HAL
archives-ouvertes.fr

# Joint Subcarrier and Power Allocation in NOMA: Optimal and Approximate Algorithms

Lou Salaun, Marceau Coupechoux, Chung Shue Chen

▶ **To cite this version:**

Lou Salaun, Marceau Coupechoux, Chung Shue Chen. Joint Subcarrier and Power Allocation in NOMA: Optimal and Approximate Algorithms. IEEE Transactions on Signal Processing, Institute of Electrical and Electronics Engineers, 2020, 68 (1), pp.2215 - 2230. 10.1109/TSP.2020.2982786 . hal-02302721v2

**HAL Id: hal-02302721**
**https://hal.archives-ouvertes.fr/hal-02302721v2**

Submitted on 18 Mar 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Joint Subcarrier and Power Allocation in NOMA: Optimal and Approximate Algorithms

Lou Salaün, *Student Member, IEEE*, Marceau Coupechoux, and Chung Shue Chen, *Senior Member, IEEE*

***Abstract*—Non-orthogonal multiple access (NOMA) is a promising technology to increase the spectral efficiency and enable massive connectivity in 5G and future wireless networks. In contrast to orthogonal schemes, such as OFDMA, NOMA multiplexes several users on the same frequency and time resource. Joint subcarrier and power allocation problems (JSPA) in NOMA are NP-hard to solve in general. In this family of problems, we consider the weighted sum-rate (WSR) objective function as it can achieve various tradeoffs between sum-rate performance and user fairness. Because of JSPA's intractability, a common approach in the literature is to solve separately the power control and subcarrier allocation (also known as user selection) problems, therefore achieving sub-optimal result. In this work, we first improve the computational complexity of existing single-carrier power control and user selection schemes. These improved procedures are then used as basic building blocks to design new algorithms, namely OPT-JSPA, $\varepsilon$-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 $\varepsilon$-JSPA. Since, no approximation has been studied in the literature, $\varepsilon$-JSPA stands out by allowing to control a tight trade-off between performance guarantee and complexity. 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.**

## I. INTRODUCTION

In multi-carrier multiple access systems, the total frequency bandwidth is divided into subcarriers and assigned to users to optimize the spectrum utilization. Orthogonal multiple access (OMA), such as orthogonal frequency-division multiple access (OFDMA) adopted in 3GPP-LTE and also 5G New Radio Phase 1 standards [1], [2], only serves one user per subcarrier in order to avoid intra-cell interference and have low-complexity signal decoding at the receiver side. OMA is known to be suboptimal in terms of spectral efficiency [3].

The principle of multi-carrier non-orthogonal multiple access (MC-NOMA) is to multiplex several users on the same subcarrier by performing signals superposition at the transmitter side. Successive interference cancellation (SIC) is applied at the receiver side to mitigate interference between superposed signals. MC-NOMA is a promising multiple access technology for 5G and beyond mobile networks as it can achieve higher spectral efficiency than OMA schemes [4], [5].

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 in practice [6], subcarrier allocation for each transmission also needs to be optimized. As a consequence, joint subcarrier and power allocation problems (JSPA) in NOMA have received much attention. In this class of problems, weighted sum-rate (WSR) maximization is especially important as it can achieve different tradeoffs between sum-rate performance and user fairness [7].

Two types of power constraints are considered in the literature. On the one hand, cellular power constraint is mostly used in downlink transmissions to represent the total transmit power budget available at the base station (BS). On the other hand, individual power constraint sets a power limit independently for each user. The latter is often considered in uplink scenarios [8], [9], nevertheless it can also be applied to the downlink [10], [11].

It is known that the equal-weight sum-rate and WSR maximization are both strongly NP-hard if we consider individual power constraints in OFDMA [12] and in MC-NOMA systems [11], [13]. Nevertheless, several algorithms have been developed to perform subcarrier and/or power allocation for MC-NOMA and this type of constraints. Fractional transmit power control (FTPC) is a simple heuristic that allocates a fraction of the total power budget to each user based on their channel conditions [4]. In [8] and [9], heuristic user pairing strategies and iterative resource allocation algorithms are studied for uplink transmissions. A time efficient two-step heuristic is introduced in [10] to solve the problem with equal weights. Reference [11] derives an upper bound on the optimal WSR and proposes a Lagrangian duality and dynamic programming (LDDP) scheme. This scheme achieves near-optimal result, assuming the power budget is divided in $J$ equal parts to be allocated. It mainly serves as benchmark due to its high computational complexity when $J$ is large in practical systems, which may not be suitable for low latency requirements.

If we consider now cellular power constraints without individual power constraints, the equal-weight sum-rate maximization is polynomial time solvable in OFDMA [14]. 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 [15, Proposition 1] proves that the subcarrier optimization is NP-hard only in the case of equal power allocation among the users. The

L. Salaün and C. S. Chen are with Bell Labs, Nokia Paris-Saclay, 91620 Nozay, France, and also with Lincs, Paris 75013, France (e-mail: lou.salaun@nokia-bell-labs.com; chung_shue.chen@nokia-bell-labs.com).

L. Salaün and M. Coupechoux are with LTCI, Telecom ParisTech, University of Paris-Saclay, Paris 75013, France (e-mail: marceau.coupechoux@telecom-paristech.fr).

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 [11] solves it optimally in pseudo-polynomial time depending on $J$. Therefore, the WSR problem with cellular power constraint is *weakly* NP-hard at most (in contrast to *strongly* NP-hard for the individual power constraints as mentioned previously). 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 [16]. In reference [15], a matching algorithm is developed to perform subcarrier allocation. A minorization-maximization algorithm is used in [17] to compute precoding vectors of a MISO-NOMA system. The authors of [18] employ monotonic optimization to develop an optimal resource allocation policy, which serves as benchmark due to its exponential complexity. The TSDP scheme is also optimal for cellular power constraint scenarios as proven in Theorem 13 of reference [11], but it has high pseudo-polynomial complexity as well.

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 extend the framework of our previous paper [19] with a fully polynomial-time approximation scheme (FPTAS) for the WSR maximization problem with cellular power constraint. In [19], we developed the following algorithms: two basic building blocks SCPC and SCUS which solve respectively the single-carrier power control and single-carrier user selection problems in polynomial time; and a heuristic JSPA scheme based on projected gradient descent, SCPC and SCUS, denoted here by GRAD-JSPA. Our contributions are as follows:

1) We improve SCPC and SCUS by performing precomputation to avoid repeated operations each time they are executed. This reduces their computational complexity by a factor proportional to the number of users.
2) The above precomputation also speeds up GRAD-JSPA, which now has low and practical computational complexity. In addition, numerical results show that GRAD-JSPA achieves near-optimal WSR, as well as significant improvement in performance over OMA.
3) We develop a new optimal algorithm, called OPT-JSPA, suitable for use as a benchmark for optimal WSR performance in simulations. We show that OPT-JSPA has lower computational complexity than existing optimal schemes [11], [18].
4) We propose a FPTAS, which is denoted by $\varepsilon$-JSPA. Its design is based on the improved SCPC and SCUS, as well as techniques from the multiple choice knapsack problem [20]. 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, $\varepsilon$-JSPA stands out by allowing to control a tight trade-off between performance guarantee and complexity.

Through the aforementioned points, our aim is to deepen the understanding of JSPA and NOMA resource allocation problems. We develop optimal, approximate and heuristic schemes which are each suitable for systems with different computational capabilities, as well as for performance benchmarking. In addition, we provide mathematical tools to study the WSR maximization problem, which can also be applied to other similar resource allocation problems.

The paper is organized as follows. In Section II, we present the system model and notations. Section III formulates the WSR problem. We consider two single-carrier subproblems in Section IV that were previously solved using SCPC and SCUS in [19]. We propose improved versions of these algorithms, namely i-SCPC and i-SCUS, which perform precomputation to reduce their complexity. Based on these basic building blocks, we develop a low complexity gradient descent based heuristic (GRAD-JSPA), a pseudo-polynomial time optimal algorithm (OPT-JSPA) and a FPTAS with $\varepsilon$-approximation guarantee ($\varepsilon$-JSPA) in Section V. We show in Section VI some numerical results, highlighting our solution's WSR performance and computational complexity. In Section VII, we discuss about how to generalize our framework to more realistic channel estimation models and multi-antenna systems. Finally, we conclude in Section VIII.

## II. SYSTEM MODEL AND NOTATIONS

We define in this section the system model and notations used throughout the paper. We consider a downlink multi-carrier NOMA system composed of one base station (BS) serving $K$ users. We denote the index set of users by $\mathcal{K} \triangleq \{1,\dots,K\}$, and the set of subcarriers by $\mathcal{N} \triangleq \{1,\dots,N\}$. 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 $W_n$.

Let $p_k^n$ denotes the transmit power from the BS to user $k \in \mathcal{K}$ on subcarrier $n \in \mathcal{N}$. User $k$ is said to be active on subcarrier $n$ if $p_k^n > 0$, and inactive otherwise. In addition, let $g_k^n$ be the channel gain between the BS and user $k$ on subcarrier $n$, and $\eta_k^n$ be the received noise power. We assume that the channel gains are perfectly known. We discuss about more realistic models with imperfect channel state information (CSI) in Section VII. 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, and $\boldsymbol{p}^n \triangleq (p_k^n)_{k\in\mathcal{K}}$ the vector of transmit powers on subcarrier $n$.

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 decoding complexity and error propagation [6]. 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 $\forall n \in \mathcal{N},\ |\mathcal{U}_n| \leq M$, where $|\cdot|$ denotes the cardinality of a finite set. Each subcarrier is modeled as a multi-user Gaussian broadcast channel [6] 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 over the active users on $n$, i.e., $\pi_n\colon \{1,\dots,|\mathcal{U}_n|\} \to \mathcal{U}_n$. However, for ease of reading, we choose to represent it by a permutation over all users $\mathcal{K}$, i.e., $\pi_n\colon \{1,\dots,K\} \to \mathcal{K}$. These two definitions are equivalent in our model since the Shannon capacity (2) does not depend on the inactive users $k \in \mathcal{K} \setminus \mathcal{U}_n$, for which $p_k^n = 0$. For $i \in \{1,\dots,K\}$, $\pi_n(i)$ returns the $i$-th decoded user's index. Conversely, user $k$'s decoding order is given by $\pi_n^{-1}(k)$.

In this work, we consider the optimal decoding order studied in [6, Section 6.2]. It consists of decoding users' signals from the highest to the lowest normalized noise power:

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

User $\pi_n(i)$ first decodes the signals of users $\pi_n(1)$ to $\pi_n(i-1)$ and subtracts them from the superposed signal before decoding its own signal. Interference from users $\pi_n(j)$ for $j > i$ is treated as noise. The maximum achievable data rate of user $k$ on subcarrier $n$ is given by Shannon capacity:

$$\begin{aligned} R_k^n(\boldsymbol{p}^n) &\triangleq W_n \log_2\left(1 + \frac{g_k^n p_k^n}{\sum_{j=\pi_n^{-1}(k)+1}^{K} g_k^n p^n_{\pi_n(j)} + \eta_k^n}\right), \\ &\overset{\text{(a)}}{=} W_n \log_2\left(1 + \frac{p_k^n}{\sum_{j=\pi_n^{-1}(k)+1}^{K} p^n_{\pi_n(j)} + \tilde{\eta}_k^n}\right), \end{aligned} \tag{2}$$

where equality (a) is obtained after normalizing by $g_k^n$. We assume perfect SIC, therefore interference from users $\pi_n(j)$ for $j < \pi_n^{-1}(k)$ is completely removed in (2).

## III. Problem Formulation

Let $\boldsymbol{w} = \{w_1,\dots,w_K\}$ be a sequence of $K$ positive weights. The main focus of this work is to solve the following JSPA optimization problem:

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

The objective of $\mathcal{P}$ is to maximize the system's WSR. As discussed in Section I, this objective function has received much attention since its weights $\boldsymbol{w}$ can be chosen to achieve different tradeoffs between sum-rate performance and fairness [7]. Note that $C1$ represents the cellular power constraint, i.e., a total power budget $P_{max}$ at the BS. In $C2$, we set a power limit of $P_{max}^n$ for each subcarrier $n$. This is a common assumption in multi-carrier systems, e.g., [12], [14]. Constraint $C3$ ensures that the allocated powers remain non-negative. Due to decoding complexity and error propagation in SIC [6], we restrict the maximum number of multiplexed users per subcarrier to $M$ in $C4$.

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

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

Let us consider the following change of variables:

$$\forall n \in \mathcal{N},\ x_i^n \triangleq \begin{cases} \sum_{j=i}^{K} p^n_{\pi_n(j)}, & \text{if } i \in \{1,\dots,K\}, \\ 0, & \text{if } i = K+1. \end{cases} \tag{3}$$

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 1** (Equivalent problem $\mathcal{P}'$)**.**
*Problem $\mathcal{P}$ is equivalent to problem $\mathcal{P}'$ formulated below:*

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

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

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

*and where $\mathcal{U}_n' \triangleq \{i \in \{1,\dots,K\}\colon x_i^n > x_{i+1}^n\}$. The constant term $A = \sum_{n\in\mathcal{N}} w_{\pi_n(K)} \log_2\left(1/\tilde{\eta}^n_{\pi_n(K)}\right)$ is chosen so that $\mathcal{P}$ and $\mathcal{P}'$ have exactly the same optimal value.*

*Proof:* The idea is to apply the change of variables (3) to problem $\mathcal{P}$. Details of the calculation can be found in Appendix A. □

The advantage of this formulation $\mathcal{P}'$ 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 $x_i^n$.

## IV. 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:

$$\begin{aligned} F^n(\bar{P}^n) = \max_{\boldsymbol{x}^n} \quad & \sum_{i=1}^{K} f_i^n(x_i^n) + A^n, \\ \text{subject to} \quad & C2'\text{–}3':\ \bar{P}^n \geq x_1^n \geq \ldots \geq x_K^n \geq 0, \\ & C4':\ |\mathcal{U}_n'| \leq M, \end{aligned} \tag{$\mathcal{P}'_{SC}(n)$}$$

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 have been introduced in our previous paper [19] to tackle respectively the single-carrier power control and single-carrier user selection sub-problems that arise from $\mathcal{P}'_{SC}(n)$. We provide technical details of these algorithms below, and we show how precomputation can further improve their computational complexity. They will be used as basic building blocks in Section V to design efficient algorithms GRAD-JSPA, OPT-JSPA and ε-JSPA, for the joint resource allocation problem.

### A. 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:

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

This simplification of notation is relevant for the analysis of SCPC and SCUS in the following subsections. Indeed, if users $j, \dots, i-1$ are not active (i.e., $j, \dots, i-1 \notin \mathcal{U}'_n$), then $x_j^n = \dots = x_i^n$, 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 $\mathcal{P}'_{SC}(n)$ only requires $O(M)$ operations.

We study the properties of $f_{j,i}^n$ in Lemma 2. Note that $f_i^n = f_{i,i}^n$, therefore Lemma 2 also holds for functions $f_i^n$. Fig. 1 shows the two general forms that can be taken by $f_{j,i}^n$.

**Lemma 2** (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:* These analytical properties can be obtained by studying the first and second derivatives of $f_{j,i}^n$. Details can be found in Appendix B. □

We present in Algorithm 1 the pseudocode ARGMAX$f$ which computes the maximum of $f_{j,i}^n$ on $\left[0, \bar{P}^n\right]$ following the result of Lemma 2. ARGMAX$f$ only requires a constant number of basic operations, therefore its complexity is $O(1)$.

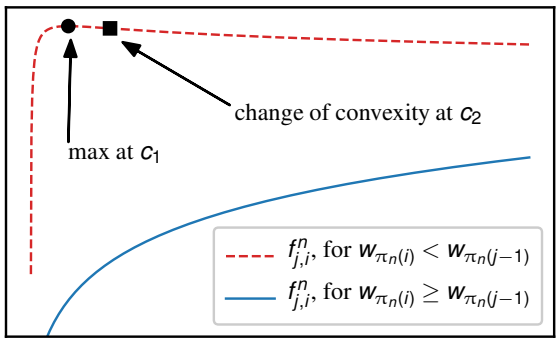


Fig. 1. The two general forms of functions $f_{j,i}^n$

**Algorithm 1** Compute maximum of $f_{j,i}^n$ on $\left[0, \bar{P}^n\right]$

**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**

### B. Single-Carrier Power Control

The single-carrier power control problem $\mathcal{P}'_{SCPC}(n)$ is equivalent to problem $\mathcal{P}'_{SC}(n)$, 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:

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

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 $i_n \in \{1_n, \dots, |\mathcal{U}'_n|_n\}$. 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 $\mathcal{P}'_{SCPC}(n)$ can be written as:

$$\sum_{i=1}^{K} f_i^n(x_i^n) + A^n = \sum_{i=1}^{|\mathcal{U}'_n|} f^n_{(i-1)_n+1,i_n}\left(x^n_{i_n}\right) + B^n, \quad (4)$$

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:

$$\tilde{f}^n_{j,i}\left(\mathcal{U}'_n, \cdot\right) \triangleq f^n_{(j-1)_n+1,i_n}(\cdot),$$
$$\text{ARGMAX}\tilde{f}\left(j, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right) \triangleq \text{ARGMAX}f\left((j-1)_n+1, i_n, \mathcal{I}^n, \bar{P}^n\right).$$

We reformulate the problem as:

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

Algorithm 2 presents the SCPC method proposed in [19]. The idea is to iterate over variables $x^n_{i_n}$ for $i = 1$ to $|\mathcal{U}'_n|$, and compute their optimal value $x^* = \text{ARGMAX}\tilde{f}(i, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n)$ 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 $x^n_{j_n} \geq \text{ARGMAX}\tilde{f}(j+1, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n)$. Then, variables $x^n_{(j+1)_n}, \ldots, x^n_{i_n}$ are set equal to $\text{ARGMAX}\tilde{f}(j+1, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n)$ at line 10. The optimality and complexity of SCPC are presented in Theorem 3.

**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: $x^* \leftarrow \text{ARGMAX}\tilde{f}\left(i, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right)$
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: $x^* \leftarrow \text{ARGMAX}\tilde{f}\left(j, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\right)$
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 3** (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 $O\left(M^2\right)$.*

*Proof:* We prove this theorem in Appendix C by mathematical induction combined with Lemma 2. □

In multi-carrier resource allocation schemes, such as GRAD-JSPA and ε-JSPA, it is often required to compute the optimal single-carrier power control and WSR for many different values of power budget $\bar{P}^n$. In these cases, running many times SCPC is actually not efficient in terms of computational 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 4** (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 $O\left(M^2\right)$. 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 $O\left(M^2 + CM\right)$.*

**Algorithm 3** Improved SCPC algorithm with precomputation

**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**

*Proof:* We show in Appendix D that subsequent evaluations of SCPC can be obtained as in line 1 of Algorithm 3. □

**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 $\mathcal{P}'_{SCPC}(n)$ as shown in Eqn. (4). If needed, the full vector $\boldsymbol{x}^n$ can be obtained by the following procedure in $O(K)$ operations:*

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

### C. Single-Carrier User Selection

Unlike in the previous subsection, we consider here furthermore the user selection $\mathcal{U}'_n$ optimization under multiplexing and SIC constraint $C4'$, i.e., we solve $\mathcal{P}'_{SC}(n)$. In [19], a dynamic programming (DP) is proposed to solve $\mathcal{P}'_{SC}(n)$. Here, we first develop a similar 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 $\mathcal{P}'_{SC}[m, j, i]$:

$$\begin{aligned} V[m,j,i] \quad \triangleq \quad \underset{\boldsymbol{x}^n}{\max} \quad & \sum_{l=j}^{K} f^n_l(x^n_l), \quad (\mathcal{P}'_{SC}[m,j,i]) \\ \text{subject to} \quad & C2', C3', C3'', \\ & C4' : \ |\mathcal{U}'_n| \leq m, \\ & C5' : \ x^n_j = \cdots = x^n_i. \end{aligned}$$

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

It is interesting to note that $V[M, 1, 1]$ is the optimal value of $\mathcal{P}'_{SC}(n)$, since the objective function is $\sum_{l=1}^{K} f^n_l(x^n_l)$ for $j = 1$ and constraint $C5'$ becomes trivially true for $j = i$. Let $x^{n*}_j, \ldots, x^{n}_K{}^*$ be the optimal solution achieving $V[m, j, i]$. We define $X[m, j, i] \triangleq x^{n*}_i$, which is also equal to $x^{n*}_j, \ldots, x^{n}_{i-1}{}^*$

**Algorithm 4** Single-carrier user selection algorithm (SCUS)

**function** SCUS$\left(\mathcal{I}^n, M, \bar{P}^n\right)$
1: ▷ Initialize arrays $V$, $X$, $U$ for $m = 0$ and $i = K$
2: **for** $i = K$ to 1 and $j = i$ to 1 **do**
3: $V\left[0, j, i\right] \leftarrow f^n_{j,K}\left(0\right)$
4: $X\left[0, j, i\right] \leftarrow 0$
5: $U\left[0, j, i\right] \leftarrow \varnothing$
6: **end for**
7: **for** $m = 1$ to $M$ and $j = K$ to 1 **do**
8: $x^* \leftarrow \text{ARGMAX}f\left(j, K, \mathcal{I}^n, \bar{P}^n\right)$
9: $V\left[m, j, K\right] \leftarrow f^n_{j,K}\left(x^*\right)$
10: $X\left[m, j, K\right] \leftarrow x^*$
11: $U\left[m, j, K\right] \leftarrow \varnothing$
12: **end for**
13: ▷ Compute $V$, $X$, $U$ for $m \in [1, M]$ and $j \leq i \leq K - 1$
14: **for** $i = K - 1$ to 1 and $m = 1$ to $M$ and $j = i$ to 1 **do**
15: $x^* \leftarrow \text{ARGMAX}f\left(j, i, \mathcal{I}^n, \bar{P}^n\right)$
16: $v_{act} \leftarrow f^n_{j,i}\left(x^*\right) + V[m-1, i+1, i+1]$
17: $v_{inact} \leftarrow V[m, j, i+1]$
18: **if** $v_{act} > v_{inact}$ and $x^* > X\left[m-1, i+1, i+1\right]$ **then**
19: $V\left[m, j, i\right] \leftarrow v_{act}$
20: $X\left[m, j, i\right] \leftarrow x^*$
21: $U\left[m, j, i\right] \leftarrow (m-1, i+1, i+1)$
22: **else**
23: $V\left[m, j, i\right] \leftarrow v_{inact}$
24: $X\left[m, j, i\right] \leftarrow X\left[m, j, i+1\right]$
25: $U\left[m, j, i\right] \leftarrow (m, j, i+1)$
26: **end if**
27: **end for**
28: ▷ Retrieve the optimal solution $\boldsymbol{x}^n$
29: $x^n_1, \ldots, x^n_K \leftarrow 0$
30: $(m, j, i) \leftarrow (M, 1, 1)$
31: **repeat**
32: $x^n_j, \ldots, x^n_i \leftarrow X\left[m, j, i\right]$
33: $(m, j, i) \leftarrow U\left[m, j, i\right]$
34: **until** $(m, j, i) = \varnothing$
35: **return** $\boldsymbol{x}^n$
**end function**

due to constraint $C5'$. The idea of SCUS is to recursively compute the elements of $V$ through the following relation:

$$V[m,j,i] = \begin{cases} v_{act}, & \text{if } v_{act} > v_{inact} \\ & \text{and } x^* > X\left[m-1, i+1, i+1\right], \\ v_{inact}, & \text{otherwise,} \end{cases} \tag{5}$$

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

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

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. (5).
If $V\left[m, j, i\right] = v_{act}$, then:

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

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:

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

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

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

A detailed analysis is given in Appendix E.

**Theorem 5** (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 $\mathcal{P}'_{SC}(n)$. Its worst case computational complexity is $O\left(MK^2\right)$.*

*Proof:* The proof is done by induction based on the principle of dynamic programming. See Appendix E. □

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 $x^n_1, \ldots, x^n_K$ 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 6.

**Theorem 6** (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 $O\left(MK^2\right)$. Any subsequent evaluation costs $O(MK)$. Hence, for $C$ different power budgets, i-*SCUS *computes their respective optimal single-carrier power control and user selection $\mathcal{P}'_{SC}(n)$ with overall complexity $O\left(MK^2 + CMK\right)$.*

*Proof:* See Appendix F. □

Table I 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 V.

**Algorithm 5** Improved SCUS algorithm with precomputation

**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 optimal solution $x_1^n, \dots, x_K^n$
4: Add $(\mathcal{U}'_n, x_1^n, \dots, x_K^n)$ to $collection$
5: **end for**
**function** i-SCUS$\big(\bar{P}^n\big)$
6: Get $(\mathcal{U}'_n, x_1^n, \dots, x_K^n)$ in $collection$ that maximizes $F^n(\mathcal{U}'_n, \bar{P}^n) = \sum_{l=1}^{|\mathcal{U}'_n|} \tilde{f}_{l,l}^n\big(\mathcal{U}'_n, \min\{x_{l_n}^n, \bar{P}^n\}\big) + B^n$
7: **return** $\min\{x_1^n, \bar{P}^n\}, \dots, \min\{x_K^n, \bar{P}^n\}$
**end function**

TABLE I
SUMMARY OF THE SINGLE-CARRIER RESOURCE ALLOCATION SCHEMES

| Algorithm | Complexity to perform $C$ evaluations |
|---|---|
| SCPC [19] | $O\big(CM^2\big)$ |
| i-SCPC | $O\big(M^2 + CM\big)$ |
| SCUS [19] | $O\big(CMK^2\big)$ |
| i-SCUS | $O\big(MK^2 + CMK\big)$ |

## V. JOINT SUBCARRIER AND POWER ALLOCATION

Recall that $F^n\big(\bar{P}^n\big)$ is the optimal value of $\mathcal{P}'_{SC}(n)$ 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 $x_1^n, \dots, x_K^n$ is the output of i-SCUS$\big(\bar{P}^n\big)$. Using this notation, the JSPA problem $\mathcal{P}'$ can be simplified as:

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

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

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

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

Problem $\mathcal{P}'_{MC}$ consists in optimizing the power budget $\bar{P}^n$ allocated to each subcarrier $n$. For a given budget $\bar{P}^n$, $F^n\big(\bar{P}^n\big)$ is computed by finding the optimal single-carrier user selection and power control using i-SCUS$\big(\bar{P}^n\big)$. The choice of $\bar{P}^n$ affects the single-carrier user selection and power control, i.e., variables $\boldsymbol{x}^n$ and $\mathcal{U}'_n$. The latter influence the value of $F^n$, which in turn has an impact on the power budget optimization. Although variables $\boldsymbol{x}^n$ and $\mathcal{U}'_n$ are hidden in $F^n$, they are nevertheless optimized jointly with $\bar{P}^n$. Indeed, we can see that $\mathcal{P}'_{MC}$ is equivalent to $\mathcal{P}'$ when replacing $F^n$ by its definition in $\mathcal{P}'_{SC}(n)$ along with its constraints $C2'$ to $C4'$.

### A. Gradient Descent Based Heuristic

GRAD-JSPA is an efficient heuristic based on projected gradient descent. Its principle is to perform a two-stage optimization as presented in Fig. 2. The first-stage is a projected gradient descent on $\bar{\boldsymbol{P}}$ in the search space $\mathcal{F}$. The gradient descent requires to evaluate $\sum_{n\in\mathcal{N}} F^n$ and its gradient at each iteration. This task is carried out by i-SCUS in the second-stage. We denote the derivative of $F^n$ at $\bar{P}^n$ by $F^{n\prime}\big(\bar{P}^n\big)$. Lemma 7 shows how to compute it. As illustrated in Fig. 2, the second-stage is called at each gradient iteration to return $F^n\big(\bar{P}^n\big)$ and $F^{n\prime}\big(\bar{P}^n\big)$ to the first-stage, for all $n \in \mathcal{N}$.

**Lemma 7** (Derivative of $F^n$)**.**
*Let $x_1^n, \dots, x_K^n$ 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:*

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

*where $l$ is the greatest index such that $x_l^n = \bar{P}^n$, and $\ln(2)$ is the natural logarithm of 2.*

*Proof:* To get $F^{n\prime}$, we first prove that $F^{n\prime}\big(\mathcal{U}'_n, \bar{P}^n\big) = f_{1,l}^{n\prime}\big(\bar{P}^n\big)$ and $F^n(\bar{P}^n) = \max_{\mathcal{U}'_n}\{F^n(\mathcal{U}'_n, \bar{P}^n)\}$, where max is taken over all active users sets in $collection$ of i-SCUS. See Appendix G for the detailed proof of semi-differentiability. □

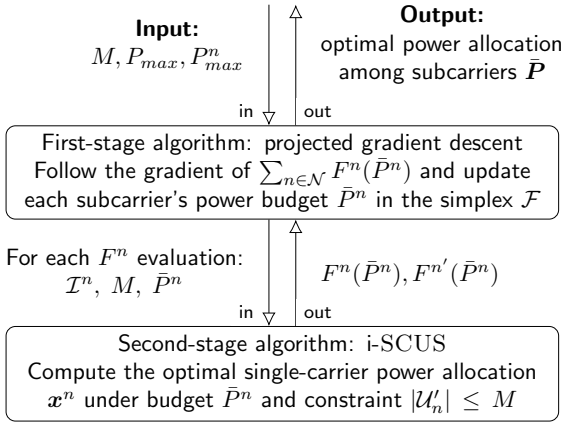


Fig. 2. Overview of GRAD-JSPA

The pseudocode of GRAD-JSPA is described in Algorithm 6. Input $\xi$ corresponds to the error tolerance at termination, as we can see at line 8. The search direction at lines 4-5 is the gradient of $\sum_{n\in\mathcal{N}} F^n$ evaluated at $\bar{\boldsymbol{P}}$. Since $F^1, \dots, F^N$ are independent, it is equal to the vector of $F^{1\prime}\big(\bar{P}^1\big), \dots, F^{N\prime}\big(\bar{P}^N\big)$. Note that the step size $\alpha$ at line 6 can be tuned by backtracking line search or exact line search [21, Section 9.2]. We adopt the latter to perform simulations. The projection of $\bar{\boldsymbol{P}} + \alpha\Delta$ on the simplex $\mathcal{F}$ at line 7 can be computed efficiently [21, Section 8.1.1], the details of its implementation are omitted here.

We showed in our previous work [19] that GRAD-JSPA worst case complexity is $O\big(\log(1/\xi)NMK^2\big)$ when SCUS is used to evaluate functions $F^n$, $n \in \mathcal{N}$. We show in Theorem 8 that the complexity of GRAD-JSPA can be reduced by the use of i-SCUS.

**Theorem 8** (Complexity of GRAD-JSPA)**.**
*Let $\xi$ be the error tolerance at termination. Algorithm* GRAD-JSPA *has complexity $O\big(NMK^2 + \log(1/\xi)NMK\big)$ when i-*SCUS *is used to evaluate functions $F^n$, $n \in \mathcal{N}$.*

*Proof:* In Appendix G, we prove that the objective function is piece-wise $\alpha$-strongly concave and $\beta$-smooth.

**Algorithm 6** Gradient descent based heuristic (GRAD-JSPA)

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

Therefore, the convergence to a local optimum follows from classical convex optimization results [22, Section 2.2.4]. □

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 VI that it achieves near-optimal WSR performance with low complexity.

### *B. Pseudo-Polynomial Time Optimal Scheme*

The JSPA problem as formulated in $\mathcal{P}'_{MC}$ has real variables $\bar{P}^n$ on a continuous search space $\mathcal{F}$. However, the study of NP-hard optimization problems and their approximation requires parameters and variables to be represented by a bounded number of bits [23], 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 $\mathcal{F}$, in the same way as in [11]. Let $\delta$ 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

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

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

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

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 [20]. 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 denote its optimal value by $F^*_{MCKP}$.

As mentioned previously, discretizing $\mathcal{P}'_{MC}$ is necessary due to the bounded precision which arises inherently from the study of algorithms and their implementation in practical systems. Besides, MCKP can be used to approach the continuous solution of $\mathcal{P}'_{MC}$ with arbitrary precision. Indeed, Theorem 9 shows that the discretization error is upper-bounded by a linear function in $\delta$ with a coefficient depending on the system's parameters.

**Theorem 9** (Discretization error between $F^*$ and $F^*_{MCKP}$)**.** *The gap between the optimal values of the continuous problem $\mathcal{P}'_{MC}$ and its discretized version MCKP with step size $\delta$ is upper-bounded by:*

$$F^* - F^*_{MCKP} \leq \delta \sum_{n\in\mathcal{N}} \max_{k\in\mathcal{K}} \left\{ \frac{W_n w_{\pi_n(k)}}{\left(\bar{P}^{n*} + \tilde{\eta}^n_{\pi_n(k)}\right)\ln(2)} \right\},$$

*where $\bar{P}^{n*}$ is the optimal power budget of $\mathcal{P}'_{MC}$ on subcarrier $n \in \mathcal{N}$.*

*Proof:* We derive the proof in Appendix H. □

The discrete problem MCKP can be solved optimally by *dynamic programming by weights* studied in [20, Section 11.5]. Based on this idea, we propose OPT-JSPA (see Algorithm 7) to solve $\mathcal{P}'_{MC}$. We first transform $\mathcal{P}'_{MC}$ 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 lines 6-7. We summarize the optimality and complexity of OPT-JSPA in Theorem 10. Detailed analysis of the dynamic programming can be found in Appendix I.

**Algorithm 7** The pseudo-polynomial time optimal scheme

**function** OPT-JSPA$\left((\mathcal{I}^n)_{n\in\mathcal{N}}, M, P_{max}, P^n_{max}, \delta\right)$
1: ▷ Compute the parameters of MCKP
2: **for** $n \in \mathcal{N}$ and $l \in [0, J]$ **do**
3: $a_{n,l} \leftarrow l\cdot\delta$
4: $c_{n,l} \leftarrow F^n(l\cdot\delta)$
5: **end for**
6: **return** optimal allocation from the *dynamic programming by weights* [20]
**end function**

**Theorem 10** (Optimality and complexity of OPT-JSPA)**.** *Given a minimum transmit power $\delta$, algorithm* OPT-JSPA *computes the optimal of $\mathcal{P}'_{MC}$ on the discrete set $\mathcal{F}'$. Its computational complexity is $O\left(NMK^2 + JNMK + J^2N\right)$, which is pseudo-polynomial in $J$.*

*Proof:* We explain the principle of dynamic programming by weights and derive its complexity in Appendix I. □

TABLE II
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 [18] | Optimal | Exponential in $K$ and $N$ |
| TSDP [11] | Optimal | $O\big(J^2NMK\big)$ |
| OPT-JSPA | Optimal | $O\big(NMK^2 + JNMK + J^2N\big)$ |
| $\varepsilon$-JSPA | FPTAS, i.e., its performance is within a factor $1-\varepsilon$ of the optimal, for any $\varepsilon > 0$ | $O\Big(NMK^2 + \min\Big\{ \log(J)\, \frac{N^2MK}{\varepsilon} + \frac{N^3}{\varepsilon^2}\,,\; JNMK + J^2N \Big\}\Big)$ |
| GRAD-JSPA | Heuristic | $O\big(NMK^2 + \log(J)NMK\big)$ |

OPT-JSPA is said to be pseudo-polynomial since it depends on the total number of power values $J$, whereas 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 $N$, $K$, $M$.

### C. Fully Polynomial-Time Approximation Scheme

We develop a FPTAS to avoid the pseudo-polynomial complexity in $J$ that is inherent to the optimal schemes OPT-JSPA and TSDP [11]. According to [24], 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 $\varepsilon$-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 J. 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:

$$L_n \triangleq \{l' \leq J, l \leq \frac{4N}{\varepsilon} - 1 : c_{n,l'} \geq l\frac{\varepsilon U}{4N} > c_{n,l'-1}\}.$$

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 $l \in \{1,\ldots,4N/\varepsilon\}$. Each $L_n$, for $n \in \mathcal{N}$, can be obtained by multi-key binary search [25]. All function evaluations required by the multi-key binary search are done by i-SCUS.

Finally, we apply the *dynamic programming by profits* [20, Section 11.5] in lines 5-6. 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 of $\varepsilon$-JSPA are summarized in Theorem 11. We provide more details on the estimation $U$ in Appendix J and the dynamic programming by profits in Appendix K.

**Theorem 11** (Performance and complexity of $\varepsilon$-JSPA)**.** *Given a minimum transmit power $\delta$ and an approximation factor $\varepsilon$, algorithm $\varepsilon$-JSPA computes an $\varepsilon$-approximation of $\mathcal{P}'_{MC}$ on the discrete set $\mathcal{F}'$. The algorithm is a FPTAS with asymptotic complexity: $O\Big(NMK^2 + \min\Big\{ \log(J)\, \frac{N^2MK}{\varepsilon} + \frac{N^3}{\varepsilon^2}\,,\; JNMK + J^2N \Big\}\Big)$.*

*Proof:* We derive this result in Appendix K, using the dynamic programming by profits and the estimation procedure studied in Appendix J. □

**Algorithm 8** The proposed FPTAS ($\varepsilon$-JSPA)

**function** $\varepsilon$-JSPA$\big((\mathcal{I}^n)_{n\in\mathcal{N}}, M, P_{max}, P^n_{max}, \delta, \varepsilon\big)$
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** $\varepsilon$-approximate allocation from the *dynamic*
6: *programming by profits* [20]
**end function**

### D. Comparison of JSPA Algorithms

In Table II, we compare the performance and complexity of the proposed algorithms with JSPA schemes in the literature. Reference [18] 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 $O\big(J^2NMK\big)$ according to [11, 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 [10] shows that in practical systems $J = \Theta\left(\min\{K, MN\}\right)$. This result is verified by simulation in Section VI. $\varepsilon$-JSPA is the proposed FTPAS. Its complexity is bounded by a polynomial in $N/\varepsilon$ and $\log(J)$. If $N/\varepsilon = O(J)$, it has lower complexity than OPT-JSPA. Otherwise, if $N/\varepsilon = \Omega(J)$, its complexity is asymptotically equivalent to OPT-JSPA's complexity. This means that for very low error rate $\varepsilon$, the complexity of $\varepsilon$-JSPA tends to that of OPT-JSPA. 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 $\xi$ 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 $J$.

## VI. NUMERICAL RESULTS

We evaluate the WSR and computational complexity of OPT-JSPA, $\varepsilon$-JSPA and GRAD-JSPA through numerical simulations. We compare them with the optimal benchmark scheme TSDP introduced in [11]. We consider a hexagonal

TABLE III
SIMULATION PARAMETERS

| 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 $\delta$ | 0.01 W |
| Number of power values $J$ | $10^3$ |
| Error tolerance $\xi$ | $10^{-4}$ |
| Parameter $M$ | 1 (OMA), 2 and 3 (NOMA) |

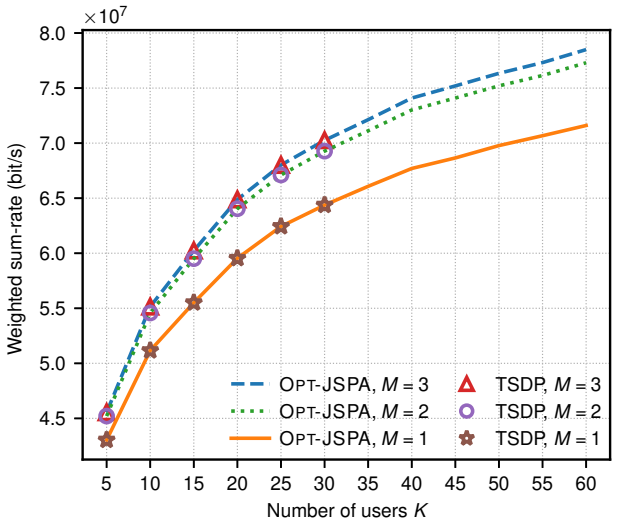


Fig. 3. WSR of the optimal schemes for different number of users $K$

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 $n$. We follow the radio propagation model of [26], including path loss, shadowing and Rayleigh fading. 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. 6 and 7 represent a single instance. The simulation parameters and channel model are summarized in Table III.

Fig. 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 complexity. We see that OPT-JSPA and TSDP achieve the same WSR performance, which is consistent with the fact that they are both optimal. Indeed, the optimality of OPT-JSPA is shown in Theorem 10, and the optimality of TSDP has been proven in [11, Theorem 13]. Although both algorithms have the same performance, we will see further on in Fig. 5 that OPT-JSPA has lower computational complexity than TSDP. The performance gain of NOMA with $M = 2$ (resp. $M = 3$) over

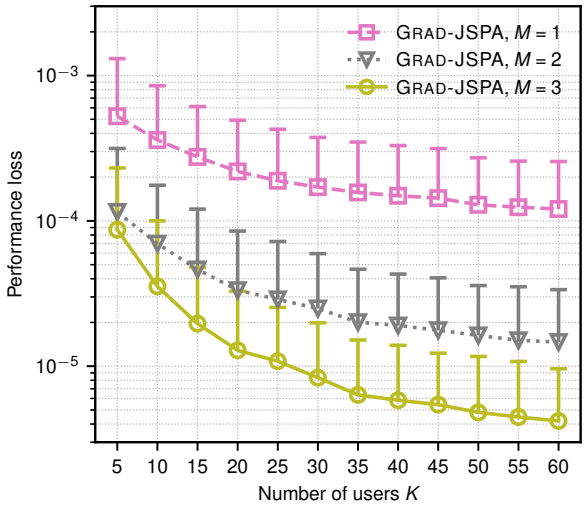


Fig. 4. Performance loss of GRAD-JSPA compared to the optimal WSR

OMA (i.e., $M = 1$) is about 8% (resp. 10%), for $K = 60$.

Fig. 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 $9 \times 10^{-4}$ of performance loss. We observe that the average performance loss is always below $6 \times 10^{-4}$. 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. 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. 6 and 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. 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 $N/\varepsilon$ increases, the approximation guarantee tends to the

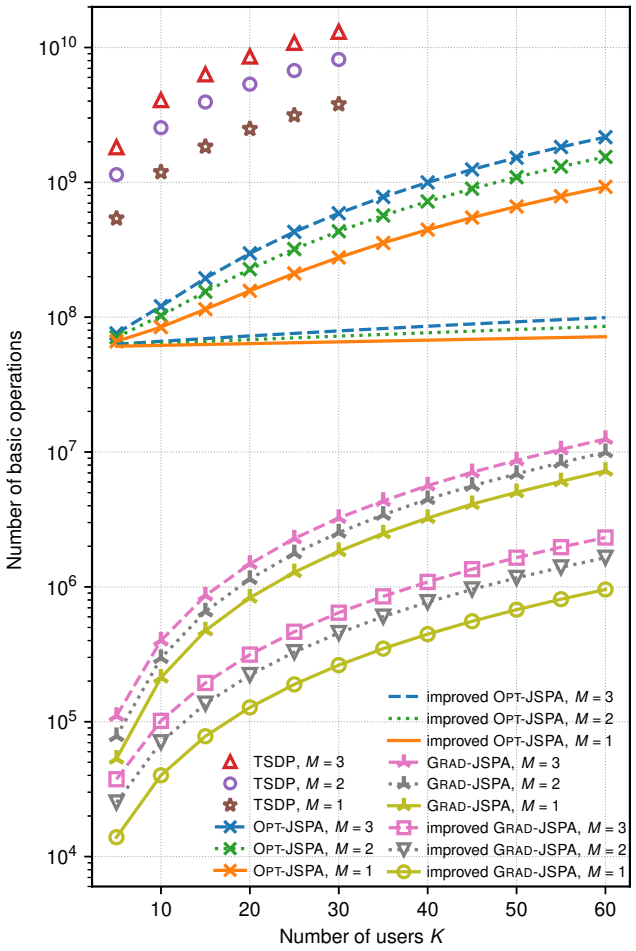


Fig. 5. Number of basic operations performed by each algorithm versus $K$

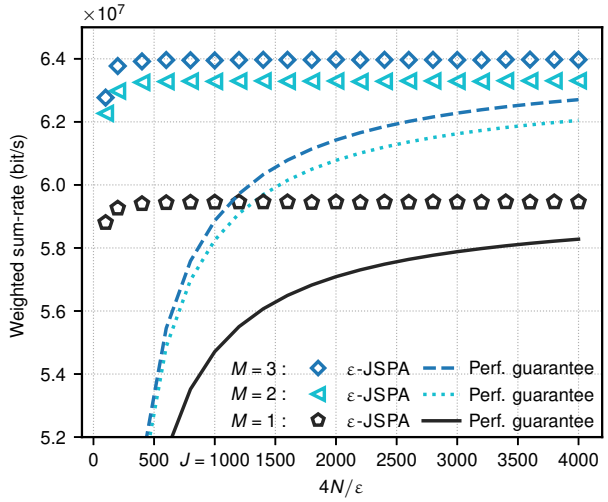


Fig. 6. WSR of $\varepsilon$-JSPA and its guaranteed performance bound versus $4N/\varepsilon$

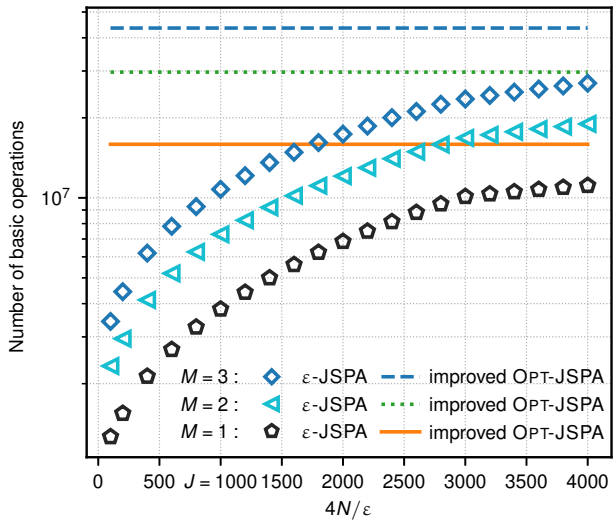


Fig. 7. Number of basic operations performed by $\varepsilon$-JSPA versus $4N/\varepsilon$

optimal. In this instance, the algorithm already achieves a near-optimal solution for $4N/\varepsilon = 400$, i.e., $\varepsilon = 0.2$. In Fig. 7, we also plot the complexity of the improved OPT-JSPA for comparison. As explained in Section V-C, the complexity increases with $N/\varepsilon$ and becomes (asymptotically) equal to that of OPT-JSPA for $N/\varepsilon = \Omega(J)$. In this regime, there is apparently no benefit of using $\varepsilon$-JSPA, since OPT-JSPA achieves the optimal with the same complexity. Nevertheless, in practice, we can see that even for $4N/\varepsilon \geq J$, $\varepsilon$-JSPA has less operations than OPT-JSPA. This is because the number of items computed by $\varepsilon$-JSPA increases slowly and smoothly as a function of $\varepsilon$. This behavior is not captured in the asymptotic complexity (big-$O$ notation). This is verified in Fig. 7 for up to $4J = 4000$. In summary, $\varepsilon$-JSPA allows us to control the trade-off between WSR and complexity with $\varepsilon$.

## VII. Discussion on Possible Generalizations

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 [27], [28], for which the channel gains are given with a known estimation error probability distribution, and *second order statistics* (SOS) adopted in [29], for which only the distances between users and BS are known. We believe that our framework can be extended to these cases by maximizing the expected WSR depending on stochastic channel gains, while the power constraints remain unchanged (i.e., non-stochastic). The challenge would be to characterize such a stochastic objective function in many scenarios. This is a possible future research direction.

As multi-antenna technologies are becoming more and more important in 5G and Beyond 5G systems [1], [2], it would be interesting to extend the current work to multi-antenna transmissions. Paper [30] states that MC-NOMA with multiple antennas is a much more complex problem which requires to develop novel low complexity solutions. One may draw inspiration from the work of Sun et al. [31] which generalizes the monotonic optimization framework of [18] to a MC-MISO-NOMA system. A similar approach might be adopted in our framework, which is explained as follows: Since the SIC decoding order in multi-antenna MC-NOMA systems does not only depend on the channel gains but also on the beamforming (BF), the user clustering (UC) and BF have to be jointly optimized to achieve optimal or approximate performance. Hence, the idea would be to extend SCUS to a joint UC and BF optimization scheme. This scheme can then be integrated in OPT-JSPA, $\varepsilon$-JSPA and GRAD-JSPA, while preserving their performance guarantees. Although optimal joint UC and BF remains a difficult open problem, existing heuristics can be adopted instead. For example, the schemes of paper [32] have shown to outperform classical (OMA-based) MIMO systems

and other multi-antenna NOMA algorithms by simulations.

## VIII. Conclusion

In this work, we investigate the WSR maximization in MC-NOMA with cellular power constraint. We improve the complexity of the single-carrier power control (SCPC) and user selection (SCUS) procedures using precomputation. These improved schemes are denoted by i-SCPC and i-SCUS. We develop three algorithms to solve the JSPA problem, based on i-SCPC and i-SCUS. Firstly, Opt-JSPA gives optimal results with lower complexity than current state-of-the-art optimal schemes, i.e., TSDP [11] and monotonic optimization [18]. Secondly, $\varepsilon$-JSPA is a FPTAS. It achieves a controllable and tight trade-off between approximation guarantee and complexity. Opt-JSPA and $\varepsilon$-JSPA are both suitable for performance benchmarking. Finally, Grad-JSPA is a heuristic. We show by simulation that it has near-optimal WSR with low and practical complexity.

## References


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

[2] V. W. Wong, R. Schober, D. W. K. Ng, and L.-C. Wang, *Key technologies for 5G wireless systems*. Cambridge university press, 2017.

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

[4] 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.

[5] 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.

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

[7] 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.

[8] 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.

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

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

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

[17] M. F. Hanif, Z. Ding, T. Ratnarajah, and G. K. Karagiannidis, "A minorization-maximization method for optimizing sum rate in the downlink of non-orthogonal multiple access systems," *IEEE Trans. Signal Process.*, vol. 64, no. 1, pp. 76–88, 2016.

[18] 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.

[19] L. Salaün, M. Coupechoux, and C. S. Chen, "Weighted sum-rate maximization in multi-carrier NOMA with cellular power constraint," in *IEEE INFOCOM*, 2019, pp. 451–459.

[20] H. Kellerer, U. Pferschy, and D. Pisinger, *Knapsack Problems*. Berlin Heidelberg: Springer-Verlag, 2004.

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

[22] Y. Nesterov, "Introductory lectures on convex programming volume I: Basic course," *Lecture notes*, 1998.

[23] M. R. Garey and D. S. Johnson, *Computers and Intractability*. W. H. Freeman New York, 2002, vol. 29.

[24] V. V. Vazirani, *Approximation Algorithms*. Springer Science & Business Media, 2013.

[25] A. Tarek, "Multi-key binary search and the related performance," in *Proc. Amer. Conf. on Appl. Math.*, ser. MATH'08. World Scientific and Engineering Academy and Society (WSEAS), 2008, pp. 104–109.

[26] GreenTouch, *Mobile communications WG architecture doc2: Reference scenarios*, May 2013.

[27] 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.

[28] J. Choi, "Joint rate and power allocation for NOMA with statistical CSI," *IEEE Trans. Commun.*, vol. 65, no. 10, pp. 4519–4528, 2017.

[29] 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.

[30] S. R. Islam, M. Zeng, O. A. Dobre, and K.-S. Kwak, "Resource allocation for downlink NOMA systems: Key techniques and open issues," *IEEE Wireless Commun.*, vol. 25, no. 2, pp. 40–47, 2018.

[31] Y. Sun, D. W. K. Ng, and R. Schober, "Optimal resource allocation for multicarrier MISO-NOMA systems," in *IEEE Int. Conf. Commun. (ICC)*, 2017.

[32] S. Ali, E. Hossain, and D. I. Kim, "Non-orthogonal multiple access (NOMA) for downlink multiuser MIMO systems: User clustering, beamforming, and power allocation," *IEEE access*, vol. 5, pp. 565–577, 2016.


## Appendix

We first provide in Lemma 12 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 in Appendices C and G.

**Lemma 12.**
*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 $x_{1_n}^n, \ldots, x_{i_n}^n$ 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., $\bar{P}^n \geq x_{1_n}^n \geq \cdots \geq x_{i_n}^n \geq 0$. $x_{1_n}^n, \ldots, x_{i_n}^n$ can be partitioned into sequences of consecutive terms with the same value. That is, sequences of the form $x_{q_n}^n, \ldots, x_{q_n'}^n$, 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 $x_{q_n}^n, \ldots, x_{q_n'}^n$ be a sequence of consecutive terms with the same value, as defined in Lemma 12. 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 $x_{q_n}^n < \textsc{Argmax}\tilde{f}\left(q, q', \mathcal{I}^n, \mathcal{U}_n', \bar{P}^n\right)$ and $q > 1$. Let $y_{1_n}^n, \ldots, y_{i_n}^n$ be an allocation defined as:

$$y_{l_n}^n \triangleq \begin{cases} \min\{x_{(q-1)_n}^n, x^*\}, & \text{if } q \leq l \leq q', \\ x_{l_n}^n, & \text{otherwise.} \end{cases} \tag{6}$$

We have the following inequalities:

$$\sum_{l=1}^{i} \tilde{f}_{l,l}^{n}\left(y_{l_n}^{n}\right) = \sum_{l\notin\{q,\ldots,q'\}} \tilde{f}_{l,l}^{n}\left(x_{l_n}^{n}\right) + \tilde{f}_{q,q'}^{n}\left(y_{l_n}^{n}\right), \tag{7}$$

$$> \sum_{l\notin\{q,\ldots,q'\}} \tilde{f}_{l,l}^{n}\left(x_{l_n}^{n}\right) + \tilde{f}_{q,q'}^{n}\left(x_{l_n}^{n}\right) = \sum_{l=1}^{i} \tilde{f}_{l,l}^{n}\left(x_{l_n}^{n}\right). \tag{8}$$

Equality (7) comes from the definition in (6). According to Lemma 2, $\tilde{f}_{q,q'}^{n}$ is increasing on $[0, x^*]$, which implies inequality (8). In summary, $y_{1_n}^{n}, \ldots, y_{i_n}^{n}$ satisfies $C2'$–$3'$ by its definition in (6), and it achieves greater value of $\sum_{l=1}^{i} \tilde{f}_{l,l}^{n}$ than $x_{1_n}^{n}, \ldots, x_{i_n}^{n}$. This is a contradiction, therefore it must be that

$$x_{q_n}^{n} \geq \textsc{Argmax}\tilde{f}\left(q, q', \mathcal{I}^n, \mathcal{U}_n', \bar{P}^n\right). \tag{9}$$

If $q = 1$, the same reasoning can be applied by replacing $\min\{x_{(q-1)_n}^{n}, x^*\}$ by $\bar{P}^n$ in Eqn. (6). We can perform a similar proof by contradiction on the case $x_{q_n}^{n} > \textsc{Argmax}\tilde{f}\left(q, q', \mathcal{I}^n, \mathcal{U}_n', \bar{P}^n\right)$ to deduce that:

$$x_{q_n}^{n} \leq \textsc{Argmax}\tilde{f}\left(q, q', \mathcal{I}^n, \mathcal{U}_n', \bar{P}^n\right). \tag{10}$$

The desired result follows from (9) and (10). □

### A. Proof of Lemma 1

The objective of $\mathcal{P}$ 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 in (2). 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 (3), we derive the equivalent problem $\mathcal{P}'$. 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. Proof of Lemma 2

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

$${f_{1,i}^{n}}'(x) = \frac{W_n w_{\pi_n(i)}}{\left(x + \tilde{\eta}_{\pi_n(i)}^{n}\right)\ln(2)}, \tag{11}$$

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:

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

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

We know that $\tilde{\eta}_{\pi_n(j-1)}^{n} \geq \tilde{\eta}_{\pi_n(i)}^{n}$ by construction of the optimal decoding order in Eqn. (1). If, in addition, we have $w_{\pi_n(i)} \geq w_{\pi_n(j-1)}$, then ${f_{j,i}^{n}}'(x) \geq 0$ and ${f_{j,i}^{n}}''(x) \leq 0$ for all $x \geq 0$. We deduce that $f_{j,i}^{n}$ is increasing and concave. This proves the first point of Lemma 2. Now suppose that $w_{\pi_n(i)} < w_{\pi_n(j-1)}$ instead. Values $c_1$ and $c_2$ defined in Lemma 2 are the unique roots of the first and second derivatives, i.e., ${f_{j,i}^{n}}'(c_1) = 0$ and ${f_{j,i}^{n}}''(c_2) = 0$. ${f_{j,i}^{n}}'$ is positive on $\left(-\tilde{\eta}_{\pi_n(j-1)}, c_1\right)$ and negative on $(c_1, \infty)$. This implies that $f_{j,i}^{n}$ is unimodal and has a unique global maximum at $c_1$ for $x > 0$. Similarly, ${f_{j,i}^{n}}''$ is negative on $\left(-\tilde{\eta}_{\pi_n(j-1)}, c_2\right)$ and positive on $(c_2, \infty)$. Therefore, $f_{j,i}^{n}$ is concave before $c_2$ and convex after $c_2$. This proves the second point of Lemma 2. □

### C. Proof of Theorem 3

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 $\sum_{i=1}^{|\mathcal{U}_n'|} i = O\left(|\mathcal{U}_n'|^2\right) = O\left(M^2\right)$.

***Optimality analysis:*** Without loss of generality, we can suppose that the $x_{i_n}^{n}$'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:

$H_1(i)$: $\sum_{l=1}^{i} \tilde{f}_{l,l}^{n}$ is maximized by $x_{1_n}^{n}, \ldots, x_{i_n}^{n}$,
$H_2(i)$: $C2'$–$3'$ is satisfied, i.e., $\bar{P}^n \geq x_{1_n}^{n} \geq \cdots \geq x_{i_n}^{n} \geq 0$.

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

***Inductive step:*** Assume that $x_{1_n}^{n}(i-1), \ldots, x_{(i-1)_n}^{n}(i-1)$ 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 $x_{1_n}^{n}, \ldots, x_{i_n}^{n}$. We consider two cases:

i) We first suppose that:

$$x^* = \textsc{Argmax}\tilde{f}(i, i, \mathcal{I}^n, \bar{P}^n) \leq x_{(i-1)_n}^{n}(i-1). \tag{12}$$

In this case, Algorithm 2 sets $x_{i_n}^{n} = x^*$ and $x_{l_n}^{n} = x_{l_n}^{n}(i-1)$, for all $l < i$. The induction hypothesis $H_2(i-1)$ states that $\bar{P}^n \geq x_{1_n}^{n} \geq \cdots \geq x_{(i-1)_n}^{n} \geq 0$. By

taking into account Eqn. (12), 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:

$$x^* = \text{ARGMAX}\tilde{f}(i, i, \mathcal{I}^n, \bar{P}^n) > x^n_{(i-1)_n}(i-1). \quad (13)$$

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 $x^n_{j_n}(i-1) \geq \text{ARGMAX}\tilde{f}(j+1, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n)$ 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 2. 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 $x^{*\prime} = \text{ARGMAX}\tilde{f}(k, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n)$. According to Lemma 2, $\tilde{f}^n_{k,i}$ is increasing on $[0, x^{*\prime}]$. Therefore, we can improve the objective function by setting $x^n_{k_n}, \ldots, x^n_{i_n} \leftarrow x^n_{(k-1)_n}$. 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 $\text{ARGMAX}\tilde{f}(j+1, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n) \leq x^n_{j_n}(i-1)$, which can be treated as in case i). Hence, variables $x^n_{(j+1)_n}, \ldots, x^n_{i_n}$ are set equal to $\text{ARGMAX}\tilde{f}(j+1, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n)$ at line 10, and it satisfies $H_1(i)$ and $H_2(i)$.

We proved that, in both cases i) and ii), the allocation $x^n_{1_n}, \ldots, x^n_{i_n}$ 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 $\mathcal{P}'_{SCPC}(n)$, which concludes the proof. □

### D. Proof of Theorem 4

***Optimality analysis:*** Let $x^n_{1_n}, \ldots, x^n_{|\mathcal{U}'_n|_n}$ 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 SCPC$\big(\mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\big)$, the value $\text{ARGMAX}\tilde{f}\big(j, i, \mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\big)$ can be replaced by $\min\{\text{ARGMAX}\tilde{f}(j, i, \mathcal{I}^n, \mathcal{U}'_n, P_{max}), \bar{P}^n\}$, since they are equal by definition. One can show, by mathematical induction on $i_n$, that the function SCPC$\big(\mathcal{I}^n, \mathcal{U}'_n, \bar{P}^n\big)$ returns $\min\{x^n_{1_n}, \bar{P}^n\}, \ldots, \min\{x^n_{|\mathcal{U}'_n|_n}, \bar{P}^n\}$. Therefore, the latter allocation is also optimal.

***Complexity analysis:*** The initialization consists in running SCPC, with complexity $O\big(M^2\big)$ (see Theorem 3). Each subsequent evaluation requires to compute $\min\{x^n_{i_n}, \bar{P}^n\}$, for $i \in \{1, \ldots, |\mathcal{U}'_n|\}$, with complexity $O(M)$. □

### E. Proof of Theorem 5

***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 $M\sum_{i=1}^{K-1}(i) = O\big(MK^2\big)$ iterations. Each iteration has a constant number of operations. Thus, the overall worst case computational complexity is $O\big(MK^2\big)$.

***Optimality analysis:*** We will prove by induction that at any iteration $m \in \{0, \ldots, M\}$, $j \in \{1, \ldots, K\}$ and $i \geq j$ of Algorithm 4, the construction of $V[m, j, i]$ is the optimal value of problem $\mathcal{P}'_{SC}[m, j, i]$. It follows directly that $V[M, 1, 1]$ is the optimal value of $\mathcal{P}'_{SC}(n)$.

***Basis:*** For $m = 0$, no user can be active due to constraint $C4'$. Thus, $V[0, j, i] = f^n_{j,K}(0)$ 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 IV-C.

***Inductive step:*** Let $m \in \{1, \ldots, M\}$ and $1 \leq j \leq i \leq K-1$. Assume that $V[m', j', i']$ is the optimal value of $\mathcal{P}'_{SC}[m', j', i']$ for any $m' \leq m$, $j' \geq j$ and $i' > i$. We denote the optimal solution of problem $\mathcal{P}'_{SC}[m, j, i]$ by $x^n_j, \ldots, x^n_K$. Let $v_{act}$ (resp. $v_{inact}$) be the optimal value of $\mathcal{P}'_{SC}[m, j, i]$, given that user $i$ is active (resp. inactive). Let $x^{n*}_{(i+1)_n} = X[m-1, i+1, i+1]$ be the optimal value of $x^n_{(i+1)_n}$ in $\mathcal{P}'_{SC}[m-1, i+1, i+1]$. If $x^* \leq x^{n*}_{(i+1)_n}$, then we can prove as in case ii) of Appendix C, 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 $\mathcal{P}'_{SC}[m, j, i]$ is achieved when user $i$ is inactive, then we have $x^n_i = x^n_{i+1}$ by definition of $\mathcal{U}'_n$. It follows from $C5'$ that $x^n_j = \cdots = x^n_{i+1}$. 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 $x^* > x^{n*}_{(i+1)_n}$ satisfies $C3'$, and the objective is separable, the optimal is obtained when maximizing independently $f^n_{j,i}$ and $\sum_{l=i+1}^{K} f^n_l$ with $m-1$ active users. That is, $V[m, j, i] = v_{act} \triangleq f^n_{j,i}(x^*) + V[m-1, i+1, i+1]$, where $x^* = \text{ARGMAX}f\big(j, i, \mathcal{I}^n, \bar{P}^n\big)$ in line 15.

Hence, $V[m, j, i]$, as computed in (5), corresponds to the optimal of $\mathcal{P}'_{SC}[m, j, i]$.

We derive, by mathematical induction, that $V[M, 1, 1]$ is the optimal value of $\mathcal{P}'_{SC}[M, 1, 1] = \mathcal{P}'_{SC}(n)$. The corresponding optimal allocation $\boldsymbol{x}^n$ is retrieved in lines 28 to 35. □

### F. Proof of Theorem 6

***Optimality analysis:*** Let $y^n_1, \ldots, y^n_K$ be the optimal solution of $\mathcal{P}'_{SC}(n)$ subject to a power constraint $\bar{P}^n$. Let $i \in \{1, \ldots, K\}$ be the unique index such that $y^n_1 = \cdots = y^n_i$ and $y^n_i > y^n_{i+1}$. We know from Lemma 2 that $y^n_1 = \cdots = y^n_i = \bar{P}^n$. Therefore, $y^n_{i+1}, \ldots, y^n_K$ are all strictly less than $\bar{P}^n$. Let $x^n_1, \ldots, x^n_K$ be the optimal solution of $\mathcal{P}'_{SC}[M, 1, i]$ in the execution of SCUS$(\mathcal{I}^n, M, P_{max})$, i.e., subject to a power budget $P_{max}$. According to Lemma 2, $x^n_1 = \cdots = x^n_i = P_{max}$. We deduce from $f$'s unimodality in Lemma 2, that $y^n_{i+1}, \ldots, y^n_K$ is the optimal solution of $\mathcal{P}'_{SC}[M, i+1, i+1]$ given any power budget no less than $\bar{P}^n$. In particular, we have $x^n_l = y^n_l$, for all $l \in \{i+1, \ldots, K\}$. Hence, $x^n_1, \ldots, x^n_K$ 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 $1 \le l \le |\mathcal{U}_n'|$.

We proved above that, for any $\bar{P}^n \le P_{max}$, 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.

***Complexity analysis:*** The initialization consists in running SCUS, with complexity $O\big(MK^2\big)$ (see Theorem 5). Each subsequent evaluation has complexity $O(MK)$. Indeed, there are $K$ active users sets $\mathcal{U}_n'$ in *collection*, one for each solution of $\mathcal{P}_{SC}'[M, 1, i]$, 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 $O(|\mathcal{U}_n'|) = O(M)$. □

### G. Proofs of Lemma 7 and Theorem 8

Let $x_1^n, \dots, x_K^n$ be the output of i-SCUS$\big(\bar{P}^n\big)$, and $\mathcal{U}_n'$ the corresponding active users set. For $i \in \{1, \dots, K\}$, there exists $q \le i$ and $q' \ge i$, such that $x_{q_n}^n = \dots = x_{q_n'}^n = \text{ARGMAX}\tilde{f}\big(q, q', \mathcal{I}^n, \mathcal{U}_n', \bar{P}^n\big)$, according to Lemma 12. We have:

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

We consider it as a function of $\bar{P}^n$. Its left derivative at $\bar{P}^n = x_{q_n}^n$ is 0, according to Lemma 2. Its right derivative at $\bar{P}^n = x_{q_n}^n$ is 0, as it is constant for $\bar{P}^n > x_{q_n}^n$. Hence, $\tilde{f}_{q,q'}^n(\mathcal{U}_n', \min\{x_{q_n}, \cdot\})$ 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. (11) of Appendix B as follows:

$$F^{n\prime}\big(\mathcal{U}_n', \bar{P}^n\big) = f_{1,l}^{n\prime}\big(\bar{P}^n\big) = \frac{W_n w_{\pi_n(l)}}{\left(\bar{P}^n + \tilde{\eta}_{\pi_n(l)}^n\right)\ln(2)}. \quad (14)$$

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. (14) is the left derivative of $F^n$. This proves Lemma 7.

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

$$\beta \le F^{n\prime\prime}\big(\bar{P}^n\big) = \frac{-W_n w_{\pi_n(l)}}{\left(\bar{P}^n + \tilde{\eta}_{\pi_n(l)}^n\right)^2 \ln(2)} \le \alpha < 0, \quad (15)$$

where $\beta$ and $\alpha$ 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 F shows that there are $K$ such intervals. Eqn. (15) implies that $F^n$ is piece-wise twice differentiable, $\alpha$-strongly concave and $\beta$-smooth. Therefore, the projected gradient descent on the simplex $\mathcal{F}$ converges in $O(\log(1/\xi))$ iterations, according to [22, Section 2.2.4]. This proves Theorem 8. □

### H. Proof of Theorem 9

Let $\bar{P}^{n*}$ be the optimal power budget of $\mathcal{P}_{MC}'$ on subcarrier $n \in \mathcal{N}$. The power budget after discretization with step size $\delta$ is denoted by $a_n^* \triangleq \lfloor \bar{P}^{n*}/\delta \rfloor \delta$. We have:

$$F^* - F_{MCKP}^* \le \sum_{n\in\mathcal{N}} \left(F^n\left(\bar{P}^{n*}\right) - F^n(a_n^*)\right), \quad (16)$$

$$\le \sum_{n\in\mathcal{N}} \max_{\mathcal{U}_n'}\{F^{n\prime}(\mathcal{U}_n', \bar{P}^{n*})\} \times \left(\bar{P}^{n*} - a_n^*\right), \quad (17)$$

$$\le \delta \sum_{n\in\mathcal{N}} \max_{k\in\mathcal{K}} \left\{ \frac{W_n w_{\pi_n(k)}}{\left(\bar{P}^{n*} + \tilde{\eta}_{\pi_n(k)}^n\right)\ln(2)} \right\}. \quad (18)$$

Inequality (16) comes from the definition of $F^*$ and the fact that $F_{MCKP}^* \ge \sum_{n\in\mathcal{N}} F^n(a_n^*)$, as $F_{MCKP}^*$ is the optimal discrete solution with step size $\delta$. We know from Appendix G that $F^n(\bar{P}^n) = \max_{\mathcal{U}_n'}\{F^n(\mathcal{U}_n', \bar{P}^n)\}$, and that $F^n\big(\mathcal{U}_n', \bar{P}^n\big)$ is twice differentiable and concave, for any $\mathcal{U}_n'$ and $n \in \mathcal{N}$. Hence, $F^n$ lies below the maximum slope tangent among the tangents of $F^n(\mathcal{U}_n', \bar{P}^{n*})$, for all $\mathcal{U}_n'$. This implies inequality (17). We obtain (18) by applying Eqn. (14), and the fact that $\bar{P}^{n*} - a_n^* \le \delta$ by construction. □

### I. Proof of Theorem 10

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 $l = 0, \dots, J$. For $n \in \mathcal{N}$ and $l = 0, \dots, J$, the recurrence relation is:

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

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

***Optimality analysis:*** Reference [20] proves that dynamic programming by weights is optimal for MCKP. Since problems $\mathcal{P}_{MC}'$ and MCKP are equivalent, the proposed OPT-JSPA based on dynamic programming by weights is also optimal for $\mathcal{P}_{MC}'$.

***Complexity analysis:*** In Algorithm 7, we first transform $\mathcal{P}_{MC}'$ to problem MCKP: from line 1 to 5, every item's profit $c_{n,l}$ is computed using i-SCUS in $O\big(NMK^2 + JNMK\big)$. Then, we perform dynamic programming by weights at lines 6-7. According to [20], its complexity is $O\big(J^2N\big)$, which is the number of items $N(J+1)$ multiplied by the number of possible power values $J+1$. Therefore, the overall complexity is $O\big(NMK^2 + JNMK + J^2N\big)$. □

### J. 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 13 that will be used for the analysis of the estimation procedure.

**Lemma 13** (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 $\mathcal{P}'_{MC}$ 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 $\mathcal{P}'_{MC}$ 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 the sublinearity of $F^*_{MCKP}$. 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 2. 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 $\mathcal{U}'_n$, $\mathcal{P}'_{SCPC}(n)$ 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 $\mathcal{P}'_{SC}(n)$ 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 $\mathcal{P}'_{MC}$ 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 [20, 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 $n \in \mathcal{N}$, $l \in \{0, \ldots, 2N\}$, be an optimal solution of this problem. In addition, we denote by $y'_{n,l}$ for $n \in \mathcal{N}$, $l \in \{0, \ldots, 2N\}$, a 1/2-approximation given by the Dyer-Zemel algorithm. On the one hand, we have:

$$\sum_{n\in\mathcal{N}}\sum_{l=1}^{2N} c_{n,l}y'_{n,l} \geq \frac{1}{2}\sum_{n\in\mathcal{N}}\sum_{l=1}^{2N} c_{n,l}y^*_{n,l}, \tag{19}$$

$$\geq \frac{1}{2}\sum_{n\in\mathcal{N}} F^n\left(\left\lceil \frac{\bar{P}^{n*}}{\lfloor J/N \rfloor \delta}\right\rceil \left\lfloor \frac{J}{N}\right\rfloor \delta\right), \tag{20}$$

$$\geq \frac{1}{2}\sum_{n\in\mathcal{N}} F^n\big(\bar{P}^{n*}\big) = \frac{1}{2}F^*_{MCKP}(P_{max}), \tag{21}$$

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. (19). The right term of Eqn. (20) 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 (20). We derive Eqn. (21) from inequality (e) and the monotonicity of $F^n$ (see Lemma 13).

We have, on the other hand:

$$\sum_{n\in\mathcal{N}}\sum_{l=1}^{2N} c_{n,l}y'_{n,l} \leq \sum_{n\in\mathcal{N}}\sum_{l=1}^{2N} c_{n,l}y^*_{n,l}, \tag{22}$$

$$\leq F^*_{MCKP}(2P_{max}), \tag{23}$$

$$\leq 2F^*_{MCKP}(P_{max}). \tag{24}$$

The optimality of $y^*_{n,l}$ implies Eqn. (22). Eqn. (23) comes from the fact that the items of MCKP' is a subset of MCKP items, given a budget $2P_{max}$. Eqn. (24) follows from the sublinearity of $F^*_{MCKP}$ (see Lemma 13).

Let $U \triangleq 2\sum_{n\in\mathcal{N}}\sum_{l=1}^{2N} c_{n,l}y'_{n,l}$. We derive from inequalities (21) and (24) the desired approximation bound:

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

### K. Proof of Theorem 11

***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 $O\big(NMK^2\big)$.

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

***iii. Lines 2-4:*** Each $L_n$, for $n \in \mathcal{N}$, is obtained by multi-key binary search [25]. 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 obtain $O\big(\min\{\log(J)\, N^2MK/\varepsilon, JNMK\}\big)$.

***iv. Lines 5-6:*** Let us first briefly explain the dynamic programming by profits [20]. 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:

$$Y[n,q] = \min_{l \in L_n} \begin{cases} Y\left[n-1, q - \left\lfloor \frac{4c_{n,l}N}{\varepsilon U} \right\rfloor\right] + a_{n,l}, \\ \qquad\qquad \text{if } \frac{q \cdot \varepsilon U}{4N} \geq c_{n,l}, \\ +\infty, \text{ otherwise.} \end{cases} \tag{25}$$

This recursion has complexity $O\big(\min\{N^3/\varepsilon^2, J^2N\}\big)$, 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. (25), $|L_n| = \min\{4N/\varepsilon, J\}$.

***Approximation analysis:*** As proved in [20, 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, $\varepsilon$-JSPA achieves $\varepsilon$-approximation with polynomial complexity in $1/\varepsilon$ and $N$, $M$, $K$. Therefore, $\varepsilon$-JSPA is a FPTAS, which concludes the proof. $\square$