case we are addressing. For downlink scenarios, the optimal decoding order obeys the following sorting [7, Section 6.2]:

$$\frac{\eta_{\pi_n(1)}^n}{g_{\pi_n(1)}^n} \geq \frac{\eta_{\pi_n(2)}^n}{g_{\pi_n(2)}^n} \geq \cdots \geq \frac{\eta_{\pi_n(K)}^n}{g_{\pi_n(K)}^n},$$

which can be written as:

$$\tilde{\eta}_{\pi_n(1)}^n \geq \tilde{\eta}_{\pi_n(2)}^n \geq \cdots \geq \tilde{\eta}_{\pi_n(K)}^n. \tag{2.13}$$

As explained in Subsection 2.2.2 and illustrated in Fig. 2.7, this decoding order is optimal in the sense that it achieves greater rates than any other decoding order. For uplink scenarios, the decoding order depends on the desired operating point in the capacity region (e.g., Fig. 2.4). Nevertheless, if the goal is to minimize the total transmit power to meet given target rates, then [7, Section 6.1] shows that the optimal decoding order consists in decoding the strongest user first (in terms of received power) and continuing towards the weakest user:

$$g_{\pi_n(1)}^n p_{\pi_n(1)}^n \geq g_{\pi_n(2)}^n p_{\pi_n(2)}^n \geq \cdots \geq g_{\pi_n(K)}^n p_{\pi_n(K)}^n. \tag{2.14}$$

This decoding order also ensures that the weakest user gets the best possible rate, which is known to be the *max-min fair* operating point in the capacity region.

Shannon capacity formula is applied to model the capacity of a communication link, i.e., the maximum achievable data rate. Regarding the downlink, the achievable data rate of user $k \in \mathcal{K}$ on subcarrier $n \in \mathcal{N}$ is given by:

$$\begin{aligned} R_k^n(\boldsymbol{p}^n) &\triangleq W_n \log_2\left(1 + \frac{g_k^n p_k^n}{\sum_{j=\pi_n^{-1}(k)+1}^{K} g_k^n p_{\pi_n(j)}^n + \eta_k^n}\right), \\ &\overset{(a)}{=} W_n \log_2\left(1 + \frac{p_k^n}{\sum_{j=\pi_n^{-1}(k)+1}^{K} p_{\pi_n(j)}^n + \tilde{\eta}_k^n}\right), \end{aligned} \tag{2.15}$$

where equality (a) is obtained after normalizing by g_k^n.
In the uplink, since the only receiver is the BS, all users transmitting on subcarrier n are subject to the same noise $\eta^n = \eta_k^n$, $k \in \mathcal{K}$. The data rate can be written as:

$$R_k^n \triangleq W_n \log_2\left(1 + \frac{g_k^n p_k^n}{\sum_{j=\pi_n^{-1}(k)+1}^{|\mathcal{U}_n|} g_{\pi_n(j)}^n p_{\pi_n(j)}^n + \eta^n}\right). \tag{2.16}$$

We assume perfect SIC, therefore interference from users $\pi_n(j)$ for $j < \pi_n^{-1}(k)$ is completely removed in Eqn. (2.15) and (2.16). In other words, user k is only subject to interference from users $\pi_n(j)$, $j > \pi_n^{-1}(k)$.

For ease of reading, let us define the following notations: $R_k \triangleq \sum_{n\in\mathcal{N}} R_k^n$ represents user k's individual data rate, while $R^n \triangleq \sum_{k\in\mathcal{K}} R_k^n$ corresponds to the sum of data rates achieved on subcarrier n. We denote by $\boldsymbol{R} \triangleq (R_k)_{k\in\mathcal{K}}$ the individual data rates vector. Data rates are function of the power allocation, nevertheless we use the notations R_k^n, R_k, R^n and $\boldsymbol{R}$ instead of $R_k^n(\boldsymbol{p})$, $R_k(\boldsymbol{p})$, $R^n(\boldsymbol{p})$, $\boldsymbol{R}(\boldsymbol{p})$, for simplicity.

We summarize the system model and related notations in Table 2.1.

Table 2.1: System model and notations

Notation	**Description**
K	Number of users
$\mathcal{K}$	Index set of all users $\mathcal{K} \triangleq \{1, \ldots, K\}$
N	Number of subcarriers
$\mathcal{N}$	Index set of all subcarriers $\mathcal{N} \triangleq \{1, \ldots, N\}$
W_n	Bandwidth of subcarrier $n \in \mathcal{N}$
W	Total bandwidth, $\sum_{n\in\mathcal{N}} W_n = W$
g_k^n	Channel gain of user $k \in \mathcal{K}$ on subcarrier $n \in \mathcal{N}$
η_k^n	Received noise power of user $k \in \mathcal{K}$ on subcarrier $n \in \mathcal{N}$. In the uplink, the noise power at the BS only depends on n, i.e., $\eta_k^n = \eta^n$, for all $k \in \mathcal{K}$
$\tilde{\eta}_k^n$	Normalized noise power of user $k \in \mathcal{K}$ on subcarrier $n \in \mathcal{N}$, $\tilde{\eta}_k^n \triangleq \eta_k^n / g_k^n$
p_k^n	In the downlink, p_k^n is the transmit power from the BS to user $k \in \mathcal{K}$ on subcarrier $n \in \mathcal{N}$. In the uplink, p_k^n is the transmit power of user $k \in \mathcal{K}$ on subcarrier $n \in \mathcal{N}$ to the BS
$\boldsymbol{p}$	Vector of all transmit powers, $\boldsymbol{p} \triangleq (p_k^n)_{k\in\mathcal{K},n\in\mathcal{N}}$
$\boldsymbol{p}^n$	Vector of all transmit powers on subcarrier n, $\boldsymbol{p}^n \triangleq (p_k^n)_{k\in\mathcal{K}}$
$\boldsymbol{p}_k$	Vector of user k's transmit powers, $\boldsymbol{p}_k \triangleq (p_k^n)_{n\in\mathcal{N}}$
$\mathcal{U}_n$	Set of active users on subcarrier $n \in \mathcal{N}$, $\mathcal{U}_n \triangleq \{k \in \mathcal{K} \colon p_k^n > 0\}$
M	Maximum number of active users per subcarrier
π_n	Decoding order on subcarrier $n \in \mathcal{N}$
R_k^n	Achievable data rate of user $k \in \mathcal{K}$ on subcarrier $n \in \mathcal{N}$
R_k	Individual data rate of user $k \in \mathcal{K}$, $R_k \triangleq \sum_{n\in\mathcal{N}} R_k^n$
R^n	Sum of data rates achieved on subcarrier $n \in \mathcal{N}$, $R^n \triangleq \sum_{k\in\mathcal{K}} R_k^n$
$\boldsymbol{R}$	Vector of all individual data rates, $\boldsymbol{R} \triangleq (R_k)_{k\in\mathcal{K}}$

Chapter 3

Radio Resource Management: an Optimization Framework

We explained in the previous section that each subcarrier can only serve a limited number of users at the same time (i.e., no more than M users), the choice of allocating a user to a subcarrier is an important combinatorial optimization problem called *subcarrier allocation*, also known as *user selection*. This can be seen as finding an $(M+1)$-dimensional matching between subcarriers and users [28], [30]–[32].

Another important problem is the *power control*, also known as *power allocation*. It consists in computing the transmit power of each signal given a limited power budget at the BS (in the downlink) or at each user's equipment (in the uplink), in order to maximize some desired system utility functions and ensure successful decoding of the signals.

Efficient RRM schemes in MC-NOMA systems require to solve jointly both optimization problems, which then becomes the *joint subcarrier and power allocation* problem (JSPA). We present in Section 3.1 the system utility functions that have been studied in the literature. In addition, we introduce a general framework which generalizes all these functions. In Section 3.2, we formulate two JSPA problems, namely the *utility maximization with individual power constraints* and the *utility maximization with cellular power constraint*. We discuss about the two types of power constraints and show their importance for practical systems. Finally, we study the computational complexity of these two JSPA problems in Section 3.3.

3.1 Utility Functions

One of the main objectives of RRM in wireless communication systems is to maximize the data rates. Since they are multiple users' data rates $\boldsymbol{R} = (R_1, \ldots, R_K)$ to optimize, a utility function $\mathcal{M}(\boldsymbol{R})$ should be chosen to characterize the desired system's performance metric. We will see later in this section that such a utility function is chosen in practice to achieve a trade-off between user fairness and the system sum rate (i.e., the sum of all users' data rates). A general utility maximization problem is of the following form:

$$\underset{\boldsymbol{p} \in \mathcal{F}}{\text{maximize}} \quad \mathcal{M}(\boldsymbol{R}(\boldsymbol{p})), \tag{$\mathcal{P}$}$$

where $\mathcal{M}$ can be any non-decreasing function of the data rates $\boldsymbol{R}$, and $\mathcal{F}$ is the search space defined by some practical constraints (see Section 3.2). However, most utilities studied in the NOMA literature belong to a smaller class of functions called *weighted generalized mean*, also known as *α-fairness* [43]. The weighted generalized mean is denoted by $\mathcal{M}_{i,\boldsymbol{w}}$, where $i \in \mathbb{R} \cup \{-\infty, +\infty\}$ is a parameter and $\boldsymbol{w}$ is a sequence of positive weights.

Definition 3 (Weighted generalized mean [44])**.**
Let $\mathcal{M}_{i,\boldsymbol{w}}$ denotes the weighted generalized mean of order $i \in \mathbb{R}\backslash\{0\}$, which is defined with a sequence of positive weights $\boldsymbol{w} = \{w_1, \ldots, w_K\}$ such that $\sum_{k=1}^{K} w_k = 1$. For K non-negative real numbers $x_1, \ldots, x_K$, we have:

$$\mathcal{M}_{i,\boldsymbol{w}}(x_1, \ldots, x_K) = \left(\sum_{k=1}^{K} w_k x_k^i \right)^{1/i}.$$

It can also be extended to $i \in \{-\infty, 0, +\infty\}$ by taking the limit, i.e., $\mathcal{M}_{i,\boldsymbol{w}}(x_1, \ldots, x_K) = \lim_{j \to i} \mathcal{M}_{j,\boldsymbol{w}}(x_1, \ldots, x_K)$.

An important property of $\mathcal{M}_{i,\boldsymbol{w}}$ is the generalized mean inequality, see below:

$$r < q \implies \mathcal{M}_{r,\boldsymbol{w}}(x_1, \ldots, x_K) \leq \mathcal{M}_{q,\boldsymbol{w}}(x_1, \ldots, x_K). \tag{3.1}$$

Note that the equality holds if and only if $x_1 = \cdots = x_K$.

In this thesis, we consider the class of weighted generalized mean maximization problems of the following form:

$$\underset{\boldsymbol{p} \in \mathcal{F}}{\text{maximize}} \quad \mathcal{M}_{i,\boldsymbol{w}}(R_1(\boldsymbol{p}), \ldots, R_k(\boldsymbol{p})), \tag{$\mathcal{P}_i$}$$

where order i and weights $\boldsymbol{w}$ are given as parameters of the problem. This class includes the following popular objective functions:

1. For $i = 1$, we have the weighted sum rate utility, namely the weighted arithmetic mean:
$$\mathcal{M}_{1,\boldsymbol{w}} = \sum_{k=1}^{K} w_k R_k.$$
If the weights are equal $\boldsymbol{w} = \boldsymbol{w}_{eq} = \{1/K, \ldots, 1/K\}$, it becomes the sum rate utility:
$$\mathcal{M}_{1,\boldsymbol{w}_{eq}} = \frac{1}{K} \sum_{k=1}^{K} R_k.$$

2. For $i = 0$, we have the weighted proportional fairness utility, namely the weighted geometric mean:
$$\mathcal{M}_{0,\boldsymbol{w}} = \left(\prod_{k=1}^{K} (R_k)^{w_k} \right)^{1/\sum_{k=1}^{K} w_k},$$
and the proportional fairness utility (geometric mean) if the weights are equal:
$$\mathcal{M}_{0,\boldsymbol{w}_{eq}} = \left(\prod_{k=1}^{K} R_k \right)^{1/K}.$$

3. For $i = -1$, we have the weighted harmonic mean utility:
$$\mathcal{M}_{-1,\boldsymbol{w}} = \sum_{k=1}^{K} w_k / \left(\sum_{k=1}^{K} \frac{w_k}{R_k} \right),$$
and harmonic mean utility if the weights are equal:
$$\mathcal{M}_{-1,\boldsymbol{w}_{eq}} = K / \left(\sum_{k=1}^{K} (R_k)^{-1} \right).$$

4. For $i = -\infty$, we have the max-min utility, which does not depend on the weights:
$$\mathcal{M}_{-\infty,\boldsymbol{w}} = \min_{k \in \mathcal{K}} \{R_k\}.$$

Note that, $i = -\infty$ is the most fair utility function which maximizes the minimum rate among all users. In contrast, $i = +\infty$ is the most unfair objective, as it maximizes only the best user's rate. In practical systems, the two desired objectives are user fairness and sum rate. As a consequence, we only consider values of i ranging from $-\infty$ to 1, achieving various trade-offs between user fairness and sum rate. In this regard, the aforementioned four utilities have indeed parameter $i \in [-\infty, 1]$. Furthermore, most utility maximization problems in the NOMA literature belong to these four utilities [19]–[22], [25], [26], [33]–[36], [45]–[50]. Nevertheless, our study of the general problem $\mathcal{P}_i$ could provide a foundation applicable to a much larger scope of similar subjects.

3.2 Problem Formulation and Constraints

We present two generalized mean maximization problems with different sets of constraints that will be studied throughout this manuscript. The first one is called *utility maximization with individual power constraints*, and it is defined as follows:

$$
\begin{aligned}
\underset{\boldsymbol{p}}{\text{maximize}} \quad & \mathcal{M}_{i,\boldsymbol{w}}(R_1(\boldsymbol{p}), \ldots, R_k(\boldsymbol{p})), \\
\text{subject to} \quad & C1: \sum_{n\in\mathcal{N}} p_k^n \leq P_{max}^k, \ k \in \mathcal{K}, \\
& C2: \ p_k^n \leq P_{max}^{n,k}, \ k \in \mathcal{K}, \ n \in \mathcal{N}, \\
& C3: \ p_k^n \geq 0, \ k \in \mathcal{K}, \ n \in \mathcal{N}, \\
& C4: \ |\mathcal{U}_n| \leq M, \ n \in \mathcal{N}.
\end{aligned}
\tag{$\mathcal{P}_i^I$}
$$

We say that $\mathcal{P}_i^I$ has individual power constraints, since each user $k \in \mathcal{K}$ has an individual power budget P_{max}^k. Constraint $C1$ states that the total transmit power of user k is no more than P_{max}^k. In $C2$, we further restrict the power of each user k, on each subcarrier n, to be less than or equal to $P_{max}^{n,k}$. This is a common assumption in multi-carrier systems, e.g., [51], [52], to avoid practical issues with power amplifiers such as high peak-to-average power ratio. $C3$ ensures that the allocated power remain non-negative, for mathematical consistency. Due to decoding complexity and error propagation in SIC [7], practical implementations have a maximum number of multiplexed users per subcarrier M, which corresponds to constraint $C4$.

The second problem is called *utility maximization with cellular power constraint*. It is defined below:

$$
\begin{aligned}
\underset{\boldsymbol{p}}{\text{maximize}} \quad & \mathcal{M}_{i,\boldsymbol{w}}(R_1(\boldsymbol{p}), \ldots, R_k(\boldsymbol{p})), \\
\text{subject to} \quad & C1: \sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{N}} p_k^n \leq P_{max}, \\
& C2: \sum_{k\in\mathcal{K}} p_k^n \leq P_{max}^n, \ n \in \mathcal{N}, \\
& C3: \ p_k^n \geq 0, \ k \in \mathcal{K}, \ n \in \mathcal{N}, \\
& C4: \ |\mathcal{U}_n| \leq M, \ n \in \mathcal{N}.
\end{aligned}
\tag{$\mathcal{P}_i^C$}
$$

Problem $\mathcal{P}_i^C$ is similar to $\mathcal{P}_i^I$, with the difference that it considers in $C1$ a total (cellular) power budget P_{max} to be allocated among all users. In addition, $C2$ represents a per subcarrier power constraint.

The two types of power constraints are used to model different systems and scenarios. On the one hand, cellular power constraint is mostly used in downlink

transmissions to represent the total transmit power budget available at the BS, e.g., [21], [25], [26], [30], [36], [45]–[47]. On the other hand, individual power constraint is particularly suitable for uplink scenarios where each UE has a maximum transmit power imposed by their power amplifier, e.g., [22], [33]–[35], [48]. Nevertheless, individual power constraint can also be applied to the downlink as in [P4], [P5], [20] for practical features of the RRM schemes, such as guaranteeing effort-fairness [53] or regulatory requirement on the transmit power towards each user device [54]. Furthermore, some papers consider both cellular and individual power constraints for full-duplex systems where users are served simultaneously on the downlink and uplink [22], [55]–[60].

In addition to these two types of power constraints, quality of service (QoS) constraints are often considered. QoS constraints require that each user achieve a minimum data rate Γ_k, i.e., $R_k(\boldsymbol{p}) \geq \Gamma_k$, for all $k \in \mathcal{K}$. This type of constraints can be added on top of the aforementioned power constraints [19], [31], [32], [48], [49]. To keep the optimization algorithm design simple, QoS constraints will not be considered in this work.

3.3 Computational Complexity

In this section, we study the computational complexity of problems $\mathcal{P}_i^I$ and $\mathcal{P}_i^C$. We first explain in Subsection 3.3.1 the concept of NP optimization (NPO) problems, and introduce a definition of NP-hardness for NPO problems. Then, we show in Subsection 3.3.2 that $\mathcal{P}_i^I$ is strongly NP-hard for any weighted generalized mean utility of order $i \in [-\infty, 1]$. This means that computing its optimal solution in the general case is intractable. Nevertheless, we present some special cases in which $\mathcal{P}_i^I$ is polynomial-time solvable. Finally, we discuss in Subsection 3.3.3 about the complexity of $\mathcal{P}_i^C$ in various scenarios.

3.3.1 Definitions and Preliminaries

Definition 4 formalizes the idea of NP optimization problems introduced in [61].

Definition 4 (NP optimization problem (NPO))**.**
A NPO problem $\mathcal{H}$ is a 4-tuple $(\mathcal{I}_\mathcal{H}, \mathcal{F}_\mathcal{H}, f, type)$ such that:

1) $\mathcal{I}_\mathcal{H}$ is the set of instances. Each instance is recognizable in polynomial time.

2) For any instance $x \in \mathcal{I}_{\mathcal{H}}$, $\mathcal{F}_{\mathcal{H}}(x)$ is the space of feasible solutions. Every solution $y \in \mathcal{F}_{\mathcal{H}}(x)$ has a size bounded by a polynomial in the size of x. Moreover, membership in $\mathcal{F}_{\mathcal{H}}$ is decidable in polynomial time.

3) f is the objective function, computable in polynomial time.

4) $type \in \{min, max\}$ indicates whether $\mathcal{H}$ is a minimization or maximization problem.

It can be verified that $\mathcal{P}_i^I$ and $\mathcal{P}_i^C$ are both NP optimization problems, as they fulfill Definition 4. As an example, let us verify it for problem $\mathcal{P}_i^I = (\mathcal{I}, \mathcal{F}, \mathcal{M}_{i,\boldsymbol{w}}, \max)$. An instance $x \in \mathcal{I}$ is only defined by the following polynomial number (in N and K) of system parameters:

$$x = (\boldsymbol{w}, K, N, M, (W_n)_{n\in\mathcal{N}}, (g_k^n)_{n\in\mathcal{N},k\in\mathcal{K}}, (\eta_k^n)_{n\in\mathcal{N},k\in\mathcal{K}}, (P_{max}^k)_{k\in\mathcal{K}}, (P_{max}^{n,k})_{n\in\mathcal{N},k\in\mathcal{K}}),$$

therefore, x is recognizable in polynomial time as required by Condition 1) in Definition 4. The feasible set $\mathcal{F}(x)$ is defined as the set of all power vectors satisfying constraints $C1$ to $C4$ in $\mathcal{P}_i^I$. Condition 2) holds, since these constraints can be verified in polynomial time. Finally, as required by condition 3), the objective function is computable in polynomial time.

Going back to the general problem $\mathcal{P}_i$, let $opt_{\mathcal{P}_i}(x)$ be the global optimal of an instance $x \in \mathcal{I}$, then the decision version of problem $\mathcal{P}_i$ consists of checking if this value is greater or equal to a given threshold T, i.e.,

$$opt_{\mathcal{P}_i}(x) \geq T. \qquad (\mathcal{D}_i)$$

In Garey and Johnson computational complexity framework [62, Chapter 5], a numerical optimization problem $\mathcal{P}_i$ is said to be NP-hard if its corresponding decision problem $\mathcal{D}_i$ is NP-hard. A discussion about strong NP-hardness and complexity preserving reductions can be found in [63]. We summarize these concepts in Definition 5.

Definition 5 (Strong NP-hardness)**.**
A decision problem $\mathcal{H}$ is said to be NP-hard if there exists a polynomial-time reduction from a NP-complete problem $\mathcal{G}$ to $\mathcal{H}$. In addition, $\mathcal{H}$ is said to be strongly NP-hard if it is still NP-hard even when all its numerical parameters are bounded by a polynomial in the size of the input.

The subclass of problems $\mathcal{P}_i$ and $\mathcal{D}_i$ in which the number of active users per subcarrier M is fixed is denoted by $\mathcal{P}_{i|M}$ and $\mathcal{D}_{i|M}$, respectively. Any instance of the optimization problem can be converted to an instance of the decision problem by appending an additional threshold parameter $T \in \mathbb{R}$, i.e.,

$$\mathcal{I}_{\mathcal{D}_{i|M}} = \mathcal{I}_{\mathcal{P}_{i|M}} \times \mathbb{R}. \tag{3.2}$$

It is known that WSR maximization in OFDMA systems (i.e., with $M = 1$) is polynomial-time solvable if we consider cellular power constraint [51] but strongly NP-hard with individual power constraints [64]. More generally, the authors of [52] proved that WSR, proportional fairness, harmonic mean utility and max-min fairness maximization problems are strongly NP-hard in OFDMA systems if we consider individual power constraints. In addition, they also proved that power minimization problems subject to minimum rate requirements (i.e., QoS constraints) are strongly NP-hard. In the same way, we will study in the following Subsections 3.3.2 and 3.3.3, the computational complexity of problem $\mathcal{P}_i^I$ under individual power constraints and $\mathcal{P}_i^C$ subject to cellular power constraint. We will also highlight three tractable special cases of $\mathcal{P}_i^I$ and $\mathcal{P}_i^C$ in Subsection 3.3.4 and discuss about possible polynomial time algorithms to solve them.

3.3.2 Complexity under Individual Power Constraints

It has been proven in [20] that the WSR maximization in MC-NOMA with individual power constraints (i.e., $\mathcal{P}_1^I$ with $i = 1$) is strongly NP-hard. Theorem 6 shows more generally that $\mathcal{P}_i^I$ is strongly NP-hard for any fixed $M \geq 1$ and $i \in [-\infty, 1]$.

Theorem 6 ($\mathcal{P}_i^I$ is strongly NP-hard[1])**.**
For any $i \in [-\infty, 1]$ and $M \geq 1$, problem $\mathcal{D}_{i|M}^I$ with objective function $\mathcal{M}_{i,\boldsymbol{w}}$ is strongly NP-hard in both downlink and uplink scenarios. In particular, the sum-rate $\mathcal{M}_{1,\boldsymbol{w}}$, proportional fairness $\mathcal{M}_{0,\boldsymbol{w}}$, harmonic mean utility $\mathcal{M}_{-1,\boldsymbol{w}}$ and max-min fairness $\mathcal{M}_{-\infty,\boldsymbol{w}}$ versions of the problem are all strongly NP-hard.

Proof. The idea of the proof is to construct a pseudo-polynomial reduction mapping any instance of the 3-Dimensional Matching Problem (3DM) to an instance of $\mathcal{D}_{i|M}^I$. 3DM is one of Karp's 21 NP-complete problems [65] and is also known to be NP-complete in the strong sense [66]. Details of the proof can be found in Appendix A. □

[1] This result has been published in [P3]

It is interesting to note that OFDMA results [51], [52], [64] correspond to the special case $M = 1$ of Theorem 6. Theorem 6 shows that computing the optimal solution of $\mathcal{P}_i^I$ in the general case is intractable, unless P = NP. Nevertheless, there may exist special cases in which the problem is solvable in polynomial time. We discuss about some special cases in Subsection 3.3.4.

3.3.3 Complexity under Cellular Power Constraint

We consider now the problem $\mathcal{P}_i^C$ with cellular power constraint. The (equal-weight) sum-rate ($\mathcal{M}_{1,\boldsymbol{w}_{eq}}$) maximization is polynomial time solvable in OFDMA [51]. The sum-rate maximization is also polynomial time solvable in MC-NOMA. Indeed, we will show in Theorem 15 of Subsection 4.3.1 that it can be solved in polynomial time by our proposed GRAD-JSPA algorithm. However, to the best of our knowledge, it is unknown whether WSR maximization in MC-NOMA under this type of constraints is polynomial time solvable or NP-hard. Reference [30, Proposition 1] proves that the subcarrier optimization is NP-hard only in the case of equal power allocation among the users. The proposed polynomial-time reduction from the NP-complete 3-dimensional matching (3DM) to the NOMA problem should have shown that all instances of 3DM can be mapped into an instance of the NOMA problem to be complete. Besides, the two-stage dynamic programming (TSDP) proposed in [20] solves it optimally in pseudo-polynomial time depending on J. Therefore, WSR maximization with cellular power constraint can only be *weakly* NP-hard at most (in contrast to *strongly* NP-hard for the individual power constraints as mentioned in the previous Subsection).

Only a few papers have developed optimization schemes in this setting, which are either heuristics with no theoretical performance guarantee or algorithms with impractical computational complexity. For example, a greedy user selection and heuristic power allocation scheme based on difference-of-convex programming is proposed in [36]. In reference [30], a matching algorithm is developed to perform subcarrier allocation. The authors of [21] employ monotonic optimization to develop an optimal resource allocation policy, which serves as benchmark due to its exponential complexity. The TSDP scheme in [20] achieves optimal solutions assuming the power budget is divided in J equal parts to be allocated. It is said to be pseudo-polynomial since its complexity depends on the total number of power values J, which is independent of the input size N, K, and M. In addition, all system's parameters and

variables are encoded in $O(\log(J))$ bits. As a consequence, TSDP mainly serves as benchmark due to its high computational complexity when J is large.

We note that, to the best of our knowledge, no polynomial-time approximation scheme (PTAS) has been proposed in the literature. Although PTAS is interesting for practical considerations of NP-hard problems, as it provides theoretical performance guarantees with controllable computational complexity. Motivated by this observation, we developed in [P1] a fully polynomial-time approximation scheme (FPTAS) for the WSR maximization problem with cellular power constraint. This algorithm will be explained later in Chapter 4. Besides, $\mathcal{P}_i^C$ can also be solved optimally in polynomial-time in some particular scenarios that will be discussed in the next subsection.

3.3.4 Polynomial Time Special Cases

We highlight here three tractable special cases of problems $\mathcal{P}_i^I$ and $\mathcal{P}_i^C$. They are obtained by relaxing the search space (removing the combinatorial constraint $C4$) or fixing some parameters (K and N). We also discuss about possible polynomial time algorithms to solve them.

1) If a user selection $\mathcal{U}_n$ is given and fixed, for each $n \in \mathcal{N}$, then problems $\mathcal{P}_i^I$ and $\mathcal{P}_i^C$ reduce to a power control problem. In downlink, the sum-rate objective function with equal weights $\boldsymbol{w}_{eq} = \{1/K, \ldots, 1/K\}$ can be rewritten as:

$$\begin{aligned}
\mathcal{M}_{1,\boldsymbol{w}_{eq}}(\boldsymbol{R}(\boldsymbol{p})) &= \sum_{n\in\mathcal{N}} \sum_{i=1}^{|\mathcal{U}_n|} R^n_{\pi_n(i)}\,, \\
&= \sum_{n\in\mathcal{N}} W_n \sum_{i=1}^{|\mathcal{U}_n|} \log_2\left(1 + \frac{g^n_{\pi_n(i)} p^n_{\pi_n(i)}}{\sum_{j=i+1}^{|\mathcal{U}_n|} g^n_{\pi_n(i)} p^n_{\pi_n(j)} + \eta^n_{\pi_n(i)}}\right), \\
&= \sum_{n\in\mathcal{N}} W_n \sum_{i=1}^{|\mathcal{U}_n|} \log_2\left(\frac{\sum_{j=i}^{|\mathcal{U}_n|} p^n_{\pi_n(j)} + \eta^n_{\pi_n(i)}/g^n_{\pi_n(i)}}{\sum_{j=i+1}^{|\mathcal{U}_n|} p^n_{\pi_n(j)} + \eta^n_{\pi_n(i)}/g^n_{\pi_n(i)}}\right), \\
&= \sum_{n=1}^{N} W_n \left(\sum_{i=1}^{|\mathcal{U}_n|-1} \log_2(\alpha^n_i(\boldsymbol{p})) + \log_2(\beta^n(\boldsymbol{p}))\right). \qquad (3.3)
\end{aligned}$$

For $n \in \mathcal{N}$ and $i < |\mathcal{U}_n|$, α^n_i is obtained by combining the numerator of $R^n_{\pi_n(i+1)}$ and the denominator of $R^n_{\pi_n(i)}$, i.e.,

$$\alpha^n_i(\boldsymbol{p}) \triangleq \frac{\sum_{j=i+1}^{|\mathcal{U}_n|} p^n_{\pi_n(j)} + \eta^n_{\pi_n(i+1)}/g^n_{\pi_n(i+1)}}{\sum_{j=i+1}^{|\mathcal{U}_n|} p^n_{\pi_n(j)} + \eta^n_{\pi_n(i)}/g^n_{\pi_n(i)}},$$

and β^n contains the numerator of R_1^n and the denominator of $R_{|\mathcal{U}_n|}^n$, i.e.,

$$\beta^n(\boldsymbol{p}) \triangleq \frac{\sum_{j=1}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \eta_{\pi_n(1)}^n / g_{\pi_n(1)}^n}{\eta_{\pi_n(|\mathcal{U}_n|)}^n / g_{\pi_n(|\mathcal{U}_n|)}^n}.$$

Assuming optimal decoding order (2.13) is applied in downlink, we have for all $i < |\mathcal{U}_n|$, $\eta_{\pi_n(i)}^n / g_{\pi_n(i)}^n \geq \eta_{\pi_n(i+1)}^n / g_{\pi_n(i+1)}^n$. It can be verified that α_i^n is a concave homographic function and β^n is linear, therefore also concave. Thus, by composition with logarithms and summation, we derive that the objective function (3.3) is concave. In addition, the feasible set defined by $C1$ to $C3$ is a convex set. Therefore, given a fixed and arbitrarily chosen subcarrier allocation, the sum-rate maximization problem can be optimally solved using classical convex programming methods [37]. The same result applies to uplink transmissions with sum-rate rewritten as:

$$\mathcal{M}_{1,\boldsymbol{w}_{eq}}(\boldsymbol{R}(\boldsymbol{p})) = \sum_{n \in \mathcal{N}} W_n \log_2 \left(\frac{\sum_{j=1}^{|\mathcal{U}_n|} g_{\pi_n(j)}^n p_{\pi_n(j)}^n + \eta^n}{\eta^n} \right).$$

In particular, if $M = K$, then all users can be multiplexed on all subcarriers. It directly follows that $\mathcal{P}_i^I$ and $\mathcal{P}_i^C$ are solvable by convex programming such as the projected gradient descent algorithm.

2) Now let's suppose that $K = 1$, then both $\mathcal{P}_i^I$ and $\mathcal{P}_i^C$ become equivalent to the following single-user problem:

$$\begin{aligned} \underset{\boldsymbol{p}}{\text{maximize}} \quad & R_1(\boldsymbol{p}_1), \\ \text{subject to} \quad & C1: \sum_{n \in \mathcal{N}} p_1^n \leq P, \\ & C2: \; p_1^n \leq P^n, \; n \in \mathcal{N}, \\ & C3: \; p_1^n \geq 0, \; n \in \mathcal{N}, \end{aligned}$$

where P corresponds to P_{max}^1 in $\mathcal{P}_i^I$, and P_{max} in $\mathcal{P}_i^C$. Similarly, P^n is equal to $P_{max}^{1,n}$ in $\mathcal{P}_i^I$, and P_{max}^n in $\mathcal{P}_i^C$. Here, we remove constraint $C4$ as it is always satisfied, i.e., at most $1 \leq M$ user is assigned to each subcarrier. The optimal solution to this single-user data rate maximization problem can be computed by the well known waterfilling power allocation [38].

3) In case there is only one subcarrier, i.e., $N = 1$, and assuming that the best M users are already selected, optimal closed-form solutions can be obtained for several subproblems. The authors of [31] solved the single-carrier power control problem for max-min fairness $\mathcal{M}_{-\infty,\boldsymbol{w}}$, sum-rate $\mathcal{M}_{1,\boldsymbol{w}_{eq}}$ and energy efficiency with

QoS constraints. The WSR problem $\mathcal{M}_{1,\boldsymbol{w}}$, with arbitrary weights $\boldsymbol{w}$, has been tackled in [18] using the Lagrange multiplier method for $M = 2$, and in [30] using geometric programming for general $M \geq 1$.

Chapter 4

WSR Maximization with Cellular Power Constraint

In this chapter, we focus on the WSR maximization problem with cellular power constraint:

$$
\begin{aligned}
\underset{\boldsymbol{p}}{\text{maximize}} \quad & \mathcal{M}_{1,\boldsymbol{w}}(\boldsymbol{R}(\boldsymbol{p})) = \sum_{k\in\mathcal{K}} w_k \sum_{n\in\mathcal{N}} R_k^n(\boldsymbol{p}^n), \\
\text{subject to} \quad & C1: \ \sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{N}} p_k^n \leq P_{max}, \\
& C2: \ \sum_{k\in\mathcal{K}} p_k^n \leq P_{max}^n, \ n\in\mathcal{N}, \\
& C3: \ p_k^n \geq 0, \ k\in\mathcal{K}, \ n\in\mathcal{N}, \\
& C4: \ |\mathcal{U}_n| \leq M, \ n\in\mathcal{N}.
\end{aligned}
\tag{$\mathcal{P}_1^C$}
$$

This problem is a particular case of $\mathcal{P}_i^C$ for which $i = 1$. Since cellular power constraint is applied to downlink systems, we compute the data rates using formula (2.15). The WSR objective function has received much attention as its weights $\boldsymbol{w}$ can be chosen to achieve different tradeoffs between sum-rate performance and fairness [67]. Moreover, when implemented in a scheduling system, the weights can also be tuned at each time slot to perform fair resource allocation [68]–[70].

For ease of reading, we summarize some system parameters of a given instance of $\mathcal{P}_1^C$, for all $n \in \mathcal{N}$, as follows:

$$
\mathcal{I}^n = (\boldsymbol{w}, \mathcal{K}, W_n, (g_k^n)_{k\in\mathcal{K}}, (\eta_k^n)_{k\in\mathcal{K}}) .
$$

In Section 4.1, we transform $\mathcal{P}_1^C$ into a finite-sum problem with separable objective functions f_i^n. These functions f_i^n satisfy particular unimodality and convexity

properties, so that their maximum can be computed in constant time (see Algorithm 1). We then formulate the *single-carrier power control* and *single-carrier user selection* sub-problems in Section 4.2, which are solved by algorithms SCPC and SCUS, respectively. We show that they can be improved by performing precomputation to avoid repeated operations each time they are executed. The improved algorithms are denoted by i-SCPC and i-SCUS. These algorithms are used as basic building blocks in Section 4.3 to design algorithms Opt-JSPA, ε-JSPA and Grad-JSPA for the joint subcarrier and power allocation problem:

- Opt-JSPA computes an optimal solution with lower complexity than current optimal schemes in the literature [20], [21]. It can be used as a benchmark for optimal WSR performance in simulations. However, its pseudo-polynomial time complexity remains impractical for real-world systems with low latency requirements.

- ε-JSPA is a fully polynomial-time approximation scheme (FPTAS), which we propose to reduce the complexity. Its design is based on techniques from the multiple choice knapsack problem [71]. By definition of FPTAS, its performance is within a factor $1-\varepsilon$ of the optimal, for any $\varepsilon > 0$. Moreover, it has polynomial complexity in both the input size and $1/\varepsilon$. Since, no approximation has been studied in the literature, ε-JSPA stands out by allowing to control a tight trade-off between performance guarantee and complexity.

- Grad-JSPA is a heuristic based on gradient descent. Numerical results in Section 4.4 show that it achieves near-optimal WSR with much lower complexity than existing optimal methods. Moreover, we prove that when the weights are all equal, the problem becomes concave. Hence, Grad-JSPA solves optimally the sum-rate maximization problem in polynomial-time.

In Section 4.4, we perform simulations with randomly deployed users and realistic channel conditions. Numerical results show the performance of the aforementioned algorithms, i.e., Opt-JSPA, ε-JSPA and Grad-JSPA, as well as their computational complexity. The results presented in this chapter have been published in [P2] and [P1].

4.1 Equivalent Separable Problem

Let us consider the following change of variables:

$$\forall n \in \mathcal{N},\ x_i^n \triangleq \begin{cases} \sum_{j=i}^{K} p_{\pi_n(j)}^n, & \text{if } i \in \{1,\dots,K\}, \\ 0, & \text{if } i = K+1. \end{cases} \tag{4.1}$$

We define $\boldsymbol{x} \triangleq (x_i^n)_{i\in\{1,\dots,K\},n\in\mathcal{N}}$ and $\boldsymbol{x}^n \triangleq (x_i^n)_{i\in\{1,\dots,K\}}$.

Lemma 7 (Equivalent problem $\mathcal{P}_1^{C\prime}$)**.**
Problem $\mathcal{P}_1^C$ is equivalent to problem $\mathcal{P}_1^{C\prime}$ formulated below:

$$\begin{aligned} \underset{\boldsymbol{x}}{\textit{maximize}} \quad & \sum_{n\in\mathcal{N}}\sum_{i=1}^{K} f_i^n(x_i^n) + A, && (\mathcal{P}_1^{C\prime}) \\ \textit{subject to} \quad & C1' : \sum_{n\in\mathcal{N}} x_1^n \le P_{max}, \\ & C2' : \ x_1^n \le P_{max}^n,\ n\in\mathcal{N}, \\ & C3' : \ x_i^n \ge x_{i+1}^n,\ i\in\{1,\dots,K\},\ n\in\mathcal{N}, \\ & C3'' : \ x_{K+1}^n = 0,\ n\in\mathcal{N}, \\ & C4' : \ |\mathcal{U}_n'| \le M,\ n\in\mathcal{N}, \end{aligned}$$

where for any $i \in \{1,\dots,K\}$ and $n \in \mathcal{N}$, we have:

$$f_i^n(x_i^n) \triangleq \begin{cases} W_n \log_2\left(\left(x_1^n + \tilde{\eta}_{\pi_n(1)}^n\right)^{w_{\pi_n(1)}}\right), & \text{if } i = 1, \\ W_n \log_2\left(\dfrac{\left(x_i^n+\tilde{\eta}_{\pi_n(i)}^n\right)^{w_{\pi_n(i)}}}{\left(x_i^n+\tilde{\eta}_{\pi_n(i-1)}^n\right)^{w_{\pi_n(i-1)}}}\right), & \text{if } i > 1, \end{cases}$$

and where $\mathcal{U}_n' \triangleq \{i \in \{1,\dots,K\} \colon x_i^n > x_{i+1}^n\}$. The constant term A is equal to:

$$A = \sum_{n\in\mathcal{N}} w_{\pi_n(K)} \log_2\left(1/\tilde{\eta}_{\pi_n(K)}^n\right),$$

and it is chosen so that $\mathcal{P}_1^C$ and $\mathcal{P}_1^{C\prime}$ have exactly the same optimal value.

Proof. See Appendix B.1. □

The advantage of this formulation $\mathcal{P}_1^{C\prime}$ is that it exhibits a separable objective function in both dimensions $i \in \{1,\dots,K\}$ and $n \in \mathcal{N}$. In other words, it can be written as a sum of functions f_i^n, each only depending on one variable x_i^n. In the following sections, we take advantage of this separability property to design efficient algorithms to solve $\mathcal{P}_1^{C\prime}$. The solution of $\mathcal{P}_1^C$ can then be obtained by solving $\mathcal{P}_1^{C\prime}$.

4.1.1 Analysis of the Separable Functions f_i^n

We introduce auxiliary functions to help us in the analysis of f_i^n and the algorithm design. For $n \in \mathcal{N}$, $i \in \{1, \dots, K\}$ and $j \leq i$, assume that the consecutive variables $x_j^n, \dots, x_i^n$ are all equal to a certain value $x \in \left[0, \bar{P}^n\right]$. We define $f_{j,i}^n$ as:

$$f_{j,i}^n(x) \triangleq \sum_{l=j}^{i} f_l^n(x) = \begin{cases} W_n \log_2\left(\left(x + \tilde{\eta}_{\pi_n(i)}^n\right)^{w_{\pi_n(i)}}\right), & \text{if } j = 1, \\ W_n \log_2\left(\frac{\left(x+\tilde{\eta}_{\pi_n(i)}^n\right)^{w_{\pi_n(i)}}}{\left(x+\tilde{\eta}_{\pi_n(j-1)}^n\right)^{w_{\pi_n(j-1)}}}\right), & \text{if } j > 1. \end{cases}$$

This simplification of notation is relevant for the analysis of $\mathcal{P}_1^{C\prime}$ and the algorithm design. Indeed, if users $j, \dots, i-1$ are not active (i.e., $j, \dots, i-1 \notin \mathcal{U}_n'$), then $x_j^n = \cdots = x_i^n$, therefore $\sum_{l=j}^{i} f_l^n$ can be replaced by $f_{j,i}^n$ and $x_{j+1}^n, \dots, x_i^n$ are redundant with x_j^n. If constraint $C4'$ is satisfied, up to M users are active on each subcarrier. Thus, evaluating the objective function of $\mathcal{P}_1^{C\prime}$ only requires $O(MN)$ operations.

We study the properties of $f_{j,i}^n$ in Lemma 8. Note that $f_i^n = f_{i,i}^n$, therefore Lemma 8 also holds for functions f_i^n. Fig. 4.1 shows the two general forms that can be taken by $f_{j,i}^n$.

Lemma 8 (Properties of $f_{j,i}^n$)**.**
Let $n \in \mathcal{N}$, $i \in \{1, \dots, K\}$, and $j \leq i$, we have:

- *If $j = 1$ or $w_{\pi_n(i)} \geq w_{\pi_n(j-1)}$, then $f_{j,i}^n$ is increasing and concave on $[0, \infty)$.*
- *Otherwise when $j > 1$ and $w_{\pi_n(i)} < w_{\pi_n(j-1)}$, $f_{j,i}^n$ is unimodal. It increases on $\left(-\tilde{\eta}_{\pi_n(j-1)}, c_1\right]$ and decreases on $[c_1, \infty)$, where*

$$c_1 = \frac{w_{\pi_n(j-1)}\tilde{\eta}_{\pi_n(i)} - w_{\pi_n(i)}\tilde{\eta}_{\pi_n(j-1)}}{w_{\pi_n(i)} - w_{\pi_n(j-1)}}.$$

  Besides, $f_{j,i}^n$ is concave on $\left(-\tilde{\eta}_{\pi_n(j-1)}, c_2\right]$ and convex on $[c_2, \infty)$, where

$$c_2 = \frac{\sqrt{w_{\pi_n(j-1)}}\tilde{\eta}_{\pi_n(i)} - \sqrt{w_{\pi_n(i)}}\tilde{\eta}_{\pi_n(j-1)}}{\sqrt{w_{\pi_n(i)}} - \sqrt{w_{\pi_n(j-1)}}} \geq c_1.$$

Proof. See Appendix B.2. □

We present in Algorithm 1 the pseudocode ARGMAXf which computes the maximum of $f_{j,i}^n$ on $\left[0, \bar{P}^n\right]$ following the result of Lemma 8. ARGMAXf only requires a constant number of basic operations, therefore its complexity is $O(1)$.

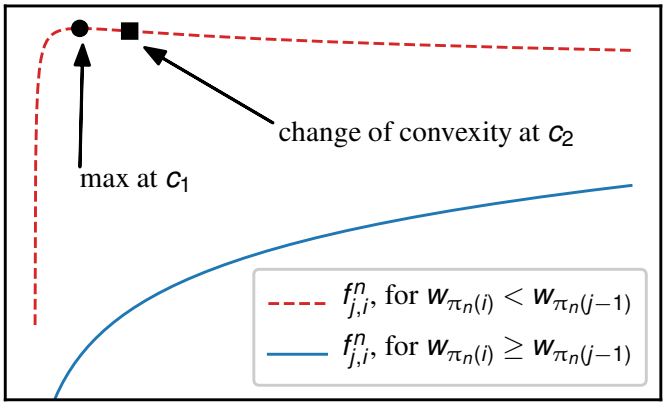

Figure 4.1: The two general forms of functions $f^n_{j,i}$

Algorithm 1 Compute maximum of $f^n_{j,i}$ on $\left[0, \bar{P}^n\right]$ (ARGMAXf)

function ARGMAX$f\left(j, i, \mathcal{I}^n, \bar{P}^n\right)$

1: $a \leftarrow \pi_n(i)$
2: $b \leftarrow \pi_n(j-1)$
3: **if** $j = 1$ or $w_a \geq w_b$ **then**
4: **return** $\bar{P}^n$
5: **else**
6: **return** $\max\left\{0, \min\left\{\frac{w_b\tilde{\eta}_a - w_a\tilde{\eta}_b}{w_a - w_b}, \bar{P}^n\right\}\right\}$
7: **end if**

end function

4.2 Single-Carrier Optimization

In this section, we focus on a simpler problem, in which there is a single subcarrier $n \in \mathcal{N}$ and a power budget $\bar{P}^n$ is given for this subcarrier:

$$\begin{aligned} F^n\left(\bar{P}^n\right) \quad = \quad & \max_{\boldsymbol{x}^n} && \sum_{i=1}^{K} f^n_i\left(x^n_i\right) + A^n, && \left(\mathcal{P}^{C\prime}_{SC}(n)\right) \\ & \text{subject to} && C2\text{–}3' : \ \bar{P}^n \geq x^n_1 \geq \ldots \geq x^n_K \geq 0, \\ & && C4' : \ |\mathcal{U}'_n| \leq M, \end{aligned}$$

where $A^n = w_{\pi_n(K)} \log_2\left(1/\tilde{\eta}^n_{\pi_n(K)}\right)$. $C2$–$3'$ is obtained by combining $C2'$, $C3'$ and $C3''$. $F^n\left(\bar{P}^n\right)$ denotes its optimal value. Algorithms SCPC and SCUS are developed to tackle respectively the single-carrier power control and single-carrier user selection sub-problems that arise from $\mathcal{P}^{C\prime}_{SC}(n)$. We provide technical details of these algorithms below, and we show how precomputation can further improve their com-

putational complexity. They will be used as basic building blocks in Section 4.3 to design efficient algorithms Grad-JSPA, Opt-JSPA and ε-JSPA for the joint subcarrier and power allocation problem.

4.2.1 Single-Carrier Power Control

The single-carrier power control problem $\mathcal{P}^{C\prime}_{SCPC}(n)$ is equivalent to problem $\mathcal{P}^{C\prime}_{SC}(n)$, with the exception that a fixed user selection $\mathcal{U}'_n$ (or equivalently $\mathcal{U}_n$) is given as input instead of being an optimization variable. It is defined below:

$$\begin{aligned} F^n\left(\mathcal{U}'_n, \bar{P}^n\right) \quad = \quad & \max_{\boldsymbol{x}^n} && \sum_{i=1}^{K} f_i^n\left(x_i^n\right) + A^n, && \left(\mathcal{P}^{C\prime}_{SCPC}(n)\right) \\ & \text{subject to} && C2\text{–}3' : \ \bar{P}^n \geq x_1^n \geq \ldots \geq x_K^n \geq 0. \end{aligned}$$

We denote its optimal value by $F^n\left(\mathcal{U}'_n, \bar{P}^n\right)$.

Since inactive users $k \notin \mathcal{U}_n$ have no contribution on the data rates, i.e., $p_k^n = 0$ and $R_k^n = 0$, we remove them for the study of this sub-problem. Without loss of generality, we index the remaining active users on subcarrier n by $i_n \in \{1_n, \ldots, |\mathcal{U}'_n|_n\}$. For example, if $\mathcal{U}'_n = \{4, 7, 10\}$, then $1_n = 4$, $2_n = 7$ and $3_n = 10$. For simplicity of notation, we add an index $0_n = 0$, which does not correspond to any user. From the definition of $\mathcal{U}'_n$, variables x_l^n with index from $l = (i-1)_n + 1$ to i_n are equal, for any $i \geq 1$. In the above example, we would have $x_1 = x_2 = x_3 = x_4 > x_5 = x_6 = x_7 > x_8 = x_9 = x_{10}$. Thus, the objective function of $\mathcal{P}^{C\prime}_{SCPC}(n)$ can be written as:

$$\sum_{i=1}^{K} f_i^n(x_i^n) + A^n = \sum_{i=1}^{|\mathcal{U}'_n|} f^n_{(i-1)_n+1,i_n}\left(x^n_{i_n}\right) + B^n, \tag{4.2}$$

where $B^n = A^n$ if the last active user's index is $|\mathcal{U}'_n|_n = K$, and $B^n = f^n_{|\mathcal{U}'_n|_n+1,K}(0) + A^n$ otherwise. For $1 \leq j \leq i \leq K$, we simplify some notations as follows:

$$\begin{aligned} \tilde{f}^n_{j,i}\left(\mathcal{U}'_n, \cdot\right) &\triangleq f^n_{(j-1)_n+1,i_n}\left(\cdot\right), \\ \textsc{Argmax}\tilde{f}\left(j, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right) &\triangleq \textsc{Argmax} f\left((j-1)_n+1, i_n, \mathcal{I}^n, \bar{P}^n\right). \end{aligned}$$

We reformulate the problem as:

$$\begin{aligned} F^n\left(\mathcal{U}'_n, \bar{P}^n\right) \quad = \quad & \max_{x^n_{i_n}} && \sum_{i=1}^{|\mathcal{U}'_n|} \tilde{f}^n_{i,i}\left(\mathcal{U}'_n, x^n_{i_n}\right) + B^n, && \left(\mathcal{P}^{C\prime}_{SCPC}(n)\right) \\ & \text{subject to} && C2\text{–}3' : \ \bar{P}^n \geq x^n_{1_n} \geq \ldots \geq x^n_{|\mathcal{U}'_n|_n} \geq 0. \end{aligned}$$

Algorithm 2 presents the SCPC method. The idea is to iterate over variables $x^n_{i_n}$ for $i = 1$ to $|\mathcal{U}'_n|$, and compute their optimal value $x^* = \textsc{Argmax}\tilde{f}(i, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n)$ at line 3. If the current allocation satisfies constraint $C3'$, then $x^n_{i_n}$ gets value x^*. Otherwise, the algorithm backtracks at line 6 and finds the highest index $j \in \{1, \ldots, i-2\}$ such that $x^n_{j_n} \geq \textsc{Argmax}\tilde{f}(j+1, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n)$. Then, variables $x^n_{(j+1)_n}, \ldots, x^n_{i_n}$ are set equal to $\textsc{Argmax}\tilde{f}(j+1, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n)$ at line 10. The optimality and complexity of SCPC are presented in Theorem 9.

Algorithm 2 Single-carrier power control algorithm (SCPC)

function SCPC$\left(\mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right)$

1: **for** $i = 1$ to $|\mathcal{U}'_n|$ **do**
2: ▷ Compute the optimal of $\tilde{f}^n_{i,i}$
3: $x^* \leftarrow \textsc{Argmax}\tilde{f}\left(i, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right)$
4: ▷ Modify x^* if this allocation violates constraint $C3'$
5: $j \leftarrow i - 1$
6: **while** $x^n_{j_n} < x^*$ and $j \geq 1$ **do**
7: $x^* \leftarrow \textsc{Argmax}\tilde{f}\left(j, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right)$
8: $j \leftarrow j - 1$
9: **end while**
10: $x^n_{(j+1)_n}, \ldots, x^n_{i_n} \leftarrow x^*$
11: **end for**
12: **return** $x^n_{1_n}, \ldots, x^n_{|\mathcal{U}'_n|_n}$

end function

Theorem 9 (Optimality and complexity of SCPC)**.**
Given subcarrier $n \in \mathcal{N}$, a set $\mathcal{U}'_n$ of M active users and a power budget $\bar{P}^n$, algorithm SCPC *computes the optimal single-carrier power control. Its worst case computational complexity is $O(M^2)$.*

Proof. See Appendix B.3. □

In JSPA schemes, such as Grad-JSPA and ε-JSPA, it is often required to compute the optimal single-carrier power control for many different values of power budget $\bar{P}^n$. Running many times SCPC is actually not efficient in terms of complexity, since several computations may be repeated. To avoid this, we propose in Algorithm 3 an improved SCPC algorithm (i-SCPC). The idea is to perform

precomputation before runtime by calling SCPC($\mathcal{I}^n$,$\mathcal{U}'_n$, P_{max}) and storing its result $x^n_{1_n}, \ldots, x^n_{|\mathcal{U}'_n|_n}$ as a global variable (also called lookup table). Any subsequent evaluation with input $\mathcal{I}^n$, $\mathcal{U}'_n$, $\bar{P}^n$, where $\bar{P}^n \leq P_{max}$, can be obtained as in line 1.

Theorem 10 (Optimality and complexity of i-SCPC)**.**
*Given subcarrier $n \in \mathcal{N}$ and a set $\mathcal{U}'_n$ of M active users, the precomputation of i-*SCPC *has complexity $O(M^2)$. Any subsequent evaluation costs $O(M)$. Hence, for C different power budgets, algorithm i-*SCPC *computes their respective optimal single-carrier power control with overall complexity $O(M^2 + CM)$.*

Proof. See Appendix B.4. □

Algorithm 3 Improved SCPC algorithm with precomputation (i-SCPC)

input: $\mathcal{I}^n$,$\mathcal{U}'_n$, P_{max}
global variable: $x^n_{1_n}, \ldots, x^n_{|\mathcal{U}'_n|_n}$
initialization: $x^n_{1_n}, \ldots, x^n_{|\mathcal{U}'_n|_n} \leftarrow$ SCPC($\mathcal{I}^n$,$\mathcal{U}'_n$, P_{max})
function i-SCPC$\left(\bar{P}^n\right)$
1: **return** $\min\{x^n_{1_n}, \bar{P}^n\}, \ldots, \min\{x^n_{|\mathcal{U}'_n|_n}, \bar{P}^n\}$
end function

Remark. *Note that* SCPC *and i-*SCPC *returns $|\mathcal{U}'_n|$ values $x^n_{1_n}, \ldots, x^n_{|\mathcal{U}'_n|_n}$ representing only the active users' variables. These values are sufficient to compute the optimal power allocation and WSR of $\mathcal{P}^{C\prime}_{SCPC}(n)$ as shown in Eqn. (4.2). If needed, the full vector $\boldsymbol{x}^n$ can be obtained by the following procedure in $O(K)$ operations:*

1: ***for*** *$i = 1$ to $|\mathcal{U}'_n|$ and $l = (i-1)_n + 1$ to i_n* ***do***
2: $\quad x^n_l \leftarrow x^n_{i_n}$
3: ***end for***
4: ***for*** *$l = |\mathcal{U}'_n|_n + 1$ to K* ***do***
5: $\quad x^n_l \leftarrow 0$
6: ***end for***

4.2.2 Single-Carrier User Selection

Unlike in the previous subsection, we consider here furthermore the user selection $\mathcal{U}'_n$ optimization under the SIC constraint $C4'$, i.e., we solve $\mathcal{P}^{C\prime}_{SC}(n)$. We first develop a dynamic programming (DP) procedure in Algorithm 4 (SCUS). Then, we propose an improved version (i-SCUS) which performs SCUS as precomputation.

The idea of SCUS is to compute recursively the elements of three arrays V, X, U. Let $m \in \{0, \ldots, M\}$, $j \in \{1, \ldots, K\}$ and $i \in \{j, \ldots, K\}$, we define $V[m, j, i]$ as the optimal value of the following problem $\mathcal{P}_{SC}^{C\prime}[m, j, i]$:

$$
\begin{aligned}
V[m,j,i] \quad \triangleq \quad & \max_{x_j^n, \ldots, x_K^n} && \sum_{l=j}^{K} f_l^n(x_l^n), && (\mathcal{P}_{SC}^{C\prime}[m,j,i]) \\
& \text{subject to} && C2\text{–}3' : \ \bar{P}^n \geq x_j^n \geq \ldots \geq x_K^n \geq 0, \\
& && C4' : \ |\mathcal{U}_n'| \leq m, \\
& && C5' : \ x_j^n = \cdots = x_i^n.
\end{aligned}
$$

This problem is more restrictive than $\mathcal{P}_{SC}^{C\prime}(n)$. The objective function only depends on variables $x_j^n, \ldots, x_K^n$. $C4'$ limits the number of active users to m. Moreover, variables $x_j^n, \ldots, x_i^n$ are equal according to $C5'$.

It is interesting to note that $V[M, 1, 1]$ is the optimal value of $\mathcal{P}_{SC}^{C\prime}(n)$, since the objective function is $\sum_{l=1}^{K} f_l^n(x_l^n)$ for $j = 1$ and constraint $C5'$ becomes trivially true for $j = i$. Let $x_j^{n*}, \ldots, x_K^{n*}$ be the optimal solution achieving $V[m, j, i]$. We define $X[m, j, i] \triangleq x_i^{n*}$, which is also equal to $x_j^{n*}, \ldots, x_{i-1}^{n*}$ due to constraint $C5'$. The idea of SCUS is to recursively compute the elements of V through the following relation:

$$
V[m,j,i] = \begin{cases} v_{act}, & \text{if } v_{act} > v_{inact} \\ & \text{and } x^* > X\,[m-1, i+1, i+1], \\ v_{inact}, & \text{otherwise,} \end{cases} \tag{4.3}
$$

where $x^* = \textsc{Argmax} f\left(j, i, \mathcal{I}^n, \bar{P}^n\right)$, and v_{act} (resp. v_{inact}) corresponds to the optimal allocation assuming user i is active (resp. inactive):

$$
\begin{aligned}
v_{act} &= f_{j,i}^n\,(x^*) + V[m-1, i+1, i+1], \\
v_{inact} &= V[m, j, i+1].
\end{aligned}
$$

During SCUS's iterations, the array U keeps track of which previous element of V has been used to compute the current value function $V[m, j, i]$. This allows us to retrieve the entire optimal vector $\boldsymbol{x}^n$ at the end of Algorithm 4 (at lines 28-35) by backtracking from index $(M, 1, 1)$ to $\varnothing$, where $\varnothing$ is set at initial indices (see lines 5 and 11) to indicate the recursion termination. To sum up, X and U have two different recurrence relations depending on the cases in Eqn. (4.3).
If $V\,[m, j, i] = v_{act}$, then:

$$
X\,[m, j, i] = x^*,
$$

Algorithm 4 Single-carrier user selection algorithm (SCUS)

function SCUS$\left(\mathcal{I}^n, M, \bar{P}^n\right)$

▷ Initialize arrays V, X, U for $m = 0$ and $i = K$
for $i = K$ to 1 and $j = i$ to 1 **do**
    $V\left[0, j, i\right] \leftarrow f_{j,K}^n\left(0\right)$
    $X\left[0, j, i\right] \leftarrow 0$
    $U\left[0, j, i\right] \leftarrow \varnothing$
end for
for $m = 1$ to M and $j = K$ to 1 **do**
    $x^* \leftarrow \textsc{Argmax} f\left(j, K, \mathcal{I}^n, \bar{P}^n\right)$
    $V\left[m, j, K\right] \leftarrow f_{j,K}^n\left(x^*\right)$
    $X\left[m, j, K\right] \leftarrow x^*$
    $U\left[m, j, K\right] \leftarrow \varnothing$
end for
▷ Compute V, X, U for $m \in [1, M]$ and $j \leq i \leq K - 1$
for $i = K - 1$ to 1 and $m = 1$ to M and $j = i$ to 1 **do**
    $x^* \leftarrow \textsc{Argmax} f\left(j, i, \mathcal{I}^n, \bar{P}^n\right)$
    $v_{act} \leftarrow f_{j,i}^n\left(x^*\right) + V[m-1, i+1, i+1]$
    $v_{inact} \leftarrow V[m, j, i+1]$
    if $v_{act} > v_{inact}$ and $x^* > X\left[m-1, i+1, i+1\right]$ **then**
        $V\left[m, j, i\right] \leftarrow v_{act}$
        $X\left[m, j, i\right] \leftarrow x^*$
        $U\left[m, j, i\right] \leftarrow (m-1, i+1, i+1)$
    else
        $V\left[m, j, i\right] \leftarrow v_{inact}$
        $X\left[m, j, i\right] \leftarrow X\left[m, j, i+1\right]$
        $U\left[m, j, i\right] \leftarrow (m, j, i+1)$
    end if
end for
▷ Retrieve the optimal solution $\boldsymbol{x}^n$
$x_1^n, \ldots, x_K^n \leftarrow 0$
$(m, j, i) \leftarrow (M, 1, 1)$
repeat
    $x_j^n, \ldots, x_i^n \leftarrow X\left[m, j, i\right]$
    $(m, j, i) \leftarrow U\left[m, j, i\right]$
until $(m, j, i) = \varnothing$
return $\boldsymbol{x}^n$
end function

$$U\left[m,j,i\right]=(m-1,i+1,i+1).$$

If $V\left[m,j,i\right]=v_{inact}$, then:

$$\begin{aligned}X\left[m,j,i\right]&=X\left[m,j,i+1\right],\\U\left[m,j,i\right]&=(m,j,i+1).\end{aligned}$$

When $m=0$, no user can be active on this subcarrier due to constraint $C4'$. Therefore, V, X, U can be initialized by:

$$\begin{aligned}V\left[0,j,i\right]&=f_{j,K}^{n}\left(0\right),\\X\left[0,j,i\right]&=0,\\U\left[0,j,i\right]&=\varnothing.\end{aligned}$$

For simplicity, we also extend V, X and U on the index $i=K$ and $j\leq K$ and initialize them as follows:

$$\begin{aligned}V\left[m,j,K\right]&=f_{j,K}^{n}\left(x^{*}\right),\\X\left[m,j,K\right]&=x^{*},\\U\left[m,j,K\right]&=\varnothing.\end{aligned}$$

A detailed analysis is given in Appendix B.5.

Theorem 11 (Optimality and complexity of SCUS)**.**
Given a subcarrier $n\in\mathcal{N}$, a power budget $\bar{P}^{n}$ and $M\geq 1$, algorithm SCUS *computes the optimal single-carrier power control and user selection of $\mathcal{P}_{SC}^{C\prime}(n)$. Its worst case computational complexity is $O(MK^{2})$.*

Proof. See Appendix B.5. □

We present i-SCUS in Algorithm 5, which performs precomputation to avoid repeating the DP procedure when multiple evaluations are required. The algorithm precomputes vectors V, X, U from SCUS$(\mathcal{I}^{n},M,P_{max})$ before runtime, at line 1. Then, in lines 2-5, it retrieves the active users set $\mathcal{U}_{n}'$ and optimal solution $x_{1}^{n},\ldots,x_{K}^{n}$ of each $V[M,1,i]$, $i\in\{1,\ldots,K\}$, and stores them in *collection*. Any subsequent evaluation with a lower budget $\bar{P}^{n}\leq P_{max}$, can be obtained by searching the best allocation among the K possibilities in *collection* (lines 6-7). Each allocation is truncated as in i-SCPC$\left(\bar{P}^{n}\right)$ to satisfy budget $\bar{P}^{n}$. The optimality and complexity of Algorithm 5 are given in Theorem 12.

Theorem 12 (Optimality and complexity of i-SCUS)**.**
*Given a subcarrier $n \in \mathcal{N}$, a power budget $\bar{P}^n$ and $M \geq 1$, the precomputation of i-*SCUS *has complexity $O(MK^2)$. Any subsequent evaluation costs $O(MK)$. Hence, for C different power budgets, i-*SCUS *computes their respective optimal power control and user selection $\mathcal{P}_{SC}^{C\prime}(n)$ with overall complexity $O(MK^2 + CMK)$.*

Proof. See Appendix B.6. □

Algorithm 5 Improved SCUS algorithm with precomputation (i-SCUS)

input: $\mathcal{I}^n, M, P_{max}$

global variable: *collection*

initialization:

1: Get V, X, U from SCUS$(\mathcal{I}^n, M, P_{max})$

2: **for** $i = 1$ to K **do**

3: Retrieve the active users set $\mathcal{U}'_n$ of $V[M, 1, i]$ and its corresponding optimal solution $x_1^n, \ldots, x_K^n$

4: Add $(\mathcal{U}'_n, x_1^n, \ldots, x_K^n)$ to *collection*

5: **end for**

function i-SCUS$\left(\bar{P}^n\right)$

6: Get $(\mathcal{U}'_n, x_1^n, \ldots, x_K^n)$ in *collection* that maximizes $F^n(\mathcal{U}'_n, \bar{P}^n)$, where

$$F^n(\mathcal{U}'_n, \bar{P}^n) = \sum_{l=1}^{|\mathcal{U}'_n|} \tilde{f}_{l,l}^n \left(\mathcal{U}'_n, \min\{x_{l_n}^n, \bar{P}^n\}\right) + B^n$$

7: **return** $\min\{x_1^n, \bar{P}^n\}, \ldots, \min\{x_K^n, \bar{P}^n\}$

end function

Table 4.1 summarizes the complexity of the single-carrier algorithms developed in this section. They will be used as basic building blocks to design JSPA schemes in Section 4.3.

Table 4.1: Summary of the single-carrier resource allocation schemes

Algorithm	**Complexity to perform C evaluations**
SCPC	$O(CM^2)$
i-SCPC	$O(M^2 + CM)$
SCUS	$O(CMK^2)$
i-SCUS	$O(MK^2 + CMK)$

4.3 Joint Subcarrier and Power Allocation

Recall that $F^n\big(\bar{P}^n\big)$ is the optimal value of the single-carrier problem $\mathcal{P}_{SC}^{C\prime}(n)$ with power budget $\bar{P}^n$. We have $F^n\big(\bar{P}^n\big) = \sum_{i=1}^{K} f_i^n(x_i^n) + A^n$, where $x_1^n, \ldots, x_K^n$ is the output of i-SCUS$\big(\bar{P}^n\big)$ or (equivalently SCUS$\big(\bar{P}^n\big)$). Using this notation, $\mathcal{P}_1^{C\prime}$ can be simplified as the following multi-carrier resource allocation problem:

$$\begin{aligned} &\underset{\bar{\boldsymbol{P}}}{\text{maximize}} && \sum_{n\in\mathcal{N}} F^n\big(\bar{P}^n\big)\,, && (\mathcal{P}_{MC}^{C\prime}) \\ &\text{subject to} && \bar{P}^n \in \mathcal{F}_{MC}, \end{aligned}$$

where $\bar{P}^n$, for $n \in \mathcal{N}$, are intermediate variables representing each subcarrier's power budget. $\bar{\boldsymbol{P}} \triangleq \big(\bar{P}^1, \ldots, \bar{P}^N\big)$ denotes the power budget vector. The feasible set

$$\mathcal{F}_{MC} \triangleq \{\bar{\boldsymbol{P}}\colon \sum_{n\in\mathcal{N}} \bar{P}^n \le P_{max} \text{ and } 0 \le \bar{P}^n \le P_{max}^n,\ n \in \mathcal{N}\}$$

is chosen to satisfy $C1'$ and $C2'$ in $\mathcal{P}_1^{C\prime}$.

4.3.1 Gradient Descent Based Heuristic: Grad-JSPA

Grad-JSPA is an efficient heuristic based on projected gradient descent. Its principle is to perform a two-stage optimization as presented in Fig. 4.2. The first-stage is a projected gradient descent on $\bar{\boldsymbol{P}}$ in the search space $\mathcal{F}_{MC}$. The evaluations of each function F^n and its derivative are done by i-SCUS in a second-stage optimization. We denote the derivative of F^n at $\bar{P}^n$ by $F^{n\prime}\big(\bar{P}^n\big)$. Lemma 13 shows how to compute it.

Lemma 13 (Derivative of F^n)**.**
Let $x_1^n, \ldots, x_K^n$ be the output of i-SCUS$\big(\bar{P}^n\big)$. The left derivative of F^n at $\bar{P}^n$, can be computed as follows:

$$F^{n\prime}\big(\bar{P}^n\big) = \frac{W_n w_{\pi_n(l)}}{\Big(x_l^n + \tilde{\eta}_{\pi_n(l)}^n\Big)\ln(2)} = \frac{W_n w_{\pi_n(l)}}{\Big(\bar{P}^n + \tilde{\eta}_{\pi_n(l)}^n\Big)\ln(2)},$$

where l is the greatest index such that $x_l^n = \bar{P}^n$.

Proof. See Appendix B.7. □

The pseudocode of Grad-JSPA is described in Algorithm 6. Input ξ corresponds to the error tolerance at termination, as we can see at line 8. The search

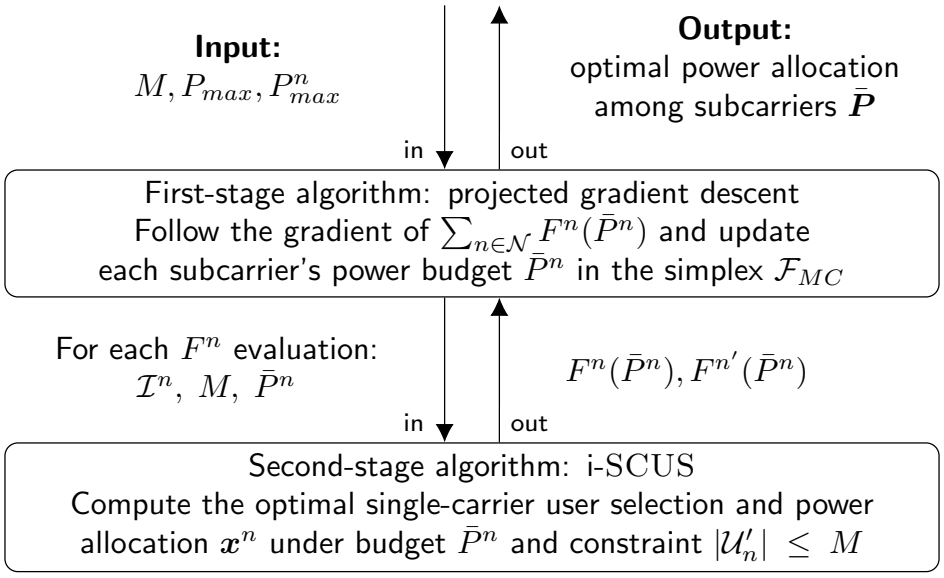

Figure 4.2: Overview of GRAD-JSPA

Algorithm 6 Gradient descent based heuristic (GRAD-JSPA)

function GRAD-JSPA$\left((\mathcal{I}^n)_{n\in\mathcal{N}}, M, P_{max}, P_{max}^n, \xi\right)$

1: Let $\bar{\boldsymbol{P}} \leftarrow (0,\ldots,0)$ be the starting point
2: **repeat**
3: Save the previous vector $\bar{\boldsymbol{P}}' \leftarrow \bar{\boldsymbol{P}}$
4: Determine a search direction $\Delta \leftarrow \left(F^{1\prime}\left(\bar{P}^1\right),\ldots,F^{N\prime}\left(\bar{P}^N\right)\right)$
5: Choose a step size α
6: Update $\bar{\boldsymbol{P}} \leftarrow$ projection of $\bar{\boldsymbol{P}} + \alpha\Delta$ on $\mathcal{F}_{MC}$
7: **until** $||\bar{\boldsymbol{P}}' - \bar{\boldsymbol{P}}|| \leq \xi$
8: **return** $\bar{\boldsymbol{P}}$

end function

direction at line 4 is the gradient of $\sum_{n\in\mathcal{N}} F^n$ evaluated at $\bar{\boldsymbol{P}}$. Since $F^1,\ldots,F^N$ are independent, it is equal to the vector of $F^{1\prime}\left(\bar{P}^1\right),\ldots,F^{N\prime}\left(\bar{P}^N\right)$. Note that the step size α at line 5 can be tuned by backtracking line search or exact line search [37, Section 9.2]. We adopt the latter to perform simulations. The projection of $\bar{\boldsymbol{P}}+\alpha\Delta$ on the simplex $\mathcal{F}_{MC}$ at line 6 can be computed efficiently [37, Section 8.1.1], the details of its implementation are omitted here.

We proved in [P2] that GRAD-JSPA worst case complexity is $O(\log(1/\xi)NMK^2)$ when SCUS is used to evaluate functions F^n, $n\in\mathcal{N}$. We now show in Theorem 14 that the complexity of GRAD-JSPA can be reduced by the use of i-SCUS.

Theorem 14 (Complexity of GRAD-JSPA[1])**.**
Let ξ be the error tolerance at termination. Algorithm GRAD-JSPA *has complexity $O(NMK^2 + \log(1/\xi)NMK)$ when i-*SCUS *is used to evaluate functions F^n, $n \in \mathcal{N}$.*

Proof. See Appendix B.7. □

Although i-SCUS (or equivalently SCUS) is optimal, the returned $F^n(\bar{P}^n)$ is not necessarily concave in $\bar{P}^n$. As a consequence, GRAD-JSPA is not guaranteed to converge to a global maximum. Nevertheless, we show by numerical results in Section 4.4 that it achieves near-optimal WSR performance with low complexity. Moreover, Theorem 15 shows that GRAD-JSPA solves optimally the sum-rate maximization problem in polynomial-time. The idea of the proof is that F^n becomes concave when the weights are all equal, for all $n \in \mathcal{N}$.

Theorem 15 (GRAD-JSPA is optimal for $\mathcal{P}_1^C$ with equal weights)**.**
Algorithm GRAD-JSPA *solves optimally $\mathcal{P}_1^C$ with equal weights $\boldsymbol{w} = \boldsymbol{w}_{eq} = \{1/K, \ldots, 1/K\}$ in polynomial-time. As a consequence, the sum-rate maximization problem with cellular power constraint is polynomial-time solvable.*

Proof. See Appendix B.8. □

4.3.2 Power Discretization

The JSPA problem as formulated in $\mathcal{P}_{MC}^{C\prime}$ has real variables $\bar{P}^n$ on a continuous search space $\mathcal{F}_{MC}$. However, the study of NP-hard optimization problems and their approximation requires parameters and variables to be represented by a bounded number of bits [62], i.e., with bounded precision. This is also a reasonable assumption in practice since MC-NOMA systems are subject to minimum transmit power limitation at the BS and floating-point arithmetic precision of the hardware. As a consequence, we discretize the search space $\mathcal{F}_{MC}$, in the same way as in [20]. Let δ be the minimum transmit power such that the variables $\bar{P}^n$ can only take value of the form $l \cdot \delta$, for $l \in \{0, 1, \ldots, \lfloor \frac{P_{max}}{\delta} \rfloor\}$. We denote the number of non-zero power values as $J = \lfloor \frac{P_{max}}{\delta} \rfloor$. The feasible set then becomes:

$$\mathcal{F}'_{MC} \triangleq \{\bar{\boldsymbol{P}} : \sum_{n \in \mathcal{N}} \bar{P}^n \leq P_{max} \text{ and } 0 \leq \bar{P}^n \leq P_{max}^n,\ n \in \mathcal{N},$$
$$\text{and } \bar{P}^n = l \cdot \delta,\ l \in \{0, \ldots, J\},\ n \in \mathcal{N}\}.$$

[1] This theorem can also be found in [P1]

We rewrite problem $\mathcal{P}_{MC}^{C\prime}$ with search space $\mathcal{F}'_{MC}$ as follows:

$$\begin{aligned}
\underset{\boldsymbol{y}}{\text{maximize}} \quad & \sum_{n\in\mathcal{N}}\sum_{l=1}^{J} c_{n,l}y_{n,l}, \qquad \text{(MCKP)}\\
\text{subject to} \quad & \sum_{n\in\mathcal{N}}\sum_{l=1}^{J} a_{n,l}y_{n,l} \leq P_{max},\\
& \sum_{l=1}^{J} a_{n,l}y_{n,l} \leq P_{max}^{n},\ n\in\mathcal{N},\\
& \sum_{l=1}^{J} y_{n,l} \leq 1,\ n\in\mathcal{N},\\
& y_{n,l}\in\{0,1\},\ n\in\mathcal{N},\ l\in[1,J],
\end{aligned}$$

where $c_{n,l} = F^n(l\cdot\delta)$ and $a_{n,l} = l\cdot\delta$. The discretized JSPA problem, denoted by MCKP, is known as the multiple choice knapsack problem [71]. It has N disjoint classes each containing J items to be packed into a knapsack of capacity P_{max}. Each item has a profit $c_{n,l}$ and a weight $a_{n,l}$, representing respectively the WSR and power consumption of this allocation on subcarrier n. The binary variable $y_{n,l}$ takes value 1 if and only if item l in class n is assigned to the knapsack. The problem consists in assigning at most one item from each class to maximize the sum of their profit. We define $\boldsymbol{y} \triangleq (y_{n,l})_{n\in\mathcal{N},l\in\{0,\dots,J\}}$, and we denote the optimal value of MCKP by F^*_{MCKP}.

According to [71, Appendix A.1], the input size of the integer multiple choice knapsack problem is linear in the number of items and in $\log_2(c)$, where c is the integer knapsack capacity. This precision is important: an algorithm is said to be *polynomial* only if its complexity is polynomial in $\log_2(c)$. Any algorithm with complexity polynomial in c is said to be *pseudo-polynomial*, e.g., the primal dynamic programming algorithm presented in [71, Section 5.2]. One may wonder if the numerical parameter P_{max} in MCKP plays the same role as c. The answer is no, since P_{max} and δ can be normalized by a factor α, while multiplying g_k^n by α for all $k\in\mathcal{K}$ and $n\in\mathcal{N}$, without changing the problem and its complexity. Nevertheless, we see that J remains constant after such a normalization and seems to be similar to c in the integer multiple choice knapsack problem. To prove this, we show below that MCKP can be transformed in polynomial-time into the following integer multiple choice knapsack problem:

$$\underset{\boldsymbol{y}}{\text{maximize}} \quad \sum_{n\in\mathcal{N}}\sum_{l=1}^{J} c_{n,l}y_{n,l}, \qquad \text{(Int-MCKP)}$$

$$\begin{aligned}
\text{subject to} \quad & \sum_{n\in\mathcal{N}}\sum_{l=1}^{J} a'_{n,l}y_{n,l} \leq J, \\
& \sum_{l=1}^{J} a'_{n,l}y_{n,l} \leq \lfloor P^n_{max}/\delta \rfloor, \ n \in \mathcal{N}, \\
& \sum_{l=1}^{J} y_{n,l} \leq 1, \ n \in \mathcal{N}, \\
& y_{n,l} \in \{0,1\}, \ n \in \mathcal{N}, \ l \in [1,J]\,.
\end{aligned}$$

Int-MCKP has the same input as MCKP except that the following parameters are normalized by δ:

- The minimum transmit power δ' is equal to 1 after normalization by δ,
- The total power budget is $J = \lfloor P_{max}/\delta \rfloor$,
- Following the first two points, the weights $a'_{n,l}$ now take integer values between 0 and J. That is, $a'_{n,l} = l\delta' = l$, for $l \in \{0, 1, \ldots, J\}$ and $n \in \mathcal{N}$.
- The per-subcarrier budget is now $\lfloor P^n_{max}/\delta \rfloor$.

Int-MCKP is obtained by a solution-preserving polynomial transformation of MCKP. MCKP is itself computed in polynomial time from the discretized version of $\mathcal{P}_i^C$. We deduce from known properties of Int-MCKP, that J is a relevant numerical input of both MCKP and the discretized version of $\mathcal{P}_i^C$, so that their input size is linear in N, K and $\log_2(J)$.

We will see in the following subsections that algorithms TSDP and Opt-JSPA are polynomial in J, therefore we will qualify their complexity as pseudo-polynomial. On the contrary, ε-JSPA is a polynomial algorithm, as its complexity is polynomial in $\log_2(J)$.

4.3.3 Pseudo-Polynomial Time Optimal Scheme: Opt-JSPA

The discrete problem MCKP can be solved optimally by *dynamic programming by weights* studied in [71, Section 11.5]. Based on this idea, we propose Opt-JSPA (see Algorithm 7) to solve optimally $\mathcal{P}^{C\prime}_{MC}$. We first transform $\mathcal{P}^{C\prime}_{MC}$ to problem MCKP: from line 1 to 5, every item's profit $c_{n,l}$ is computed using i-SCUS. Then, we perform dynamic programming by weights at line 6. We summarize the optimality and complexity of Opt-JSPA in Theorem 16. Detailed analysis of the dynamic programming can be found in Appendix B.9.

Theorem 16 (Optimality and complexity of Opt-JSPA)**.**
Given a minimum transmit power δ, algorithm Opt-JSPA *computes the optimal of $\mathcal{P}_{MC}^{C\prime}$ on the discrete set $\mathcal{F}'_{MC}$. Its computational complexity is $O(NMK^2 + JNMK + J^2N)$, which is pseudo-polynomial in J.*

Proof. See Appendix B.9. □

Opt-JSPA is said to be pseudo-polynomial since it depends on the total number of power values J, which is independent of the input size N, K, and M. In addition, all system's parameters and variables are encoded in $O(\log(J))$ bits. As a consequence, in practical systems, the contribution of J to the computation time is way higher than parameters N, K, M.

Algorithm 7 The pseudo-polynomial time optimal scheme (Opt-JSPA)

function Opt-JSPA$\left((\mathcal{I}^n)_{n\in\mathcal{N}}, M, P_{max}, P^n_{max}, \delta\right)$

▷ Compute the parameters of MCKP
for $n \in \mathcal{N}$ and $l \in [0, J]$ **do**
    $a_{n,l} \leftarrow l \cdot \delta$
    $c_{n,l} \leftarrow F^n(l \cdot \delta)$
end for
return optimal allocation from the *dynamic programming by weights* [71]
end function

4.3.4 Fully Polynomial Time Approximation: ε-JSPA

We develop a FPTAS to avoid the pseudo-polynomial complexity in J that is inherent to the optimal schemes Opt-JSPA and TSDP [20]. According to [72], an algorithm is said to be a FPTAS if it outputs a solution within a factor $1-\varepsilon$ of the optimal, for any $\varepsilon > 0$. Moreover, its running time is bounded by a polynomial in both the input size and $1/\varepsilon$. A FPTAS is the best trade-off one can hope for an NP-hard optimization problem in terms of performance guarantee and complexity, assuming P $\neq$ NP.

The proposed FPTAS, called ε-JSPA (see Algorithm 8), is based on dynamic programming with scaled profits. Scaling the profits is a common technique to reduce the number of items computed in MCKP. First, we compute an estimation U of MCKP's optimal value, such that $U \geq F^*_{MCKP} \geq U/4$. We explain the estimation

procedure in Appendix B.10. Then, instead of computing all JN profit values $c_{n,l}$, we only consider the subset L_n of items on each subcarrier n such that:

$$L_n \triangleq \left\{ l' \in \{1,\dots,J\} \colon \exists\, l \in \left\{0,\dots,\frac{4N}{\varepsilon}-1\right\},\, c_{n,l'} \geq l\frac{\varepsilon U}{4N} > c_{n,l'-1} \right\}.$$

This can be seen as considering only one profit value per interval of the form $[(l-1)\cdot \varepsilon U/4N, l\cdot \varepsilon U/4N]$, for $l \in \{1,\dots,4N/\varepsilon\}$. Each L_n, for $n \in \mathcal{N}$, can be obtained by multi-key binary search [73]. All function evaluations required by the multi-key binary search are done by i-SCUS.

Finally, we apply the *dynamic programming by profits* [71, Section 11.5] in line 5. It is known that the optimal solution obtained by dynamic programming by profits considering only items in L_n, differs from F^*_{MCKP} by at most a factor $1-\varepsilon$. The performance and complexity of ε-JSPA are summarized in Theorem 17. We provide more details on the estimation U in Appendix B.10 and the dynamic programming by profits in Appendix B.11.

Algorithm 8 The proposed FPTAS (ε-JSPA)

function ε-JSPA$\left((\mathcal{I}^n)_{n\in\mathcal{N}}, M, P_{max}, P^n_{max}, \delta, \varepsilon\right)$

1: Compute an estimation U of F^*_{MCKP}
2: **for** $n \in \mathcal{N}$ **do**
3: Get $a_{n,l}$, $c_{n,l}$, for $l \in L_n$ by multi-key binary search
4: **end for**
5: **return** ε-approximate allocation from the *dynamic programming by profits* [71]

end function

Theorem 17 (Performance and complexity of ε-JSPA)**.**
Given a minimum transmit power δ and an approximation factor ϵ, algorithm ϵ-JSPA computes an ϵ-approximation of $\mathcal{P}^{C\prime}_{MC}$ on the discrete set $\mathcal{F}'_{MC}$. The algorithm is a FPTAS with asymptotic complexity:

$$O\left(NMK^2 + \min\left\{ \log(J)\,\frac{N^2MK}{\epsilon} + \frac{N^3}{\epsilon^2},\, JNMK + J^2N \right\}\right).$$

Proof. See Appendix B.11. □

4.3.5 Comparison of JSPA Algorithms

In Table 4.2, we compare the performance and complexity of the proposed algorithms with JSPA schemes in the literature. Reference [21] studied an optimal

monotonic optimization framework, which has exponential complexity in K and N. The two-stage dynamic programming algorithm (TSDP) proposed by Lei et al. has complexity $O(J^2NMK)$ according to [20, Theorem 13]. Both TSDP and the proposed Opt-JSPA are optimal. However, Opt-JSPA has lower complexity than TSDP. Indeed, the right term J^2N is lower by a factor MK, the middle term $JNMK$ by a factor J. The left term NMK^2 also improves the complexity, since reference [P4] shows that in practical systems $J \geq \Theta(\min\{K, MN\})$. This result is verified by simulation in Section 4.4. ε-JSPA is the proposed FTPAS. Its complexity is polynomial in N/ε and $\log(J)$. If $N/\varepsilon < J$, it has a lower complexity than Opt-JSPA. Otherwise, if $N/\varepsilon \geq J$, its complexity exceeds Opt-JSPA's complexity. Thus Opt-JSPA can be applied instead to achieve optimal result. Finally, Grad-JSPA is a heuristic. Its performance is evaluated through simulation in the next section. When applied in a discrete setting, the error tolerance or precision ξ is related to $\delta = 2\xi$. Hence, its complexity is proportional to $\log(J)$, which is way lower than the optimal schemes with pseudo-polynomial complexity due to J.

Table 4.2: Comparison of some JSPA schemes proposed in this work and in the literature

Algorithm	**Performance guarantee**	**Complexity for J discrete power values**
Monotonic optimization with outer polyblock approximation [21]	Optimal	Exponential in K and N
TSDP [20]	Optimal	$O(J^2NMK)$
Opt-JSPA	Optimal	$O(NMK^2 + JNMK + J^2N)$
ε-JSPA	FPTAS	$O\Big(NMK^2 + \min\Big\{\log(J)\,\frac{N^2MK}{\varepsilon} + \frac{N^3}{\varepsilon^2},\; JNMK + J^2N\Big\}\Big)$
Grad-JSPA	Heuristic	$O(NMK^2 + \log(J)NMK)$

4.4 Numerical Results

We evaluate the WSR and computational complexity of Opt-JSPA, ε-JSPA and Grad-JSPA through numerical simulations. We compare them with the optimal benchmark scheme TSDP introduced in [20]. We consider a hexagonal cell of diameter 1000 meters, with one BS located at its center and K users distributed uniformly at random in the cell. The users' weights are generated uniformly at random in $[0, 1]$. The number of users K varies from 5 to 60, and the number of subcarriers is $N = 20$. We assume a system bandwidth of $W = 5$ MHz and $W_n = W/N$ for all n. The minimum transmit power is $\delta = 0.01$ W. The cellular power budget is $P_{max} = 10$ W, therefore the number of power values is $J = 10^3$. Each point in the following figures is the average value obtained over 1000 random instances. Only Fig. 4.6 and 4.7 represent a single instance. The simulation parameters and channel model are summarized in Table 4.3.

Table 4.3: Simulation parameters (cellular power constraint)

Parameter	**Value**
Cell radius	1000 m
Min. distance from user to BS	35 m
Carrier frequency	2 GHz
Path loss model	$128.1 + 37.6 \log_{10} d$ dB, d is in km
Shadowing	Log-normal, 10 dB standard deviation
Fading	Rayleigh fading with variance 1
Noise power spectral density	-174 dBm/Hz
System bandwidth W	5 MHz
Number of subcarriers N	20
Number of users K	5 to 60
Total power budget P_{max}	10 W
Minimum transmit power δ	0.01 W
Number of power values J	10^3
Error tolerance ξ	10^{-4}
Parameter M	1 (OMA), 2 and 3 (NOMA)

Fig. 4.3 shows the WSR performance of Opt-JSPA and TSDP, for $M = 1$, 2 and 3. We only simulate TSDP for $K = 5$ to 30, due to its high running time. We see that Opt-JSPA and TSDP achieve the same WSR performance, which

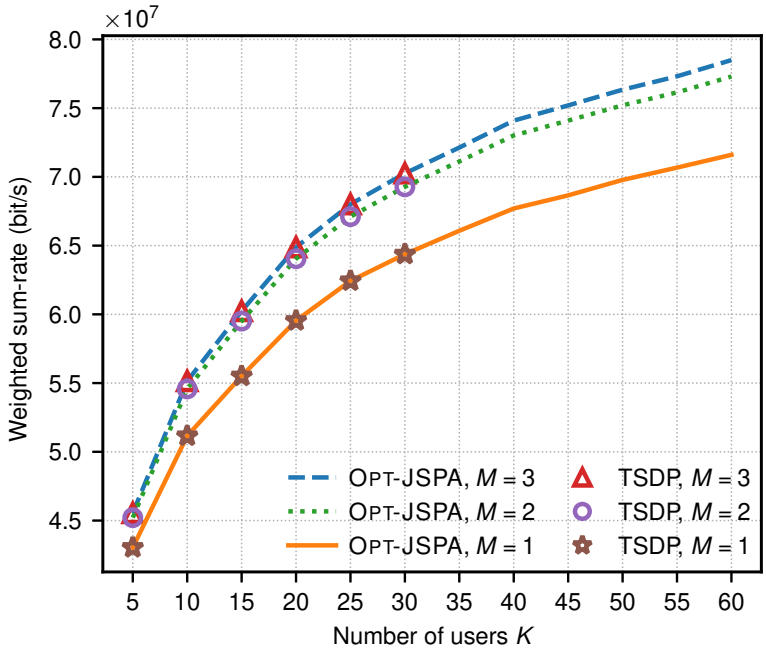

Figure 4.3: WSR of the optimal schemes for different number of users K

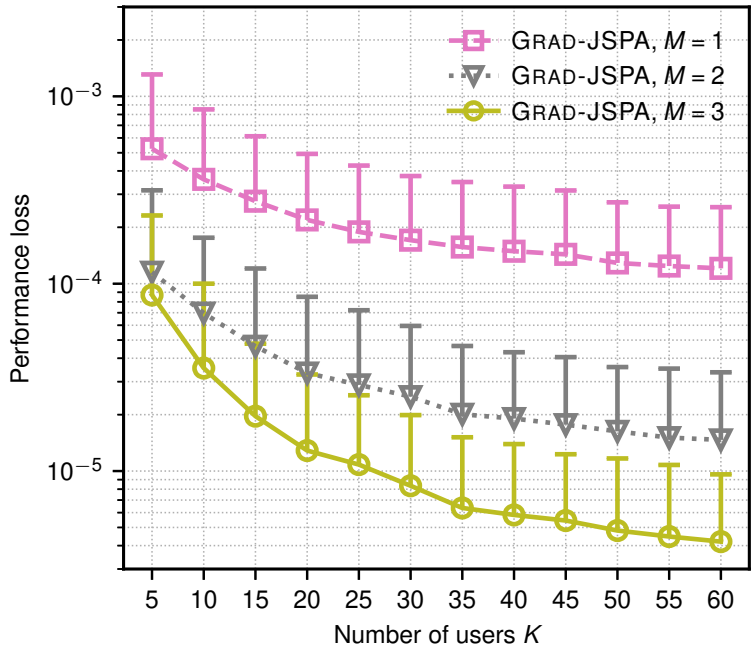

Figure 4.4: Performance loss of GRAD-JSPA compared to the optimal WSR

is consistent with the fact that they are both optimal. Indeed, the optimality of OPT-JSPA is shown in Theorem 16, and the optimality of TSDP has been proven in [20, Theorem 13]. Although both algorithms have the same performance, we will

see further on in Fig. 4.5 that Opt-JSPA has lower computational complexity than TSDP. The performance gain of NOMA with $M = 2$ (resp. $M = 3$) over OMA (i.e., $M = 1$) is about 8% (resp. 10%), for $K = 60$.

Fig. 4.4 illustrates the performance loss of Grad-JSPA compared to the optimal, for $M = 1$, 2 and 3. The performance loss is defined as:

$$\frac{\text{Optimal WSR} - \text{Grad-JSPA WSR}}{\text{Optimal WSR}}.$$

The markers represent the average performance loss, while the upper intervals indicate the 90th percentile. For example, for $K = 10$ and $M = 1$, 90% of Grad-JSPA results have less than 9×10^{-4} of performance loss. We observe that the average performance loss is always below 6×10^{-4}. Hence, our proposed heuristic Grad-JSPA achieves near-optimal solutions in these simulation settings. It is also suitable for large systems, since the performance loss decreases as K or M increases.

In Fig. 4.5, we count the number of basic operations (additions, multiplications, comparisons) performed by each algorithm, which reflects their computational complexity. The term "improved" in the legend represents the complexity of Opt-JSPA and Grad-JSPA when using i-SCPC and i-SCUS instead of SCPC and SCUS. There is a significant speed up by employing i-SCPC and i-SCUS as basic building blocks. Indeed, for $K = 60$ and $M = 1$, 2 or 3, there is a factor of at least 10 between Opt-JSPA and its improved version. Besides, the improved Opt-JSPA outperforms TSDP in terms of complexity. For instance, Opt-JSPA reduces the complexity by a factor 330, for $K = 30$ and $M = 3$. Finally, Grad-JSPA has low complexity, which makes it a good choice for practical implementation.

Fig. 4.6 and 4.7 present the WSR and complexity of ε-JSPA versus $4N/\varepsilon$. We choose such a normalized x-axis, as it is equal to the number of items evaluated in each subcarrier, i.e., $|L_n| = 4N/\varepsilon$. It can be directly compared to J, which is the total number of items in each subcarrier in the discretized problem MCKP. Here, we simulate a single instance with $K = 60$ users to show how ε-JSPA behaves as a function of ε. In Fig. 4.6, we also present its performance guarantee. Recall that the performance guarantee is $1 - \varepsilon$ times the optimal. As expected, ε-JSPA is always above its performance guarantee. As N/ε increases, the approximation guarantee tends to the optimal. In this instance, the algorithm already achieves a near-optimal solution for $4N/\varepsilon = 400$, i.e., $\varepsilon = 0.2$. In Fig. 4.7, we also plot the complexity of the improved Opt-JSPA for comparison. As explained in Section 4.3.4, the complexity increases with N/ε and becomes (asymptotically) equal to that of Opt-JSPA

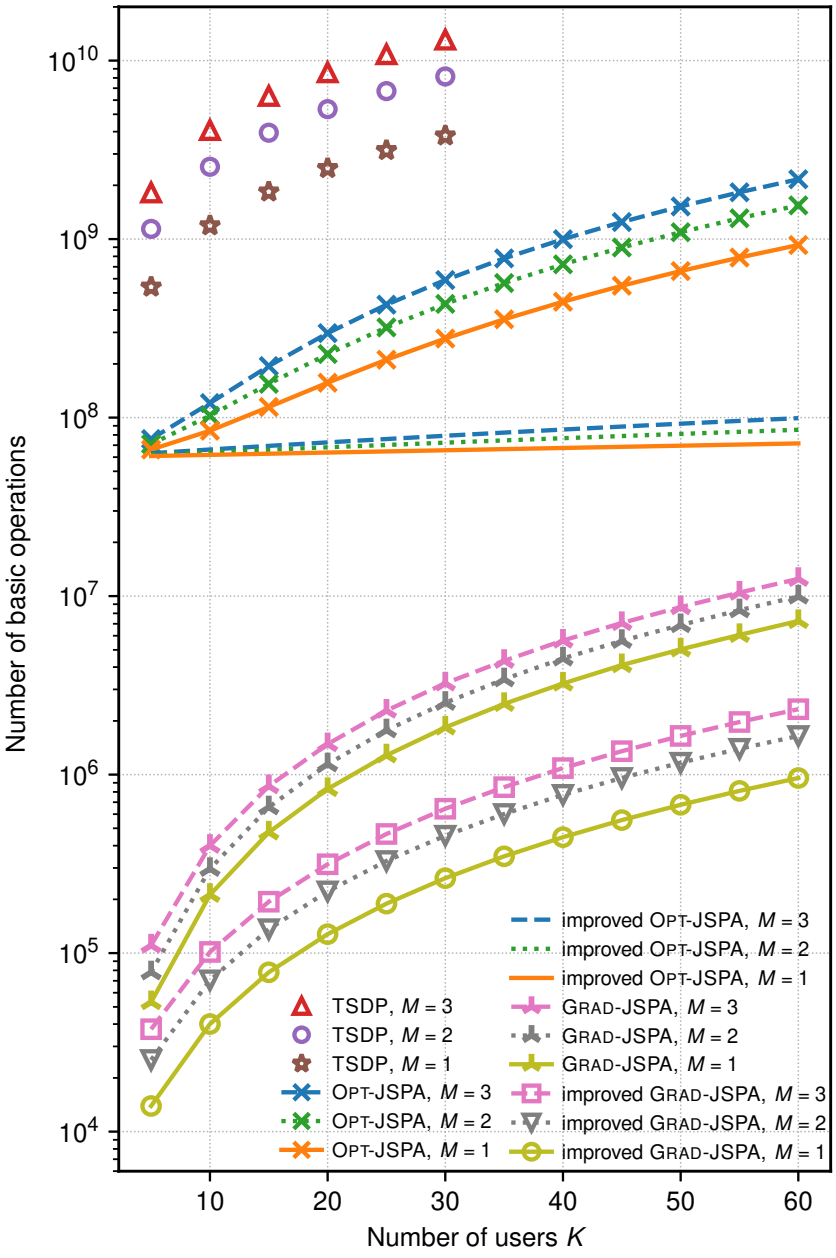

Figure 4.5: Number of basic operations performed by each algorithm versus K

for $N/\varepsilon = \Omega(J)$. In this regime, there is apparently no benefit of using ε-JSPA, since OPT-JSPA achieves the optimal with the same complexity. Nevertheless, in practice, we can see that even for $4N/\varepsilon \geq J$, ε-JSPA has less operations than OPT-JSPA. This is because the number of items computed by ε-JSPA increases slowly and smoothly as a function of ε. This behavior is not captured in the asymptotic complexity (big-O notation). This is verified in Fig. 4.7 for up to $4J = 4000$. In summary, ε-JSPA allows us to control the trade-off between WSR and complexity with ε.

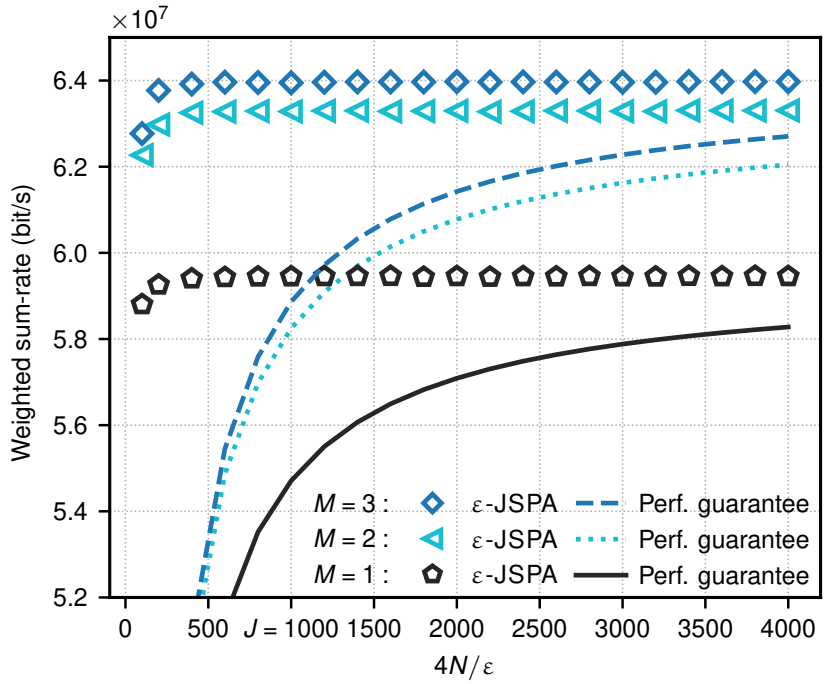

Figure 4.6: WSR of ε-JSPA and its guaranteed performance bound versus $4N/\varepsilon$

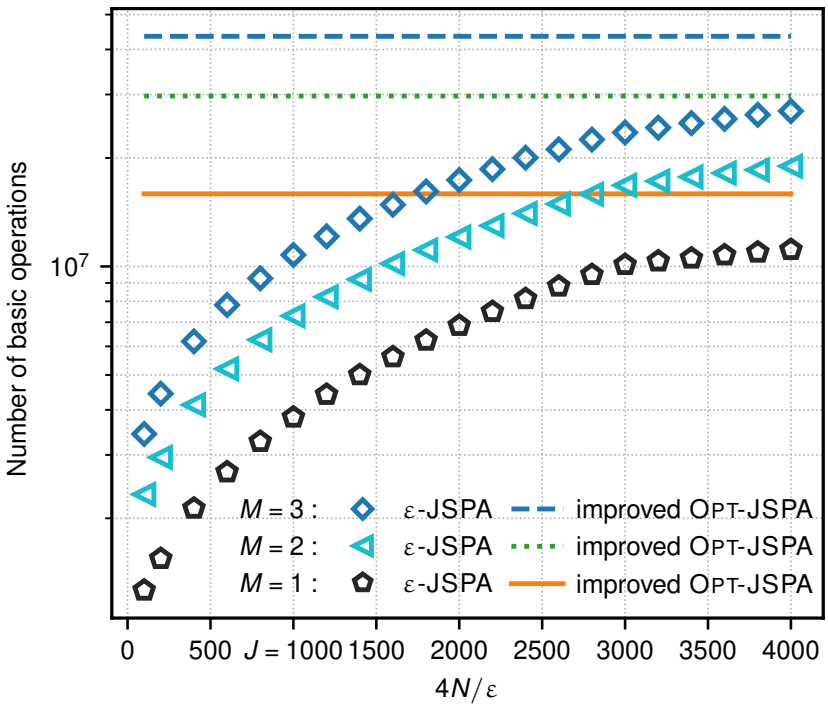

Figure 4.7: Number of basic operations performed by ε-JSPA versus $4N/\varepsilon$

Chapter 5

Sum-Rate Maximization with Individual Power Constraints

In this chapter, we consider the sum-rate maximization problem with individual power constraints:

$$
\begin{aligned}
\underset{\boldsymbol{p}}{\text{maximize}} \quad & \mathcal{M}_{1,\boldsymbol{w}_{eq}}(\boldsymbol{R}(\boldsymbol{p})) = \sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{N}} R_k^n(\boldsymbol{p}^n), \\
\text{subject to} \quad & C1: \sum_{n\in\mathcal{N}} p_k^n \leq P_{max}^k,\ k\in\mathcal{K}, \\
& C2: p_k^n \leq P_{max}^{n,k},\ k\in\mathcal{K},\ n\in\mathcal{N}, \\
& C3: p_k^n \geq 0,\ k\in\mathcal{K},\ n\in\mathcal{N}, \\
& C4: |\mathcal{U}_n| \leq M,\ n\in\mathcal{N}.
\end{aligned}
\tag{$\mathcal{P}_1^I$}
$$

We explained in Chapter 4 that problem $\mathcal{P}_1^C$ with cellular power constraint can be transformed into a separable problem $\mathcal{P}_1^{C\prime}$. In contrast, $\mathcal{P}_1^I$ is not separable due to the individual power constraints in $C1$, which makes it a challenging problem and requires to develop different optimization tools than those presented in Chapter 4. For this reason, we restrict our study to the equal-weight case ($\boldsymbol{w}_{eq}$), which ensures that the objective function $\mathcal{M}_{1,\boldsymbol{w}_{eq}}$ is concave. We take advantage of the objective function's concavity to develop centralized and distributed schemes.

In Section 5.1, we solve optimally the power control sub-problem (given a fixed subcarrier allocation) with a centralized gradient descent algorithm, denoted here by GA. In Section 5.2, we propose a distributed game theoretic approach in which each user optimizes its own data rate using local information only (the user's own channel conditions and received interference). The advantage of this approach compared to the centralized one is that it reduces the computational complexity and

requires only local information. This can be used to reduce the control signaling overhead in uplink systems for example. The downside is that the distributed power control achieves suboptimal sum-rate performance. We first prove that this problem is a concave game [74], [75] which has a unique Nash equilibrium. Then, we study in Subsection 5.2.1 a distributed pseudo-gradient descent algorithm called PGA to compute the equilibrium. Finally, we propose in Subsection 5.2.2 the synchronous iterative waterfilling algorithm (SIWA), and we show that it converges to the equilibrium if $M = 2$.

A heuristic three-step methodology is introduced in Section 5.3 to perform joint subcarrier and power allocation. This three-step methodology requires to solve two power control sub-problems, which can be done by any of the aforementioned schemes. When either GA, PGA or SIWA is applied twice, we get respectively the double gradient algorithm (DGA), the double pseudo-gradient algorithm (DPGA) and the double iterative waterfilling algorithm (DIWA). We show numerical results of these schemes in Section 5.5. The content of this chapter has been published in [P4] and [P5].

For ease of reading, we consider downlink transmissions as in the previous chapter. The distributed approach studied in Section 5.2 may not be relevant to the downlink, in which the allocation is computed centrally at the BS. Nevertheless, it is relevant for the uplink optimization, which can be performed similarly to the downlink optimization by replacing the decoding order with (2.14), and computing the data rates using formula (2.16). We summarize some system parameters of an instance of $\mathcal{P}_1^I$ as $\mathcal{I} = (\mathcal{K}, (W)_{n\in\mathcal{N}}, (g_k^n)_{k\in\mathcal{K},\, n\in\mathcal{N}}, (\eta_k^n)_{k\in\mathcal{K},\, n\in\mathcal{N}})$.

5.1 Centralized Power Control with Fixed Subcarrier Allocation

Given a fixed subcarrier allocation $\mathcal{U}_n$, for all $n \in \mathcal{N}$, constraint $C4$ can be relaxed, thus problem $\mathcal{P}_1^I$ can be written as the following power control sub-problem:

$$
\begin{aligned}
\underset{\boldsymbol{p}}{\text{maximize}} \quad & \sum_{k\in\mathcal{K}} \sum_{n\in\mathcal{N}} R_k^n\left(\boldsymbol{p}^n\right), \\
\text{subject to} \quad & C1: \ \sum_{n\in\mathcal{N}} p_k^n \le P_{max}^k, \ k \in \mathcal{K}, \\
& C2: \ p_k^n \le P_{max}^{n,k}, \ k \in \mathcal{K}, \ n \in \mathcal{N}, \\
& C3: \ p_k^n \ge 0, \ k \in \mathcal{K}, \ n \in \mathcal{N}.
\end{aligned}
\qquad (\mathcal{P}_1^I(\mathcal{U}_n))
$$

The set of feasible powers for user $k \in \mathcal{K}$ is denoted by:

$$\mathcal{P}_k \triangleq \{\boldsymbol{p}_k : \sum_{n\in\mathcal{N}} p_k^n \leq P_{max}^k \text{ and } 0 \leq p_k^n \leq P_{max}^{n,k},\, n \in \mathcal{N}_k\}. \tag{5.1}$$

We also define $\mathcal{P}$ as the joint feasible set of all users' powers, which can be expressed by the Cartesian product of all $\mathcal{P}_k$, for $k \in \mathcal{K}$, i.e,

$$\mathcal{P} \triangleq \prod_{k\in\mathcal{K}} \mathcal{P}_k. \tag{5.2}$$

We prove the concavity of problem $\mathcal{P}_1^I(\mathcal{U}_n)$ in Theorem 18.

Theorem 18 ($\mathcal{P}_1^I(\mathcal{U}_n)$ is a concave maximization problem)**.**
$\mathcal{P}_1^I(\mathcal{U}_n)$ is a concave maximization problem over the non-empty simplex (convex) feasible region $\mathcal{P}$.

Proof. See Appendix C.1. □

A direct implication of Theorem 18 is that $\mathcal{P}_1^I(\mathcal{U}_n)$ can be optimally solved through classical convex programming methods [37]. Since its feasible region $\mathcal{P}$ is a standard simplex on $\mathbb{R}_{0+}^{K\times N}$, we choose the projected gradient descent method [76] to solve $\mathcal{P}_1^I(\mathcal{U}_n)$. The pseudocode is given in Algorithm 9.

Algorithm 9 Projected gradient descent algorithm (GA)

function GA$\left(\mathcal{I}, (\mathcal{U}_n)_{n\in\mathcal{N}}, \left(P_{max}^k\right)_{k\in\mathcal{K}}, \left(P_{max}^{n,k}\right)_{n\in\mathcal{N},k\in\mathcal{K}}, \xi\right)$

1: Let $\boldsymbol{p} \leftarrow (0,\ldots,0)$ be the starting point
2: **repeat**
3: Save the previous vector $\boldsymbol{p}' \leftarrow \boldsymbol{p}$
4: Determine a search direction $\Delta \leftarrow \nabla\left(\sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{N}} R_k^n\right)(\boldsymbol{p}')$
5: Choose a step size α
6: Update $\boldsymbol{p} \leftarrow$ projection of $\boldsymbol{p} + \alpha\Delta$ on $\mathcal{P}$
7: **until** $||\boldsymbol{p}' - \boldsymbol{p}|| \leq \xi$
8: **return** $\boldsymbol{p}$

end function

In Algorithm 9, the operator ∇ represents the gradient with respect to the power vector $\boldsymbol{p}$. The step size α at line 5 can be tuned by backtracking line search or exact line search [37, Section 9.2]. We adopt the latter to perform simulations. ξ is a small constant chosen for the termination condition. At line 6 we perform the euclidian

projection on the feasible set $\mathcal{P}$. Note that the projection can be obtained as in [37, Section 8.1.1]. The details of the projection or omitted here. The convergence analysis of Algorithm 9 can be found in [77, Section 2.2.4], and [37], [76].

5.2 Distributed Power Control with Fixed Subcarrier Allocation

In this section, we study the power control sub-problem from a game theoretic point of view. That is, we consider a distributed version of $\mathcal{P}_1^I(\mathcal{U}_n)$, in which each user allocates power to its assigned subcarriers so as to maximize its individual data rate R_k. Each user treats the interference from other users after SIC as additional white Gaussian noise (AWGN). In this case, the power control problem can be seen as a K-player game [74], where $\boldsymbol{p}_k$ is the power allocation strategy of player k and $\mathcal{P}_k$ in (5.1) is its feasible strategy set. $\mathcal{P}$ represents the joint strategy space of all players in the system and is defined in (5.2). Note that the terms *user* and *player* both refer to elements of $\mathcal{K}$. They will be used interchangeably in this section.

For $k \in \mathcal{K}$, let $\boldsymbol{p}_{-k} \triangleq (\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_{k-1}, \boldsymbol{p}_{k+1}, \ldots, \boldsymbol{p}_K)$. We denote by $\mathcal{N}_k$ the set of subcarriers allocated to user $k \in \mathcal{K}$, i.e., $\mathcal{N}_k \triangleq \{n \in \mathcal{N} \colon k \in \mathcal{U}_n\}$. We define:

$$u_k(\boldsymbol{p}_k, \boldsymbol{p}_{-k}) \triangleq R_k(\boldsymbol{p}_k, \boldsymbol{p}_{-k}) = \sum_{n \in \mathcal{N}_k} R_k^n(\boldsymbol{p}_k, \boldsymbol{p}_{-k}),$$

as the *utility function* of user $k \in \mathcal{K}$. Additionally, let $\boldsymbol{u} \triangleq (u_1, u_2, \ldots, u_K)$. This game is characterized by the pair $(\mathcal{P}, \boldsymbol{u})$. A common equilibrium concept in game theory is the Nash equilibrium, which is defined in Definition 19.

Definition 19 (Nash equilibrium)**.**
A power allocation strategy $\tilde{\boldsymbol{p}} \triangleq (\tilde{\boldsymbol{p}}_1, \tilde{\boldsymbol{p}}_2, \ldots, \tilde{\boldsymbol{p}}_K) \in \mathcal{P}$ is called a Nash equilibrium if the following holds for all $k \in \mathcal{K}$ and any $\boldsymbol{p}_k \in \mathcal{P}_k$,

$$u_k(\tilde{\boldsymbol{p}}_k, \tilde{\boldsymbol{p}}_{-k}) \geq u_k(\boldsymbol{p}_k, \tilde{\boldsymbol{p}}_{-k},).$$

At a Nash equilibrium, no player can increase its data rate by unilaterally changing its strategy, i.e, no user has incentive to change its strategy at a Nash equilibrium. Theorem 20 shows the existence of Nash equilibrium in concave games. This theorem is a corollary of the fundamental theorem in game theory [78]–[80], a proof and further details can be found in [74]. All three conditions in the theorem are

satisfied by our problem, thus $(\mathcal{P}, \boldsymbol{u})$ is a concave game and it has at least one Nash equilibrium.

Theorem 20 (Existence of Nash equilibrium in concave games [74])**.**
If for each $k \in \mathcal{K}$, the following three statements hold:

(i) $\mathcal{P}_k$ is compact and convex,

(ii) u_k is continuous in $\boldsymbol{p}$,

(iii) u_k is concave in $\boldsymbol{p}_k$ for any given $\boldsymbol{p}_{-k}$,

then $(\mathcal{P}, \boldsymbol{u})$ is called a concave game, and it has at least one Nash equilibrium.

Next, we prove that the Nash equilibrium of $(\mathcal{P}, \boldsymbol{u})$ is unique. Let $D \triangleq \sum_{k\in\mathcal{K}} |\mathcal{N}_k|$, and $\Phi : \mathcal{P} \to \mathbb{R}^D$ be the pseudo-gradient of $\boldsymbol{u}$ as defined in [74]:

$$\forall\, \boldsymbol{p} \in \mathcal{P},\ \Phi(\boldsymbol{p}) \triangleq \left(\nabla_1 u_1(\boldsymbol{p}), \ldots, \nabla_K u_K(\boldsymbol{p})\right), \tag{5.3}$$

in which ∇_k denotes the gradient with respect to user k's power vector $\boldsymbol{p}_k$. That is, $\nabla_k u_k$ is the vector of u_k's derivatives with respect to p_k^n, for each $n \in \mathcal{N}_k$:

$$\nabla_k u_k(\boldsymbol{p}) = \left(\frac{W_n}{\left(\sum_{j=\pi_n^{-1}(k)}^{|\mathcal{U}_n|} p^n_{\pi_n(j)} + \tilde{\eta}_k^n\right)\ln(2)} \right)_{n\in\mathcal{N}_k}. \tag{5.4}$$

Since the uniqueness of Nash equilibrium is related to the monotonicity of Φ [74], we first show in Theorem 21 that Φ is strongly monotone. The uniqueness then follows in Theorem 22. The definition of strong monotonicity can be found in [75].

Theorem 21 (Strong monotonicity of Φ)**.**
Φ is strongly monotone, i.e., there exists a constant $c < 0$ such that for all $\boldsymbol{p}, \boldsymbol{p}' \in \mathcal{P}$,

$$(\Phi(\boldsymbol{p}) - \Phi(\boldsymbol{p}'))^T \cdot (\boldsymbol{p} - \boldsymbol{p}') \leq c||\boldsymbol{p} - \boldsymbol{p}'||_2^2.$$

Proof. See Appendix C.2. □

Theorem 22 (Uniqueness of the Nash equilibrium)**.**
The game $(\mathcal{P}, \boldsymbol{u})$ has a unique Nash equilibrium.

Proof. Theorem 21 shows that the pseudo-gradient of this game, Φ, is strongly monotone. Hence, the game has a unique Nash equilibrium according to [74], [75]. □

5.2.1 Pseudo-Gradient Descent Method

We know from Theorem 22 that the K-player concave game $(\mathcal{P}, \boldsymbol{u})$ has a unique Nash equilibrium. Therefore, the pseudo-gradient descent method converges to the unique Nash equilibrium according to [74]. We state in Algorithm 10 the pseudo-code of the distributed pseudo-gradient descent procedure performed by each user $k \in \mathcal{K}$. Besides, we define for $k \in \mathcal{K}$ and $n \in \mathcal{N}_k$:

$$\tilde{I}_k^n \triangleq \sum_{j=\pi_n^{-1}(k)+1}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_k^n, \tag{5.5}$$

as the normalized interference plus noise received by user $k \in \mathcal{K}$ on subcarrier $n \in \mathcal{N}$. In addition, we define $\tilde{\boldsymbol{I}}_k \triangleq (\tilde{I}_k^n)_{n \in \mathcal{N}_k}$. We see from Eqn. (5.4) that $\nabla_k u_k$ only depends on $\boldsymbol{p}_k$ and $\tilde{\boldsymbol{I}}_k$, thus we can write equivalently $\nabla_k u_k\big(\boldsymbol{p}_k, \tilde{\boldsymbol{I}}_k\big)$. Similarly, we use notation $R_k^n(p_k^n, \tilde{I}_k^n)$ and $R_k\big(\boldsymbol{p}_k, \tilde{\boldsymbol{I}}_k\big)$ for the data rates of user $k \in \mathcal{K}$.

Algorithm 10 Pseudo-gradient descent algorithm (PGA)

1: **procedure of user k**
2: **initialization:** Let $\boldsymbol{p}_k \leftarrow (0, \ldots, 0)$ be the starting point
3: **when** $\tilde{\boldsymbol{I}}_k$ changes or $||\boldsymbol{p}'_k - \boldsymbol{p}_k|| > \xi$ **do**
4: Save the previous power vector $\boldsymbol{p}'_k \leftarrow \boldsymbol{p}_k$
5: Determine a search direction $\Delta_k \leftarrow \nabla_k u_k\big(\boldsymbol{p}_k, \tilde{\boldsymbol{I}}_k\big)$
6: Choose a step size α_k
7: Update $\boldsymbol{p}_k \leftarrow$ projection of $\boldsymbol{p}_k + \alpha_k \Delta_k$ on $\mathcal{P}_k$
8: **end**
9: **end procedure**

An iteration of the gradient descent is triggered at line 3 either when the interference $\tilde{\boldsymbol{I}}_k$ changes or if the pseudo-gradient has not converged yet, i.e. $||\boldsymbol{p}'_k - \boldsymbol{p}_k|| > \xi$. Variable $\boldsymbol{p}'_k$ is chosen initially so that $||\boldsymbol{p}'_k - \boldsymbol{p}_k|| > \xi$.

The pseudo-gradient descent algorithm is distributed in the sense that each user $k \in \mathcal{K}$ only needs to know the channel gains between itself and the BS, P_{max}^k, $P_{max}^{n,k}$ and $\tilde{\boldsymbol{I}}_k$. These information are available locally or can be obtained using channel estimation techniques between the user and the BS. Indeed, at line 5 of Algorithm 10, $\nabla_k u_k$ only depends on $\boldsymbol{p}_k$, $\tilde{\boldsymbol{I}}_k$, and implicitly $\tilde{\eta}_k^n$. At line 6, the step size α_k, computed by line search or other methods, depends on $R_k\big(\boldsymbol{p}_k, \tilde{\boldsymbol{I}}_k\big)$ and its derivatives with respect to $\boldsymbol{p}_k$. The projection on $\mathcal{P}_k$ at line 7 only requires the information of the individual power budgets P_{max}^k and $P_{max}^{n,k}$.

5.2.2 Synchronous Iterative Waterfilling Algorithm

We propose here a heuristic, the synchronous iterative waterfilling algorithm (SIWA), based on the well known waterfilling algorithm [38, Sections 10.4 and 10.5]. The waterfilling algorithm computes the optimal power allocation for a user k assuming the other users' power allocation is fixed, i.e., $\tilde{\boldsymbol{I}}_k$ is fixed. This optimal power allocation is also called the waterfilling power allocation. Details are given in Theorem 23.

Theorem 23 (The waterfilling power control maximizes the individual data rate [38])**.** *Suppose that $\tilde{I}_k^n$ for $n \in \mathcal{N}_k$ are fixed. Then, $\boldsymbol{p}_k^* \in \mathcal{P}_k$ maximizes $R_k\big(\boldsymbol{p}_k^*, \tilde{\boldsymbol{I}}_k\big)$ if and only if there exists $\omega_k \in \mathbb{R}_{0+}$ such that:*

$$p_k^{n*} = \big[\omega_k - \tilde{I}_k^n\big]_0^{P_{max}^{n,k}}, \quad \text{for } n \in \mathcal{N}_k, \tag{5.6}$$

where

$$[x]_a^b = \begin{cases} a, & \text{if } x \leq a, \\ x, & \text{if } a \leq x \leq b, \\ b, & \text{if } b \leq x, \end{cases}$$

and

$$\sum_{n \in \mathcal{N}_k} p_k^{n*} = P_{max}^k. \tag{5.7}$$

The lowest value $w_k \geq 0$ satisfying Eqn. (5.6) and (5.7) is called the water level.

We define $\varphi_k \colon \prod_{k' \neq k} \mathcal{P}'_k \to \mathcal{P}_k$ as the waterfilling function of user k which takes as input $\boldsymbol{p}_{-k}$ and returns the waterfilling power allocation $(p_k^{n*})_{n \in \mathcal{N}_k}$ defined in Theorem 23. Since Theorem 23 only requires the received interference and not the full power vector $\boldsymbol{p}_{-k}$, one can also write $\varphi_k\big(\tilde{\boldsymbol{I}}_k\big)$ instead of $\varphi_k(\boldsymbol{p}_{-k})$. Furthermore, we define $\varphi \colon \mathcal{P} \to \mathcal{P}$ as the joint waterfilling function of all users, which is given as follows:

$$\varphi(\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_K) \triangleq (\varphi_k(\boldsymbol{p}_{-k}))_{k \in \{1,K\}}.$$

The idea of the iterative waterfilling algorithm (IWA) is that each user $k \in \mathcal{K}$ applies iteratively φ_k to update its power allocation (see Algorithm 11). Since this procedure is performed in a distributed manner, it may not always converge to a fixed point. The synchronous iterative waterfilling algorithm (SIWA) presented in Algorithm 12 further assumes that users are synchronized and that they update their power allocation simultaneously. We denote the power of user $k \in \mathcal{K}$ on subcarrier

Algorithm 11 Iterative waterfilling algorithm (IWA)

procedure of user k
  initialization: Let $\boldsymbol{p}_k \leftarrow (0, \ldots, 0)$ be the starting point
  when user is activated **do**
    Get the normalized interference plus noise $\tilde{\boldsymbol{I}}_k$
    Update $\boldsymbol{p}_k \leftarrow \varphi_k\big(\tilde{\boldsymbol{I}}_k\big)$
  end
end procedure

Algorithm 12 Synchronous iterative waterfilling algorithm (SIWA)

procedure of user k
  initialization: Let $\boldsymbol{p}_k(0) \leftarrow (0, \ldots, 0)$ be the starting point
  when user is activated at iteration t **do**
    Get the normalized interference plus noise $\tilde{\boldsymbol{I}}_k(t-1)$
    Update $\boldsymbol{p}_k(t) \leftarrow \varphi_k\big(\tilde{\boldsymbol{I}}_k(t-1)\big) \triangleq \varphi_k(\boldsymbol{p}_{-k}(t-1))$
  end
end procedure

$n \in \mathcal{N}_k$ at iteration t by $p_k^n(t)$ and the individual power vector by $\boldsymbol{p}_k(t)$. From a centralized point of view, SIWA's iterations are given as follows:

$$(\boldsymbol{p}_1(t+1), \boldsymbol{p}_2(t+1) \ldots, \boldsymbol{p}_K(t+1)) = \varphi(\boldsymbol{p}_1(t), \boldsymbol{p}_2(t), \ldots, \boldsymbol{p}_K(t)),$$

with $\boldsymbol{p}_k(0) = (0, \ldots, 0)$ for all $k \in \mathcal{K}$. Asynchronous versions of the IWA [81], [82] would be more suitable for practical applications, as they do not require simultaneous updates. However, synchronicity in distributed systems is out of the scope of this thesis, therefore we consider SIWA for simplicity. We show in Theorem 24 that SIWA is guaranteed to converge when $|\mathcal{U}_n| \leq 2$, for all $n \in \mathcal{N}$.

Theorem 24 (Convergence of SIWA)**.**
If $|\mathcal{U}_n| \leq 2$ for all $n \in \mathcal{N}$, then SIWA converges.

Proof. See Appendix C.3. □

5.3 Joint Subcarrier and Power Allocation as a Three-Step Heuristic

We propose a three-step heuristic to tackle the JSPA problem $\mathcal{P}_1^I$. Step 1 relaxes the problem by setting $\mathcal{U}_n = \mathcal{K}$, for all $n \in \mathcal{N}$. The resulting power control sub-problem $\mathcal{P}_1^I(\mathcal{K})$ is solved using either GA, PGA or SIWA. Step 2 is a heuristic which determines the final subcarrier allocation $\widehat{\mathcal{U}}_n$ satisfying constraint $C4$ and taking into account the data rates obtained in Step 1. In Step 3, the powers are optimized again, by either GA, PGA or SIWA, considering the subcarrier allocation $\widehat{\mathcal{U}}_n$. We illustrate this three-step method in Fig. 5.1 and give the pseudo-code below.

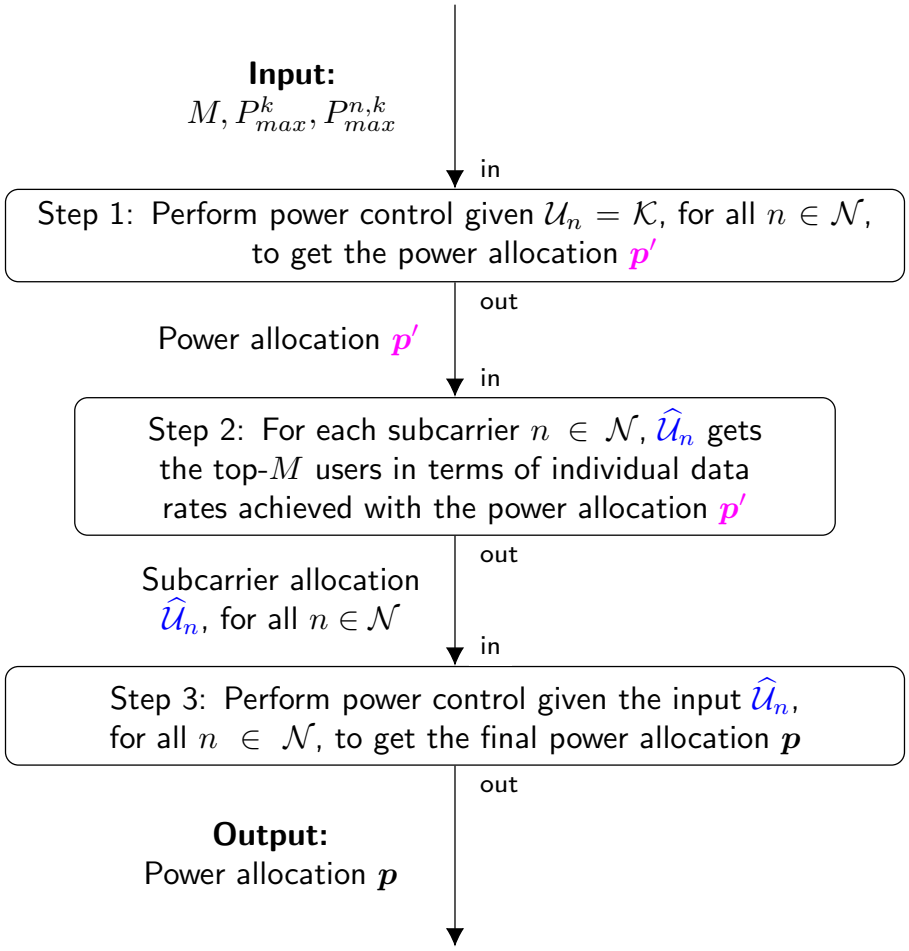

Figure 5.1: Overview of the three-step heuristic

Step 1: Solve the power control sub-problem $\mathcal{P}_1^I(\mathcal{K})$ given the subcarrier allocation $\mathcal{U}_n = \mathcal{K}$. The obtained power vector is denoted by $\boldsymbol{p}'$.

Step 2: Allocate subcarriers to users based on the data rates $\boldsymbol{R}(\boldsymbol{p}')$ obtained in Step 1. For each subcarrier $n \in \mathcal{N}$:

- If M or more users have positive power on subcarrier n, allocate subcarrier n to the M users who have the top-M highest individual data rates on subcarrier n, with ties broken arbitrarily.
- If less than M (but not 0) users have positive power on subcarrier n, allocate subcarrier n only to these users.
- If no user has positive power on subcarrier n, allocate subcarrier n to user k^*, where $k^* \triangleq \operatorname{argmin}_{k\in\mathcal{K}} \tilde{\eta}_k^n$, with ties broken arbitrarily.

The resulting subcarrier allocation is denoted by $\widehat{\mathcal{U}}_n$. It satisfies constraint $C4$, i.e., $|\widehat{\mathcal{U}}_n| \leq M$ for all $n \in \mathcal{N}$.

Step 3: Solve the power control sub-problem $\mathcal{P}_1^I(\widehat{\mathcal{U}}_n)$ using the subcarrier allocation $\widehat{\mathcal{U}}_n$ obtained in Step 2. The final power allocation is denoted by $\boldsymbol{p}$ in Fig. 5.1.

Either algorithm GA, PGA or SIWA can be used to perform power control in Step 1 and Step 3. When GA is applied in both Step 1 and Step 3, this three-step method is called the *double gradient algorithm* (DGA). Similarly, we refer to *double pseudo-gradient algorithm* (DPGA) and *double iterative waterfilling algorithm* (DIWA), when PGA and SIWA are used, respectively.

5.4 Computational Complexity Analysis

In this subsection, we analyze the asymptotic computational complexity of the proposed centralized and distributed power allocation algorithms, which are the key components of our designed three-step resource allocation framework. Besides, we also discuss the complexity of the benchmark schemes: LDDP [20] and FTPC [23], [24], [26]. We summarize these results in Table 5.1.

1) *Complexity of* GA: According to Theorem 18, the optimization problem $\mathcal{P}_1^I(\mathcal{U}_n)$ is concave over a simplex feasible region. One can prove as in Appendix B.7, that the objective function is furthermore α-strongly concave and β-smooth, for some parameters α and β. Therefore, $\mathcal{P}_1^I(\mathcal{U}_n)$ can be solved in a centralized manner by Algorithm 9 in $O(\log(1/\xi))$ iterations [77, Section 2.2.4], where ξ is the error tolerance at termination. Each iteration requires to compute the gradient $\Delta \leftarrow \nabla \left(\sum_{k\in\mathcal{K}} \sum_{n\in\mathcal{N}} R_k^n\right)(\boldsymbol{p}')$ with respect to the power vector $\boldsymbol{p}'$. For each $n \in \mathcal{N}$

Table 5.1: Complexity comparison of the proposed algorithms and benchmark schemes

Proposed Algorithms	**Number of iterations**	**Time complexity of each iteration in Step 1**	**Time complexity of each iteration in Step 3**
Projected gradient descent (centralized)	$O(\log(1/\xi))$	$O(NK^2)$	$O(NM^2)$
Pseudo-gradient descent (distributed)	$O(\log(1/\xi))$	$O(NK)$	$O(NM)$
SIWA (distributed)	Exponentially fast in ξ (empirical)	$O(NK)$	$O(NM)$
Benchmark Algorithms	**Time complexity**		
LDDP	$O(CNMKJ^2)$		
FTPC	$O(NK)$		

and $k \in \mathcal{K}$, the (k,n)-th element of this gradient is:

$$\Delta_{n,k} = \sum_{j \le \pi_n^{-1}(k)} \frac{\partial R^n_{\pi_n(j)}}{\partial p^n_k}, \tag{5.8}$$

where of $\frac{\partial \cdot}{\partial p^n_k}$ is the partial derivative with respect to p^n_k. Equation (5.8) can be computed by at most $O(|\mathcal{U}_n|)$ operations, therefore each iteration has $O(N|\mathcal{U}_n|^2)$ time complexity. In our three-step heuristic, the complexity in Step 1 and Step 3 are respectively $O(NK^2)$ and $O(NM^2)$.

2) *Complexity of* PGA: According to [74] and Theorem 22, Algorithm 10 converges to the unique fixed point within $O(\log(1/\xi))$ iterations. In line 5 of Algorithm 10, each user $k \in \mathcal{K}$ only computes its pseudo-gradient $\nabla_k u_k\big(\boldsymbol{p}_k, \tilde{\boldsymbol{I}}_k\big)$ which consists of $|\mathcal{N}_k| \le N$ elements. Thus, each iteration's time complexity is $O(N|\mathcal{U}_n|)$. When applied in Step 1 of our three-step method, PGA has complexity $O(NK)$. In Step 3, its complexity reduces to $O(NM)$.

3) *Complexity of SIWA*: By simulation results in Section 5.5, we observe that SIWA converges exponentially fast in ξ. Since in Step 1 all users can use N subcarriers simultaneously, each waterfilling step has $O(NK)$ time complexity. The time complexity becomes $O(NM)$ in Step 3, as no more than M users are active on each

subcarrier.

4) *Complexity of LDDP*: As seen in Subsections 3.3.3 and 4.3.5, the complexity of LDDP is $O(J^2NMK)$ for cellular power constraint. According to [20], the complexity of LDDP for individual power constraints is $O(CJ^2NMK)$, where the additional parameter C represents the number of sub-gradient optimization iterations upon termination. Recall that the power budget P_{max}^k is divided in J discrete power steps of value P_{max}^k/J and power allocation is performed by distributing these discrete power pieces among users and subcarriers. In Subsection 5.5.4, we study the trade-off between sum-rate performance and computational complexity of LDDP for different values of J. We determine that for the system described in Table 5.2, LDDP with $J = 10K$ is an appropriate choice as a near-optimal benchmark since its sum-rate does not improve significantly by further increasing J. In addition, we analyze how the performance gain in LDDP's optimization depends on J, N, K and M. We deduce that $J = O(\min\{K, MN\})$ is a good choice achieving near-optimal sum-rate in practice.

5) *Complexity of FTPC*: The complexity of FTPC is $O(NK)$, see [23], [24], [26].

5.5 Numerical Results

We evaluate the performance of the proposed algorithms by simulations. The radius R of the cell is set to 250 meters. Within the cell, there is one BS located at the center and K users uniformly distributed inside it. The system bandwidth W is assumed to be 5 MHz and $W_n = W/N$ for $n \in \mathcal{N}$, where $N = 10$. The noise power spectral density is assumed to be -174 dBm/Hz. In the radio propagation model, we include the distance-dependent path loss, shadow fading and small-scale fading. The distance-dependent path loss is given by $128.1 + 37.6 \log_{10} d$, in which d is the distance between the transmitter and the receiver in km. Log-normal shadowing has a standard deviation of 8 dB. For small-scale fading, each user experiences independent Rayleigh fading with variance 1. Each point in the following figures is the average value obtained over 2000 random instances. Only figures 5.2, 5.3 and 5.4 represent a single instance. The simulation parameters are summarized in Table 5.2.

We will compare the performance of the following schemes:

- Double iterative waterfilling algorithm (DIWA): we adopt the proposed three-step optimization framework and use SIWA in both Step 1 and 3.

Table 5.2: Simulation parameters (individual power constraints)

Parameter	**Value**
Cell radius	250 m
Min. distance from user to BS	35 m
Path loss model	$128.1 + 37.6 \log_{10} d$ dB, d is in km
Shadowing	Log-normal, 8 dB standard deviation
Fading	Rayleigh fading with variance 1
Noise power spectral density	-174 dBm/Hz
System bandwidth, W	5 MHz
Number of subcarriers N	10
Number of users K	3 to 20
Individual power budget P_{max}^k	0.5 W
Error tolerance ξ	10^{-4}
Number of power values J for LDDP	$10K$ (10 for each user)
Decay factor of FTPC	0.4
Parameter M	1 (OMA), 2 to 6 (NOMA)

- Double gradient algorithm (DGA): the projected gradient descent algorithm (Algorithm 9) is considered in both Step 1 and Step 3.

- Double pseudo-gradient algorithm (DPGA): we apply pseudo-gradient descent algorithm (Algorithm 10) in both Step 1 and Step 3.

- NOMA-SIWA-FTPC: in this scheme SIWA is applied in Step 1 to perform subcarrier allocation in Step 2. Then, FTPC optimizes the power control in Step 3.

- LDDP: the near-optimal scheme proposed in [20].

- OMA-FTPC: we apply FTPC in OMA system, that is $M = 1$.

5.5.1 Convergence of the Power Control Schemes

We investigate here the convergence of SIWA, DGA and DPGA in both Step 1 and Step 3 of our three-step scheme. The number of users is $K = 10$ and the number of active users per subcarrier is set to $M = 2$.

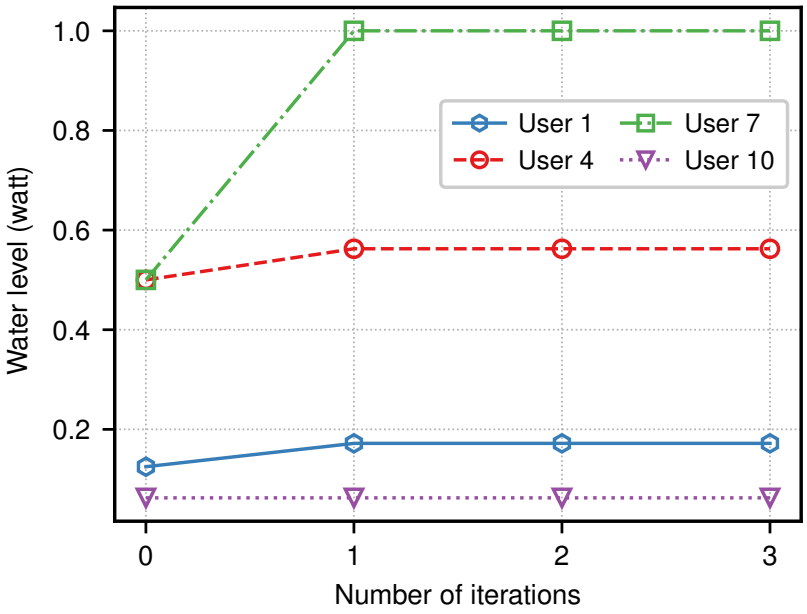

Figure 5.2: Convergence of SIWA in Step 1

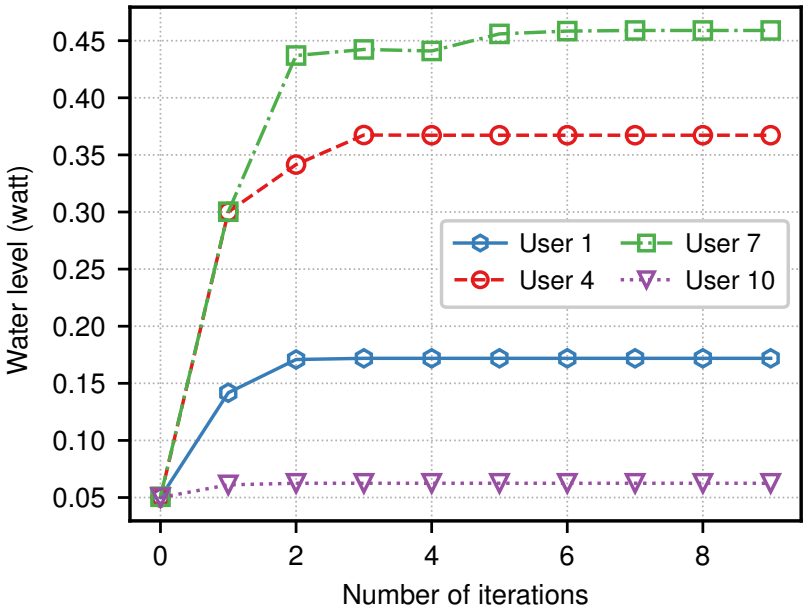

Figure 5.3: Convergence of SIWA in Step 3

Fig. 5.2 and Fig. 5.3 show the convergence of SIWA in Step 1 and Step 3, respectively. We choose four users on an arbitrary chosen instance and use their water levels at each iteration to demonstrate the convergence. As expected from the proof of Theorem 24 and from the fact that $M = 2$, the water level of each user in Step 3

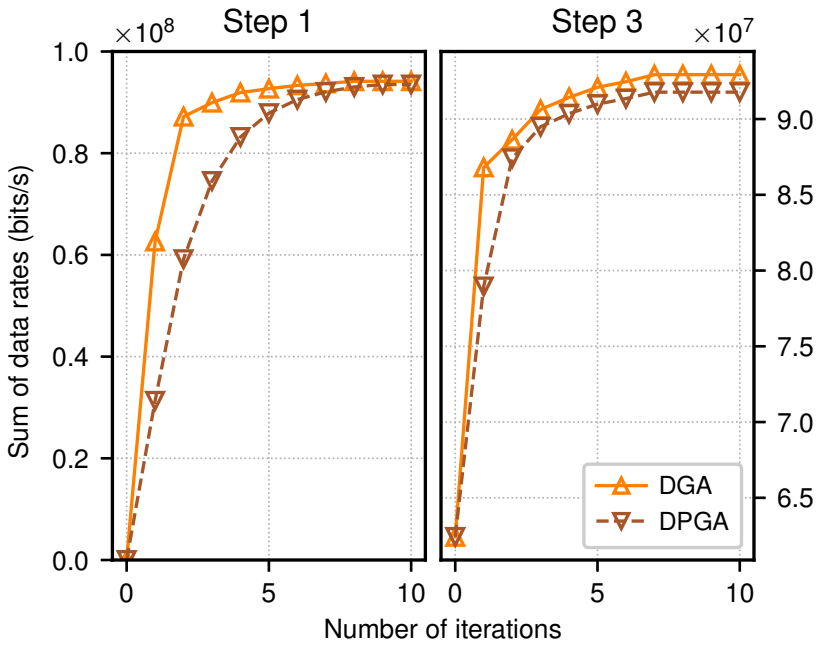

Figure 5.4: Convergence of DGA and DPGA

is monotonically increasing.

Fig. 5.4 shows the sum-rate over the iterations of DGA and DPGA in Step 1 and Step 3. In this example, these gradient descent based power control algorithms converge rapidly, which is consistent with the convergence analysis in Section 5.4.

5.5.2 Sum-Rate Performance

In Fig. 5.5 and 5.6, we compare the sum-rate performance of LDDP and DGA in NOMA system, as well as the performance of OMA-FTPC in OMA system. Parameter M is set to 2 and 3 in Fig. 5.5 and Fig. 5.6, respectively. In Fig. 5.5, we see that the sum-rate of each scheme increases with the number of users K. For any given K, LDDP has the best performance among the three schemes. Nevertheless, DGA achieves comparable performance to that of LDDP, see for example, when $K = 20$ and $M = 2$, the proposed DGA has a performance loss of only 3.14% when compared with LDDP. In addition, OMA-FTPC has the worst system performance among all. Similar conclusions can be drawn from Fig. 5.6. It is worth mentioning that the proposed scheme becomes more efficient when M increases. For example, when $K = 20$ and $M = 3$, the performance loss of DGA compared to LDDP is reduced to 1.71%.

In Fig. 5.7 and 5.8, we show the performance loss of our three-step heuristics:

DGA, DPGA, DIWA, and NOMA-SIWA-FTPC. The sum-rate performance loss when compared to the near-optimal LDDP is defined as follows:

$$\frac{\sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{N}} R_k^n(LLDP) - \sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{N}} R_k^n}{\sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{N}} R_k^n(LLDP)}, \tag{5.9}$$

where $\sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{N}} R_k^n(LLDP)$ and $\sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{N}} R_k^n$ refer to the solutions of LDDP and the three-step algorithm, respectively. In Fig. 5.7 we set $M = 2$, and in Fig. 5.8 we set $M = 3$. In both figures, DGA has the smallest performance loss among the four schemes. Meanwhile, DIWA has a similar performance to that of DPGA. In addition, NOMA-SIWA-FTPC has the largest performance loss among all.

Fig. 5.9 presents the sum-rate of LDDP and our proposed heuristics versus the number of multiplexed users M. The number of users K is fixed and equal to 6. It can be seen that when M increases, the performance gap between LDDP and our proposed schemes decreases. For example, when $M = 4$, the proposed DGA and SIWA have almost the same performance to that of LDDP, i.e., a performance loss of approximately 0.0016% and 0.0074%, respectively, while DPGA stagnates at 0.036% and NOMA-SIWA-FTPC has a performance loss of 0.37%.

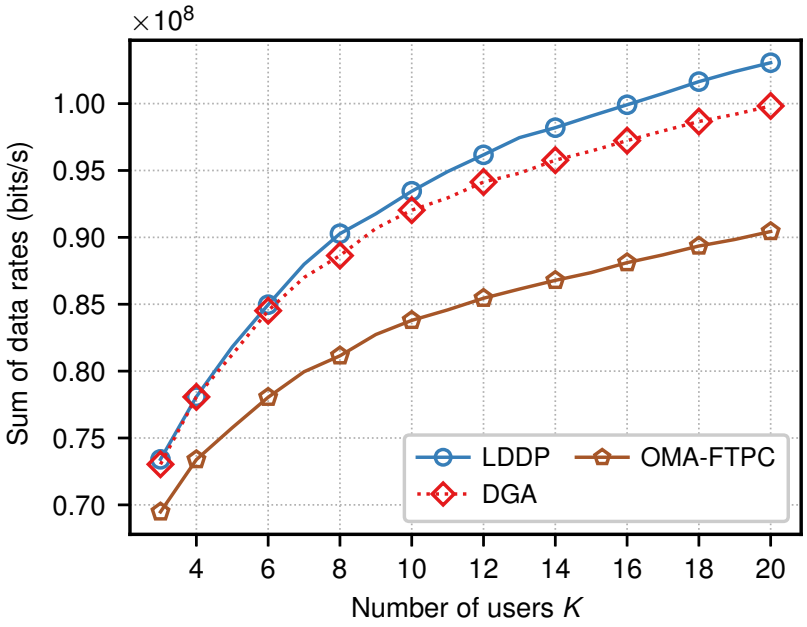

Figure 5.5: Sum-rate vs. K, for $M = 2$

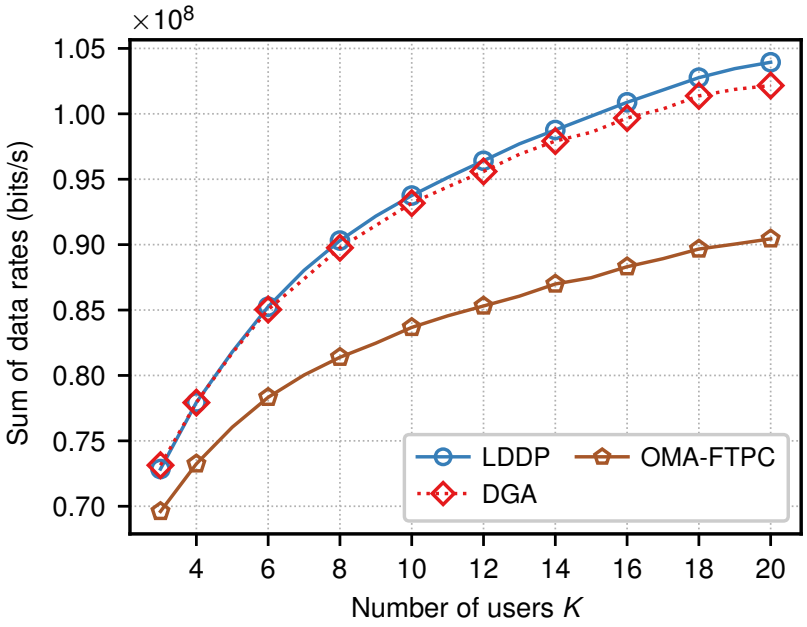

Figure 5.6: Sum-rate vs. K, for $M = 3$

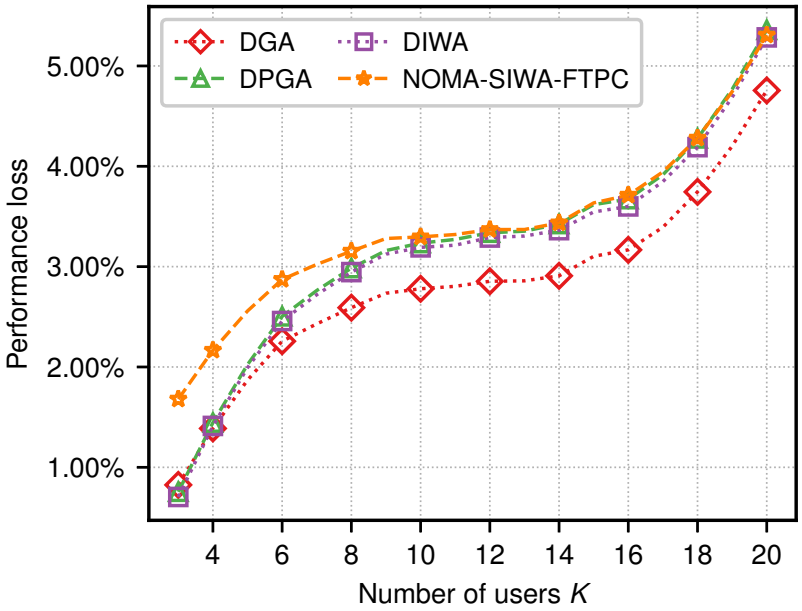

Figure 5.7: Performance loss vs. K, for $M = 2$

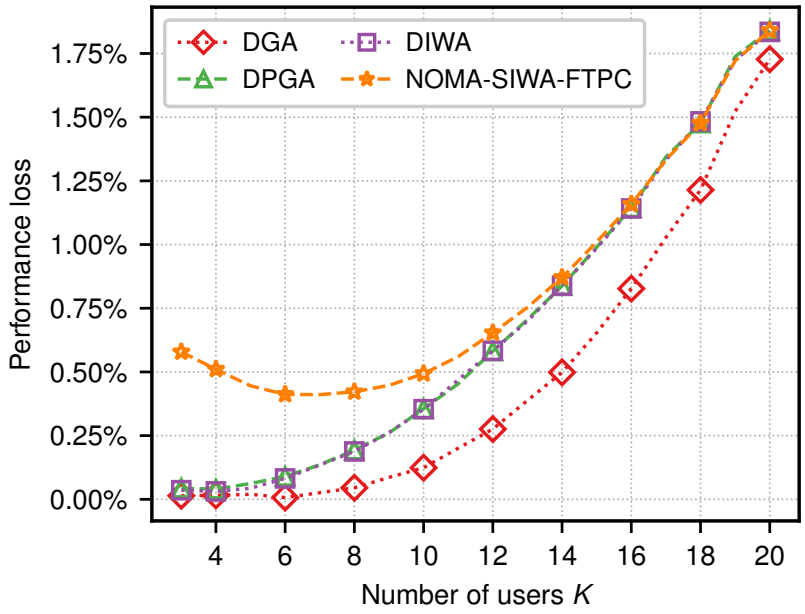

Figure 5.8: Performance loss vs. K, for $M = 3$

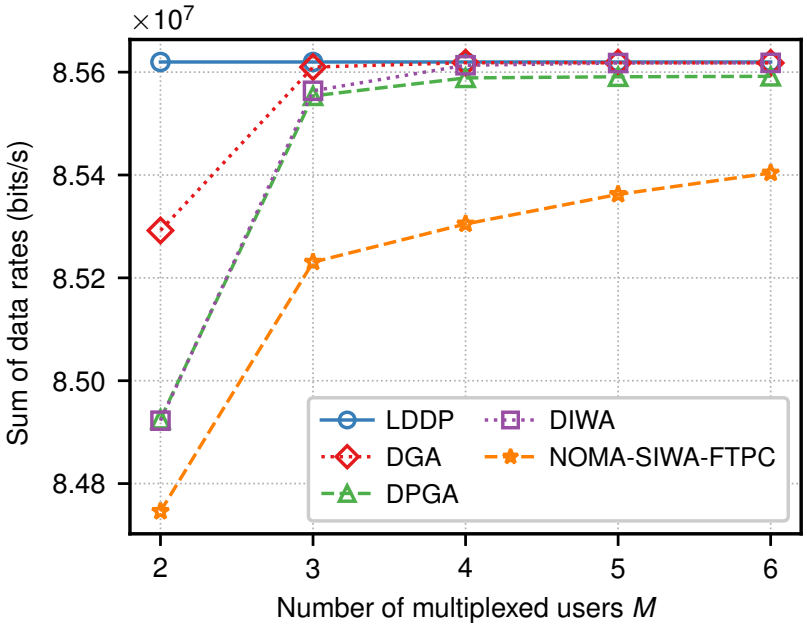

Figure 5.9: Sum-rate vs. M

5.5.3 Number of Operations

In this subsection, we evaluate the number of basic operations (additions, multiplications, comparisons) used by each algorithm, which reflects their computational complexity. In figures 5.10 and 5.11, the number of multiplexed users M is fixed to 2 and 3, respectively. The number of operations performed by each scheme increases with K. LDDP has the highest computational complexity. OMA-FTPC performs the fewest operations of all. In addition, the proposed DIWA uses approximately the same number of operations as NOMA-SIWA-FTPC and slightly more operations than OMA-FTPC but much less operations than DPGA and DGA. For any K, all of the proposed schemes are much more time efficient than LDDP especially when the number of users is large. For example, when $K = 20$ and $M = 2$, the number of operations required by DIWA, DPGA and DGA is less than 0.01%, 0.2%, and 0.27% of that required by LDDP, respectively. Besides, when $K = 20$ and $M = 3$, the number of operations of DIWA, DPGA and DGA is around 0.01%, 0.07%, and 0.1% of that needed by LDDP, respectively.

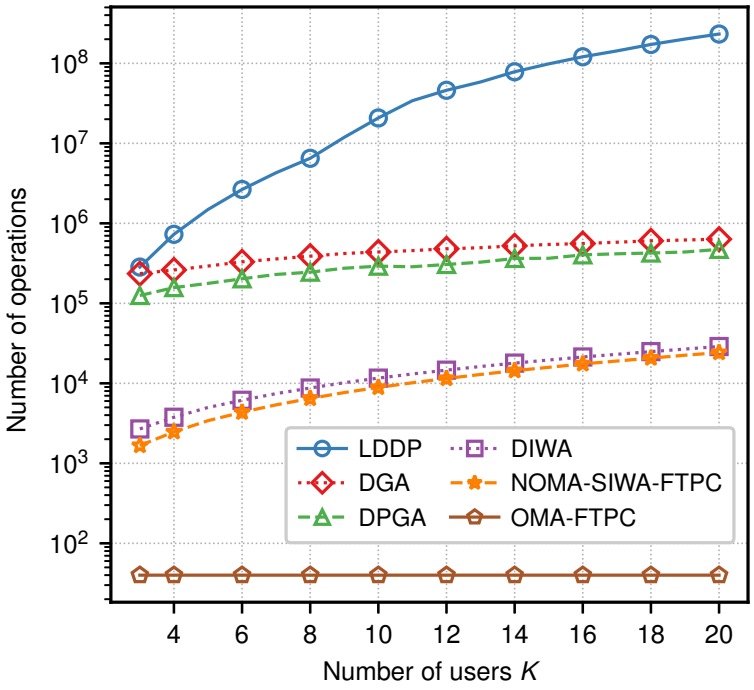

Figure 5.10: Number of operations vs. K, for $M = 2$

Fig. 5.12 shows the number of operations versus different values of M given a fixed number of users $K = 6$. The number of operations performed by LDDP is the highest of all. In addition, when M increases, the computational complexity of LDDP has the fastest growth rate among all the schemes.

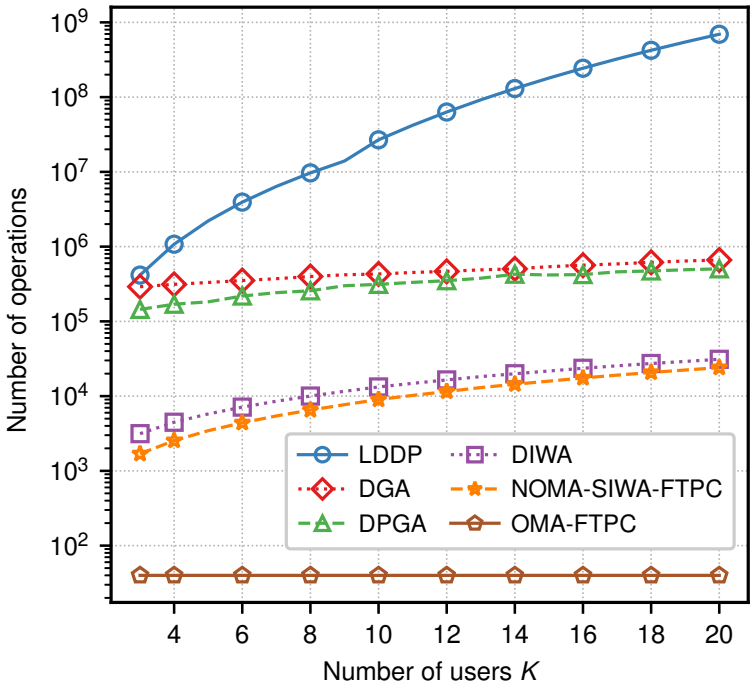

Figure 5.11: Number of operations vs. K, for $M = 3$

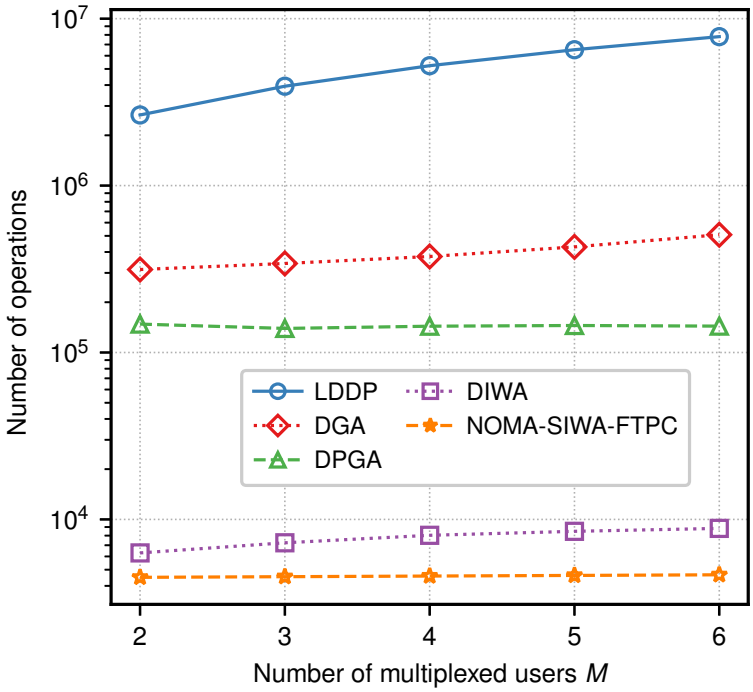

Figure 5.12: Number of operations vs. M

5.5.4 Impact of J on LDDP's Performance

We show in this subsection how the sum-rate and complexity of LDDP depends on the number of power levels J and the system's parameters. We present in figures 5.13 and 5.14 the sum-rate and number of operations of LDDP for different number of power values $J = 10, 20, 50, 200$ and $10K$, and compare them to the performance of DGA. As expected from [20], when J increases, the sum-rate of LDDP increases and tends the optimal, while the computational complexity increases in $O(CNMKJ^2)$. The number of power levels considered here is always greater than 10, otherwise LDDP would not have enough discrete power steps to serve all $N = 10$ subcarriers.

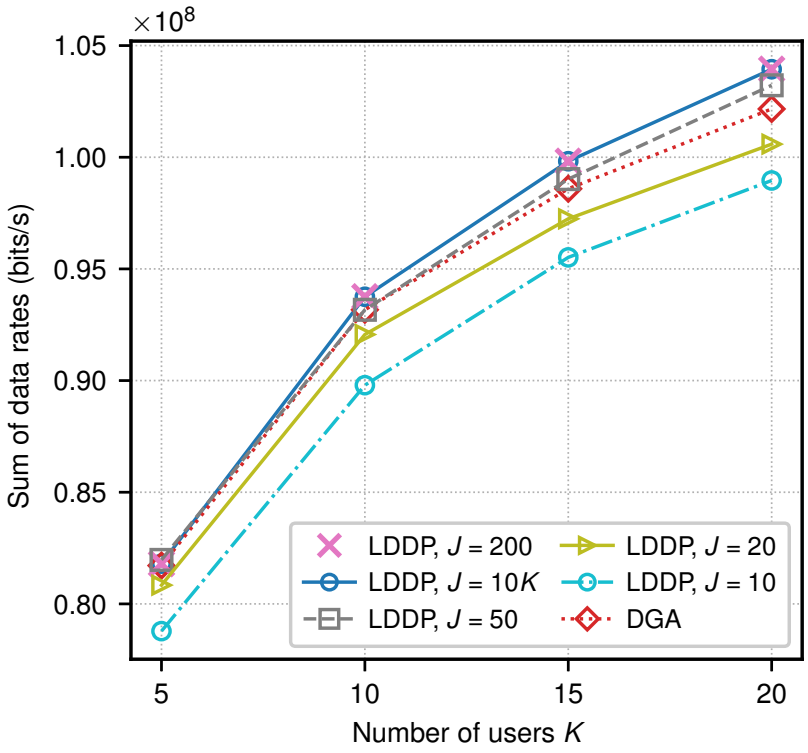

Figure 5.13: Sum-rate for different values of J, and $M = 3$

We observe in Fig. 5.13 that LDDP's sum-rate for $J = 200$ and $J = 10K$ are very close, i.e. they have less than 0.01% of difference for any number of users. Hence, LDDP's sum-rate does not improve significantly by further increasing J over $10K$. Furthermore, the sum-rate is reduced significantly as J decreases. This justifies the choice of LDDP with 10 power steps per user, i.e. $J = 10K$, as a benchmark near-optimal scheme in our simulations. Of course this value only holds for the system's parameters described in Table 5.2. We discuss below how J can be set in other systems depending on N, K and M.

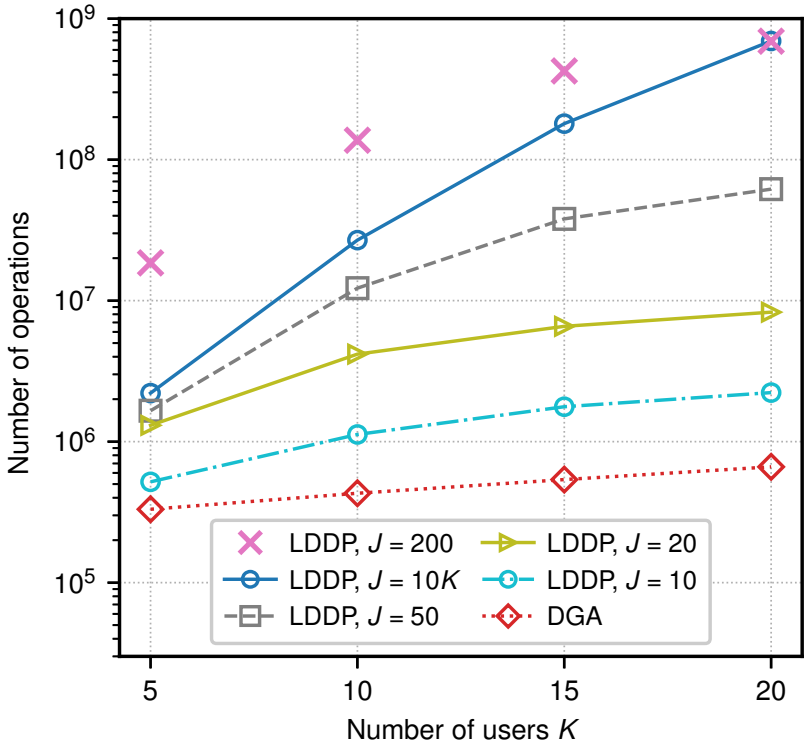

Figure 5.14: Number of operations for different values of J, and $M = 3$

As the number of resource increases (N, K and M), J should also increase in order to achieve similar gain in the joint subcarrier and power allocation. The idea is that each allocated user should have enough power steps to optimize the allocation of their individual power budget P_{max}^k among the subcarriers in LDDP. Given that the number of allocated user is at most $\min\{K, MN\}$, we choose empirically $J = \Theta(\min\{K, MN\})$, which achieves good performance by simulation. For example, we choose $J = 10K$ in Fig. 5.13, since $K < MN = 30$.

In Fig. 5.14, all LDDP schemes have higher complexity than DGA. Among these schemes, only LDDP with $J = 50, 200, 10K$ achieve better sum-rate than DGA. The performance gains compared to DGA, for $K = 20$ users, are respectively 1.04% and 1.74% for $J = 50$ and $J = 10K = 200$, while the increase in complexity are approximately 100 and 1000 folds. Moreover, DGA outperforms both LDDP with $J = 10$ and $J = 20$ in terms of higher sum-rate and lower complexity. For $K = 20$ users, the performance loss of LDDP with $J = 10$ and $J = 20$ compared to DGA are respectively 3.20% and 1.54%.

In summary, DGA has close to the best LDDP's sum-rate, while requiring less computational complexity. The latter statement can be explained by comparing the

asymptotic complexity of LDDP iteration $O(NMKJ^2)$ and DGA iteration $O(NK^2)$. As discussed previously, we can assume $J = O(\min\{K, MN\})$. If $K \leq MN$ (which is the case considered in our simulations), LDDP iteration's complexity can be written as $O(NMK^3)$. In this case, DGA requires MK times less operations than LDDP, which explains the results shown in Fig. 5.10, 5.11, 5.12 and Fig. 5.14. Otherwise if $K > MN$, then LDDP iteration's complexity becomes $O(N^3M^3K)$. This complexity is in practice higher than DGA's iteration complexity. Indeed, let's consider a 3GPP-LTE use case [83] with a bandwidth of 10 MHz divided in $N = 50$ resource blocks, and let $M = 2$, DGA remains advantageous in terms of computational complexity as long as $K < N^2M^3$, i.e. up to $20,000$ connected users.

5.5.5 User Fairness

Previous sections reveal the tradeoff between sum-rate performance and computational complexity achieved by the proposed resource allocation algorithms. We show in this subsection that various tradeoffs between sum-rate performance and user fairness can be obtained by adjusting the individual power budgets P_{max}^k, for $k \in \mathcal{K}$. This is particularly suitable for downlink systems, in which the BS can arbitrarily set the individual power constraints, as long as $\sum_{k\in\mathcal{K}} P_{max}^k \leq P_{max}$, where P_{max} is the total power budget available at the BS.

We adopt Jain's fairness index [84] to evaluate the user fairness in Fig. 5.15. It is defined as:

$$\frac{(\sum_{k=1}^{K} R_k)^2}{K\sum_{k=1}^{K} R_k^2}.$$

For simplicity, we choose DGA as an example to demonstrate the fairness performance. Similar conclusions can be obtained for the other three-steps heuristics. We choose $M = 2$ and we consider two individual power constraints strategies. In the first strategy, denoted by *DGA equal power constraints*, each user has the same power budget, i.e., $P_{max}^k = P_{max}/K$, where $k \in \mathcal{K}$. In the other strategy, denoted by *DGA proportional power constraints*, the power budget of user k is given as follows:

$$\bar{P}_{max}^k = \frac{(\sum_{n\in\mathcal{N}} g_k^n)^{-1}}{\sum_{j\in\mathcal{K}}(\sum_{n\in\mathcal{N}} g_j^n)^{-1}} P_{max}. \tag{5.10}$$

We see in Fig. 5.15 that DGA under proportional power constraints has higher fairness index than that of DGA with equal power constraints. For example, for $K = 20$, DGA with proportional power constraints improves Jain's fairness index by 60% when compared to that of equal power constraints scheme.

Fig. 5.16 shows the sum-rate performance of the two aforementioned strategies. DGA under equal power constraints achieves higher sum-rate than DGA with proportional power constraints since the latter scheme assigns more power to users with weak channel conditions. When $K = 20$, DGA with proportional power constraints has 7.3% performance loss compared to the equal power strategy.

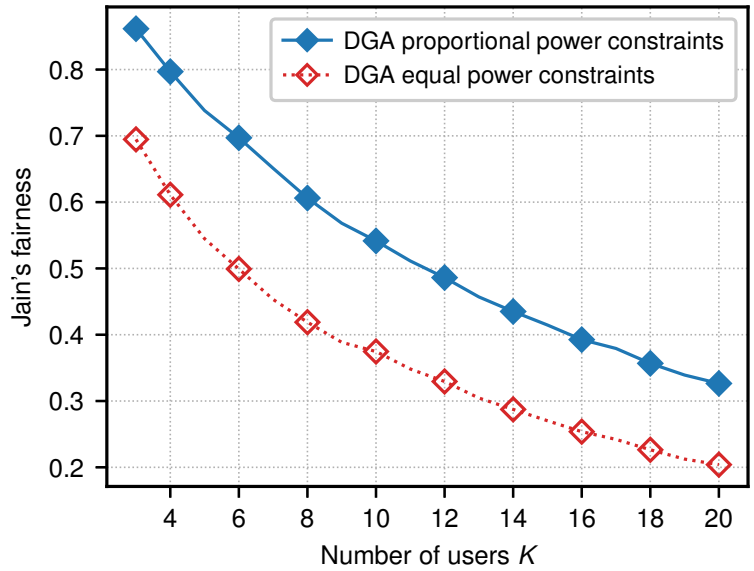

Figure 5.15: Fairness index vs. K, for $M = 2$

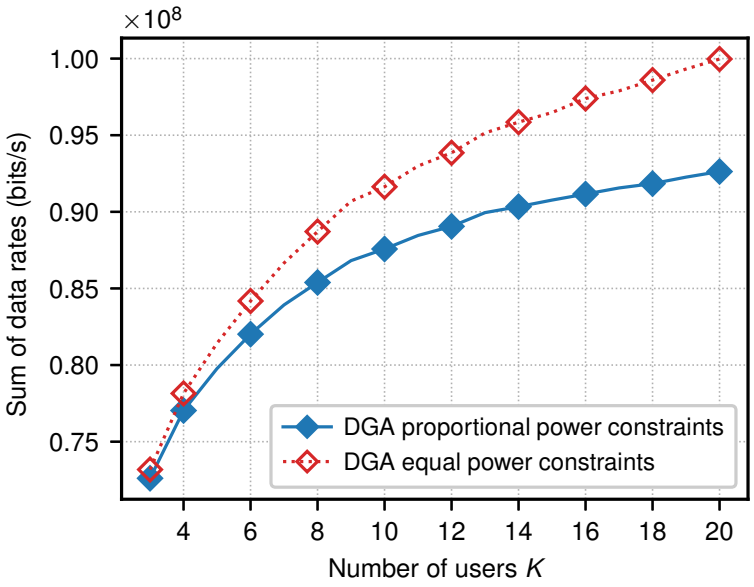

Figure 5.16: Sum-rate vs. K, for $M = 2$

Chapter 6

Conclusion

In this thesis, we introduce a novel optimization framework to tackle a general class of utility maximization JSPA problems. We show that these problems are NP-hard to solve in general. Nevertheless, we discuss about special cases in which optimal solutions can be computed in polynomial time. Then, we apply more specifically our framework to two constrained optimization problems: the WSR maximization with individual power constraints and the sum-rate maximization with cellular power constraint.

We propose three new algorithms for the WSR maximization with individual power constraints, namely Opt-JSPA, ε-JSPA and Grad-JSPA. Opt-JSPA computes an optimal solution with lower complexity than current optimal schemes in the literature, which makes it a suitable benchmark for optimal WSR performance in simulations. However, its pseudo-polynomial time complexity remains impractical for real-time applications. To further reduce the complexity, we propose a fully polynomial-time approximation scheme called ε-JSPA. It stands out by allowing to control a tight trade-off between performance guarantee and complexity. To the best of our knowledge, ε-JSPA is the first polynomial-time approximation scheme proposed for this problem. Finally, Grad-JSPA is a heuristic based on gradient descent. Numerical results show that it achieves near-optimal WSR with much lower complexity than existing optimal methods.

Regarding the sum-rate maximization with cellular power constraint, we first solve the power control sub-problem by an optimal gradient descent algorithm called GA. Furthermore, we propose a distributed game theoretic variant of this sub-problem in which each user optimizes its power allocation using local information only. We develop a pseudo-gradient descent method (PGA) which converges to

the unique Nash equilibrium, and a synchronous iterative waterfilling algorithm (SIWA). Finally, we extend these three power control algorithms to perform joint subcarrier and power allocation. The resultant schemes are called DGA (for the centralized problem), DPGA and DIWA (in the distributed settings). Numerical results show that these schemes have less than 5.5% of performance loss compared to the optimal in our simulations settings. Since DGA is centralized, simulations show that it achieves better performance than DPGA and DIWA. However, DPGA and DIWA require respectively around 30% and 90% less computations than DGA, while their average sum-rate is no less than 99% of that achieved by DGA. Finally, we show that various tradeoffs between sum-rate performance and user fairness can be obtained by adjusting the individual power budgets.

6.1 Future Work and Open Problems

In this section, we discuss about possible future research directions related to NOMA resource allocation problems. The first idea would be to apply the tools developed in our framework to new objective functions. For example, max-min fairness [85], [86] and connectivity maximization [87], [88] are of particular interest to increase the number of connected devices and enable mMTC. Our framework can also be extended to new constraints, such as the QoS constraints mentioned in Section 3.2.

In this thesis, as well as in many papers in the literature, the SIC decoding capability of the system is modeled as a limit M on the number of active users per subcarrier (see constraint $C4$ in $\mathcal{P}_i^I$ and $\mathcal{P}_i^C$). The superposition of at most M signals is assumed to be always perfectly decoded. A more realistic model would be to consider error propagation issues in the successive decoding procedure. Imperfect SIC has been studied in [47], [89]–[92]. It remains an open problem in many MC-NOMA scenarios and objectives, such as WSR and connectivity maximization.

In this work, we assume that the channel gains are perfectly known. Two more realistic models using only partial CSI can be considered instead: *imperfect CSI* studied in [10], [13], [93], for which the channel gains are given with a known estimation error probability distribution, and *second order statistics* (SOS) adopted in [94], for which only the distances between users and BS are known. In both cases, JSPA becomes a stochastic optimization problem, and it remains to be solved in many scenarios.

Another interesting direction is to further develop distributed resource allocation

algorithms. In multi-cell heterogeneous networks, the control signaling between BSs can be prohibitive for URLLC use cases. Hence, resource allocation can be done by each BS in a distributed manner, using mostly local information, and through managing the interference caused to other cells. Uplink grant-free NOMA is another important use case, which has received much attention [40], [41], [95].

The NP-complexity class is not enough to determine whether a problem can be solved efficiently or not. Indeed, as we show in Chapter 4, although computing an optimal solution of JSPA is intractable, good approximations can be obtained with low complexity. Thus, we believe that approximation algorithms and approximability classes [96] can have a significant impact on the design of RRM schemes.

Besides, machine learning and deep learning are promising methods to handle the increasing complexity of RRM in 5G and future wireless networks. They have already found many applications in the physical layer design of NOMA [91], [95], [97]–[100]. Consider modulation and channel detection as an example, the idea is to represent the system composed of a transmitter, the channel and the receiver as an autoencoder neural network [101]. However, only a few papers have considered deep learning for NOMA resource allocation problems. The authors of [49] use deep reinforcement learning (DRL) to solve the subcarrier allocation problem for sum-rate maximization with QoS constraint and for max-min fairness utility subject to cellular power constraint. In this case, the impact of deep learning is limited as these two problems can be tackled efficiently by classical optimization algorithms. Moreover, the proposed DRL requires to run a training phase each time the system changes, i.e., when the number of users and subcarriers varies, while these are simple inputs of the classical algorithms. We discuss here some examples in which deep learning has an advantage over classical optimization methods. First, deep learning can be applied to communication systems with a big number of complex parameters. In this respect, reference [102] developed a DRL approach for power allocation in a cache-aided NOMA system, and [92] considers a heterogeneous IoT system accessing the channel from a cognitive radio perspective and taking into account imperfect SIC. Secondly, machine learning methods are suitable for online resource allocation. The work in [103] optimizes the decoding order and power allocation of a satellite NOMA downlink system based on each user's packet buffer and time-varying channel state over the long-term. Finally, the authors of [104] use reinforcement learning to perform power allocation in the presence of smart jamming. This shows that machine learning approaches can be applied in adversarial environment.

Appendix A

Proofs of $\mathcal{P}_i^I$ Strong NP-Hardness

First, let us define the 3-Dimensional Matching Problem (3DM) as follows.

Definition 25 (3-Dimensional Matching Problem)**.**
The 3-dimensional matching problem (3DM) takes four finite sets as inputs: X, Y, Z, and S such that $|X| = |Y| = |Z|$ and $S \subseteq X \times Y \times Z$. Let $\mathcal{I}_{3DM}$ denotes the set of all possible inputs. The problem consists of deciding whether there exists a 3-dimensional matching $S' \subseteq S$ such that no two distinct triplets (x_1, y_1, z_1), $(x_2, y_2, z_2) \in S'$ overlap, i.e.,

$$(x_1, y_1, z_1) \neq (x_2, y_2, z_2) \implies x_1 \neq x_2, y_1 \neq y_2, z_1 \neq z_2,$$

and all elements are covered by S', i.e.,

$$|S'| = |X|. \tag{A.1}$$

For this NP-hardness proof, we consider pseudo-polynomial reductions $t_M : \mathcal{I}_{3DM} \mapsto \mathcal{I}_{\mathcal{D}_{i|M}^I}$ mapping any instance of 3DM to an instance of $\mathcal{D}_{i|M}^I$, for $M \geq 1$, $i \in [-\infty, 1]$. Pseudo-polynomial transformations preserve NP-hardness in the strong sense [63] and are defined for all instances $x_{3DM} \in \mathcal{I}_{3DM}$ as:

(i) x_{3DM} has a matching $\iff opt_{\mathcal{P}_i}(t_M(x_{3DM})) \geq T$,

(ii) t_M is polynomial time computable in the size of x_{3DM},

(iii) The largest numerical value of $t_M(x_{3DM})$ is lower and upper bounded by polynomials in the size of x_{3DM}.

In the following subsections, we will prove that $\mathcal{D}_{i|M}^I$ is strongly NP-hard for any fixed $M \geq 1$ and $i \in [-\infty, 1]$ by constructing the aforementioned pseudo-polynomial

reduction t_M. To this end, we first prove it in Lemma 26 for the sum-rate objective function $\mathcal{M}_{1,\boldsymbol{w}}$ with no more than $M = 1$ active user per subcarrier. Then, we extend this proof in Lemma 27 to any $M \geq 1$. Finally, we generalize it to any objective functions $\mathcal{M}_{i,\boldsymbol{w}}$, $i \leq 1$, which proves Theorem 6.

A.1 Sum-Rate Maximization with $M = 1$

Lemma 26. *For $M = 1$, problem $\mathcal{D}^I_{1|M}$ with sum-rate objective function $\mathcal{M}_{1,\boldsymbol{w}}$ is strongly NP-hard in both downlink and uplink scenarios.*

Proof. The idea of the proof is to construct a reduction $t_1 \colon \mathcal{I}_{3DM} \mapsto \mathcal{I}_{\mathcal{D}^I_{1|1}}$ mapping any instance of 3DM to an instance of $\mathcal{D}^I_{1|1}$ in which no more than one user is allocated to each subcarrier. We first detail t_1 and show that it satisfies conditions (ii) and (iii). Then, we prove condition (i) for $T = 3$. As a result, t_1 is a well defined pseudo-polynomial reduction, and it follows from 3DM's strong NP-hardness [66] that $\mathcal{D}^I_{1|1}$ is also strongly NP-hard.

Let $x_{3DM} = (X, Y, Z, S) \in \mathcal{I}_{3DM}$. Without loss of generality, we can assume that $|S| \geq |X|$, otherwise x_{3DM} has trivially no matching according to (A.1). The corresponding instance $t_1(x_{3DM}) \in \mathcal{I}_{\mathcal{D}^I_{1|1}}$ is given by:

- $K = |S|$ users. There is a bijective mapping between users $k \in \mathcal{K}$ and triplets $(x_k, y_k, z_k) \in S$.
- $N = |S| + 2|X|$ subcarriers divided into four groups $\mathcal{N}_X$, $\mathcal{N}_Y$, $\mathcal{N}_Z$ and $\mathcal{N}_R$.

The first three groups $\mathcal{N}_X$, $\mathcal{N}_Y$, $\mathcal{N}_Z$ are called *primary* subcarriers and are in bijection with X, Y and Z respectively. For notational simplicity, we index them by their corresponding set, e.g., $n_x \in \mathcal{N}_X$ corresponds to $x \in X$. The same goes for Y and Z. This way, we have $N_X = N_Y = N_Z = |X|$ subcarriers in each of these primary groups. The set $\mathcal{N}_R$ is called the *residual* group, it contains $N_R = |S| - |X|$ subcarriers.

The channel gains of user $k \in \mathcal{K}$ whose corresponding triplet is $(x_k, y_k, z_k) \in S$ are set as follows:

$$\forall n \in \mathcal{N},\ g_k^n = \begin{cases} 1 & \text{if } n \in \{n_{x_k}, n_{y_k}, n_{z_k}\}, \\ 1 & \text{if } n \in \mathcal{N}_R, \\ 0 & \text{otherwise.} \end{cases}$$

And the noise powers are set as follows:

$$\forall n \in \mathcal{N},\ \eta_k^n = \begin{cases} 3/7 & \text{if } n \in \mathcal{N}_R, \\ 1 & \text{otherwise.} \end{cases}$$

Noise powers are the same for all users on a given subcarrier, therefore both downlink and uplink scenarios are covered in this proof. We further consider equal weights $\boldsymbol{w} = \{1/K, \ldots, 1/K\}$ in the objective function, assume that $W_n = 1$ for all $n \in \mathcal{N}$ and $\bar{P}_k = 3$ for all $k \in \mathcal{K}$. Moreover, its allocated power on each subcarrier $n \in \mathcal{N}$ is subject to constraint $C2$ such that:

$$\bar{p}_k^n = \begin{cases} 1 & \text{if } n \in \mathcal{N}_X \cup \mathcal{N}_Y \cup \mathcal{N}_Z, \\ 3 & \text{if } n \in \mathcal{N}_R. \end{cases} \tag{A.2}$$

For this reduction, we set the decision problem's threshold (3.2) to be $T = 3$. We have characterized above the transformed instance $t_1(x_{3DM})$ and its parameters. The number of parameters is polynomially bounded in the size of x: there are $|S|$ users, $|S|+2|X|$ subcarriers, and so on. Thus, by construction our reduction satisfies property (ii). Condition (iii) is also satisfied, since all numerical values are constant, regardless of the size of x. It only remains to prove (i) in order to conclude that t_1 is indeed a pseudo-polynomial reduction, i.e.,

$$x_{3DM} \text{ has a matching } \iff opt_{\mathcal{P}_1}(t_1(x_{3DM})) \geq 3. \tag{A.3}$$

No more than one user can be served on each subcarrier according to $M = 1$ in $C4$. If we suppose that the total system power $\sum_{k\in\mathcal{K}} \bar{P}_k = 3K$ can be distributed among the N subcarriers without constraint $C1$, then the optimal is obtained by the following waterfilling power allocation [38]:

$$\forall n \in \mathcal{N}_R,\ R^n = \log_2(1 + \frac{3}{3/7}) = 3, \tag{A.4}$$

$$\forall n \in \mathcal{N}_X \cup \mathcal{N}_Y \cup \mathcal{N}_Z,\ R^n = \log_2(1 + \frac{1}{1}) = 1. \tag{A.5}$$

The best solution consists in having the maximum allowable power on every subcarrier while meeting the constraint $C2$. The corresponding user allocation allocates one user on every primary subcarrier with maximum power 1 and one user per residual subcarrier with maximum power 3. According to the problem setting, there is no other optimal power and subcarrier allocation. There are $3|X|$ primary subcarriers and $|S| - |X|$ residual subcarriers, thus the sum-rate objective is:

$$\mathcal{M}_{1,\boldsymbol{w}}(\boldsymbol{R}) = \frac{3|X| \times 1 + (|S| - |X|) \times 3}{K} = 3. \tag{A.6}$$

Since our problem is constrained by $C1$, the optimal cannot be greater than (A.6), i.e., $opt_{\mathcal{P}_1}(t_1(x_{3DM})) \leq 3$. It follows that the equivalence (A.3) to prove together with the derived upper bound can be rewritten as:

$$x_{3DM} \text{ has a matching} \iff opt_{\mathcal{P}_1}(t_1(x_{3DM})) = 3. \tag{A.7}$$

Proof of the part $\Longleftarrow$: Let $x_{3DM} = (X, Y, Z, S)$ be an instance of 3DM. Assume that the corresponding instance $t_1(x_{3DM})$ has a power and subcarrier allocation which is optimal and equal to 3. We have seen that the only possibility to achieve this optimum is to allocate every triplet of subcarriers $(n_x, n_y, n_z) \in X \times Y \times Z$ to a user for which channel gain is 1 with power 1 and every residual subcarrier to the remaining $|S| - |X|$ users with power 3. Now, let us define $S' \subset S$ such that $(x, y, z) \in S'$ iff n_x, n_y and n_z are allocated to the same user. By construction, S' is a matching for x_{3DM}.

Proof of the part $\Longrightarrow$: Let $x_{3DM} = (X, Y, Z, S)$ be an instance of 3DM for which there exists a matching S'. Consider the following power and subcarrier allocation: for every indexes $(x_k, y_k, z_k) \in S'$ allocate subcarriers n_{x_k}, n_{y_k}, n_{z_k} to user k with power 1; allocate the remaining users to the residual subcarriers with power 3. Then the objective function is exactly 3, which is also an upper bound. As a consequence, $opt(t_1(x_{3DM})) = 3$. □

A.2 Sum-Rate Maximization with $M \geq 1$

Lemma 27. *For any $M \geq 1$, problem $\mathcal{D}^I_{1|M}$ with sum-rate objective function $\mathcal{M}_{1,\boldsymbol{w}}$ is strongly NP-hard in both downlink and uplink scenarios.*

Proof. The idea of this proof is to extend Lemma 26's reduction to any $M \geq 1$ by adding $N(M-1)$ dummy users. For each subcarrier $n \in \mathcal{N}$, we create $M-1$ dummy users, denoted by the index set $\mathcal{D}^n = \{d^n_1, \ldots, d^n_{M-1}\}$. Thus, the set of all users becomes $\mathcal{K}' = \mathcal{K} \cup \mathcal{D}^1 \cup \cdots \cup \mathcal{D}^N$, where $\mathcal{K}$ is the users set defined in Lemma 26's proof. All parameters of the transformation $t_M(x_{3DM}) \in \mathcal{I}_{\mathcal{D}^I_{1|M}}$ related to user $k \in \mathcal{K}$ and subcarriers $n \in \mathcal{N}$ remain as $t_1(x_{3DM})$ in Lemma 26's proof. In addition, we keep equal weights $\boldsymbol{w} = \{1/|\mathcal{K}'|, \ldots, 1/|\mathcal{K}'|\}$.
The following construction aims to guarantee that dummy users in $\mathcal{D}^n$ can only be active on subcarrier n. For any $j \in \{1, \ldots, M-1\}$, parameters of user d^n_j on

subcarrier $n' \in \mathcal{N}$ are set as follows:

$$g_{d_j^n}^{n'} = \begin{cases} 1 & \text{if } n' = n, \\ 0 & \text{otherwise.} \end{cases} \tag{A.8}$$

and,

$$\bar{p}_{d_j^n}^{n'} = \begin{cases} \bar{P}_{d_j^n} & \text{if } n' = n, \\ 0 & \text{otherwise.} \end{cases} \tag{A.9}$$

The total power constraint $C1$ is extended as follows:

$$\bar{P}_{d_j^n} = \begin{cases} 14 \times 8^{M-j-1} & \text{if } n \in \mathcal{N}_X \cup \mathcal{N}_Y \cup \mathcal{N}_Z, \\ 24 \times 8^{M-j-1} & \text{if } n \in \mathcal{N}_R. \end{cases} \tag{A.10}$$

Let $\boldsymbol{p}^* = (p_k^{n*})_{k \in \mathcal{K}', n \in \mathcal{N}}$ denotes the optimal power allocation of $t_M(x_{3DM})$. Let $n \in \mathcal{N}$, since dummy users in $\mathcal{D}^n$ have greater power budget than any other user in $\mathcal{K}$ (compare (A.10) to (A.2)), it is straightforward to see that the optimal is achieved when all $M-1$ dummy users in $\mathcal{D}^n$ are multiplexed on subcarrier n with the following power allocation:

$$\forall j \in \{1, \ldots, M-1\},\ p_{d_j^n}^{n*} = \bar{P}_{d_j^n}. \tag{A.11}$$

We consider the following decoding order:

$$\forall j \in \{1, \ldots, M-1\},\ \pi_n(j) = d_j^n. \tag{A.12}$$

This decoding order satisfies (2.13) and (2.14), therefore both downlink and uplink scenarios are covered in this proof. It is interesting to note that any desired decoding order can be achieved by adjusting the above dummy users' channel gains, noise powers and power budgets.

It remains that subcarrier n can be allocated to an additional non-dummy user $k \in \mathcal{K}$, while respecting constraints $C4$. In this case, according to (2.13) and (2.14), user k is decoded last, i.e., $\pi_n(M) = k$. Thus, k is not subject to interference from the dummy users on subcarrier n. Furthermore, at the optimal, no more than one user in $\mathcal{K}$ can be multiplexed on each subcarrier n. It follows that the optimal subcarrier and power allocation of users $\mathcal{K}$ in $t_M(x_{3DM})$ is the same as in $t_1(x_{3DM})$ and we have:

$$opt_{\mathcal{P}_1}(t_1(x_{3DM})) = 3 \iff opt_{\mathcal{P}_1}(t_M(x_{3DM})) = \frac{3K + \sum_{n \in \mathcal{N}} \sum_{j=1}^{M-1} R_{d_j^n}(\boldsymbol{p}^*)}{|\mathcal{K}'|}. \tag{A.13}$$

Using (A.8-A.12), we can compute the optimal data rate of dummy user d_j^n, for any $j \in \{1, \dots, M-1\}$, on primary subcarriers $n \in \mathcal{N}_X \cup \mathcal{N}_Y \cup \mathcal{N}_Z$ as

$$\begin{aligned} R_{d_j^n} &= \log_2(1 + \frac{\bar{P}_{d_j^n}}{\sum_{j'=j+1}^{M-1} \bar{P}_{d_j^n} + 2}) \\ &= \log_2(1 + \frac{14 \times 8^{M-j-1}}{\sum_{j'=j+1}^{M-1} 14 \times 8^{M-j'-1} + 2}) \\ &= \log_2(1 + \frac{14 \times 8^{M-j-1}}{14(1-8^{M-j-1})/(1-8) + 2}) \qquad \text{(A.14)} \\ &= 3 \qquad \text{(A.15)} \end{aligned}$$

where (A.14) is obtained by calculating the partial sum of the geometric sequence $\sum_{j'=j+1}^{M-1} 14 \times 8^{M-j'-1}$ with ratio 8 and $M-j-1$ terms. In the same way, we prove that for all residual subcarriers $n \in \mathcal{N}_R$,

$$R_{d_j^n} = 3. \qquad \text{(A.16)}$$

Combining (A.15) and (A.16), equivalence (A.13) then becomes

$$opt_{\mathcal{P}_1}(t_1(x_{3DM})) = 3 \iff opt_{\mathcal{P}_1}(t_M(x_{3DM})) = \frac{3K + 3N(M-1)}{|\mathcal{K}'|} = 3. \qquad \text{(A.17)}$$

Last equality is deduced from $|\mathcal{K}'| = |\mathcal{K} \cup \mathcal{D}^1 \cup \cdots \cup \mathcal{D}^N| = K + N(M-1)$. Equivalence (A.18) follows from (A.17) and (A.7), which implies that t_M is a pseudo-polynomial reduction.

$$x_{3DM} \text{ has a matching} \iff opt_{\mathcal{P}_1}(t_M(x_{3DM})) = 3. \qquad \text{(A.18)}$$

We then conclude from (A.18) and Lemma 26 that $\mathcal{D}^I_{1|M}$ is also strongly NP-hard, for any $M \geq 1$. □

A.3 Generalized Mean Utility Maximization with $M \geq 1$

Theorem 6. *For any $i \in [-\infty, 1]$ and $M \geq 1$, problem $\mathcal{D}^I_{i|M}$ with objective function $\mathcal{M}_{i,\boldsymbol{w}}$ is strongly NP-hard in both downlink and uplink scenarios. In particular, the sum-rate $\mathcal{M}_{1,\boldsymbol{w}}$, proportional fairness $\mathcal{M}_{0,\boldsymbol{w}}$, harmonic mean utility $\mathcal{M}_{-1,\boldsymbol{w}}$ and max-min fairness $\mathcal{M}_{-\infty,\boldsymbol{w}}$ versions of the problem are all strongly NP-hard.*

Proof. Let $i \in [-\infty, 1)$, $M \geq 1$ and $x_{3DM} \in \mathcal{I}_{3DM}$ be an instance of 3DM. Using Lemma 27's reduction t_M, we showed that finding a 3-dimensional matching of x_{3DM} is equivalent to verifying $opt_{\mathcal{P}_1}(t_M(x_{3DM})) = 3$, i.e., (A.18). More precisely, when (A.18) is satisfied, all users achieve the same data rate. Indeed, for any user $k \in \mathcal{K}$, there are three possibilities:

- k is a dummy user then $R_k = 3$ according to (A.15) and (A.16), or
- k is not a dummy user and it is active on a residual subcarrier $n \in \mathcal{N}_R$ with power 3 so that $R_k = R_k^n = \log_2(1 + \frac{3}{3/7}) = 3$, i.e., (A.4), or
- k is not a dummy user and it is active on three primary subcarriers $n_{x_k} \in \mathcal{N}_X$, $n_{y_k} \in \mathcal{N}_Y$ and $n_{z_k} \in \mathcal{N}_Z$ so that $R_k = R_k^{n_{x_k}} + R_k^{n_{y_k}} + R_k^{n_{z_k}} = 3\log_2(1 + \frac{1}{1}) = 3$, i.e., (A.5).

It follows that:

$$x_{3DM} \text{ has a matching} \iff \forall k \in \mathcal{K},\ R_k(\boldsymbol{p}^*) = 3, \tag{A.19}$$

where $\boldsymbol{p}^*$ is the optimal power allocation. Since $i < 1$, the generalized mean inequality (3.1) implies that $opt_{\mathcal{P}_i}(t_M(x_{3DM}))$ is also upper bounded by 3 and the equality holds when all individual data rates are equal to 3, i.e.,

$$opt_{\mathcal{P}_i}(t_M(x_{3DM})) = opt_{\mathcal{P}_1}(t_M(x_{3DM})) = 3 \iff \forall k \in \mathcal{K},\ R_k(\boldsymbol{p}^*) = 3, \tag{A.20}$$

where $\boldsymbol{p}^*$ is an optimal power allocation of either $\mathcal{P}_i$ or $\mathcal{P}_1$ (this choice does not matter, as they are equal when (A.20) is satisfied). Finally, we derive equivalence (A.21) from (A.19) and (A.20), which proves that $\mathcal{D}_{i|M}^I$ is strongly NP-hard.

$$x_{3DM} \text{ has a matching} \iff opt_{\mathcal{P}_i}(t_M(x_{3DM})) \geq 3. \tag{A.21}$$

□

Appendix B

Proofs of Chapter 4

B.1 Proof of Lemma 7

Proof. The objective of $\mathcal{P}_1^C$ can be written as:

$$\begin{aligned}
\sum_{k\in\mathcal{K}} w_k \sum_{n\in\mathcal{N}} R_k^n\left(\boldsymbol{p}^n\right) &= \sum_{n\in\mathcal{N}}\sum_{k\in\mathcal{K}} w_k R_k^n\left(\boldsymbol{p}^n\right), \\
&\overset{(b)}{=} \sum_{n\in\mathcal{N}} W_n \sum_{i=1}^{K} w_{\pi_n(i)} \log_2\left(\frac{\sum_{j=i}^{K} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(i)}^n}{\sum_{j=i+1}^{K} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(i)}^n}\right), \\
&\overset{(c)}{=} \sum_{n\in\mathcal{N}} W_n \sum_{i=1}^{K} \log_2\left(\frac{\left(\sum_{j=i}^{K} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(i)}^n\right)^{w_{\pi_n(i)}}}{\left(\sum_{j=i+1}^{K} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(i)}^n\right)^{w_{\pi_n(i)}}}\right), \\
&\overset{(d)}{=} \sum_{n\in\mathcal{N}} W_n \Bigg[w_{\pi_n(1)} \log_2\left(\sum_{j=1}^{K} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(1)}^n\right) \\
&\quad + \sum_{i=2}^{K} \log_2\left(\frac{\left(\sum_{j=i}^{K} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(i)}^n\right)^{w_{\pi_n(i)}}}{\left(\sum_{j=i}^{K} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(i-1)}^n\right)^{w_{\pi_n(i-1)}}}\right) \\
&\quad + w_{\pi_n(K)} \log_2\left(\frac{1}{\tilde{\eta}_{\pi_n(K)}^n}\right)\Bigg].
\end{aligned}$$

Equality (b) comes from the definition of R_k^n in (2.15). At (c), the weights $w_{\pi_n(i)}$ are put inside the logarithm. Finally, (d) is obtained by combining the numerator of the i-th term with the denominator of the $(i-1)$-th term, for $i \in \{2, \ldots, K\}$.

By applying the change of variables shown in (4.1), we derive the equivalent problem $\mathcal{P}_1^{C\prime}$. The constant term is $A = \sum_{n\in\mathcal{N}} w_{\pi_n(K)} \log_2\left(1/\tilde{\eta}_{\pi_n(K)}^n\right)$. Constraints $C1'$ and $C2'$ are respectively equivalent to $C1$ and $C2$ since $x_1^n = \sum_{j=1}^{K} p_{\pi_n(j)}^n = \sum_{k\in\mathcal{K}} p_k^n$, for $n \in \mathcal{N}$. Constraints $C3'$ and $C3''$ come from $C3$ and the fact that

$x_i^n - x_{i+1}^n = p_{\pi_n(i)}^n$, for any $i \in \{1, \ldots, K\}$ and $n \in \mathcal{N}$. In the same way, the active users set in $C4'$ is defined as $\mathcal{U}_n' \triangleq \{i \in \{1, \ldots, K\} \colon x_i^n > x_{i+1}^n\}$. □

B.2 Proof of Lemma 8

Proof. We study the first and second derivatives of $f_{j,i}^n$, denoted by $f_{j,i}^{n\,\prime}$ and $f_{j,i}^{n\,\prime\prime}$. If $j = 1$, then we have:

$$f_{1,i}^{n\,\prime}(x) = \frac{W_n w_{\pi_n(i)}}{\left(x + \tilde{\eta}_{\pi_n(i)}^n\right) \ln(2)}, \tag{B.1}$$

which is strictly positive and decreasing for $x \geq 0$. Hence, $f_{1,i}^n$ is increasing and concave. For $j > 1$, the first and second derivatives are as follows:

$$f_{j,i}^{n\,\prime}(x) = \frac{W_n}{\ln(2)} \left(\frac{w_{\pi_n(i)}}{x + \tilde{\eta}_{\pi_n(i)}^n} - \frac{w_{\pi_n(j-1)}}{x + \tilde{\eta}_{\pi_n(j-1)}^n} \right),$$

$$f_{j,i}^{n\,\prime\prime}(x) = \frac{W_n}{\ln(2)} \left(\frac{w_{\pi_n(j-1)}}{\left(x + \tilde{\eta}_{\pi_n(j-1)}^n\right)^2} - \frac{w_{\pi_n(i)}}{\left(x + \tilde{\eta}_{\pi_n(i)}^n\right)^2} \right).$$

We know that $\tilde{\eta}_{\pi_n(j-1)}^n \geq \tilde{\eta}_{\pi_n(i)}^n$ by definition of the optimal downlink decoding order in Eqn. (2.13). If, in addition, we have $w_{\pi_n(i)} \geq w_{\pi_n(j-1)}$, then $f_{j,i}^{n\,\prime}(x) \geq 0$ and $f_{j,i}^{n\,\prime\prime}(x) \leq 0$ for all $x \geq 0$. We deduce that $f_{j,i}^n$ is increasing and concave. This proves the first point of Lemma 8. Now suppose that $w_{\pi_n(i)} < w_{\pi_n(j-1)}$ instead. Values c_1 and c_2 defined in Lemma 8 are the unique roots of the first and second derivatives, i.e., $f_{j,i}^{n\,\prime}(c_1) = 0$ and $f_{j,i}^{n\,\prime\prime}(c_2) = 0$. $f_{j,i}^{n\,\prime}$ is positive on $\left(-\tilde{\eta}_{\pi_n(j-1)}, c_1\right)$ and negative on (c_1, ∞). This implies that $f_{j,i}^n$ is unimodal and has a unique global maximum at c_1 for $x > 0$. Similarly, $f_{j,i}^{n\,\prime\prime}$ is negative on $\left(-\tilde{\eta}_{\pi_n(j-1)}, c_2\right)$ and positive on (c_2, ∞). Therefore, $f_{j,i}^n$ is concave before c_2 and convex after c_2. This proves the second point of Lemma 8. □

B.3 Proof of Theorem 9

Proof. The complexity and optimality proofs of SCPC are presented below.

Complexity analysis: At each for loop iteration i, the while loop at line 6 has at most i iterations. Thus, the worst case complexity is proportional to $\sum_{i=1}^{|\mathcal{U}_n'|} i = O(|\mathcal{U}_n'|^2) = O(M^2)$.

Optimality analysis: Without loss of generality, we can suppose that the $x^n_{i_n}$'s are initialized to zero. We will prove by induction that at the end of each iteration i at line 10 of Algorithm 2, the following loop invariants are true:

$H_1(i)$: $\sum_{l=1}^{i} \tilde{f}^n_{l,l}$ is maximized by $x^n_{1_n}, \ldots, x^n_{i_n}$,

$H_2(i)$: $C2$–$3'$ is satisfied, i.e., $\bar{P}^n \geq x^n_{1_n} \geq \cdots \geq x^n_{i_n} \geq 0$.

Basis: For $i = 1$, x^* computed at line 3 is indeed the optimal of $\tilde{f}^n_{1,1}$. The while loop has no effect since $j = 0 < 1$, therefore $x^n_{1_n} \leftarrow x^*$ and statements $H_1(1)$ and $H_2(1)$ are both true.

Inductive step: Assume that $x^n_{1_n}(i-1), \ldots, x^n_{(i-1)_n}(i-1)$ are the variables verifying $H_1(i-1)$ and $H_2(i-1)$ at iteration $i - 1 < K$. Let the variables at iteration i be $x^n_{1_n}, \ldots, x^n_{i_n}$. We consider two cases:

i) We first suppose that:

$$x^* = \textsc{Argmax} \tilde{f}(i, i, \mathcal{I}^n, \bar{P}^n) \leq x^n_{(i-1)_n}(i-1). \tag{B.2}$$

In this case, Algorithm 2 sets $x^n_{i_n} = x^*$ and $x^n_{l_n} = x^n_{l_n}(i-1)$, for all $l < i$. The induction hypothesis $H_2(i-1)$ states that $\bar{P}^n \geq x^n_{1_n} \geq \cdots \geq x^n_{(i-1)_n} \geq 0$. By taking into account Eqn. (B.2), this inequality becomes $\bar{P}^n \geq x^n_{1_n} \geq \cdots \geq x^n_{(i-1)_n} \geq x^* = x^n_{i_n} \geq 0$. Thus, $H_2(i)$ is satisfied. In addition, we know from $H_1(i-1)$ that $x^n_{1_n}, \ldots, x^n_{(i-1)_n}$ maximizes $\sum_{l=1}^{i-1} \tilde{f}^n_{l,l}$. Since, the objective is separable and $x^n_{i_n} = x^*$ maximizes $\tilde{f}^n_{i,i}$ by construction, $H_1(i)$ is true.

ii) Now, suppose that we have the opposite:

$$x^* = \textsc{Argmax} \tilde{f}(i, i, \mathcal{I}^n, \bar{P}^n) > x^n_{(i-1)_n}(i-1). \tag{B.3}$$

In this case, the allocation mentioned above would violate constraint $C2$–$3'$. The algorithm finds the highest index $j \in \{1, \ldots, i-2\}$ such that $x^n_{j_n}(i-1) \geq \textsc{Argmax} \tilde{f}(j+1, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n)$ in the while loop at line 6. Such an index exists since all variables are upper bounded by $\bar{P}^n$ and $x^n_{1_n} = \bar{P}^n$ due to Lemma 8. Let us show by contradiction that $H_1(i)$ and $H_2(i)$ are only satisfied if $x^n_{(j+1)_n} = \cdots = x^n_{i_n}$. If it is not the case, let $k > j+1$ be the last index such that $x^n_{k_n} = x^n_{(k+1)_n} = \cdots = x^n_{i_n}$ and $x^n_{(k-1)_n} > x^n_{k_n}$. We know from the while condition that $x^n_{(k-1)_n} < x^{*\prime}$, with $x^{*\prime} = \textsc{Argmax} \tilde{f}(k, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n)$. According to Lemma 8, $\tilde{f}^n_{k,i}$ is increasing on $[0, x^{*\prime}]$. Therefore, we can improve the objective function

by setting $x^n_{k_n}, \dots, x^n_{i_n} \leftarrow x^n_{(k-1)_n}$. This is a contradiction with $x^n_{(k-1)_n} > x^n_{k_n}$, we have thus $x^n_{(j+1)_n} = \cdots = x^n_{i_n}$. Furthermore, at the termination of the while loop, we have $\textsc{Argmax}\tilde{f}(j+1, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n) \leq x^n_{j_n}(i-1)$, which can be treated as in case i). Hence, variables $x^n_{(j+1)_n}, \dots, x^n_{i_n}$ are set equal to $\textsc{Argmax}\tilde{f}(j+1, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n)$ at line 10, and it satisfies $H_1(i)$ and $H_2(i)$.

We proved that, in both cases i) and ii), the allocation $x^n_{1_n}, \dots, x^n_{i_n}$ computed by Algorithm 2 satisfies $H_1(i)$ and $H_2(i)$. Therefore, by mathematical induction, the allocation returned at line 12 satisfies $H_1(|\mathcal{U}'_n|)$ and $H_2(|\mathcal{U}'_n|)$. We note that $H_1(|\mathcal{U}'_n|)$ and $H_2(|\mathcal{U}'_n|)$ are equivalent to an optimal solution of $\mathcal{P}^{C\prime}_{SCPC}(n)$, which concludes the proof. □

B.4 Proof of Theorem 10

Proof. The complexity and optimality proofs of i-SCPC are presented below.

Complexity analysis: The initialization consists in running SCPC, with complexity $O(M^2)$ (see Theorem 9). Each subsequent evaluation requires to compute $\min\{x^n_{i_n}, \bar{P}^n\}$, for $i \in \{1, \dots, |\mathcal{U}'_n|\}$, with complexity $O(M)$.

Optimality analysis: Let $x^n_{1_n}, \dots, x^n_{|\mathcal{U}'_n|_n}$ be the optimal allocation of SCPC with budget P_{max}. We consider now a lower budget $\bar{P}^n \leq P_{max}$. At each iteration i of the loop in $\text{SCPC}\big(\mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\big)$, the value $\textsc{Argmax}\tilde{f}\big(j, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\big)$ can be replaced by $\min\{\textsc{Argmax}\tilde{f}(j, i, \mathcal{I}^n, \mathcal{U}'_n, P_{max}), \bar{P}^n\}$, since they are equal by definition. One can show, by mathematical induction on i_n, that the function $\text{SCPC}\big(\mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\big)$ returns $\min\{x^n_{1_n}, \bar{P}^n\}, \dots, \min\{x^n_{|\mathcal{U}'_n|_n}, \bar{P}^n\}$. Therefore, the latter allocation is also optimal. □

B.5 Proof of Theorem 11

Proof. The complexity and optimality proofs of SCUS are presented below.

Complexity analysis: The complexity mainly comes from the computation of V, X and U in the for loop from lines 13 to 27, which requires $M\sum_{i=1}^{K-1}(i) = O(MK^2)$ iterations. Each iteration has a constant number of operations. Thus, the overall worst case computational complexity is $O(MK^2)$.

Optimality analysis: We will prove by induction that at any iteration $m \in \{0, \dots, M\}$, $j \in \{1, \dots, K\}$ and $i \geq j$ of Algorithm 4, the construction of $V[m, j, i]$

is the optimal value of problem $\mathcal{P}_{SC}^{C\prime}[m,j,i]$. It follows directly that $V[M,1,1]$ is the optimal value of $\mathcal{P}_{SC}^{C\prime}(n)$.

Basis: For $m=0$, no user can be active due to constraint $C4'$. Thus, $V[0,j,i] = f_{j,K}^n(0)$ and $X[0,j,i]$ is initialized to zero. Furthermore, $U[0,j,i] = \varnothing$ to indicate that there is no previous index in the recursion. For simplicity of the algorithm, V, X, U are also initialized for $j \leq i = K$ as explained in Section 4.2.2.

Inductive step: Let $m \in \{1,\dots,M\}$ and $1 \leq j \leq i \leq K-1$. Assume that $V[m',j',i']$ is the optimal value of $\mathcal{P}_{SC}^{C\prime}[m',j',i']$ for any $m' \leq m$, $j' \geq j$ and $i' > i$. We denote the optimal solution of problem $\mathcal{P}_{SC}^{C\prime}[m,j,i]$ by $x_j^n,\dots,x_K^n$. Let v_{act} (resp. v_{inact}) be the optimal value of $\mathcal{P}_{SC}^{C\prime}[m,j,i]$, given that user i is active (resp. inactive). Let $x_{(i+1)_n}^{n*} = X[m-1,i+1,i+1]$ be the optimal value of $x_{(i+1)_n}^n$ in $\mathcal{P}_{SC}^{C\prime}[m-1,i+1,i+1]$. If $x^* \leq x_{(i+1)_n}^{n*}$, then we can prove as in case ii) of Appendix B.3, that user i is inactive in the optimal solution. In this case, $V[m,j,i] = v_{inact}$. Otherwise, the optimal is $V[m,j,i] = \max\{v_{act}, v_{inact}\}$. Values v_{act} and v_{inact} are computed as follows:

- Case v_{inact}: Suppose that the optimal solution of problem $\mathcal{P}_{SC}^{C\prime}[m,j,i]$ is achieved when user i is inactive, then we have $x_i^n = x_{i+1}^n$ by definition of $\mathcal{U}_n'$. It follows from $C5'$ that $x_j^n = \cdots = x_{i+1}^n$. We obtain, by definition, $V[m,j,i] = V[m,j,j+1]$, which we denote by v_{inact}.

- Case v_{act}: Suppose now that user i is active. Since $x^* > x_{(i+1)_n}^{n*}$ satisfies $C3'$, and the objective is separable, the optimal is obtained when maximizing independently $f_{j,i}^n$ and $\sum_{l=i+1}^K f_l^n$ with $m-1$ active users. That is, $V[m,j,i] = v_{act} \triangleq f_{j,i}^n(x^*) + V[m-1,i+1,i+1]$, where $x^* = \textsc{Argmax} f\left(j,i,\mathcal{I}^n,\bar{P}^n\right)$ in line 15.

Hence, $V[m,j,i]$, as computed in (4.3), corresponds to the optimal of $\mathcal{P}_{SC}^{C\prime}[m,j,i]$. We derive, by mathematical induction, that $V[M,1,1]$ is the optimal value of $\mathcal{P}_{SC}^{C\prime}[M,1,1]$, which is equivalent to $\mathcal{P}_{SC}^{C\prime}(n)$. The corresponding optimal allocation $\boldsymbol{x}^n$ is retrieved in lines 28 to 35. □

B.6 Proof of Theorem 12

Proof. The complexity and optimality proofs of i-SCUS are presented below.

Optimality analysis: Let $y_1^n,\dots,y_K^n$ be the optimal solution of $\mathcal{P}_{SC}^{C\prime}(n)$ subject to a power constraint $\bar{P}^n$. Let $i \in \{1,\dots,K\}$ be the unique index such that $y_1^n = \cdots = y_i^n$ and $y_i^n > y_{i+1}^n$. We know from Lemma 8 that $y_1^n = \cdots = y_i^n = \bar{P}^n$.

Therefore, $y_{i+1}^n, \dots, y_K^n$ are all strictly less than $\bar{P}^n$. Let $x_1^n, \dots, x_K^n$ be the optimal solution of $\mathcal{P}_{SC}^{C\prime}[M, 1, i]$ in the execution of SCUS$(\mathcal{I}^n, M, P_{max})$, i.e., subject to a power budget P_{max}. According to Lemma 8, $x_1^n = \cdots = x_i^n = P_{max}$. We deduce from f's unimodality in Lemma 8, that $y_{i+1}^n, \dots, y_K^n$ is the optimal solution of $\mathcal{P}_{SC}^{C\prime}[M, i+1, i+1]$ given any power budget no less than $\bar{P}^n$. In particular, we have $x_l^n = y_l^n$, for all $l \in \{i+1, \dots, K\}$. Hence, $x_1^n, \dots, x_K^n$ and $y_1^n, \dots, y_K^n$ correspond to the same user selection $\mathcal{U}_n'$, and we derive $y_{l_n}^n = \min\{x_{l_n}^n, \bar{P}^n\}$, for $1 \leq l \leq |\mathcal{U}_n'|$.

We proved above that, for any $\bar{P}^n \leq P_{max}$, there exists $(\mathcal{U}_n', x_1^n, \dots, x_K^n)$ in *collection*, such that the optimal allocation subject to the power constraint $\bar{P}^n$ is $\min\{x_{1_n}^n, \bar{P}^n\}, \dots, \min\{x_{|\mathcal{U}_n'|_n}^n, \bar{P}^n\}$. Thus, the optimal user selection and power control is the one maximizing $F^n(\mathcal{U}_n', \bar{P}^n) = \sum_{l=1}^{|\mathcal{U}_n'|} \tilde{f}_{l,l}^n\left(\mathcal{U}_n', \min\{x_{l_n}^n, \bar{P}^n\}\right) + B^n$ over all elements in *collection*, as shown at line 6 of Algorithm 5.

Complexity analysis: The initialization consists in running SCUS, with complexity $O(MK^2)$ (see Theorem 11). Each subsequent evaluation has complexity $O(MK)$. Indeed, there are K active users sets $\mathcal{U}_n'$ in *collection*, one for each solution of $\mathcal{P}_{SC}^{C\prime}[M, 1, i]$, for $i \in \{1, \dots, K\}$. For each of the K possible active users set $\mathcal{U}_n'$ in *collection*, we compute $F^n(\mathcal{U}_n', \bar{P}^n)$ with complexity $O(|\mathcal{U}_n'|) = O(M)$. □

B.7 Proofs of Lemma 13 and Theorem 14

We first provide in Lemma 29 an important property on the solution maximizing $\sum_{l=1}^{i} \tilde{f}_{l,l}^n$ subject to $C2$–$3'$, for $i \leq |\mathcal{U}_n'|$. This Lemma will be used subsequently in the proofs of Lemma 13 and Theorem 14.

Lemma 29.

Assume we are given a subcarrier $n \in \mathcal{N}$, a set $\mathcal{U}_n'$ of active users, a power budget $\bar{P}^n$, and an index i. Let $x_{1_n}^n, \dots, x_{i_n}^n$ be the allocation maximizing $\sum_{l=1}^{i} \tilde{f}_{l,l}^n\left(\mathcal{U}_n', x_{l_n}^n\right)$, while also satisfying $C2$–$3'$, i.e., $\bar{P}^n \geq x_{1_n}^n \geq \cdots \geq x_{i_n}^n \geq 0$. $x_{1_n}^n, \dots, x_{i_n}^n$ can be partitioned into sequences of consecutive terms with the same value. That is, sequences of the form $x_{q_n}^n, \dots, x_{q_n'}^n$, where $x_{q_n}^n = \cdots = x_{q_n'}^n$ and $1 \leq q \leq q' \leq |\mathcal{U}_n'|$, $q = 1$ or $x_{(q-1)_n}^n > x_{q_n}^n$, $q' = |\mathcal{U}_n'|$ or $x_{q_n'}^n < x_{(q'+1)_n}^n$. Any such sequence satisfies:

$$x_{q_n}^n = \cdots = x_{q_n'}^n = \textsc{Argmax} \tilde{f}\left(q, q', \mathcal{I}^n, \mathcal{U}_n', \bar{P}^n\right).$$

Proof. In this proof, we simplify notation $\tilde{f}_{l,l}^n\left(\mathcal{U}_n', \cdot\right)$ as $\tilde{f}_{l,l}^n\left(\cdot\right)$. Let $x_{q_n}^n, \dots, x_{q_n'}^n$ be a sequence of consecutive terms with the same value, as defined in Lemma 29. Assume, for the sake of contradiction, that $x_{q_n}^n = \cdots = x_{q_n'}^n \neq x^*$, where $x^* =$

$\textsc{Argmax}\tilde{f}\left(q, q', \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right)$. Without loss of generality, we consider the case $x^n_{q_n} < \textsc{Argmax}\tilde{f}\left(q, q', \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right)$ and $q > 1$. Let $y^n_{1_n}, \ldots, y^n_{i_n}$ be an allocation defined as:

$$y^n_{l_n} \triangleq \begin{cases} \min\{x^n_{(q-1)_n}, x^*\}, & \text{if } q \le l \le q', \\ x^n_{l_n}, & \text{otherwise.} \end{cases} \tag{B.4}$$

We have the following inequalities:

$$\sum_{l=1}^{i} \tilde{f}^n_{l,l}\left(y^n_{l_n}\right) = \sum_{l \notin \{q,\ldots,q'\}} \tilde{f}^n_{l,l}\left(x^n_{l_n}\right) + \tilde{f}^n_{q,q'}\left(y^n_{l_n}\right), \tag{B.5}$$

$$> \sum_{l \notin \{q,\ldots,q'\}} \tilde{f}^n_{l,l}\left(x^n_{l_n}\right) + \tilde{f}^n_{q,q'}\left(x^n_{l_n}\right) = \sum_{l=1}^{i} \tilde{f}^n_{l,l}\left(x^n_{l_n}\right). \tag{B.6}$$

Equality (B.5) comes from the definition in (B.4). According to Lemma 8, $\tilde{f}^n_{q,q'}$ is increasing on $[0, x^*]$, which implies inequality (B.6). In summary, $y^n_{1_n}, \ldots, y^n_{i_n}$ satisfies $C2$–$3'$ by its definition in (B.4), and it achieves greater value of $\sum_{l=1}^{i} \tilde{f}^n_{l,l}$ than $x^n_{1_n}, \ldots, x^n_{i_n}$. This is a contradiction, therefore it must be that

$$x^n_{q_n} \ge \textsc{Argmax}\tilde{f}\left(q, q', \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right). \tag{B.7}$$

If $q = 1$, the same reasoning can be applied by replacing $\min\{x^n_{(q-1)_n}, x^*\}$ by $\bar{P}^n$ in Eqn. (B.4). We can perform a similar proof by contradiction on the case $x^n_{q_n} > \textsc{Argmax}\tilde{f}\left(q, q', \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right)$ to deduce that:

$$x^n_{q_n} \le \textsc{Argmax}\tilde{f}\left(q, q', \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right). \tag{B.8}$$

The desired result follows from (B.7) and (B.8). □

The proofs of Lemma 13 and Theorem 14 are given below.

Proof. Let $x^n_1, \ldots, x^n_K$ be the output of i-SCUS$\left(\bar{P}^n\right)$, and $\mathcal{U}'_n$ the corresponding active users set. For $i \in \{1, \ldots, K\}$, there exists $q \le i$ and $q' \ge i$, such that $x^n_{q_n} = \cdots = x^n_{q'_n} = \textsc{Argmax}\tilde{f}\left(q, q', \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right)$, according to Lemma 29. We have:

$$\tilde{f}^n_{q,q'}\left(\mathcal{U}'_n, \min\{x^n_{q_n}, \bar{P}^n\}\right) = \begin{cases} \tilde{f}^n_{q,q'}\left(\mathcal{U}'_n, \bar{P}^n\right), & \text{if } \bar{P}^n \le x^n_{q_n}, \\ \tilde{f}^n_{q,q'}\left(\mathcal{U}'_n, x^n_{q_n}\right), & \text{if } \bar{P}^n > x^n_{q_n}. \end{cases}$$

We consider it as a function of $\bar{P}^n$. Its left derivative at $\bar{P}^n = x^n_{q_n}$ is 0, according to Lemma 8. Its right derivative at $\bar{P}^n = x^n_{q_n}$ is 0, as it is constant for $\bar{P}^n > x^n_{q_n}$. Hence, $\tilde{f}^n_{q,q'}(\mathcal{U}'_n, \min\{x_{q_n}, \cdot\})$ is continuously differentiable on $[0, P_{max}]$.

Let l be the greatest index such that $x_l^n = \bar{P}^n$. The function $F^n(\mathcal{U}_n', \bar{P}^n)$ can be written as $f_{1,l}^n\left(\bar{P}^n\right) + \sum_{i=l+1}^{K} f_{i,i}^n\left(x_i^n\right) + B^n$. Its derivative can be obtained by applying Eqn. (B.1) of Appendix B.2 as follows:

$$F^{n\prime}\left(\mathcal{U}_n', \bar{P}^n\right) = f_{1,l}^{n\prime}\left(\bar{P}^n\right) = \frac{W_n w_{\pi_n(l)}}{\left(\bar{P}^n + \tilde{\eta}_{\pi_n(l)}^n\right)\ln(2)}. \tag{B.9}$$

As $F^n(\bar{P}^n) = \max_{\mathcal{U}_n'}\{F^n(\mathcal{U}_n', \bar{P}^n)\}$, where max is taken over all active users sets in *collection* of i-SCUS, and the max operator only preserves semi-differentiability, Eqn. (B.9) is the left derivative of F^n. This proves Lemma 13.

In addition, the second left derivative of F^n satisfies:

$$\beta \leq F^{n\prime\prime}\left(\bar{P}^n\right) = \frac{-W_n w_{\pi_n(l)}}{\left(\bar{P}^n + \tilde{\eta}_{\pi_n(l)}^n\right)^2 \ln(2)} \leq \alpha < 0, \tag{B.10}$$

where β and α are constant and defined as:

$$\beta = \frac{-W_n w_{\pi_n(l)}}{\left(\tilde{\eta}_{\pi_n(l)}^n\right)^2 \ln(2)},$$
$$\alpha = \frac{-W_n w_{\pi_n(l)}}{\left(P_{max} + \tilde{\eta}_{\pi_n(l)}^n\right)^2 \ln(2)}.$$

Although F^n is only semi-differentiable at some points, it is twice differentiable on each interval where the optimal user selection $\mathcal{U}_n'$ does not change. Appendix B.6 shows that there are K such intervals. Eqn. (B.10) implies that F^n is piece-wise twice differentiable, α-strongly concave and β-smooth. Therefore, the projected gradient descent on the simplex $\mathcal{F}_{MC}$ converges in $O(\log(1/\xi))$ iterations, according to [77, Section 2.2.4]. This proves Theorem 14. □

B.8 Proof of Theorem 15

Proof. We first deduce from Eqn. (2.15) that for each subcarrier $n \in \mathcal{N}$, the rates R_k^n of all users $k \in \mathcal{K}$ can be sorted in the same order as the decoding order π_n. That is, for any power allocation $\boldsymbol{p}^n$ on subcarrier $n \in \mathcal{N}$, we have:

$$R_{\pi_n(1)}^n(\boldsymbol{p}^n) \leq R_{\pi_n(2)}^n(\boldsymbol{p}^n) \leq \cdots \leq R_{\pi_n(K)}^n(\boldsymbol{p}^n).$$

If the weights $\boldsymbol{w} = \boldsymbol{w}_{eq} = \{1/K, \ldots, 1/K\}$ are all equals, we then derive:

$$w_{\pi_n(1)} R_{\pi_n(1)}^n(\boldsymbol{p}^n) \leq w_{\pi_n(2)} R_{\pi_n(2)}^n(\boldsymbol{p}^n) \leq \cdots \leq w_{\pi_n(K)} R_{\pi_n(K)}^n(\boldsymbol{p}^n).$$

Hence, the optimal active user set on subcarrier $n \in \mathcal{N}$ maximizing the objective function $\mathcal{M}_{1,\boldsymbol{w}_{eq}}(\boldsymbol{R}(\boldsymbol{p})) = \sum_{k\in\mathcal{K}} w_k \sum_{n\in\mathcal{N}} R_k^n(\boldsymbol{p}^n)$, with no more than M users, is:

$$\begin{aligned} \mathcal{U}_n^* &= \{\pi_n(K-M+1), \ldots, \pi_n(K)\}, && \text{in the initial problem formulation } \mathcal{P}_1^C, \\ \mathcal{U}_n'^* &= \{K-M+1, \ldots, K\}, && \text{in the separable problem formulation } \mathcal{P}_1^{C\prime}. \end{aligned}$$

We know by definition that $F^n(\bar{P}^n) = \max_{\mathcal{U}_n'}\{F^n(\mathcal{U}_n', \bar{P}^n)\}$. Eqn (B.9) in Appendix B.7 shows that $F^n(\mathcal{U}_n', \bar{P}^n)$ is twice differentiable, α-strongly concave and β-smooth, for any user selection $\mathcal{U}_n'$. However, $F^n(\bar{P}^n)$ is only piece-wise concave in general due to the max operator. In our case, since $\mathcal{U}_n'^*$ is the optimal user selection, we can remove the max operator and write $F^n(\bar{P}^n) = F^n(\mathcal{U}_n'^*, \bar{P}^n)$. Thus, $F^n(\bar{P}^n)$ is twice differentiable, α-strongly concave and β-smooth. The same can be said for $\sum_{n\in\mathcal{N}} F^n(\bar{P}^n)$, which is the objective of $\mathcal{P}_{MC}^{C\prime}$. We conclude that the projected gradient descent on the simplex $\mathcal{F}_{MC}$ converges to the global optimum in $O(\log(1/\xi))$ iterations, according to [77, Section 2.2.4]. □

B.9 Proof of Theorem 16

Proof. Let us first briefly explain the principle of dynamic programming by weights. Let Z be a 2D-array such that $Z[n,l]$ is defined as the optimal value of MCKP restricted to the first n classes and with restricted capacity $l \cdot \delta$. It is initialized as $Z[0,l] = 0$, for any $l = 0, \ldots, J$. For $n \in \mathcal{N}$ and $l = 0, \ldots, J$, $Z[n,l]$ is defined by the following recurrence relation:

$$Z[n,l] = \max_{l' \le l}\{Z[n-1, l-l'] + c_{n,l'}\}.$$

The complexity and optimality proofs of Opt-JSPA are presented below.

Complexity analysis: In Algorithm 7, we first transform $\mathcal{P}_{MC}^{C\prime}$ to MCKP: from line 1 to 5, every item's profit $c_{n,l}$ is computed using i-SCUS in $O(NMK^2 + JNMK)$. Then, we perform dynamic programming by weights at line 6. According to [71], its complexity is $O(J^2N)$, which is the number of items $N(J+1)$ multiplied by the number of possible power values $J+1$. Therefore, the overall complexity is $O(NMK^2 + JNMK + J^2N)$.

Optimality analysis: Reference [71] proves that dynamic programming by weights is optimal for MCKP. Since problems $\mathcal{P}_{MC}^{C\prime}$ and MCKP are equivalent, the proposed Opt-JSPA based on dynamic programming by weights is also optimal for $\mathcal{P}_{MC}^{C\prime}$. □

B.10 Estimation U in Algorithm 8

In this section, we denote by $F^*_{MCKP}(P_{max})$ the optimal value of MCKP with cellular power budget P_{max}. We provide some properties in Lemma 30 that will be used for the analysis of the estimation procedure.

Lemma 30 (Monotonicity and sublinearity of F^*_{MCKP})**.**
F^*_{MCKP} is a non-decreasing and sublinear function of P_{max}. That is, for any $P_1 < P_2$, $F^*_{MCKP}(P_1) \leq F^*_{MCKP}(P_2)$ and $F^*_{MCKP}(P_1 + P_2) \leq F^*_{MCKP}(P_1) + F^*_{MCKP}(P_2)$.

Proof. We first prove the monotonicity of F^*_{MCKP}. Let $\mathcal{F}'_1$ and $\mathcal{F}'_2$ be two feasible sets of $\mathcal{P}^{C\prime}_{MC}$ with power budget P_1 and P_2 respectively. Assuming $P_1 < P_2$, then any solution of $\mathcal{F}'_1$ is also a solution of $\mathcal{F}'_2$, i.e., $\mathcal{F}'_1 \subset \mathcal{F}'_2$. Since $\mathcal{P}^{C\prime}_{MC}$ is a maximization problem over $\mathcal{F}'$, we have $F^*_{MCKP}(P_1) \leq F^*_{MCKP}(P_2)$. This proves that F^*_{MCKP} is non-decreasing.

Now, let us tackle its sublinearity. We first prove that the $f^n_{j,i}$ are sublinears. If $j = 1$ or $w_{\pi_n(i)} \geq w_{\pi_n(j-1)}$, then $f^n_{j,i}$ is concave according to Lemma 8. Therefore, it is also sublinear. Otherwise, $f^n_{j,i}$ is concave before c_2 and decreasing after $c_1 \leq c_2$. In this case, $f^n_{j,i}$ is thus also sublinear. Secondly, for any subcarrier n and user selection $\mathcal{U}'_n$, $\mathcal{P}^{C\prime}_{SCPC}(n)$ consists in maximizing a sum of separable sublinear functions $f^n_{j,i}$ subject to a budget constraint $\bar{P}^n$. Hence, $F^n\big(\mathcal{U}'_n, \bar{P}^n\big)$ is sublinear in $\bar{P}^n$. Thirdly, the optimal of $\mathcal{P}^{C\prime}_{SC}(n)$ can be seen as the best allocation over all possible user selections, i.e., $F^n(\bar{P}^n) = \max_{\mathcal{U}'_n}\{F^n\big(\mathcal{U}'_n, \bar{P}^n\big)\}$. The max operator preserves sublinearity. Therefore, $F^n(\bar{P}^n)$ is sublinear in $\bar{P}^n$. Finally, F^*_{MCKP} is sublinear in P_{max}, since $\mathcal{P}^{C\prime}_{MC}$ is a separable sum maximization of F^n subject to budget constraint P_{max}. □

Let us introduce a variant of MCKP, denoted by MCKP'. The differences are as follows. Its cellular power budget is $2P_{max}$. The item's weights can only take value of the form $a_{n,l} = l\lfloor J/N \rfloor \delta$ for $n \in \mathcal{N}$, $l \in \{0, \ldots, 2N\}$. The profits values are defined similarly as $c_{n,l} = F^n(a_{n,l})$. Consequently, MCKP' only contains $2N + 1$ items per class. The idea of the proof is to show that a greedy solution of MCKP' is a constant factor approximation of MCKP optimal value. The value of U is then easily obtained using the greedy Dyer-Zemel algorithm [71, Section 11.2]. In this case, the complexity is independent of J and negligible compared to the rest of the algorithm. One could also get an estimation by applying the Dyer-Zemel algorithm

directly to MCKP. However, the complexity would be proportional to $O(J)$ which is against the idea of polynomial-time approximation.

Let $y^*_{n,l}$, for $n \in \mathcal{N}$, $l \in \{0, \dots, 2N\}$, be an optimal solution of this problem. In addition, we denote by $y'_{n,l}$ for $n \in \mathcal{N}$, $l \in \{0, \dots, 2N\}$, a 1/2-approximation given by the Dyer-Zemel algorithm. On the one hand, we have:

$$\sum_{n\in\mathcal{N}} \sum_{l=1}^{2N} c_{n,l} y'_{n,l} \geq \frac{1}{2} \sum_{n\in\mathcal{N}} \sum_{l=1}^{2N} c_{n,l} y^*_{n,l}, \tag{B.11}$$

$$\geq \frac{1}{2} \sum_{n\in\mathcal{N}} F^n \left(\left\lceil \frac{\bar{P}^{n*}}{\lfloor J/N \rfloor \delta} \right\rceil \left\lfloor \frac{J}{N} \right\rfloor \delta \right), \tag{B.12}$$

$$\geq \frac{1}{2} \sum_{n\in\mathcal{N}} F^n\left(\bar{P}^{n*}\right) = \frac{1}{2} F^*_{MCKP}(P_{max}), \tag{B.13}$$

where $\bar{P}^{n*}$ is the power allocated to subcarrier n in $F^*_{MCKP}(P_{max})$. The 1/2-approximation of $y'_{n,l}$ translates into Eqn. (B.11). The right term of Eqn. (B.12) corresponds to a valid allocation of MCKP', with item $l = \lceil \bar{P}^{n*}/(\lfloor J/N \rfloor \delta) \rceil$ allocated in class n. Indeed, by definition of the ceiling and floor functions, we have:

$$\bar{P}^{n*} \overset{\text{(e)}}{\leq} \left\lceil \frac{\bar{P}^{n*}}{\lfloor J/N \rfloor \delta} \right\rceil \left\lfloor \frac{J}{N} \right\rfloor \delta < \bar{P}^{n*} + \left\lfloor \frac{J}{N} \right\rfloor \delta \leq \bar{P}^{n*} + \frac{P_{max}}{N}.$$

Therefore,

$$\sum_{n\in\mathcal{N}} \left\lceil \frac{\bar{P}^{n*}}{\lfloor J/N \rfloor \delta} \right\rceil \left\lfloor \frac{J}{N} \right\rfloor \delta < \sum_{n\in\mathcal{N}} \left(\bar{P}^{n*} + \frac{P_{max}}{N} \right) = 2P_{max}.$$

In other words, the power budget is also satisfied. As it is a valid allocation for MCKP', it must have a total profit not greater than the optimal profit:

$$\sum_{n\in\mathcal{N}} \sum_{l=1}^{2N} c_{n,l} y^*_{n,l},$$

which proves inequality (B.12). We derive Eqn. (B.13) from inequality (e) and the monotonicity of F^n (see Lemma 30).

We have, on the other hand:

$$\sum_{n\in\mathcal{N}} \sum_{l=1}^{2N} c_{n,l} y'_{n,l} \leq \sum_{n\in\mathcal{N}} \sum_{l=1}^{2N} c_{n,l} y^*_{n,l}, \tag{B.14}$$

$$\leq F^*_{MCKP}(2P_{max}), \tag{B.15}$$

$$\leq 2F^*_{MCKP}(P_{max}). \tag{B.16}$$

The optimality of $y^*_{n,l}$ implies Eqn. (B.14). Eqn. (B.15) comes from the fact that the items of MCKP' is a subset of MCKP items, given a budget $2P_{max}$. Eqn. (B.16) follows from the sublinearity of F^*_{MCKP} (see Lemma 30).

Let $U \triangleq 2\sum_{n\in\mathcal{N}}\sum_{l=1}^{2N} c_{n,l}y'_{n,l}$. We derive from inequalities (B.13) and (B.16) the desired approximation bound:

$$U \geq F^*_{MCKP}(P_{max}) \geq U/4.$$

B.11 Proof of Theorem 17

Proof. The complexity and optimality proofs of ε-JSPA are presented below.

Complexity analysis: We divide the complexity analysis of Algorithm 8 in four parts as follows. The overall complexity can be obtained by summing the complexity of each part.

i. Precomputation: The precomputation required for setting up i-SCUS on each subcarrier has complexity $O(NMK^2)$.

ii. Line 1: The estimation procedure presented in Appendix B.10, consists in $O(N^2)$ function evaluations and $O(N^2)$ iterations of the Dyer-Zemel algorithm. Each function evaluation is computed by i-SCUS, therefore the complexity of this part is $O(N^2MK)$.

iii. Lines 2-4: Each L_n, for $n \in \mathcal{N}$, is obtained by multi-key binary search [73]. For each L_n, we need to find $4N/\varepsilon$ keys in an array $\{c_{n,1}, \ldots, c_{n,J}\}$ of length J. Since repetition is not allowed, the binary search returns at most $\min\{4N/\varepsilon, J\}$ items. More precisely, it computes each of the $4N/\varepsilon$ keys in time $\log(J)$, with at most J function evaluations in total. Therefore, the binary search performs $O(\min\{\log(J)\, N/\varepsilon, J\})$ function evaluations. Multiplied by the complexity of each function evaluation on each subcarrier, we get $O(\min\{\log(J)\, N^2MK/\varepsilon, JNMK\})$.

iv. Lines 5-6: Let us first briefly explain the dynamic programming by profits [71]. Let Y be the DP array such that $Y[n,q]$ denotes the minimal weight, i.e., minimal power budget, required to achieve WSR $q\cdot\varepsilon U/4N$ when problem MCKP is restricted to the first n classes. It is initialized as $Y[0,0] = 0$ and $Y[0,q] = +\infty$, for $q = 1, \ldots, \lfloor 4N/\varepsilon \rfloor$. For $n \in \mathcal{N}$ and $q = 0, \ldots, \lfloor 4N/\varepsilon \rfloor$, the recurrence relation is:

$$Y[n,q] = \min_{l\in L_n} \begin{cases} Y\left[n-1, q - \left\lfloor \frac{4c_{n,l}N}{\varepsilon U} \right\rfloor\right] + a_{n,l}, & \text{if } \frac{q\cdot\varepsilon U}{4N} \geq c_{n,l}, \\ +\infty, \text{ otherwise.} \end{cases} \tag{B.17}$$

This recursion has complexity $O(\min\{N^3/\varepsilon^2, J^2N\})$, which is the number of all considered items $\sum_{n\in\mathcal{N}} |L_n| = \min\{4N^2/\varepsilon, JN\}$ multiplied by the number of comparisons in Eqn. (B.17), $|L_n| = \min\{4N/\varepsilon, J\}$.

Approximation analysis: As proved in [71, Section 11.9], the optimal solution obtained by dynamic programming by profits considering only items in L_n, differs from F^*_{MCKP} by at most a factor $1-\varepsilon$.

In summary, ε-JSPA achieves ε-approximation with polynomial complexity in $1/\varepsilon$ and N, M, K. Therefore, ε-JSPA is a FPTAS, which concludes the proof. □

Appendix C

Proofs of Chapter 5

C.1 Proof of Theorem 18

Proof. The basic idea is to check that $\mathcal{P}$ is a non-empty simplex, and that the objective function is concave. The feasible power region is non-empty as the zero vector satisfies constraints $C1$ to $C3$. We deduce from (5.1) and (5.2) that $\mathcal{P}$ is a standard simplex, thus it is also convex.

It remains to show that $\sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{N}} R_k^n(\boldsymbol{p})$ is a concave function of $\boldsymbol{p}$:

$$
\begin{aligned}
\sum_{k\in\mathcal{K}}\sum_{n\in\mathcal{N}} R_k^n(\boldsymbol{p}) &= \sum_{n=1}^{N} W_n \sum_{k=1}^{K} \log_2\left(1+\frac{p_k^n}{\sum_{j=\pi_n^{-1}(k)+1}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_k^n}\right), \\
&= \sum_{n=1}^{N} W_n \sum_{i=1}^{|\mathcal{U}_n|} \log_2\left(1+\frac{p_{\pi_n(i)}^n}{\sum_{j=i+1}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(i)}^n}\right), \\
&= \sum_{n=1}^{N} W_n \sum_{i=1}^{|\mathcal{U}_n|} \log_2\left(\frac{\sum_{j=i}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(i)}^n}{\sum_{j=i+1}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(i)}^n}\right), \\
&= \sum_{n=1}^{N} W_n \log_2\left(\frac{\sum_{j=1}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(1)}^n}{\sum_{j=2}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(1)}^n}\right. \\
&\qquad \left.\times \frac{\sum_{j=2}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(2)}^n}{\sum_{j=3}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(2)}^n} \times \cdots \times \frac{p_{\pi_n(|\mathcal{U}_n|)}^n + \tilde{\eta}_{\pi_n(|\mathcal{U}_n|)}^n}{\tilde{\eta}_{\pi_n(|\mathcal{U}_n|)}^n}\right), \\
&= \sum_{n=1}^{N} W_n \log_2\left(\frac{\sum_{j=1}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(1)}^n}{\tilde{\eta}_{\pi_n(|\mathcal{U}_n|)}^n}\right. \\
&\qquad \left.\times \frac{\sum_{j=2}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(2)}^n}{\sum_{j=2}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(1)}^n} \times \cdots \times \frac{p_{\pi_n(|\mathcal{U}_n|)}^n + \tilde{\eta}_{\pi_n(|\mathcal{U}_n|)}^n}{p_{\pi_n(|\mathcal{U}_n|)}^n + \tilde{\eta}_{\pi_n(|\mathcal{U}_n|-1)}^n}\right),
\end{aligned}
$$

$$= \sum_{n=1}^{N} W_n \sum_{i=1}^{|\mathcal{U}_n|-1} \log_2 \left(\frac{\sum_{j=i+1}^{|\mathcal{U}_n|} p^n_{\pi_n(j)} + \tilde{\eta}^n_{\pi_n(i+1)}}{\sum_{j=i+1}^{|\mathcal{U}_n|} p^n_{\pi_n(j)} + \tilde{\eta}^n_{\pi_n(i)}} \right),$$

$$+ \sum_{n=1}^{N} W_n \log_2 \left(\frac{\sum_{j=1}^{|\mathcal{U}_n|} p^n_{\pi_n(j)} + \tilde{\eta}^n_{\pi_n(1)}}{\tilde{\eta}^n_{\pi_n(|\mathcal{U}_n|)}} \right),$$

$$= \sum_{n=1}^{N} W_n \left(\sum_{i=1}^{|\mathcal{U}_n|-1} \log_2(\alpha^n_i(\boldsymbol{p})) + \log_2(\beta^n(\boldsymbol{p})) \right), \tag{C.1}$$

where $\alpha^n_i(\boldsymbol{p}) \triangleq \frac{\sum_{j=i+1}^{|\mathcal{U}_n|} p^n_{\pi_n(j)} + \tilde{\eta}^n_{\pi_n(i+1)}}{\sum_{j=i+1}^{|\mathcal{U}_n|} p^n_{\pi_n(j)} + \tilde{\eta}^n_{\pi_n(i)}}$ and $\beta^n(\boldsymbol{p}) \triangleq \frac{\sum_{j=1}^{|\mathcal{U}_n|} p^n_{\pi_n(j)} + \tilde{\eta}^n_{\pi_n(1)}}{\tilde{\eta}^n_{\pi_n(|\mathcal{U}_n|)}}$ in Eqn. (C.1). Furthermore, α^n_i can be written as:

$$\alpha^n_i = 1 + \frac{\tilde{\eta}^n_{\pi_n(i+1)} - \tilde{\eta}^n_{\pi_n(i)}}{\sum_{j=i+1}^{|\mathcal{U}_n|} p^n_{\pi_n(j)} + \tilde{\eta}^n_{\pi_n(i)}}.$$

It follows from $\tilde{\eta}^n_{\pi_n(i+1)} - \tilde{\eta}^n_{\pi_n(i)} \leq 0$ in Eqn. (2.13) that α^n_i is a concave function of $\boldsymbol{p}$ and β^n is linear to $\boldsymbol{p}$. Hence, both α^n_i and β^n are concave functions of $\boldsymbol{p}$. Since the logarithm is concave and non-decreasing, $\log_2(\alpha^n_i)$ and $\log_2(\beta^n)$ are concave by function composition. Concavity is also preserved by summation, we deduce that $\sum_{k\in\mathcal{K}} \sum_{n\in\mathcal{N}} R^n_k(\boldsymbol{p})$ is concave. This completes the proof. □

C.2 Proof of Theorem 21

We first introduce the auxiliary Lemma 31, which will be used in the proof of Theorem 21.

Lemma 31.

Let $(b_i)_{i\in\{1\cdots Q\}} \in \mathbb{R}^Q_+$ be a non-increasing sequence with Q positive real values, i.e., $b_1 \geq b_2 \geq \cdots \geq b_{Q-1} \geq b_Q > 0$. Then, for all $(x_i)_{i\in\{1\cdots Q\}} \in \mathbb{R}^Q$,

$$\sum_{i=1}^{Q} \sum_{j=i}^{Q} \frac{x_i x_j}{b_i} \geq \frac{\sum_{i=1}^{Q} x_i^2}{2b_i} \geq \frac{\sum_{i=1}^{Q} x_i^2}{2b_1}. \tag{C.2}$$

Proof. The second inequality in (C.2) follows directly from the fact that $b_1 \geq b_2 \geq \cdots \geq b_{Q-1} \geq b_Q$. We will prove the first inequality by mathematical induction.

Basis: For $Q = 1$, we have directly:

$$\frac{x_1 x_1}{b_1} \geq \frac{x_1^2}{2b_1}.$$

Inductive step: Assume that Eqn. (C.2) is true for $Q = q$, i.e.,

$$\sum_{i=1}^{q} \sum_{j=i}^{q} \frac{x_i x_j}{b_i} \geq \frac{\sum_{i=1}^{q} x_i^2}{2b_i}. \tag{C.3}$$

Consider the case where $Q = q + 1$. According to Eqn. (C.2), we have:

$$\begin{aligned}
&\sum_{i=1}^{q+1}\sum_{j=i}^{q+1}\frac{x_i x_j}{b_i} - \sum_{i=1}^{q+1}\frac{x_i^2}{2b_i} \\
&\quad= \frac{x_{q+1}^2}{b_{q+1}} + \frac{x_q^2}{b_q} + \frac{x_q x_{q+1}}{b_q} + \sum_{i=1}^{q-1}\sum_{j=i}^{q+1}\frac{x_i x_j}{b_i} - (\frac{x_{q+1}^2}{2b_{q+1}} + \frac{x_q^2}{2b_q} + \sum_{i=1}^{q-1}\frac{x_i^2}{2b_i}), \\
&\quad\geq \frac{x_{q+1}^2}{2b_q} + \frac{x_q^2}{2b_q} + \frac{x_q x_{q+1}}{b_q} + \sum_{i=1}^{q-1}\sum_{j=i}^{q+1}\frac{x_i x_j}{b_i} - \sum_{i=1}^{q-1}\frac{x_i^2}{2b_i}, \\
&\quad= \frac{(x_{q+1}+x_q)^2}{2b_q} + \sum_{i=1}^{q-1}\sum_{j=i}^{q+1}\frac{x_i x_j}{b_i} - \sum_{i=1}^{q-1}\frac{x_i^2}{2b_i}.
\end{aligned} \tag{C.4}$$

Note that the inequality in (C.4) holds because $b_{q+1} \leq b_q$. We then define an auxiliary variable x_i' as follows:

$$x_i' \triangleq \begin{cases} x_i & \text{if } \ i < q, \\ x_{q+1} + x_q & \text{if } \ i = q. \end{cases} \tag{C.5}$$

We deduce from Eqn. (C.4) and (C.5) the following inequality:

$$\begin{aligned}
\sum_{i=1}^{q+1}\sum_{j=i}^{q+1}\frac{x_i x_j}{b_i} - \sum_{i=1}^{q+1}\frac{x_i^2}{2b_i} &= \frac{(x_{q+1}+x_q)^2}{2b_q} + \sum_{i=1}^{q-1}\sum_{j=i}^{q+1}\frac{x_i x_j}{b_i} - \sum_{i=1}^{q-1}\frac{x_i^2}{2b_i}, \\
&= \frac{x_q'^2}{2b_q} + \left(\sum_{i=1}^{q}\sum_{j=i}^{q}\frac{x_i' x_j'}{b_i} - \frac{x_q'^2}{b_q}\right) - \sum_{i=1}^{q-1}\frac{x_i'^2}{2b_i}, \\
&= -\frac{x_q'^2}{2b_q} + \sum_{i=1}^{q}\sum_{j=i}^{q}\frac{x_i' x_j'}{b_i} - \sum_{i=1}^{q-1}\frac{x_i'^2}{2b_i}, \\
&\geq -\frac{x_q'^2}{2b_q} + \frac{x_q'^2}{2b_q}, \qquad \text{by induction hypothesis} \\
&= 0.
\end{aligned}$$

Therefore, the statement (C.2) is true for $Q = q + 1$, which completes the proof by induction. □

The proof of Theorem 21 is given below.

Proof. For all $\boldsymbol{p}, \boldsymbol{p}' \in \mathcal{P}$, according to (5.3) and (5.4), we have:

$$(\Phi(\boldsymbol{p}) - \Phi(\boldsymbol{p}'))^T \cdot (\boldsymbol{p} - \boldsymbol{p}') = \frac{W_n}{\ln(2)}\sum_{k=1}^{K}\sum_{n=1}^{N}(\frac{1}{\hat{I}_k^n} - \frac{1}{\hat{I}_k'^n})(p_k^n - p_k'^n), \tag{C.6}$$

where

$$\hat{I}_k^n = \sum_{j=\pi_n^{-1}(k)}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_k^n \quad \text{and} \quad \hat{I}_k^{'n} = \sum_{j=\pi_n^{-1}(k)}^{|\mathcal{U}_n|} p_{\pi_n(j)}^{'n} + \tilde{\eta}_k^n.$$

We re-arrange Eqn. (C.6) as follows:

$$\begin{aligned}(\Phi(\boldsymbol{p}) - \Phi(\boldsymbol{p}'))^T \cdot (\boldsymbol{p} - \boldsymbol{p}') &= \frac{W_n}{\ln(2)} \sum_{k=1}^{K} \sum_{n=1}^{N} \frac{(p_k^n - p_k^{'n}) \left(\sum_{j=\pi_n^{-1}(k)}^{|\mathcal{U}_n|} (p_{\pi_n(j)}^{'n} - p_{\pi_n(j)}^n) \right)}{\hat{I}_k^n \hat{I}_k^{'n}}, \\ &= \frac{W_n}{\ln(2)} \sum_{n=1}^{N} \sum_{i=1}^{|\mathcal{U}_n|} \frac{(p_{\pi_n(i)}^n - p_{\pi_n(i)}^{'n}) \left(\sum_{j=i}^{|\mathcal{U}_n|} (p_{\pi_n(j)}^{'n} - p_{\pi_n(j)}^n) \right)}{\hat{I}_{\pi_n(i)}^n \hat{I}_{\pi_n(i)}^{'n}}, \\ &= -\frac{W_n}{\ln(2)} \sum_{n=1}^{N} \sum_{i=1}^{|\mathcal{U}_n|} \sum_{j=i}^{|\mathcal{U}_n|} \frac{x_i^n x_j^n}{b_i^n}, \end{aligned} \tag{C.7}$$

where, for $n \in \mathcal{N}$ and $i \in \{1, 2, \ldots, |\mathcal{U}_n|\}$, $x_i^n = p_{\pi_n(i)}^n - p_{\pi_n(i)}^{'n}$ and

$$b_i^n = \hat{I}_{\pi_n(i)}^n \hat{I}_{\pi_n(i)}^{'n} = \left(\sum_{j=i}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(i)}^n \right) \left(\sum_{j=i}^{|\mathcal{U}_n|} p_{\pi_n(j)}^{'n} + \tilde{\eta}_{\pi_n(i)}^n \right).$$

We define $\frac{C_n}{2} \triangleq \left(\sum_{j=1}^{|\mathcal{U}_n|} \min\{P_{max}^{\pi_n(j)}, P_{max}^{n,\pi_n(j)}\} + \tilde{\eta}_{\pi_n(1)}^n \right)^2$, which gives us the following upper bound for b_1^n:

$$b_1^n = \left(\sum_{j=1}^{|\mathcal{U}_n|} p_{\pi_n(j)}^n + \tilde{\eta}_{\pi_n(1)}^n \right) \left(\sum_{j=1}^{|\mathcal{U}_n|} p_{\pi_n(j)}^{'n} + \tilde{\eta}_{\pi_n(1)}^n \right) \leq \frac{C_n}{2}.$$

Based on Lemma 31 and the previous upper bound, we have:

$$-\sum_{n=1}^{N} \frac{W_n}{\ln(2)} \sum_{i=1}^{|\mathcal{U}_n|} \sum_{j=i}^{|\mathcal{U}_n|} \frac{x_i^n x_j^n}{b_i^n} \leq -\sum_{n=1}^{N} \frac{W_n}{\ln(2) \cdot 2b_1^n} \sum_{i=1}^{|\mathcal{U}_n|} (x_i^n)^2 \leq -\sum_{n=1}^{N} \frac{W_n}{\ln(2) \cdot C_n} \sum_{i=1}^{|\mathcal{U}_n|} (x_i^n)^2.$$

We then obtain from Eqn. (C.7):

$$(\Phi(\boldsymbol{p}) - \Phi(\boldsymbol{p}'))^T \cdot (\boldsymbol{p} - \boldsymbol{p}') \leq c||\boldsymbol{p} - \boldsymbol{p}'||_2^2,$$

where c is a constant such that $-\sum_{n=1}^{N} \frac{W_n}{\ln(2) \cdot C_n} \leq c < 0$. This completes the proof. □

C.3 Proof of Theorem 24

Let $\omega_k(t)$ be the water level of user k at time t. From Eqn (5.6) and (5.7), we have:

$$\sum_{n \in \mathcal{N}_k} \left[\omega_k - \tilde{I}_k^n \right]_0^{P_{max}^{n,k}} = P_{max}^k. \tag{C.8}$$

Note that $\omega_k(t+1)$ can be seen as a function of $\tilde{\boldsymbol{I}}_k(t)$, and we denote it by:

$$\omega_k(t+1) \triangleq h_k(\tilde{\boldsymbol{I}}_k(t)). \tag{C.9}$$

We consider two waterfilling scenarios for user k. The normalized interference on subcarrier n in the two scenarios are denoted by $\tilde{I}_k^n$ and $\tilde{I}_k^{n\prime}$, respectively. After performing waterfilling, we denote the water levels in the two scenarios by ω_k and ω_k', respectively. With this setting, we have the following lemma:

Lemma 32. *For any $k \in \mathcal{K}$, if $\tilde{I}_k^n \geq \tilde{I}_k^{n\prime}$ for all $n \in \mathcal{N}_k$, then $h_k(\tilde{\boldsymbol{I}}_k) \geq h_k(\tilde{\boldsymbol{I}}_k')$.*

Proof. Let $\tilde{I}_k^n \geq \tilde{I}_k^{n\prime}$ for all $n \in \mathcal{N}_k$, and $\omega_k \triangleq h_k(\tilde{\boldsymbol{I}}_k)$, $\omega_k' \triangleq h_k(\tilde{\boldsymbol{I}}_k')$. Let us assume by contradiction that $\omega_k < \omega_k'$. Since w_k' is the lowest non-negative value satisfying Eqn. (5.6) and (5.7), and the power budgets are positive (i.e., $P_{max}^k > 0$ and $P_{max}^{n,k} > 0$, for $n \in \mathcal{N}$), we know that there exists at least one subcarrier $m \in \mathcal{N}$ such that $0 < \omega_k' - \tilde{I}_k^{m\prime} \leq P_{max}^{m,k}$. It follows that:

$$\left[\omega_k' - \tilde{I}_k^{m\prime}\right]_0^{P_{max}^{m,k}} = \omega_k' - \tilde{I}_k^{m\prime} > \left[\omega_k - \tilde{I}_k^{m\prime}\right]_0^{P_{max}^{m,k}} \geq \left[\omega_k - \tilde{I}_k^{m}\right]_0^{P_{max}^{m,k}}, \tag{C.10}$$

where the first strict inequality comes from $\omega_k - \tilde{I}_k^{m\prime} < \omega_k' - \tilde{I}_k^{m\prime} \leq P_{max}^{m,k}$, and the second inequality is due to $\tilde{I}_k^n \geq \tilde{I}_k^{n\prime}$. For any other subcarrier $n \in \mathcal{N}$, $n \neq m$, we have the following non-strict inequalities:

$$\left[\omega_k' - \tilde{I}_k^{n\prime}\right]_0^{P_{max}^{n,k}} \geq \left[\omega_k - \tilde{I}_k^{n\prime}\right]_0^{P_{max}^{n,k}} \geq \left[\omega_k - \tilde{I}_k^{n}\right]_0^{P_{max}^{n,k}}. \tag{C.11}$$

By summing inequalities (C.10) and (C.11) over all subcarriers, we obtain:

$$\sum_{n\in\mathcal{N}_k}\left[\omega_k' - \tilde{I}_k^{n\prime}\right]_0^{P_{max}^{n,k}} > \sum_{n\in\mathcal{N}_k}\left[\omega_k - \tilde{I}_k^{n\prime}\right]_0^{P_{max}^{n,k}} \geq \sum_{n\in\mathcal{N}_k}\left[\omega_k - \tilde{I}_k^{n}\right]_0^{P_{max}^{n,k}}.$$

According to (C.8), the inequalities in (C.3) are equivalent to:

$$P_{max}^k > \sum_{n\in\mathcal{N}_k}\left[\omega_k - \tilde{I}_k^{n\prime}\right]_0^{P_{max}^{n,k}} \geq P_{max}^k,$$

which is a contradiction, therefore $\omega_k \geq \omega_k'$. □

The proof of Theorem 24 is given below.

Proof. Since $|\mathcal{U}_n| \leq 2$, there are at most two active users in subcarrier n. We denote the subset of subcarriers in which user $k \in \mathcal{K}$ is decoded first by $\mathcal{T}_k$, and the subset of subcarriers in which user k is decoded secondly by $\mathcal{S}_k$, i.e., $\mathcal{S}_k = \mathcal{N}_k \setminus \mathcal{T}_k$. For

$n \in \mathcal{T}_k$, we define $-k_n$ as the index of the user that shares subcarrier n with user k (i.e., the secondly decoded user).

Since each user has limited power due to constraints $C1$ and $C2$, $\tilde{I}_k^n(t)$ is bounded from above for all $k \in \mathcal{K}$ and $n \in \mathcal{N}$. Therefore, according to (C.8), $\omega_k(t)$ is also bounded from above for all k. The convergence of SIWA is established by proving by induction that $\boldsymbol{\omega}(t) \triangleq (\omega_1(t), \omega_2(t), \ldots, \omega_K(t))$ is monotonically increasing, i.e., for any $t \geq 1$,

$$\boldsymbol{\omega}(t+1) \succeq \boldsymbol{\omega}(t), \tag{C.12}$$

where $\succeq$ indicates that $\omega_k(t+1) \geq \omega_k(t)$ for all k.

Basis: Since $\boldsymbol{p}_k^{(0)} = 0$ for all k, we have $\tilde{I}_k^n(0) = \tilde{\eta}_k^n$ for all k and n. Thus, $\tilde{I}_k^n(1) \geq \tilde{I}_k^n(0)$. Lemma 32 and (C.9) imply $\boldsymbol{\omega}(2) \succeq \boldsymbol{\omega}(1)$.

Inductive step: Suppose (C.12) holds for $t = L$, i.e.,

$$\boldsymbol{\omega}(L+1) \succeq \boldsymbol{\omega}(L). \tag{C.13}$$

First, consider $n \in \mathcal{S}_k$. By the definition of $\mathcal{S}_k$, $\tilde{I}_k^n(t) = \tilde{\eta}_k^n$ for all t, which implies:

$$\tilde{I}_k^n(L+1) = \tilde{I}_k^n(L), \quad \text{for } n \in \mathcal{S}_k. \tag{C.14}$$

Next, consider $n \in \mathcal{T}_k$. According to (5.5), for any t, we have:

$$\tilde{I}_k^n(t) = p_{-k_n}^n(t) + \tilde{\eta}_k^n, \quad \text{for } n \in \mathcal{T}_k. \tag{C.15}$$

By the definition of $\mathcal{T}_k$, user $-k_n$ experiences no intra-band interference in subcarrier n. The waterfilling method gives:

$$p_{-k_n}^n(t) = [\omega_{-k_n}(t) - \tilde{\eta}_{-k_n}^n]_0^{P_{max}^{n,-k_n}}.$$

Substituting it back to (C.15), we obtain:

$$\tilde{I}_k^n(t) = [\omega_{-k_n}(t) - \tilde{\eta}_{-k_n}^n]_0^{P_{max}^{n,-k_n}} + \tilde{\eta}_k^n, \quad \text{for } n \in \mathcal{T}_k,$$

which, for $t = L+1$ and together with the inductive hypothesis in (C.13), implies:

$$\tilde{I}_k^n(L+1) \geq \tilde{I}_k^n(L), \quad \text{for } n \in \mathcal{T}_k. \tag{C.16}$$

Invoking Lemma 32 with (C.14) and (C.16) and using (C.9), we obtain $\boldsymbol{\omega}(L+2) \succeq \boldsymbol{\omega}(L+1)$, which completes the proof. □

Bibliography

[1] M Series, "IMT vision–framework and overall objectives of the future development of IMT for 2020 and beyond," *Recommendation ITU–R M.2083–0*, 2015.

[2] H. Chen, R. Abbas, P. Cheng, M. Shirvanimoghaddam, W. Hardjawana, W. Bao, Y. Li, and B. Vucetic, "Ultra-reliable low latency cellular networks: Use cases, challenges and approaches," *IEEE Commun. Mag.*, vol. 56, no. 12, pp. 119–125, 2018.

[3] E. Dahlman, S. Parkvall, and J. Skold, *5G NR: The Next Generation Wireless Access Technology*. Academic Press, 2018.

[4] 3GPP TR 21.916 version 0.1.0 Release 16, 2018. [Online]. Available: https://www.3gpp.org/release-16.

[5] "White paper: 5G new radio network, use cases, spectrum, technologies and architecture," Nokia, Tech. Rep., 2019.

[6] M Series, "IMT traffic estimates for the years 2020 to 2030," *Report ITU–R M.2370–0*, 2015.

[7] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge university press, 2005.

[8] L. Lei, D. Yuan, and P. Värbrand, "On power minimization for non-orthogonal multiple access (NOMA)," *IEEE Commun. Lett.*, vol. 20, no. 12, pp. 2458–2461, 2016.

[9] X. Li, C. Li, and Y. Jin, "Dynamic resource allocation for transmit power minimization in OFDM-based NOMA systems," *IEEE Commun. Lett.*, vol. 20, no. 12, pp. 2558–2561, 2016.

[10] Z. Wei, D. W. K. Ng, J. Yuan, and H.-M. Wang, "Optimal resource allocation for power-efficient MC-NOMA with imperfect channel state information," *IEEE Trans. Commun.*, vol. 65, no. 9, pp. 3944–3961, 2017.

[11] Y. Zhang, H.-M. Wang, T.-X. Zheng, and Q. Yang, "Energy-efficient transmission design in non-orthogonal multiple access," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2852–2857, 2016.

[12] F. Fang, H. Zhang, J. Cheng, and V. C. Leung, "Energy efficiency of resource scheduling for non-orthogonal multiple access (NOMA) wireless network," in *IEEE Int. Conf. Commun. (ICC)*, 2016.

[13] F. Fang, H. Zhang, J. Cheng, S. Roy, and V. C. Leung, "Joint user scheduling and power allocation optimization for energy-efficient NOMA systems with imperfect CSI," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 12, pp. 2874–2885, 2017.

[14] D. Zhai, R. Zhang, L. Cai, B. Li, and Y. Jiang, "Energy-efficient user scheduling and power allocation for NOMA-based wireless networks with massive IoT devices," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1857–1868, 2018.

[15] M. Zeng, A. Yadav, O. A. Dobre, and H. V. Poor, "Energy-efficient power allocation for uplink NOMA," in *IEEE Global Commun. Conf. (GLOBECOM)*, 2018.

[16] M. Zeng, N.-P. Nguyen, O. A. Dobre, Z. Ding, and H. V. Poor, "Spectral-and energy-efficient resource allocation for multi-carrier uplink NOMA systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 9293–9296, 2019.

[17] M. S. Ali, H. Tabassum, and E. Hossain, "Dynamic user clustering and power allocation for uplink and downlink non-orthogonal multiple access (NOMA) systems," *IEEE Access*, vol. 4, pp. 6325–6343, 2016.

[18] G. Nain, S. S. Das, and A. Chatterjee, "Low complexity user selection with optimal power allocation in downlink NOMA," *IEEE Wireless Commun. Lett.*, vol. 7, no. 2, pp. 158–161, 2017.

[19] Z. Yang, W. Xu, C. Pan, Y. Pan, and M. Chen, "On the optimality of power allocation for NOMA downlinks with individual QoS constraints," *IEEE Commun. Lett.*, vol. 21, no. 7, pp. 1649–1652, 2017.

[20] L. Lei, D. Yuan, C. K. Ho, and S. Sun, "Power and channel allocation for non-orthogonal multiple access in 5G systems: Tractability and computation," *IEEE Trans. Wireless Commun.*, vol. 15, no. 12, pp. 8580–8594, 2016.

[21] Y. Sun, D. W. K. Ng, Z. Ding, and R. Schober, "Optimal joint power and subcarrier allocation for MC-NOMA systems," in *IEEE Global Commun. Conf.*, 2016.

[22] ——, "Optimal joint power and subcarrier allocation for full-duplex multicarrier non-orthogonal multiple access systems," *IEEE Trans. Commun.*, vol. 65, no. 3, pp. 1077–1091, 2017.

[23] N. Otao, Y. Kishiyama, and K. Higuchi, "Performance of non-orthogonal access with SIC in cellular downlink using proportional fair-based resource allocation," in *IEEE Int. Symp. on Wireless Commun. Syst. (ISWCS)*, 2012, pp. 476–480.

[24] Y. Saito, Y. Kishiyama, A. Benjebbour, T. Nakamura, A. Li, and K. Higuchi, "Non-orthogonal multiple access (NOMA) for cellular future radio access," in *IEEE 77th Veh. Technol. Conf.*, 2013.

[25] Y. Saito, A. Benjebbour, Y. Kishiyama, and T. Nakamura, "System-level performance evaluation of downlink non-orthogonal multiple access (NOMA)," in *IEEE 24th Int. Symp. on Personal Indoor and Mobile Radio Commun. (PIMRC)*, 2013, pp. 611–615.

[26] Z. Ding, P. Fan, and H. V. Poor, "Impact of user pairing on 5G non orthogonal multiple-access downlink transmissions," *IEEE Trans. Veh. Technol.*, vol. 65, no. 8, pp. 6010–6023, 2016.

[27] M.-R. Hojeij, J. Farah, C. A. Nour, and C. Douillard, "Resource allocation in downlink non-orthogonal multiple access (NOMA) for future radio access," in *IEEE 81st Veh. Technol. Conf. (VTC Spring)*, 2015.

[28] B. Di, S. Bayat, L. Song, and Y. Li, "Radio resource allocation for downlink non-orthogonal multiple access (NOMA) networks using matching theory," in *IEEE Global Commun. Conf.(GLOBECOM)*, 2015.

[29] M. Al-Imari, P. Xiao, and M. A. Imran, "Receiver and resource allocation optimization for uplink NOMA in 5G wireless networks," in *IEEE 11th Int. Symp. on Wireless Commun. Syst. (ISWCS)*, 2015, pp. 151–155.

[30] B. Di, L. Song, and Y. Li, "Sub-channel assignment, power allocation, and user scheduling for non-orthogonal multiple access networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7686–7698, 2016.

[31] J. Zhu, J. Wang, Y. Huang, S. He, X. You, and L. Yang, "On optimal power allocation for downlink non-orthogonal multiple access systems," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 12, pp. 2744–2757, 2017.

[32] J. Zhu, J. Wang, Y. Huang, S. He, and X. You, "Multichannel resource allocation for downlink non-orthogonal multiple access systems," in *IEEE Global Commun. Conf. (GLOBECOM)*, 2017.

[33] S. Chen, K. Peng, and H. Jin, "A suboptimal scheme for uplink NOMA in 5G systems," in *Int. Wireless Commun. and Mobile Computing Conf. (IWCMC)*, 2015, pp. 1429–1434.

[34] M. Al-Imari, P. Xiao, M. A. Imran, and R. Tafazolli, "Uplink non-orthogonal multiple access for 5G wireless networks," in *IEEE 11th Int. Symp. on Wireless Commun. Syst. (ISWCS)*, 2014, pp. 781–785.

[35] Y. Cheng, K. H. Li, K. C. Teh, and S. Luo, "Joint user pairing and subchannel allocation for multisubchannel multiuser nonorthogonal multiple access systems," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8238–8248, 2018.

[36] P. Parida and S. S. Das, "Power allocation in OFDM based NOMA systems: A DC programming approach," in *IEEE Globecom Workshops*, 2014.

[37] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge university press, 2004.

[38] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley & Sons, 2012.

[39] L. Dai, B. Wang, Y. Yuan, S. Han, I Chih-Lin, and Z. Wang, "Non-orthogonal multiple access for 5G: Solutions, challenges, opportunities, and future research trends," *IEEE Commun. Mag.*, vol. 53, no. 9, pp. 74–81, 2015.

[40] B. Wang, L. Dai, Y. Zhang, T. Mir, and J. Li, "Dynamic compressive sensing-based multi-user detection for uplink grant-free NOMA," *IEEE Communications Letters*, vol. 20, no. 11, pp. 2320–2323, 2016.

[41] R. Abbas, M. Shirvanimoghaddam, Y. Li, and B. Vucetic, "On the performance of massive grant-free NOMA," in *IEEE 28th Int. Symp. on Personal Indoor and Mobile Radio Commun. (PIMRC)*, 2017.

[42] M. Shirvanimoghaddam, M. Dohler, and S. J. Johnson, "Massive non-orthogonal multiple access for cellular IoT: Potentials and limitations," *IEEE Commun. Mag.*, vol. 55, no. 9, pp. 55–61, 2017.

[43] E. Altman, K. Avrachenkov, and A. Garnaev, "Generalized α-fair resource allocation in wireless networks," in *IEEE Conf. on Decision and Control*, 2008, pp. 2414–2419.

[44] G. H. Hardy, J. E. Littlewood, and G. Pólya, *Inequalities*. Cambridge university press, 1952.

[45] A. Benjebbour, A. Li, Y. Saito, Y. Kishiyama, A. Harada, and T. Nakamura, "System-level performance of downlink NOMA for future LTE enhancements," in *IEEE Globecom Workshops*, 2013, pp. 66–70.

[46] S. N. Datta and S. Kalyanasundaram, "Optimal power allocation and user selection in non-orthogonal multiple access systems," in *IEEE Wireless Commun. and Networking Conf. (WCNC)*, 2016.

[47] X. Wang, R. Chen, Y. Xu, and Q. Meng, "Low-complexity power allocation in NOMA systems with imperfect SIC for maximizing weighted sum-rate," *IEEE Access*, vol. 7, pp. 94 238–94 253, 2019.

[48] F. Guo, H. Lu, D. Zhu, and Z. Gu, "Joint user association, grouping and power allocation in uplink NOMA systems with QoS constraints," in *IEEE Int. Conf. Commun. (ICC)*, 2019.

[49] C. He, Y. Hu, Y. Chen, and B. Zeng, "Joint power allocation and channel assignment for NOMA with deep reinforcement learning," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2200–2210, 2019.

[50] Z. Ding, Z. Yang, P. Fan, and H. V. Poor, "On the performance of non-orthogonal multiple access in 5G systems with randomly deployed users," *IEEE Signal Process. Lett.*, vol. 21, no. 12, pp. 1501–1505, 2014.

[51] Y.-F. Liu, "Complexity analysis of joint subcarrier and power allocation for the cellular downlink OFDMA system," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 661–664, 2014.

[52] Y.-F. Liu and Y.-H. Dai, "On the complexity of joint subcarrier and power allocation for multi-user OFDMA systems," *IEEE Trans. Signal Process.*, vol. 62, no. 3, pp. 583–596, 2014.

[53] D. A. Eckhardt and P. Steenkiste, "Effort-limited fair (ELF) scheduling for wireless networks," in *IEEE Int. Conf. on Computer Commun. (Infocom)*, vol. 3, 2000, pp. 1097–1106.

[54] L. Wei and Y. Pan, *Resource allocation in next generation wireless networks*, 1st ed. Nova Publishers, 2006, vol. 5.

[55] Y. Sun, D. W. K. Ng, J. Zhu, and R. Schober, "Robust and secure resource allocation for full-duplex MISO multicarrier NOMA systems," *IEEE Trans. Commun.*, vol. 66, no. 9, pp. 4119–4137, 2018.

[56] H. Huang, J. Xiong, J. Yang, G. Gui, and H. Sari, "Rate region analysis in a full-duplex-aided cooperative nonorthogonal multiple-access system," *IEEE Access*, vol. 5, pp. 17 869–17 880, 2017.

[57] X. Yue, Y. Liu, S. Kang, A. Nallanathan, and Z. Ding, "Outage performance of full/half-duplex user relaying in NOMA systems," in *IEEE Int. Conf. Commun. (ICC)*, 2017.

[58] Z. Ding, P. Fan, and H. V. Poor, "On the coexistence between full-duplex and NOMA," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 692–695, 2018.

[59] X. Yue, Y. Liu, R. Liu, A. Nallanathan, and Z. Ding, "Full/half-duplex relay selection for cooperative NOMA networks," in *IEEE Global Commun. Conf. (GLOBECOM)*, 2017.

[60] M. B. Shahab and S. Y. Shin, "Time shared half/full-duplex cooperative NOMA with clustered cell edge users," *IEEE Commun. Lett.*, vol. 22, no. 9, pp. 1794–1797, 2018.

[61] D. S. Hochbaum, *Approximation Algorithms for NP-Hard Problems*. PWS Publishing Co., 1996.

[62] M. R. Garey and D. S. Johnson, *Computers and Intractability*. W. H. Freeman New York, 2002, vol. 29.

[63] ——, "Strong NP-completeness results: Motivation, examples, and implications," *Journal of the ACM*, vol. 25, no. 3, pp. 499–508, 1978.

[64] S. Hayashi and Z.-Q. Luo, "Spectrum management for interference-limited multiuser communication systems," *IEEE Trans. Inf. Theory*, vol. 55, no. 3, pp. 1153–1175, 2009.

[65] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of computer computations*, Springer, 1972, pp. 85–103.

[66] M. R. Garey and D. S. Johnson, "Complexity results for multiprocessor scheduling under resource constraints," *SIAM Journal on Computing*, vol. 4, no. 4, pp. 397–411, 1975.

[67] P. C. Weeraddana, M. Codreanu, M. Latva-aho, A. Ephremides, C. Fischione, *et al.*, "Weighted sum-rate maximization in wireless networks: A review," *Found. and Trends in Netw.*, vol. 6, no. 1–2, pp. 1–163, 2012.

[68] A. Eryilmaz and R Srikant, "Fair resource allocation in wireless networks using queue-length-based scheduling and congestion control," *IEEE/ACM Trans. Netw.*, vol. 15, no. 6, pp. 1333–1344, 2007.

[69] X. Chen, A. Benjebbour, A. Li, and A. Harada, "Multi-user proportional fair scheduling for uplink non-orthogonal multiple access (NOMA)," in *IEEE 79th Veh. Technol. Conf. (VTC Spring)*, 2014.

[70] E. Okamoto, "An improved proportional fair scheduling in downlink non-orthogonal multiple access system," in *IEEE 82nd Veh. Technol. Conf. (VTC Fall)*, 2015.

[71] H. Kellerer, U. Pferschy, and D. Pisinger, *Knapsack Problems*. Berlin Heidelberg: Springer-Verlag, 2004.

[72] V. V. Vazirani, *Approximation Algorithms*. Springer Science & Business Media, 2013.

[73] A. Tarek, "Multi-key binary search and the related performance," in *Proc. Amer. Conf. on Appl. Math.*, ser. MATH'08, World Scientific, Engineering Academy, and Society (WSEAS), 2008, pp. 104–109.

[74] J. B. Rosen, "Existence and uniqueness of equilibrium points for concave n-person games," *Econometrica*, vol. 33, no. 3, pp. 520–534, 1965.

[75] G. Scutari, D. P. Palomar, F. Facchinei, and J. S. Pang, "Convex optimization, game theory, and variational inequality theory," *IEEE Signal Processing Magazine*, vol. 27, no. 3, pp. 35–49, 2010.

[76] P. H. Calamai and J. J. Moré, "Projected gradient methods for linearly constrained problems," *Mathematical Programming*, vol. 39, no. 1, pp. 93–116, 1987.

[77] Y. Nesterov, "Introductory lectures on convex programming volume I: Basic course," *Lecture notes*, 1998.

[78] I. L. Glicksberg, "A further generalization of the Kakutani fixed point theorem with application to Nash equilibrium points," *National Academy of Sciences*, vol. 38, pp. 170–174, 1952.

[79] K. Fan, "Fixed points and minimax theorems in locally convex topological linear spaces," *National Academy of Sciences*, vol. 38, pp. 121–126, 1952.

[80] G. Debreu, "A social equilibrium existence theorem," *National Academy of Sciences*, vol. 38, pp. 886–893, 1952.

[81] K. W. Shum, K. K. Leung, and C. W. Sung, "Convergence of iterative waterfilling algorithm for Gaussian interference channels," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 6, pp. 1091–1100, 2007.

[82] G. Scutari, D. P. Palomar, and S. Barbarossa, "Asynchronous iterative waterfilling for Gaussian frequency-selective interference channels: A unified framework," *IEEE Trans. Inf. Theory*, vol. 54, no. 7, pp. 2868–2878, 2008.

[83] ETSI TR 136 942, *LTE; evolved universal terrestrial radio access (E-UTRA); radio frequency (RF) system scenarios*, 3GPP TR 36.942 version 15.0.0 Release 15, 2018.

[84] R. Jain, D. Chiu, and W. Hawe, *A quantitative measure of fairness and discrimination for resource allocation in shared computer systems*. Eastern Research Lab, Digital Equipment Corporation, DEC Technical Report 301, 1984.

[85] S. Timotheou and I. Krikidis, "Fairness for non-orthogonal multiple access in 5G systems," *IEEE Signal Process. Lett.*, vol. 22, no. 10, pp. 1647–1651, 2015.

[86] J. Choi, "Power allocation for max-sum rate and max-min rate proportional fairness in NOMA," *IEEE Commun. Lett.*, vol. 20, no. 10, pp. 2055–2058, 2016.

[87] A. E. Mostafa, Y. Zhou, and V. W. Wong, "Connection density maximization of narrowband IoT systems with NOMA," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4708–4722, 2019.

[88] ——, "Connectivity maximization for narrowband IoT systems with NOMA," in *IEEE Int. Conf. Commun. (ICC)*, 2017.

[89] D. Tweed, S. Parsaeefard, M. Derakhshani, and T. Le-Ngoc, "Dynamic resource allocation for MC-NOMA VWNs with imperfect SIC," in *IEEE 28th Int. Symp. on Personal Indoor and Mobile Radio Commun. (PIMRC)*, 2017.

[90] D. Tweed and T. Le-Ngoc, "Dynamic resource allocation for uplink MIMO NOMA VWN with imperfect SIC," in *IEEE Int. Conf. Commun. (ICC)*, 2018.

[91] J.-M. Kang, I.-M. Kim, and C.-J. Chun, "Deep learning-based MIMO-NOMA with imperfect SIC decoding," *IEEE Syst. J.*, 2019.

[92] M. Liu, T. Song, and G. Gui, "Deep cognitive perspective: Resource allocation for NOMA-based heterogeneous IoT with imperfect SIC," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2885–2894, 2018.

[93] J. Choi, "Joint rate and power allocation for NOMA with statistical CSI," *IEEE Trans. Commun.*, vol. 65, no. 10, pp. 4519–4528, 2017.

[94] Z. Yang, Z. Ding, P. Fan, and G. K. Karagiannidis, "On the performance of non-orthogonal multiple access systems with partial channel information," *IEEE Trans. Commun.*, vol. 64, no. 2, pp. 654–667, 2015.

[95] N. Ye, X. Li, H. Yu, A. Wang, W. Liu, and X. Hou, "Deep learning aided grant-free NOMA toward reliable low-latency access in tactile internet of things," *IEEE Trans. Ind. Informat.*, vol. 15, no. 5, pp. 2995–3005, 2019.

[96] V. Kann, "On the approximability of NP-complete optimization problems," PhD thesis, Royal Institute of Technology Stockholm, 1992.

[97] H. Huang, S. Guo, G. Gui, Z. Yang, J. Zhang, H. Sari, and F. Adachi, "Deep learning for physical-layer 5G wireless techniques: Opportunities, challenges and solutions," *arXiv preprint arXiv:1904.09673*, 2019.

[98] M. Kim, N.-I. Kim, W. Lee, and D.-H. Cho, "Deep learning-aided SCMA," *IEEE Commun. Lett.*, vol. 22, no. 4, pp. 720–723, 2018.

[99] G. Gui, H. Huang, Y. Song, and H. Sari, "Deep learning for an effective nonorthogonal multiple access scheme," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8440–8450, 2018.

[100] L. Jiang, X. Li, N. Ye, and A. Wang, "Deep learning-aided constellation design for downlink NOMA," in *Int. Wireless Commun. and Mobile Computing Conf. (IWCMC)*, 2019, pp. 1879–1883.

[101] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. on Cogn. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, 2017.

[102] K. N. Doan, M. Vaezi, W. Shin, H. V. Poor, H. Shin, and T. Q. Quek, "Power allocation in cache-aided NOMA systems: Optimization and deep reinforcement learning approaches," *IEEE Trans. Commun.*, 2019.

[103] Y. Sun, Y. Wang, J. Jiao, S. Wu, and Q. Zhang, "Deep learning-based long-term power allocation scheme for NOMA downlink system in S-IoT," *IEEE Access*, vol. 7, pp. 86 288–86 296, 2019.

[104] L. Xiao, Y. Li, C. Dai, H. Dai, and H. V. Poor, "Reinforcement learning-based NOMA power allocation in the presence of smart jamming," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 3377–3389, 2017.

INSTITUT POLYTECHNIQUE DE PARIS

Titre : Allocation des ressources et optimisation pour l'accès multiple non-orthogonal

Mots clés : NOMA, SIC, allocation des ressources, optimisation convexe et combinatoire

Résumé : L'accès multiple non-orthogonal (NOMA) est une technologie d'accès radio prometteuse pour améliorer l'efficacité spectrale et augmenter massivement la connectivité des réseaux sans fil. Contrairement aux méthodes d'accès orthogonal, telles que l'OFDMA, NOMA peut multiplexer en puissance plusieurs signaux sur une même ressource radio. Un des principaux défis de NOMA est de résoudre le problème d'allocation des sous-porteuses et de la puissance (JSPA).

Dans cette thèse, nous présentons un nouveau cadre théorique permettant l'étude d'une classe de problèmes d'optimisation JSPA. Nous considérons diverses contraintes réalistes et une fonction d'objectif générique pouvant représenter, entre autres, les fonctions suivantes : somme pondérée des débits (WSR), équité proportionnelle, moyenne harmonique et équité max-min. Nous prouvons que JSPA est NP-difficile en général. De plus, nous étudions sa complexité et son approximabilité dans divers cas particuliers et pour différentes fonctions et contraintes.

Nous tentons d'abord de maximiser la WSR avec des contraintes de puissances cellulaires. Nous proposons trois algorithmes : OPT-JSPA est optimal et sa complexité est moindre que les méthodes optimales existantes. Étant donné sa complexité pseudo-polynomiale, il ne peut pas être exécuté en temps réel, mais convient pour des simulations. Afin de réduire cette complexité, nous développons un schéma d'approximation entièrement en temps polynomial (FPTAS) appelé ε-JSPA. Celui-ci garantit d'obtenir une solution optimale à un facteur $1+\varepsilon$ en temps polynomial. A notre connaissance, ε-JSPA est le premier FPTAS proposé pour ce problème. Finalement, GRAD-JSPA est une heuristique fondée sur la descente de gradient. Des simulations montrent que GRAD-JSPA atteint des performances proches de l'optimum avec une faible complexité.

Nous étudions des contraintes de puissances individuelles dans un second temps. L'allocation des puissances est optimisée par descente de gradient et est optimale. Ensuite, nous développons trois heuristiques pour l'allocation des sous-porteuses dans le problème JSPA : DGA qui est centralisé, ainsi que DPGA et DIWA qui sont répartis. Leurs performance et complexité sont évaluées par simulations.

Title : Resource allocation and optimization for the non-orthogonal multiple access

Keywords : NOMA, SIC, resource allocation, convex and combinatorial optimization

Abstract : Non-orthogonal multiple access (NOMA) is a promising technology to increase the spectral efficiency and enable massive connectivity in future wireless networks. In contrast to orthogonal schemes, such as OFDMA, NOMA can serve multiple users on the same frequency and time resource by superposing their signal in the power domain. One of the key challenges for radio resource management (RRM) in NOMA systems is to solve the joint subcarrier and power allocation (JSPA) problem.

In this thesis, we present a novel optimization framework to study a general class of JSPA problems. This framework employs a generic objective function which can be used to represent the popular weighted sum-rate (WSR), proportional fairness, harmonic mean and max-min fairness utilities. Our work also integrates various realistic constraints. We prove under this framework that JSPA is NP-hard to solve in general. In addition, we study its computational complexity and approximability in various special cases, for different objective functions and constraints.

In this framework, we first consider the WSR maximization problem subject to cellular power constraint. We propose three new algorithms: OPT-JSPA computes an optimal solution with lower complexity than current optimal schemes in the literature. It can be used as an optimal benchmark in simulations. However, its pseudo-polynomial time complexity remains impractical for real-world systems with low latency requirements. To further reduce the complexity, we propose a fully polynomial-time approximation scheme called ε-JSPA, which allows tight trade-offs between performance guarantee and complexity. To the best of our knowledge, ε-JSPA is the first polynomial-time approximation scheme proposed for this problem. Finally, GRAD-JSPA is a heuristic based on gradient descent. Numerical results show that it achieves near-optimal WSR with much lower complexity than existing optimal methods.

As a second application of our framework, we study individual power constraints. Power control is solved optimally by gradient descent methods. Then, we develop three heuristics: DGA, DPGA and DIWA, which solve the JSPA problem for centralized and distributed settings. Their performance and computational complexity are compared through simulations.

Institut Polytechnique de Paris
91120 Palaiseau, France