






Decentralized Joint Pilot and Data Power Control Based on Deep Reinforcement Learning for the Uplink of Cell-Free Systems

Iran Mesquita Braga Jr. , Roberto Pinto Antonioli , Gábor Fodor , *Senior Member, IEEE*,
Yuri C. B. Silva , and Walter C. Freitas Jr. 

Abstract—While the problem of jointly controlling the pilot-and-data power in cell-based systems has been extensively studied, this problem is difficult to solve in cell-free systems due to two reasons. First, both the large- and small-scale fading are markedly different between a served user and the multiple serving access points. Second, due to the user-centric architecture, there is a need for decentralized algorithms that scale well in the cell-free environment. In this work, we study the impact of joint pilot-and-data power control and receive filter design in the uplink of cell-free systems. The problem is formulated as optimization tasks considering two different objectives: 1) maximization of the minimum spectral efficiency (SE) and 2) maximization of the total SE. Since these problems are non-convex, we resort to successive convex approximation and geometric programming to obtain a local optimal centralized solution for benchmarking purposes. We also propose a decentralized solution based on actor-critic deep reinforcement learning, in which each user acts as an agent to locally obtain the best policy relying on minimum information exchange. Practical signaling aspects are provided for such a decentralized solution. Finally, numerical results indicate that the decentralized solution performs very close to the centralized one and outperforms state-of-the-art algorithms in terms of minimum SE and total system SE.

Index Terms—Cell-free, pilot-and-data power control, successive convex approximation, geometric programming, deep reinforcement learning.

I. INTRODUCTION

CURRENTLY, the cell-free concept has garnered several efforts in the industry and academia and is considered

Manuscript received 25 August 2021; revised 11 April 2022 and 25 July 2022; accepted 25 September 2022. Date of publication 4 October 2022; date of current version 16 January 2023. The work of Gábor Fodor was supported by the Celtic Project 6G for Connected Sky, Project ID: C2021/1-9. This work was supported in part by Ericsson Research, Technical Cooperation Contract UFC.48, in part by the Brazilian National Council for Scientific and Technological Development (CNPq), in part by FUNCAP, and in part by CAPES/PRINT under Grant 88887.311965/2018-00. The review of this article was coordinated by Dr. Antonella Molinaro. (*Corresponding author: Iran Mesquita Braga Junior.*)

Iran Mesquita Braga, Yuri C. B. Silva, and Walter C. Freitas are with the Wireless Telecom Research Group (GTEL), Federal University of Ceará, Fortaleza 60455-760, Brazil (e-mail: iran@gtel.ufc.br; yuri@gtel.ufc.br; wal-ter@gtel.ufc.br).

Roberto Pinto Antonioli is with the Wireless Telecom Research Group (GTEL), Federal University of Ceará, Fortaleza 60455-760, Brazil, and also with the Instituto Atlântico, Fortaleza 60811-341, Brazil (e-mail: antonioli@gtel.ufc.br).

Gábor Fodor is with the Ericsson Research, 16480 Stockholm, Sweden, and also with the Division of Decision and Control, KTH Royal Institute of Technology, 11428 Stockholm, Sweden (e-mail: gabor.fodor@ericsson.com).

Digital Object Identifier 10.1109/TVT.2022.3211908

a promising technology for beyond fifth generation (5G) networks. It consists in an architecture in which a large number of distributed access points (APs), connected to a central processing unit (CPU) via high capacity fronthaul, serve a group of users in a large service area using the same time/frequency resource. Hence, it can offer a higher coverage probability compared to traditional cellular networks and thus improves the service quality in a given geographical area [1].

In cell-free systems the communication burdens on the fronthaul increase significantly, as all signal processing is performed at the CPU [2]. To overcome this issue, a commonly used approach consists in performing channel estimation and data detection at each AP and, next, the data estimates are passed to the CPU for final decoding. Moreover, the system performance can be improved by designing the receive filter coefficients at the CPU by using only the channel statistics [3].

Also, in general, cell-free systems employ uplink pilots for channel estimation. Ideally, the pilot sequence assigned to a given user should be mutually orthogonal to other users' pilot sequences. However, this is not always possible, especially when the coherence interval is short and/or the number of users is large [4]. Then, non-orthogonal pilot sequences have to be employed by the users, which causes pilot contamination and significantly degrades the system performance [5]. Also, due to pilot contamination, improving the pilot signal-to-interference-plus-noise ratio (SINR) for one user may cause pilot SINR degradation for users using the same pilot. Moreover, the trade-off between pilot-and-data power allocation in cell-free systems is much more complicated than in cell-based systems because each serving AP must estimate its wireless channel to its served users based on the same uplink pilot signal, and, in general, a set of users is served by a different set of APs. Thus, as highlighted in [6], further studies on joint pilot-and-data power control (JPDPC) in cell-free systems that quantify the inherent trade-off between pilot-and-data power as well as between spectral efficiency and fairness are needed.

Machine learning has made great strides in several areas, including wireless communications [7]. Specifically, deep reinforcement learning (DRL) was shown to be a promising and powerful technique for improving the performance of wireless communications. Indeed, DRL is reward-based, which allows obtaining solutions for convex and non-convex problems without training data sets. In addition, a small number of simple

operations are needed to obtain an output, thus, in general, DRL has low computational complexity. Another advantage is its robustness to incomplete and/or imperfect information [8], [9]. Thus, multiple agents can act on the environment and locally obtain the best optimal policy with minimum or without information exchange among each other. In other words, it allows the development of decentralized solutions with reduced signaling overhead.

A. Related Works and Main Contributions

The impact of JPDPC has been largely studied in the uplink of traditional cellular systems. In [10] the power minimization problem subject to target SINR constraints for a multi-cell system was studied, where a JPDPC scheme using geometric programming (GP) was proposed. In [11] the weighted max-min fairness and weighted sum spectral efficiency (SE) problems were studied in the uplink of a single-cell system using maximum ratio combining (MRC) and zero-forcing (ZF) detection. Moreover, the authors showed that a JPDPC is specially important for cell-edge users. On the other hand, in [12] the authors derived a closed-form expression for the mean squared error (MSE) of the uplink received data symbols in a multi-cell system considering pilot contamination and proposed two decentralized solutions based on non-cooperative game to minimize the sum MSE.

Initial studies on cell-free systems, such as those in [5], [13], [14], focused only on the data power control by assuming that the pilot signals are transmitted using a fixed power. However, such solutions incur in poor channel estimations since users in worse channel conditions can be affected by users with better channel conditions. Moreover, reusing pilot sequences over the coverage area causes pilot contamination. Thus, the system performance can be significantly degraded [5]. To deal with this issue, the authors in [15] proposed a pilot power control scheme to minimize the maximum channel estimation error among the users and used the data power allocation scheme proposed in [5] to solve a max-min fairness problem. The authors in that paper, however, did not consider JPDPC. Such a problem was considered in [16], where the max-min fairness problem was formulated subject to an energy budget per coherence interval. The authors also proposed a solution based on GP. However, the APs are equipped with a single antenna and the pilot contamination effects were ignored. Moreover, all APs are used to serve all users, which is shown to be suboptimal in [14]. The impact of JPDPC in cell-free systems with user-centric clustering, pilot contamination and multi-antenna AP was considered in [17] but only centralized solutions were proposed.

Furthermore, in cell-free systems, the system performance can be improved by optimizing the receive filter coefficients at the CPU using only channel statistics. In [18] an asymptotic approximation for the SINR of the minimum mean squared error (MMSE) receiver was derived as a function of large scale fading coefficients only. Moreover, the authors derived the receive filter coefficients that maximize the SINR of each user and used the bisection method for solving the max-min fairness power allocation problem. In [19] the authors investigated the max-min fairness problem and proposed an alternating optimization method

in which the receive filter coefficients design was formulated as a generalized eigenvalue problem, and the power control problem was solved using GP. Due to the high-complexity of the GP method, the authors in [20] applied a smoothing technique in combination with an accelerated projected gradient method to solve the power allocation problem. Although these works have considered the receive filter coefficients design, the JPDPC was ignored by them. Indeed, as far as we know there is no work in the literature considering JPDPC and receive filter coefficients design in cell-free systems.

Recently, some works have focused on machine learning for power control in cell-free systems. In [21] a data power control based on supervised learning was proposed for solving the max-min fairness and sum SE problems. However, supervised learning requires a large training data set which is not always available. To deal with this issue, in [22] and [23] unsupervised learning was proposed, but they also focused on data power control and only centralized solutions were provided. DRL was also considered by other works [24], [25], [26], however, the JPDPC and receive filter coefficients design was ignored.

It is intuitively clear that works on DRL, and especially on distributed DRL, still need to be exploited in cell-free systems. Therefore, in this paper, we investigate the impact of JPDPC and receive filter coefficients design in cell-free systems and, motivated by the benefits of DRL, we propose a decentralized solution based on multiple agent DRL. The main contributions of this paper can be summarized as follows:

- 1) Investigation of the JPDPC and receive filter coefficients design addressing two different objectives: i) max-min fairness and ii) maximization of the sum SE, in which we consider pilot contamination and multi-antenna APs. Moreover, we also assume a user-centric approach in which only a subset of the APs simultaneously serve each user;
- 2) To handling the non-convexity of the formulated optimization problem, we develop a centralized solution based on successive convex approximation (SCA) and GP to find local optimal solutions of the original problems for benchmarking purposes;
- 3) Although joint pilot-and-data power control is decisive for the system-wide spectral and energy efficiency, its complexity in cell-free systems is prohibitive in practice. Therefore, in this paper we seek DRL-based approaches that can be implemented by realistic signaling protocols. In this context, a novel decentralized solution based on actor-critic DRL, where we also present a signaling scheme for deployments in practical cell-free systems;
- 4) Performance evaluation by means of simulations, where we compare the proposed solutions with state-of-the-art algorithms considering different scenarios and show that the proposed decentralized solution performs very close to the centralized one, outperforming the state-of-the-art algorithms in terms of minimum spectral efficiency and total system throughput.

Organization: The remainder of the paper is organized as follows. Section II introduces the system model. Then, the JPDPC and receive filter design problems are presented in Section III. The proposed centralized and decentralized

solutions are described in Sections IV and V, respectively, with the latter also discussing the involved signaling aspects. Section VI provides the numerical results along with discussions and, finally, Section VII highlights the main conclusions, as well as perspectives for future works.

Notation: Throughout the paper, matrices and vectors are presented by boldface upper and lower case letters, respectively. \mathbf{X}^T , \mathbf{X}^H and \mathbf{X}^{-1} stand, respectively, for transpose, Hermitian and inverse of a matrix \mathbf{X} . $\{x_i\}_{\forall i}$ denotes the set of elements x_i for the values of i denoted by the subscript expression. \mathbf{I} is the identity matrix. Expected value of a random variable is denoted by $\mathbb{E}[\cdot]$. In addition, $|\mathcal{X}|$ denotes the cardinality of set \mathcal{X} , $x \sim \mathcal{CN}(0, a)$ represents a zero-mean circularly symmetric complex Gaussian random variable with variance a and a random variable x that follows the Beta distributions with shape parameters a and b is given by $x \sim \mathcal{Beta}(a, b)$.

II. SYSTEM MODEL

We consider the uplink of a cell-free system consisting of M APs, each AP equipped with N antennas, and K single-antenna users. We define \mathcal{M} and \mathcal{K} as the sets of APs and users, respectively. Furthermore, all APs are connected via fronthaul links to a CPU. This model can capture that each user k is served only by a subset of APs, denoted as \mathcal{M}_k . We remark that clustering and user association algorithms (which set of APs serve each user) are out of the scope of this paper. Such clustering algorithms have been proposed in the literature (e.g. [25], [27]), and the proposed power control schemes work with any of such clustering algorithm.

The channel vector $\mathbf{g}_{m,k} \in \mathbb{C}^{N \times 1}$ between user k and AP m is composed by the large-scale fading $\beta_{m,k}$ and the small-scale fading vector $\mathbf{h}_{m,k} \in \mathbb{C}^{N \times 1}$: $\mathbf{g}_{m,k} = \sqrt{\beta_{m,k}} \mathbf{h}_{m,k}$. Note that $\mathbf{h}_{m,k} \sim \mathcal{CN}(0, \mathbf{I}_N)$, $\forall (m, k)$ are independent and identically distributed random variables [5]. We assume that only the large scale fading coefficients, $\{\beta_{m,k}\}_{\forall (m,k)}$, are known at the CPU, as they vary slowly and can be easily estimated. Indeed, the large scale coefficients are kept constant during T coherence intervals or time slots. Also, τ_c denotes the length of the coherence interval (in samples), which is equal to the product of the coherence time and the coherence bandwidth.

A. Pilot Transmission and Channel Estimation

We assume that τ_p mutually orthogonal pilot sequences $\sqrt{\tau_p} \boldsymbol{\varphi} \in \mathbb{C}^{\tau_p \times 1}$ are used for channel estimation with $\|\boldsymbol{\varphi}\|^2 = 1$. Thus, let $\sqrt{\tau_p p_k^{(p)}} \boldsymbol{\varphi}_k$ be the pilot sequence assigned to user k , for $k = 1, \dots, K$, and $p_k^{(p)}$ is the transmit power of the pilot symbol for user k . The received pilot signal matrix $\mathbf{Y}_m^p \in \mathbb{C}^{N \times \tau_p}$ at the AP m is given by

$$\mathbf{Y}_m^p = \sum_{k=1}^K \sqrt{\tau_p p_k^{(p)}} \mathbf{g}_{m,k} \boldsymbol{\varphi}_k^H + \mathbf{N}_m^p, \quad (1)$$

where $\mathbf{N}_m^p \in \mathbb{C}^{N \times \tau_p}$ is the receiver noise with independent $\mathcal{CN}(0, \sigma^2)$ entries, in which σ^2 is the noise power. After obtaining the projection of \mathbf{Y}_m^p onto $\boldsymbol{\varphi}_k$, given by $\check{\mathbf{y}}_{m,k}^p = \mathbf{Y}_m^p \boldsymbol{\varphi}_k$, the

minimum mean squared error estimate of $\{\mathbf{g}_{m,k}\}_{\forall (m,k)}$ is [15]

$$\hat{\mathbf{g}}_{m,k} = \sqrt{\tau_p p_k^{(p)}} \beta_{m,k} \psi_{m,k}^{-1} \check{\mathbf{y}}_{m,k}^p, \quad (2)$$

where

$$\psi_{m,k} = \sum_{j=1}^K \tau_p p_j^{(p)} \beta_{m,j} |\boldsymbol{\varphi}_j^H \boldsymbol{\varphi}_k|^2 + \sigma^2. \quad (3)$$

Note that $\psi_{m,k} \mathbf{I}_N$ is the correlation matrix of $\check{\mathbf{y}}_{m,k}^p$. The estimated channel $\hat{\mathbf{g}}_{m,k}$ and the channel estimation error $\tilde{\mathbf{g}}_{m,k} = \mathbf{g}_{m,k} - \hat{\mathbf{g}}_{m,k}$ are independent vectors distributed as $\hat{\mathbf{g}}_{m,k} \sim \mathcal{CN}(\mathbf{0}, \gamma_{m,k} \mathbf{I}_N)$ and $\tilde{\mathbf{g}}_{m,k} \sim \mathcal{CN}(\mathbf{0}, c_{m,k} \mathbf{I}_N)$, where $\gamma_{m,k} = \tau_p p_k^{(p)} \beta_{m,k}^2 \psi_{m,k}^{-1}$ and $c_{m,k} = \beta_{m,k} - \gamma_{m,k}$.

B. Uplink Data Transmission and Achievable SE

We assume that $\tau_d = \tau_c - \tau_p$ symbols are used for uplink transmission. In addition, all K users simultaneously send their data on the same time-frequency resource. Thus, the received signal at the m -th AP $\mathbf{y}_m^d \in \mathbb{C}^{N \times 1}$ is modeled as

$$\mathbf{y}_m^d = \sum_{k=1}^K \sqrt{p_k^{(d)}} \mathbf{g}_{m,k} x_k + \mathbf{n}_m^d, \quad (4)$$

where $x_k \in \mathbb{C}$, with $\mathbb{E}\{|x_k|^2\} = 1$, is the transmitted data symbol by user k , $\mathbf{n}_m^d \sim \mathcal{CN}(0, \sigma^2 \mathbf{I}_N)$ is the noise on the received data signal, and $p_k^{(d)}$ is the transmit power of the data symbol.

In this work, each AP is able to perform local data detections that are passed to the CPU for final decoding. By employing MRC detection, the local estimate of x_k at AP m is given by

$$\tilde{x}_{m,k} = \hat{\mathbf{g}}_{m,k}^H \mathbf{y}_m^d = \sum_{j=1}^K \sqrt{p_j^{(d)}} \hat{\mathbf{g}}_{m,k}^H \mathbf{g}_{m,j} x_j + \hat{\mathbf{g}}_{m,k}^H \mathbf{n}_m^d. \quad (5)$$

Once the local estimates are sent to the CPU, they are multiplied by a receive filter coefficient $\omega_{m,k}$, i.e., $\hat{x}_k = \sum_{m \in \mathcal{M}_k} \omega_{m,k} \tilde{x}_{m,k}$, to obtain

$$\begin{aligned} \hat{x}_k &= \sum_{m \in \mathcal{M}_k} \sqrt{p_k^{(d)}} \omega_{m,k} \hat{\mathbf{g}}_{m,k}^H \mathbf{g}_{m,k} x_k \\ &+ \sum_{m \in \mathcal{M}_k} \omega_{m,k} \hat{\mathbf{g}}_{m,k}^H \mathbf{n}_m^d \\ &+ \sum_{\substack{j=1 \\ j \neq k}}^K \sum_{m \in \mathcal{M}_k} \sqrt{p_j^{(d)}} \omega_{m,k} \hat{\mathbf{g}}_{m,k}^H \mathbf{g}_{m,j} x_j. \end{aligned} \quad (6)$$

Note that the receive filter coefficients can be optimized to maximize the SE using only channel statistics since the CPU does not have knowledge of the channel estimates. Moreover, the achievable SE can be computed using the following lemma.¹

¹It is worth mentioning that the exact ergodic capacity of uplink multiuser channels with channel uncertainty is unknown and only a lower bound is provided here.

Lemma 1 (See [19], Theorem 1): The closed form expression for the achievable SE of user k using MRC is given by:

$$R_k = (\tau_d/\tau_c) \log_2(1 + \Upsilon_k), \quad (7)$$

where Υ_k is the SINR of user k given (8), shown at the bottom of this page.

III. PROBLEM FORMULATION

In this work, we study the maximization of a given utility function $U(R_1, \dots, R_K)$ subject to maximum energy budget constraints, for which the general problem is modeled as follows

$$\underset{p_k^{(p)}, p_k^{(d)}, \omega_{m,k}}{\text{maximize}} \quad U(R_1, \dots, R_K) \quad (9a)$$

$$\text{subject to} \quad \tau_p p_k^{(p)} + \tau_d p_k^{(d)} \leq E_{\max}, \quad \forall k, \quad (9b)$$

$$\sum_{m \in \mathcal{M}_k} |\omega_{m,k}|^2 = 1, \quad \forall k, \quad (9c)$$

$$p_k^{(p)} \geq 0, p_k^{(d)} \geq 0, \quad \forall k, \quad (9d)$$

where the pilot and data powers, as well as the receive filter coefficients, are the optimization variables, $U(\cdot)$ can be any function that is monotonically increasing in every argument and E_{\max} is the maximum energy budget.

We consider two objective functions: (1) maximization of the minimum SE (max-min SE) and (2) maximization of the sum SE (max-sum SE). The max-min SE and max-sum SE problems are two extreme cases, where the max-min SE is totally fair and the max-sum SE ignores fairness to achieve a high system throughput. Next, each problem is described in detail.

A. Maximization of the Minimum SE

The main goal of the max-min SE consists in providing a fair SE for all users. Moreover, this problem corresponds to the case in which the utility function is given as $U(R_1, \dots, R_K) = \min_k R_k$. Thus, the max-min SE problem can be written as

$$\underset{p_k^{(p)}, p_k^{(d)}, \omega_{m,k}}{\text{maximize}} \quad \min_k R_k \quad (10a)$$

$$\text{subject to} \quad (9b), (9c) \text{ and } (9d). \quad (10b)$$

Note that the coefficient τ_d/τ_c is equal for all users and $\log_2(1 + x)$ is a monotonically increasing function. Thus, we can ignore the coefficient τ_d/τ_c and remove the logarithm from the objective function. Using the epigraph form [28], we can rewrite the max-min SE problem as

$$\underset{p_k^{(p)}, p_k^{(d)}, \omega_{m,k}, \epsilon}{\text{maximize}} \quad \epsilon \quad (11a)$$

$$\text{subject to} \quad \epsilon \leq \Upsilon_k, \quad \forall k, \quad (11b)$$

$$(9b), (9c) \text{ and } (9d). \quad (11c)$$

B. Maximization of the Sum SE

The max-sum SE aims at maximizing the total system throughput. Thus, the utility function chosen for this case is $U(R_1, \dots, R_K) = \sum_{k=1}^K R_k$ and the sum SE problem can be formulated as

$$\underset{p_k^{(p)}, p_k^{(d)}, \omega_{m,k}}{\text{maximize}} \quad \sum_{k=1}^K R_k \quad (12a)$$

$$\text{subject to} \quad (9b), (9c) \text{ and } (9d). \quad (12b)$$

The coefficient τ_d/τ_c can also be ignored. Moreover, using the property of logarithm functions in which $\sum_{\forall x} \log x = \log(\prod_{\forall x} x)$ and $\log(\prod_{\forall x} (1 + x))$ is a monotonically increasing function, the max-sum SE problem can be written, in its epigraph form, as

$$\underset{p_k^{(p)}, p_k^{(d)}, \omega_{m,k}, \epsilon_k}{\text{maximize}} \quad \prod_{k=1}^K \epsilon_k \quad (13a)$$

$$\text{subject to} \quad \epsilon_k \leq (1 + \Upsilon_k), \quad \forall k, \quad (13b)$$

$$(9b), (9c) \text{ and } (9d). \quad (13c)$$

IV. PROPOSED CENTRALIZED SOLUTION BASED ON SCA AND GP

Problem (11) is non-convex [16] and it is well-known that the power control to maximize the sum SE (i.e., problem (13)) is non-polynomial time (NP)-hard, even under perfect channel knowledge [29]. Fortunately, problems (11) and (13) can be approximated as GP problems. For that, after a series of mathematical manipulations, it can be obtained (14), (15), (16), (17).

$$\sum_{m \in \mathcal{M}_k} \omega_{m,k} \gamma_{m,k} = \frac{\sum_{m \in \mathcal{M}_k} \omega_{m,k} \tau_p p_k^{(p)} \beta_{m,k}^2 \prod_{q \neq m} \psi_{q,k}}{\prod_{m \in \mathcal{M}_k} \psi_{m,k}} = \frac{\theta_k}{\lambda_k}, \quad (14)$$

$$\sum_{m \in \mathcal{M}_k} |\omega_{m,k}|^2 \gamma_{m,k} = \frac{\sum_{m \in \mathcal{M}_k} |\omega_{m,k}|^2 \tau_p p_k^{(p)} \beta_{m,k}^2 \prod_{q \neq m} \psi_{q,k}}{\prod_{m \in \mathcal{M}_k} \psi_{m,k}} = \frac{\vartheta_k}{\lambda_k}, \quad (15)$$

$$\sum_{m \in \mathcal{M}_k} |\omega_{m,k}|^2 \gamma_{m,k} \beta_{m,j}$$

$$\Upsilon_k = \frac{N p_k^{(d)} \left| \sum_{m \in \mathcal{M}_k} \omega_{m,k} \gamma_{m,k} \right|^2}{N \sum_{\substack{j=1 \\ j \neq k}}^K p_j^{(d)} \left| \sum_{m \in \mathcal{M}_k} \omega_{m,k} \gamma_{m,k} \frac{\sqrt{p_j^{(p)}} \beta_{m,j}}{\sqrt{p_k^{(p)}} \beta_{m,k}} \varphi_j^H \varphi_k \right|^2 + \sum_{j=1}^K p_j^{(d)} \sum_{m \in \mathcal{M}_k} |\omega_{m,k}|^2 \gamma_{m,k} \beta_{m,j} + \sigma^2 \sum_{m \in \mathcal{M}_k} |\omega_{m,k}|^2 \gamma_{m,k}} \quad (8)$$

$$= \frac{\sum_{m \in \mathcal{M}_k} |\omega_{m,k}|^2 \tau_p p_k^{(p)} \beta_{m,k}^2 \beta_{m,j} \prod_{q \neq m} \psi_{q,k}}{\prod_{m \in \mathcal{M}_k} \psi_{m,k}} = \frac{\xi_{k,j}}{\lambda_k}, \quad (16)$$

$$\begin{aligned} & \sum_{m \in \mathcal{M}_k} \omega_{m,k} \gamma_{m,k} \sqrt{\frac{p_j^{(p)} \beta_{m,j}}{p_k^{(p)} \beta_{m,k}}} \\ &= \frac{\sum_{m \in \mathcal{M}_k} \omega_{m,k} \tau_p p_k^{(p)} \beta_{m,k}^2 \sqrt{p_j^{(p)} \beta_{m,j}} \prod_{q \neq m} \left(\sqrt{p_k^{(p)} \beta_{q,k}} \psi_{q,k} \right)}{\prod_{m \in \mathcal{M}_k} \sqrt{p_k^{(p)} \beta_{m,k}} \prod_{m \in \mathcal{M}_k} \psi_{m,k}} \\ &= \frac{\eta_{k,j}}{\chi_k \lambda_k}. \end{aligned} \quad (17)$$

Then, the SINR expression in (8), can be rewritten as

$$\begin{aligned} \Upsilon_k &= \frac{N p_k^{(d)} |\theta_k \chi_k|^2}{N \sum_{\substack{j=1 \\ j \neq k}}^K p_j^{(d)} |\eta_{k,j} \varphi_j^H \varphi_k|^2 + \sum_{k=1}^K p_j^{(d)} \xi_{k,j} \lambda_k \chi_k^2 + \sigma^2 \vartheta_k \lambda_k \chi_k^2} \\ &\triangleq \frac{q_k(\mathbf{x})}{w_k(\mathbf{x})}. \end{aligned} \quad (18)$$

Note that $q_k(\mathbf{x})$ and $w_k(\mathbf{x})$ are posynomial functions and \mathbf{x} is composed by $\{\theta_k, \lambda_k, \vartheta_k, \xi_{k,j}, \chi_k, \eta_{k,j}, p_k^{(d)}\}_{\forall k,j}$. Thus, problems (11) and (13) can be written as a signomial geometric programming (SGP) problem as follows

$$\underset{\mathbf{x}, \varsigma}{\text{minimize}} \quad f_0(\varsigma) \quad (19a)$$

$$\text{subject to} \quad f_k(\mathbf{x}, \varsigma) \leq 1, \quad \forall k, \quad (19b)$$

$$(9b), (9c) \text{ and } (9d). \quad (19c)$$

where $\varsigma = f_0(\varsigma) = \epsilon^{-1}$ and $f_k(\mathbf{x}, \varsigma) = \frac{w_k(\mathbf{x})\varsigma}{q_k(\mathbf{x})}$ for problem (11), whereas $\varsigma = \epsilon_k$, $f_0(\varsigma) = (\prod_{\forall k} \epsilon_k)^{-1}$ and $f_k(\mathbf{x}, \varsigma) = \frac{w_k(\mathbf{x})\varsigma}{q_k(\mathbf{x}) + w_k(\mathbf{x})}$ for problem (13). However, the global optimal solution of such a problem is computationally difficult to be obtained. Therefore, to achieve a practical solution while preserving an efficient performance, we resort to an approximation approach.

Note that the difficulties lie on constraints (9c) and (19b). Indeed, (9c) are generalized posynomial equality constraints and (19b) are not valid posynomial inequality constraints, thus, the problem is very difficult to solve, at least globally [30], [31]. To deal with this issue, we relax constraint (9c) and resort to the SCA approach in which, at each iteration l , we approximate the denominator of $f_k(\mathbf{x}, \varsigma)$ (denoted as $z_k(\mathbf{x})$) with a monomial $\tilde{z}_k^{(l)}$, but leaving the numerator as a posynomial. This can be efficiently done using the following lemma:

Lemma 2 (See [31], Lemma 1): For any posynomial function $z_k(\mathbf{x}) = \sum_{\forall i} \mu_i(\mathbf{x})$, it holds for any α_i that

$$z_k(\mathbf{x}) \geq \tilde{z}_k(\mathbf{x}) = \prod_{\forall i} \left(\frac{\mu_i(\mathbf{x})}{\alpha_i} \right)^{\alpha_i}, \quad (20)$$

where $\mu_i(\mathbf{x})$ is the i -th monomial of $z_k(\mathbf{x})$. In addition, if $\alpha_i = \frac{\mu_i(\mathbf{x}^*)}{z_k(\mathbf{x}^*)}$, $\forall i$, for any fixed positive \mathbf{x}^* , then $\tilde{z}_k(\mathbf{x}^*) = z_k(\mathbf{x}^*)$, and

$\tilde{z}_k(\mathbf{x}^*)$ is the best local monomial approximation to $z_k(\mathbf{x}^*)$ near \mathbf{x}^* in the sense of first order Taylor approximation.

Relying on Lemma 2, we can now prove the following proposition, which will be instrumental in the sequel.

Proposition 1: Problem (19) can be presented in the l -th iteration in the form of a GP problem and it is proven that this relaxation is tight

$$\underset{\mathbf{x}, \varsigma}{\text{minimize}} \quad f_0(\varsigma) \quad (21a)$$

$$\text{subject to} \quad \tilde{f}_k(\mathbf{x}, \varsigma) \leq 1, \quad \forall k, \quad (21b)$$

$$\sum_{m \in \mathcal{M}_k} (\omega_{m,k})^2 \leq 1, \quad \forall k, \quad (21c)$$

$$(9b), \quad (21d)$$

where $\tilde{f}_k(\mathbf{x}, \varsigma) = \frac{w_k(\mathbf{x})\varsigma}{\tilde{z}_k(\mathbf{x})}$.

Proof: Note that constraints (21c) and (21d) are now posynomial upper bound inequality constraints and by using Lemma (2), we approximate the posynomial function $z_k(\mathbf{x})$ with a monomial function $\tilde{z}_k(\mathbf{x})$, then a lower bound on $f(\mathbf{x})$ becomes an upper bound on a monomial, which is allowed in the standard form of GP. In addition, maximizing a monomial is equivalent to minimizing its reciprocal, which is another monomial.

Furthermore, observe that $f_0(\varsigma)$ and $f_k(\mathbf{x}, \varsigma)$ are monotonically decreasing in $\omega_{m,k}$, i.e., if we increase $\omega_{m,k}$ (holding all other variables constant), $f_k(\mathbf{x}, \varsigma)$ decreases or remain constant, and the generalized posynomial function $\sum_{k=1}^K (\omega_{m,k})^2$ is monotonically and strictly increasing in $\omega_{m,k}$, i.e., if we increase $\omega_{m,k}$ (holding all other variables constant), the function $\sum_{k=1}^K (\omega_{m,k})^2$ increases. Then, as shown in [30, Section 7.4], the GP problem in Proposition 1 is a tight approximation of the original SGP problem, i.e., the solution of the relaxed problem (21) is equivalent to the solution of the original problem (19), which completes the proof. ■

Now, problem (21)² can be efficiently solved by using a standard solver, such as MOSEK [32] and CVXPY [33]. Furthermore, the proposed solution is an iterative process in which

$$\alpha_i^{(l)} = \frac{\mu_i(\mathbf{x}^{(l-1)})}{z_k(\mathbf{x}^{(l-1)})}, \quad (22)$$

where $\mathbf{x}^{(l-1)}$ is the solution from the previous iteration. The complete SCA algorithm can be seen in Algorithm 1.

It is worth mentioning that the global optimality of the solution achieved by Algorithm 1 cannot be guaranteed, which occurs due to the iterative linear approximation procedure employed by the SCA method [30], [34], [35]. However, we can obtain a Karush-Kuhn-Tucker (KKT) point of problem (11) by satisfying the conditions from [34], given in Lemma 3. Furthermore, due to the positiveness constraints imposed by GP, none of the users are assigned zero power, even though some users might be assigned with powers very close to zero. Then, when plotting the rate of

²Note that in the standard GP problem form, the positiveness constraints of the power variables are implicitly considered, as shown in [30, Section 2.2]. This is the reason why we do not explicitly include them when writing the standard GP problem.

Algorithm 1: Centralized Solution Based On SCA And GP.

-
- 1: Initialize $\mathbf{x}^{(0)}$;
 - 2: Set $l = 1$
 - 3: **repeat**
 - 4: Compute $\alpha_i^{(l)}$, $\forall i$ using (22);
 - 5: Approximate the SINR constraints using Lemma 2;
 - 6: Solve the l -th approximated problem (21) to obtain $\mathbf{x}^{(l)}$;
 - 7: $l \leftarrow l + 1$;
 - 8: **until** Convergence has been reached or $l > L_{\max}$.
-

those users in the results, it might seem that their rate is zero, but they are actually just very close to zero. Nevertheless, it is also worth highlighting that we consider the pilot-and-data power allocation of a certain number of subcarriers (e.g., 12 subcarriers as in a physical resource block) over a coherence time interval. Therefore, in practice, in long term evolution (LTE) or new radio (NR) systems [36], [37], [38], if the rate of a given user happens to be close to zero for the considered subcarriers over a coherence time interval, that user can still be assigned with some rate in other subcarriers on the same coherence time interval.

Lemma 3: By constructing a family of functions satisfying the following properties:

- 1) $f(\mathbf{x}) \leq f^{(l)}(\mathbf{x})$, $\forall \mathbf{x}$ in the feasible set,
- 2) $f(\mathbf{x}^{(l-1)}) \leq f^{(l)}(\mathbf{x}^{(l-1)})$
- 3) $\nabla f(\mathbf{x}^{(l-1)}) = f^{(l)}(\mathbf{x}^{(l-1)})$,

and optimizing the problem by replacing $f(\mathbf{x})$ with $f^{(l)}(\mathbf{x})$ in the l -th iteration, if the series of solutions converge, then it converges to a KKT point of the original problem.

Proof: It follows from [34] and is, therefore, omitted. \square

Proposition 2: Algorithm 1 converges and the approximation $f^{(l)}(\mathbf{x}) = \frac{w_k(\mathbf{x})}{\tilde{z}_k^{(l)}(\mathbf{x})}$ satisfies the three conditions in Lemma (3). Thus, Algorithm 1 converges to a KKT point of problem (11).

Proof: As shown in [34], we have that either the solution of the SCA subproblem is a solution of the original problem or the objective is monotonically improved. Since the objective function is bounded by the power constraints, we can claim the convergence of Algorithm 1. Moreover, from Lemma 3 we have that the conditions (1) and (2) are satisfied since $z_k(\mathbf{x}) > \tilde{z}_k(\mathbf{x})$ and $\tilde{z}_k(\mathbf{x}^*) = z_k(\mathbf{x}^*)$. Finally, the condition (3) can be verified by taking the derivatives of $z_k(\mathbf{x})$ and $\tilde{z}_k(\mathbf{x})$ with respect to \mathbf{x} . This completes the proof. \square

V. PROPOSED DECENTRALIZED SOLUTION BASED ON MULTI-AGENT DRL

As shown in the previous section, we can use GP to solve problems (10) and (12). However, GP requires very high complexity compared to other standard convex problems. Hence, the method presented in Section IV may not be suitable to large scale problems [20]. In this context, we propose a decentralized solution for problems (10) and (12) based on multi-agent DRL. In particular we focus on the actor-critic method.

A. An Overview of the Actor-Critic Method

DRL approaches are characterized by one or more agents interacting with the surrounding environment in order to learn an optimal policy. The learning is done by trial and error, where the agent gets a reward for each taken action [39]. Let \mathcal{S} be a set of possible states, \mathcal{A} be a set of actions, $\pi(a|s)$ be the policy, which can be either deterministic or stochastic,³ and $v_\pi(s)$ be the state-value function denoted as the expected return when starting in s and following π thereafter. The actor-critic method aims to learn approximations to both policy, $\pi(a|s, \Theta)$ and state-value function, $v_\pi(s, \Omega)$, where Θ and Ω are the neural network (NN) parameters of $\pi(a|s)$ and $v_\pi(s)$, respectively. Thus, the agent is composed by two parts: the *actor* and the *critic*. The actor is responsible to generate actions according to the observed environment state by exploring the policy. The critic, on the other hand, has as role to estimate the state-value function besides evaluating and criticizing the current policy by processing the rewards received from the environment. Moreover, the critic updates the parameters of $v_\pi(s, \Omega)$ and, next, the actor updates the policy distribution (i.e., the parameters of $\pi(a|s, \Theta)$) in the direction suggested by the critic.

Therefore, assuming discrete time steps, iteratively, the agent observes, at time t , the current state $s_t \in \mathcal{S}$ from the environment. Then, the actor part selects an action $a_t \in \mathcal{A}$ based on the policy $\pi(a^{(t)}|s^{(t)}, \Theta)$. Next, the environment moves to state $s^{(t+1)} \in \mathcal{S}$ and the agent gets a reward $r^{(t)}$, which characterizes its benefit from taking action $a^{(t)}$ at state $s^{(t)}$. Once the action is taken and the feedback from the environment is obtained, the critic computes the temporal difference (TD) error, as follows

$$\delta^{(t)} = r^{(t)} + \zeta v(s^{(t+1)}, \Omega) - v(s^{(t)}, \Omega), \quad (23)$$

where $0 \leq \zeta \leq 1$ is the discount rate, and updates the parameters of $v_\pi(s, \Omega)$ by minimizing the least squares temporal difference, i.e., by minimizing the loss function

$$L_v(\Omega^{(t)}) = \left(\delta^{(t)}\right)^2. \quad (24)$$

After that, the actor updates the parameters of $\pi(a|s, \Theta)$ using the policy gradient [39], [40] with the TD error. In other words, we must minimize the following loss function

$$L_\pi(\Theta) = -\log \left(\pi \left(a^{(t)} | s^{(t)}, \Theta^{(t)} \right) \right) \delta^{(t)}. \quad (25)$$

Note that (24) and (25) can be minimized by employing the gradient descent or similar. Finally, this process will be repeated until the optimal policy, π^* , is obtained.

B. Proposed Multi-Agent DRL Solution

In this section, we present a multi-agent DRL-based framework to solve problems (10) and (12). Differently from the solution presented in Section IV, we focus on a decentralized solution. Then, we assume that each user is an agent, i.e., there is a total of K agents in the system so that the actions are taken distributedly. Each agent k is composed by two NNs $\pi_k(s|a, \Theta_k)$

³In this paper, we assume a stochastic policy, thus, $\pi(a|s)$ is denoted as a probability distribution of taking an action a given a state s .

(actor) and $v_{\pi,k}(s, \boldsymbol{\Omega}_k)$ (critic) used to estimate the policy and state-value function, respectively.

We also assume that the agents are responsible for computing the power values, while the CPU computes the values of the receive coefficients, i.e., $\omega_{m,k}$ for all $k \in \mathcal{K}$ and $m \in \mathcal{M}_k$. The motivation is to reduce the complexity at the user side and the signaling overhead. In addition, assuming fixed power values, it is possible to compute the receive filter coefficients, $\{\omega_{m,k}\}_{m \in \mathcal{M}_k}$, that maximize the effective SINR in (8) for each user k using the following corollary:⁴

Corollary 1: The receive filter coefficients that maximize the effective SINR in (8) of user k are

$$\boldsymbol{\omega}_k = \frac{\left(N \sum_{\substack{j=1 \\ j \neq k}}^K p_j^{(d)} \boldsymbol{\nu}_{k,j} \boldsymbol{\nu}_{k,j}^H + \sum_{j=1}^K p_j^{(d)} \mathbf{D}_{k,j} + \mathbf{C}_k \right)^{-1} \boldsymbol{\nu}_{k,k}}{\left\| \left(N \sum_{\substack{j=1 \\ j \neq k}}^K p_j^{(d)} \boldsymbol{\nu}_{k,j} \boldsymbol{\nu}_{k,j}^H + \sum_{j=1}^K p_j^{(d)} \mathbf{D}_{k,j} + \mathbf{C}_k \right)^{-1} \boldsymbol{\nu}_{k,k} \right\|}, \quad (26)$$

where $\boldsymbol{\omega}_k = [\omega_{1,k}, \dots, \omega_{|\mathcal{M}_k|,k}]^T$, $\boldsymbol{\nu}_{k,j} = [\gamma_{1,k}, \dots, \gamma_{|\mathcal{M}_k|,k}]^T$, $\mathbf{C}_k = \text{diag}([\gamma_{1,k}, \dots, \gamma_{|\mathcal{M}_k|,k}])$, $\mathbf{D}_{k,j} = \text{diag}([\gamma_{1,k} \beta_{1,j}, \dots, \gamma_{|\mathcal{M}_k|,k} \beta_{|\mathcal{M}_k|,j}])$ and

$$\boldsymbol{\nu}_{k,j} = \left[\frac{\gamma_{1,k} \sqrt{p_j^{(p)}} \beta_{1,j}}{\sqrt{p_k^{(p)}} \beta_{1,k}}, \dots, \frac{\gamma_{|\mathcal{M}_k|,k} \sqrt{p_j^{(p)}} \beta_{|\mathcal{M}_k|,j}}{\sqrt{p_k^{(p)}} \beta_{|\mathcal{M}_k|,k}} \right]^T |\varphi_j^H \varphi_k|^2.$$

Proof: Based on [19], [20] and by assuming fixed power allocation, the receiver coefficient design can be formulated as a generalized eigenvalue problem, for which, according to [41], the solution is given by (26). \square

During $T_e \leq T$ time slots, denoted as exploration phase, the actions are taken while the NN' parameters are updated. Obviously, the initial actions (or the initial power allocations) may not be the best possible solutions since the best policy is not still learned. However, the proposed method is able to quickly learn to take good actions and to improve the system performance. Moreover, once the exploration phase is finalized, the updates of the NNs' parameters stop and the best power allocation found so far can be employed in the next time slots until a new exploration phase is required.⁵

That said, we define $s_k^{(t)} \in \mathcal{S}$ as the state of agent k at time slot t , which is composed by the estimated powers $\tilde{p}_k^{(p)}$ and $\tilde{p}_k^{(d)}$ computed by the actor based on the action taken in the previous time slot, the estimated SE achieved by user k in the previous time slot, $\tilde{R}_k^{(t-1)}$ and the reward obtained in the previous time

⁴By using the closed-form expression in (26), the alternating optimization method could be employed in the centralized solution based on SCA and GP, such as was done in [19]. In fact, using such an approach would reduce the number of variables of the proposed optimization-based solution compared to the case in which we jointly optimize the linear receiver filters and pilot-and-data powers. However, it was not possible to solve the problem of joint pilot-and-data power allocation for large scale cell-free systems even for fixed linear receiver filters, as shown in [17]. Moreover, the centralized optimization problems are mainly used for benchmarking purposes in comparison to the DRL-based solutions. Reducing the complexity of the centralized solutions is out of the scope of this paper and left to future works.

⁵A new exploration phase can be required when the large-scale fading coefficients change or the system performance drops below a certain threshold.

slot, $r^{(t-1)}$. In addition, we assume that each agent has the knowledge of the estimated power values of some interfering users. However, to avoid excessive overhead, the number of interfering users is limited to the number of users using the same pilot sequence of user k . Thus, the estimated powers of the interfering users are given by $\tilde{p}_j^{(p)}$ and $\tilde{p}_j^{(d)}$, where $|\varphi_j^H \varphi_k|^2 = 1$. We define those powers as estimated powers because, as we will see later, these powers may not be used for pilot and data transmissions. Note that $\tilde{R}_k^{(t-1)}$ is computed using (26) and $\{\tilde{p}_k^{(p)}, \tilde{p}_k^{(d)}\}_{\forall k}$ obtained in the previous time slot. Therefore, let $\mathbf{p}_k^{(t-1)}$ be a vector composed by all estimated powers at time slot $t-1$ known by agent k , including its own estimated powers. Then, the state s_k is given by

$$s_k^{(t)} = \{\mathbf{p}_k^{(t-1)}, \tilde{R}_k^{(t-1)}, r^{(t-1)}\}. \quad (27)$$

After observing the state, each agent k selects an action $a_k^{(t)} \in \mathcal{A}$ based on policy $\pi_k(a^{(t)} | s^{(t)}, \boldsymbol{\Theta}_k)$. In this paper, the action consists in selecting the fraction of energy allocated to pilots and data, as well as the fraction of saved energy. Thus, we define $\boldsymbol{\phi}_k = [\phi_k^{(p)}, \phi_k^{(d)}, \phi_k^{(s)}]$ as a vector in which $\phi_k^{(p)}$, $\phi_k^{(d)}$ and $\phi_k^{(s)}$ are the fraction of allocated energy to pilots, data and saved energy, respectively. Hence, the action of agent k is given by $\boldsymbol{\phi}_k$. The reason for selecting the fraction of saved energy is that allocating the full energy may not be optimal [17]. For instance, to maximize the minimum SE, the users (mainly those with better channel conditions) must save energy to minimize the interference to users with worse channel conditions. Moreover, it is worth mentioning that using the fraction of saved energy is also a way to generalize the proposed solution for those scenarios in which energy consumption is critical. In fact, a simple way to try to achieve energy-efficient solutions would be to change the reward function. However, energy-efficient solutions are out of the scope of this paper and left for future works.

Furthermore, in order to obtain continuous actions, we assume a stochastic policy in which we learn statistics of a given distribution from which the actions are obtained. In general, previous works have assumed the Gaussian policy. However, the Gaussian policy can be problematic when the actions have a limited range, which is our case. Indeed, the Gaussian policy has an infinite support, i.e., even with a small variance value, the actions sampled from the Gaussian distribution can deviate a lot from the mean, which introduces a bias and affects the learning process. To overcome this issue, we use the Beta policy, which has a $[0, 1]$ support. Moreover, the authors in [42] showed that the Beta policy is bias-free and provides significantly faster convergence and higher scores than the Gaussian policy. Then, we have that

$$\pi_k(a | s, \boldsymbol{\Theta}_k) = \frac{\Gamma(\Lambda(s, \boldsymbol{\Theta}_k) + B(s, \boldsymbol{\Theta}_k))}{\Gamma(\Lambda(s, \boldsymbol{\Theta}_k))\Gamma(B(s, \boldsymbol{\Theta}_k))} \times a^{\Lambda(s, \boldsymbol{\Theta}_k)-1} (1-a)^{B(s, \boldsymbol{\Theta}_k)-1}, \quad (28)$$

where $\Gamma(x)$ is the gamma function and $\Lambda(s, \boldsymbol{\Theta}_k)$ and $B_k(s, \boldsymbol{\Theta}_k)$ are the shape parameters of user k , which we must approximate. As shown before, the action consists in a vector containing the fraction of allocated energy to pilots, data and saved energy, thus, we must approximate the shape parameters for each of the

fractions. Therefore, we define $\Lambda^{(p)}(s, \Theta_k)$ and $B^{(p)}(s, \Theta_k)$, $\Lambda^{(d)}(s, \Theta_k)$ and $B^{(d)}(s, \Theta_k)$ and $\Lambda^{(s)}(s, \Theta_k)$ and $B^{(s)}(s, \Theta_k)$, as the shape parameters to the fraction of allocated energy to pilots, data and saved energy, respectively, given a state s . Then, we can obtain the values of $\phi_k^{(p)}$, $\phi_k^{(d)}$ and $\phi_k^{(s)}$ from the Beta distribution as follows

$$\phi_k^{(p)} \sim \text{Beta}(\Lambda^{(p)}(s^{(t)}, \Theta_k), B^{(p)}(s^{(t)}, \Theta_k)), \quad (29)$$

$$\phi_k^{(d)} \sim \text{Beta}(\Lambda^{(d)}(s^{(t)}, \Theta_k), B^{(d)}(s^{(t)}, \Theta_k)), \quad (30)$$

$$\phi_k^{(s)} \sim \text{Beta}(\Lambda^{(s)}(s^{(t)}, \Theta_k), B^{(s)}(s^{(t)}, \Theta_k)). \quad (31)$$

Note that the energy budget constraints can be violated since the sum of the fractions can be higher than 1. This can be easily solved by normalizing ϕ_k , i.e., dividing ϕ_k by its norm. After that, the estimated powers $\tilde{p}_k^{(p)}$ and $\tilde{p}_k^{(d)}$ can be computed as follows

$$\tilde{p}_k^{(p)} = \phi_k^{(p)} \frac{E_{\max}}{\tau_p}, \quad (32)$$

$$\tilde{p}_k^{(d)} = \phi_k^{(d)} \frac{E_{\max}}{\tau_d}. \quad (33)$$

These estimated powers should not be directly employed for pilot and data transmissions. Indeed, this is a poor approach since the fractions of energy are randomly sampled as shown in (29), (30) and (31) and must vary around the mean. To solve this problem, we propose to store the estimated powers that achieved the highest reward (which will be defined later) until the time slot t . Thus, let $p_{k,\max}^{(p)}$ and $p_{k,\max}^{(d)}$ be the stored powers and $r_{\max}^{(t)}$ the highest reward obtained until time slot t . We only use $\tilde{p}_k^{(p)}$ and $\tilde{p}_k^{(d)}$ for pilot and data transmissions if the reward obtained is higher than $r_{\max}^{(t)}$, otherwise, we use $p_{k,\max}^{(p)}$ and $p_{k,\max}^{(d)}$.

Now, we need to define the reward, which should be designed to maximize the objective function that we desire to optimize. Thus, the obvious choice is

$$r^{(t)} = U(R_1, \dots, R_K). \quad (34)$$

Consequently, $r^{(t)} = \min_k R_k$ when the focus is on the max-min SE problem and $r^{(t)} = \sum_{k=1}^K R_k$ when the aim is at maximizing the total throughput. Note that the reward is computed using the estimated powers and is equal to all agents.

Finally, each agent k observes the next state, $s_k^{(t+1)}$, computes the TD error using (23) and updates Ω_k and Θ_k by minimizing the loss functions in (24) and (25), respectively, using the gradient descent. This process is repeated until the optimal policy is obtained or the exploration phase is finalized. The complete DRL algorithm can be seen in Algorithm 2.

To initialize the algorithm, we assume that each agent randomly initializes the NNs' parameters and set $p_{k,\max}^{(p)}$, $p_{k,\max}^{(d)}$ and $r_{\max}^{(0)}$ equal to zero. The CPU sets $p_k^{(p)}$ and $p_k^{(d)}$ equal to E_{\max}/τ_c and computes $\{\omega_{m,k}\}_{\forall k,m \in \mathcal{M}_k}$, $\{\tilde{R}_k^{(0)}\}_{\forall k}$ and $r^{(0)}$ using (26), (7) and (34), respectively. After that, the CPU sends $\mathbf{p}_k^{(0)}$, $\tilde{R}_k^{(0)}$ and $r^{(0)}$ to each agent k , which will be used to compute the state of agent k . Also, Algorithm 2 can execute τ_s iterations to update the NNs' parameters before the pilot and data transmissions. This can be advantageous because it reduces the exploration phase.

Algorithm 2: Decentralized Solution Based On DRL.

- 1: UE: Initialize Θ_k and Ω_k randomly and set $p_{k,\max}^{(p)} \leftarrow 0$, $p_{k,\max}^{(d)} \leftarrow 0$ and $r_{\max}^{(t)} \leftarrow 0$;
 - 2: CPU: Send $\mathbf{p}_k^{(0)}$, $\tilde{R}_k^{(0)}$ and $r^{(0)}$ to each agent k using uplink signaling;
 - 3: SET $t \leftarrow 1$;
 - 4: **for** $e = 1, \dots, T_e$ **do**
 - 5: **for** $l = 1, \dots, \tau_s$ **do**
 - 6: UE: Observe the current state $s_k^{(t)}$ as shown in (27)
 - 7: UE: Get an action using (29), (30) and (31);
 - 8: UE: Compute $\tilde{p}_k^{(p)}$ and $\tilde{p}_k^{(d)}$ using (32) and (33), respectively;
 - 9: UE: Send $\tilde{p}_k^{(p)}$ and $\tilde{p}_k^{(d)}$ to the CPU using uplink signaling;
 - 10: CPU: Set $p_k^{(p)} \leftarrow \tilde{p}_k^{(p)}$ and $p_k^{(d)} \leftarrow \tilde{p}_k^{(d)} \forall k \in \mathcal{K}$;
 - 11: CPU: Compute $\{\omega_{m,k}\}_{\forall k,m \in \mathcal{M}_k}$, $\{\tilde{R}_k^{(t)}\}_{\forall k}$ and $r^{(t)}$ using (26), (7) and (34), respectively;
 - 12: CPU: Send $\mathbf{p}_k^{(t)}$, $\tilde{R}_k^{(t)}$ and $r^{(t)}$ to each agent k using downlink signaling;
 - 13: **if** $r^{(t)} > r_{\max}^{(t)}$ **then**
 - 14: UE: Set $p_{k,\max}^{(p)} \leftarrow \tilde{p}_k^{(p)}$, $p_{k,\max}^{(d)} \leftarrow \tilde{p}_k^{(d)}$ and $r_{\max}^{(t)} \leftarrow r^{(t)}$;
 - 15: **end if**
 - 16: UE: Observe the next state $s_k^{(t+1)}$ as shown in (27)
 - 17: UE: Compute the TD error using (23);
 - 18: UE: Update Ω_k and Θ_k by minimizing the loss functions in (24) and (25), respectively;
 - 19: SET $t \leftarrow t + 1$;
 - 20: **end for**
 - 21: UE: Use $p_{k,\max}^{(p)}$ and $p_{k,\max}^{(d)}$ for pilot and data transmissions;
 - 22: CPU: Set $p_k^{(p)} \leftarrow p_{k,\max}^{(p)}$ and $p_k^{(d)} \leftarrow p_{k,\max}^{(d)} \forall k \in \mathcal{K}$;
 - 23: CPU: Compute $\{\omega_{m,k}\}_{\forall k,m \in \mathcal{M}_k}$ using (26);
 - 24: CPU: Perform channel estimation and data decoding;
 - 25: **end for**
-

C. Signaling Aspects

In this section we propose a signaling framework for practical implementation of the proposed decentralized solution. Observe that, to compute its state, each agent k requires its estimated SE achieved in the previous time slot, the reward and the estimated power values in the previous time slot of users using the same pilot sequence of user k , as shown in (27). This information requires some knowledge about the interfering users. However, only local information is available at each user. On the other hand, the CPU requires the power values of all users to compute the receive filter coefficients and the reward and, consequently, the SE achieved for each user.

In this context, we propose that all users send the power values to the CPU that, in its turn, computes $\mathbf{p}_k^{(t)}$, $\tilde{R}_k^{(t)}$ and $r^{(t)}$ and sends back this information to the users. Thus, during the execution of Algorithm 2, that information is transmitted and received by means of an over-the-air signaling between

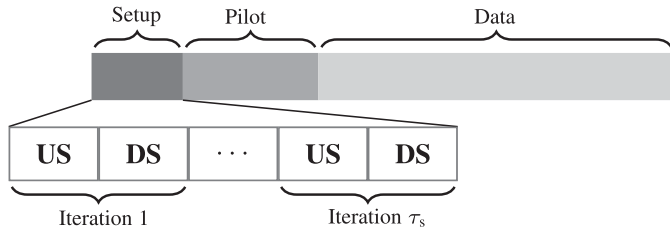


Fig. 1. Frame structure.

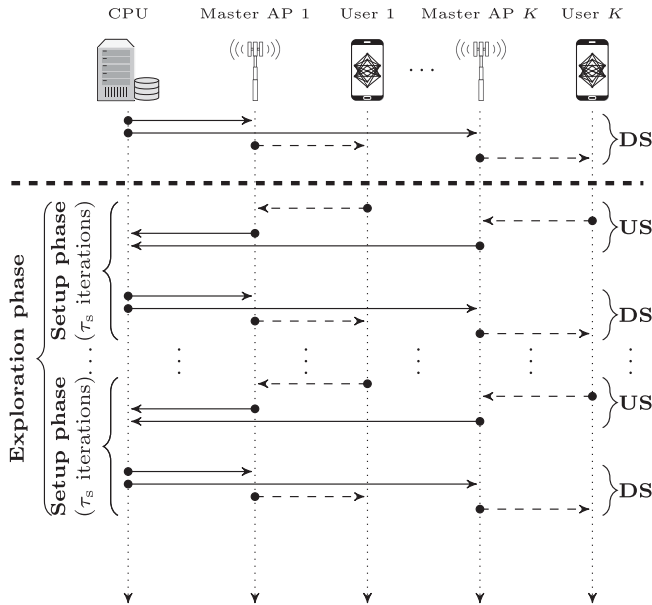


Fig. 2. Signaling exchange used in the decentralized solution.

users and their respective master AP, which is responsible to forward information from users to the CPU and vice-versa, while the communication between the master APs and the CPU is performed by fronthaul signaling. In this paper, the master AP of user k is the one with highest large scale fading coefficient, i.e., $m = \arg \max_{m \in \mathcal{M}} \beta_{m,k}$.

Given these considerations, Fig. 1 presents the proposed frame structure. Note that the frame structure is split into three parts: setup, pilot transmission and data transmission phases. The over-the-air and fronthaul signaling are performed in the setup phase, which is divided into phases: i) uplink signaling, denoted as **US**, which corresponds to line 9 of Algorithm 2, where the users send the power values to the CPU; and ii) downlink signaling, named **DS**, which occurs in line 12 of Algorithm 2, where the CPU sends information to the users, such as the reward. The signaling exchange is illustrated in Fig. 2.

Note that the setup phase occurs at the beginning of each coherence interval. Moreover, each iteration occupies one sample of the coherence interval. Hence, we now have that $\tau_d = \tau_c - \tau_p - \tau_s$, thus, the effective SE achieved by user k , when we take into account the signaling overhead of τ_s iterations during the setup phase, is given by

$$R_k^{\text{eff}} = (1 - (\tau_p + \tau_s)/\tau_c) \log_2(1 + \Upsilon_k). \quad (35)$$

D. computational Complexity and Signaling Overhead

The solution proposed in Algorithm 1 consists of a centralized approach in which the CPU is responsible for computing both pilot and data powers, as well as the receive filter coefficients. Then, the CPU informs the values of $p_k^{(p)}$ and $p_k^{(d)}$ to the users. Moreover, the per-iteration computational complexity of Algorithm 1 is dictated by solving the GP problem in line 6, which has a complexity equivalent to $O((K(M+2))^{3.5})$ [19].

Regarding Algorithm 2, the complexity is dictated by computing the actions and training the weights of the neural network on the user side. Since we use neural networks composed of fully-connected layers, the complexity of the proposed algorithm is $O(ul \log(u))$, where l is the number of layers and u is the number of units per layer [43], [44].

Furthermore, the proposed solution can be implemented in a distributed fashion by adopting an over-the-air signaling scheme, which is described in Section V-C. In that signaling scheme, each iteration has an associated overhead due to the exchange of information between CPU and users. Based on [45] we can measure the communication overhead by the number of orthogonal pilot symbols needed for each iteration, which is given by $\omega = 2\tau_s K$, where τ_s is the number of iterations in the setup phase. Thus, the minimal number of orthogonal pilots increases with the number of users and iterations. Therefore, increasing the number of inner iterations of Algorithm 2 incurs in a higher signaling overload. However, in order to obtain a practical implementation of Algorithm 2 with minimal signaling overhead, this number of iterations can be limited to a maximum of 10 iterations per data frame, as suggested in [45], at the cost of a possibly lower performance in some situations. Indeed, by increasing the number of inner iterations we can obtain a faster convergence of the proposed algorithm, which is an important aspect in high-mobility scenarios. Alternatively, the system operator could use an off-line training by collecting a large training data set, such that the signaling of the training phase is not needed anymore.

VI. NUMERICAL RESULTS

A. Simulation Setup

We consider the uplink of a cell-free system in which APs and users are uniformly distributed within a square of size 1×1 km². A wrap-around technique is applied to imitate a network with an infinite area. Moreover, a random pilot assignment is used, i.e., each user randomly selects a pilot from a predefined set of orthogonal pilots. The large-scale coefficients are modeled by the path loss and correlated shadowing as follows (in dB): $\beta_{m,k} = \text{PL}_{m,k} + \sigma_{\text{sh}} \mathcal{Z}_{m,k}$, where $\sigma_{\text{sh}} \mathcal{Z}_{m,k}$ is the shadow fading with the standard deviation σ_{sh} and $\mathcal{Z}_{m,k} \sim \mathcal{N}(0, 1)$, while $\text{PL}_{m,k}$ is the path loss that is modeled by the three slope model [5], [15]:

$$\text{PL}_{m,k} = \begin{cases} -L - 10 \log_{10}(d_{m,k}^{3.5}), & \text{if } d_{m,k} > d_1 \\ -L - 10 \log_{10}(d_1^{1.5} d_{m,k}^2), & \text{if } d_0 < d_{m,k} \leq d_1 \\ -L - 10 \log_{10}(d_1^{1.5} d_0^2), & \text{if } d_{m,k} \leq d_0 \end{cases} \quad (36)$$

TABLE I
SYSTEM PARAMETERS

Parameter	Description	Value
f_c	Carrier Frequency	1.9 GHz
Δ	System Bandwidth	20 MHz
d_0, d_1	Path loss parameters	50 m and 10 m
h_{AP}, h_u	AP and user heights	15 m and 1.65 m
M	Num. of APs	100
K	Num. of users	{8, 20}
N	Num. of antennas per AP	8
M_k	Cluster size	{3, 10}
τ_p	Number of pilots	{4, 10}
τ_c	Coherence interval	200 (in samples)
$\sigma_{sh}, d_{decorr}, \varepsilon$	Shadowing parameters	8 dB, 100 m, 0.5
σ_F	Noise figure	9 dB
κ_B	Boltzmann constant	$1.381 \cdot 10^{-23}$ Joule/Kelvin
T_0	Noise temperature	290 Kelvin

TABLE II
DRL PARAMETERS

Setting	Actor	Critic
Input layer	linear, L	linear, L
Hidden layer	ReLU, 64	ReLU, 16
Output layer	Softplus, 6	Linear, 1

in which $d_{m,k}$ is the distance between AP m and user k in kilometers and

$$L \triangleq 46.3 + 33.9 \log_{10}(f_c) - 13.83 \log_{10}(h_{AP}) - (1.1 \log_{10}(f_c) - 0.7)h_u + (1.56 \log_{10}(f_c) - 0.8), \quad (37)$$

where f_c is the carrier frequency (in MHz), h_{AP} and h_u are the AP and user antenna heights (in m), respectively. There is no shadowing if $d_{m,k} \leq d_1$.

For the shadow fading coefficients, we consider a model with two components [5], [46]: $\varkappa_{m,k} = \sqrt{\varepsilon}a_m + \sqrt{1-\varepsilon}b_k$, where $a_m \sim \mathcal{N}(0, 1)$ and $b_k \sim \mathcal{N}(0, 1)$ are independent random variables, and $0 \leq \varepsilon \leq 1$ is a fitting parameter with

$$\mathbb{E}[a_m a_n] = 2^{-\frac{d_a(m,n)}{d_{decorr}}}, \quad \mathbb{E}[b_k b_j] = 2^{-\frac{d_u(k,j)}{d_{decorr}}}, \quad (38)$$

where $d_a(m, n)$ is the distance between APs m and n , $d_u(k, j)$ is the distance between users k and j and d_{decorr} is the decorrelation distance, which depends on the environment.

We use the largest-large-scale-fading-based AP selection scheme from [14] to form \mathcal{M}_k and define $E_{\max} = \frac{100\tau_c\sigma^2}{\tilde{\beta}}$, where $\tilde{\beta}$ is the value of $\beta_{m,k}$ when the distance between a given AP and user is lower than 10 m. This is equivalent to providing a median signal-to-noise ratio (SNR) of 20 dB at the region close to the APs. Also, numerical results are obtained by 500 random realizations of APs and users locations. The noise power is given by

$$\sigma^2 = \Delta \times \kappa_B \times T_0 \times \sigma_F. \quad (39)$$

where κ_B is the Boltzmann constant, T_0 is the noise temperature and σ_F is the noise figure. The complete list of system parameters can be seen in Table I.

With respect to the DRL-based solution, we have that each agent is composed by two NNs: actor and critic. Both actor and critic were implemented using Tensorflow [47] assuming one input layer, one hidden layer and one output layer. The input size is $L = 2 + 2\varpi$ in which $\varpi = K/\tau_p$ is the factor of pilot reuse. Moreover, we use the Adam algorithm [48] with a learning rate equal to 0.001. We also assume a low mobility scenario for which $T_e \ll T$. Unless otherwise stated, we define T_e equal to

10,000 coherence intervals and $\tau_s = 1$. Moreover, we consider only the cases in which the Beta distribution is concave and unimodal, i.e., $\Lambda(s, \Theta)$, $B(s, \Theta) > 1$ and set ζ equal to zero. In Table II, the architectures of the actor and critic NNs and the hyper-parameter settings are listed in detail.

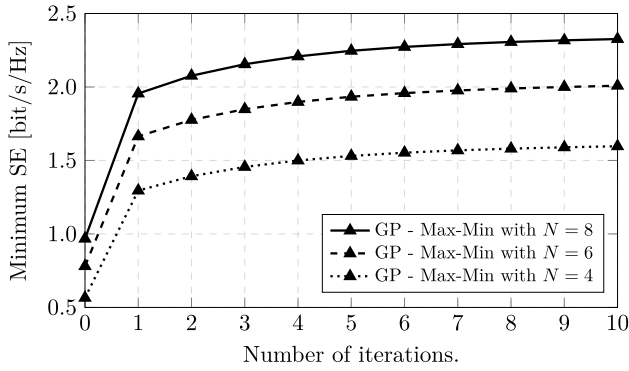
In the plots, the solution of problems (10) and (12) using SCA and GP are called GP - Max-Min and GP - Sum-SE, respectively. Similarly, the solution of problems (10) and (12) based on DRL are marked as DRL - Max-Min and DRL - Sum-SE, respectively. Moreover, we have two benchmarking solutions to evaluate the performance of our algorithms. The first is the naive solution, where we set $p_k^{(p)} = p_k^{(d)} = E_{\max}/\tau_c$ and $\omega_{m,k}$ equal to 1 for all $k \in \mathcal{K}$ and $m \in \mathcal{M}$, which is identified as Naive in the plots. The second benchmark is the solution proposed in [15], which relies on the bisection method and thus has a per-iteration complexity in the order of $\mathcal{O}(K^4)$ and is named as Mai in the plots.⁶

B. Results

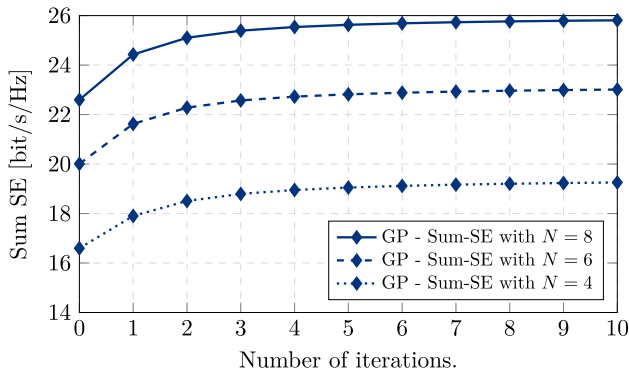
Our discussion starts in Fig. 3, which shows the evolution of Algorithm 1 for different numbers of antennas per AP. As we can see, Algorithm 1 is able to solve both problems (10) and (12), converging to a local optimal solution in a few iterations. We also observe that both the minimum SE and sum-SE increase as the number of antennas per AP increases.

Regarding the DRL-based decentralized solutions, an exploration phase is required to update the NNs' parameters. Thus, in Fig. 4 we plot the utility function values during the exploration phase for different numbers of iterations in the setup phase. To obtain a fair comparison, we consider the same amount of updates of the NNs' parameters in all cases, i.e., the product between the number of time slots and the number of iterations in the setup phase is equal for all cases. Specifically, we consider $T_e \cdot \tau_s = 10,000$. Then, as we can observe, the minimum SE and sum SE of GP - Max-Min, GP - Sum-SE, Naive and Mai solutions, are kept constant during the exploration phase. The reason is that these solutions are performed only once every T time slots. On the other hand, we observe that the DRL - Max-Min solution improves the minimum SE in Fig. 4(a) with time. Similarly, the DRL - Sum-SE solution improves the sum SE in Fig. 4(b). This is expected because as the time passes, the agents are able to learn a better policy, which aims to maximize the obtained reward. In other words, the users learn to perform a better pilot-and-data power allocation over time. Note that the DRL - Max-Min decreases the sum SE with time because this solution.

⁶Since Mai's solution does not consider the energy budget constraint, we adapt this solution by assuming the maximum pilot and data powers as E_{\max}/τ_c .



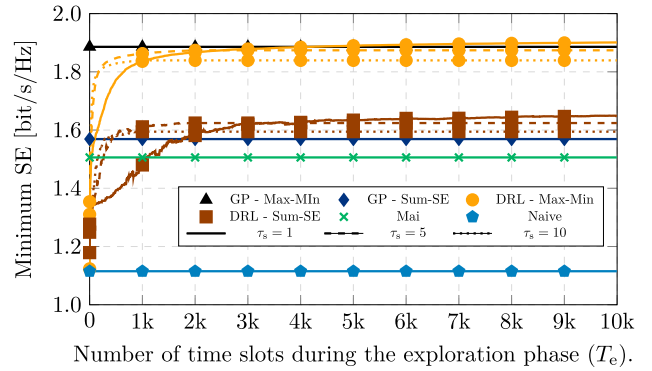
(a)



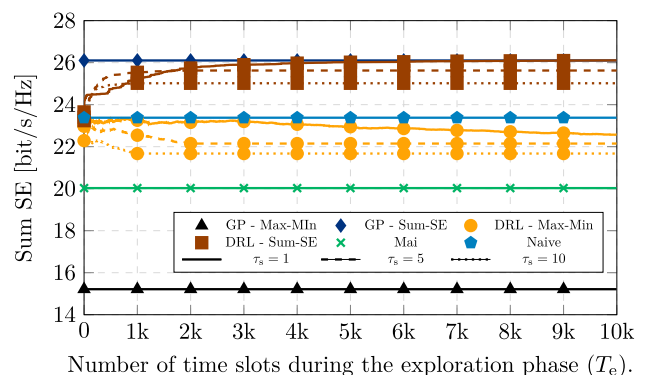
(b)

Fig. 3. Convergence of Algorithm 1 with $\{M, K, |\mathcal{M}_k|, \tau_p\} = \{100, 8, 3, 4\}$. (a) GP - Max-Min. (b) GP - Sum-SE.

Furthermore, it can be noted that the convergence of the DRL - Max-Min and DRL - Sum-SE solutions depend on the number of iterations performed during the setup phase. Indeed, the greater is the number of iterations in the setup phase, the faster is the convergence. This occurs because more updates of the NNs' parameters can be performed at each coherence interval before pilot and data transmissions. However, this comes at the cost of more signaling, consequently, less samples are used for data transmissions, which affects the system performance. Based on Fig. 4(a) and (b), we have a loss of approximately 2% and 5% of the system performance when 5 and 10 iterations are used, respectively, in the setup phase. However, this can be an important aspect to be employed in scenarios with higher mobility, where the number of time slots dedicated to exploration phase must be reduced. Also, we observe that the DRL-based decentralized solutions are able to outperform the benchmark solutions in a few time slots. Considering $\tau_s = 1$ we observe that the DRL-based decentralized solutions achieve similar performance to those centralized solutions using SCA and GP with only 5,000 time slots. In addition, focusing on Fig. 4(a), the DRL - Max-Min solution presents a gain of approximately 21% and 65% compared to the Mai and Naive solutions, respectively, while in Fig. 4(b) the DRL - Sum-SE solution presents a gain of approximately 25% and 8% compared to the Mai and Naive solutions, respectively, with only 1,000 time slots.



(a)

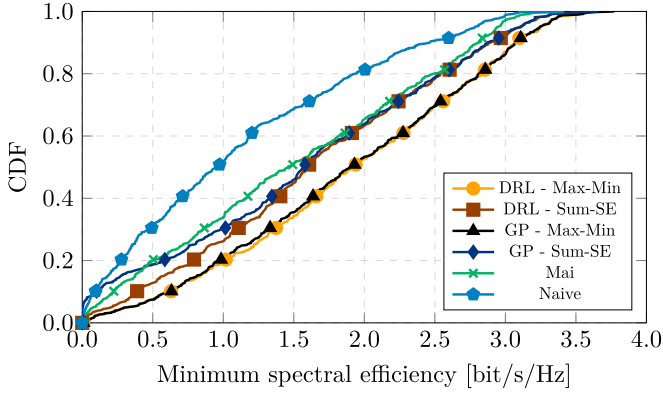


(b)

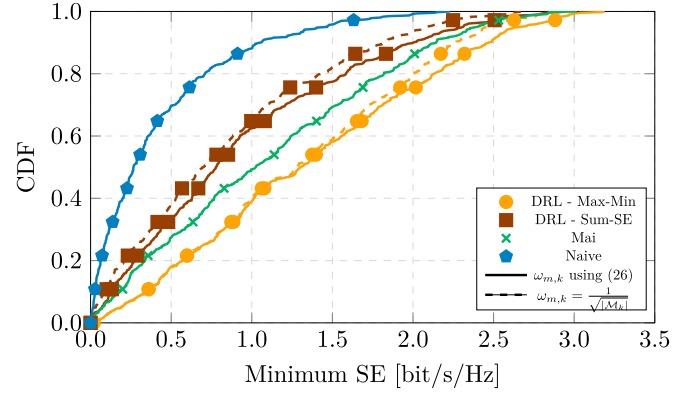
Fig. 4. Utility function during the exploration phase with $\{M, K, N, |\mathcal{M}_k|, \tau_p\} = \{100, 8, 8, 3, 4\}$. (a) Max-min SE. (b) Max-sum SE.

To further evaluate the performance of the proposed solutions, Fig. 5 presents the cumulative distribution function (CDF) of the utility function for all solutions after the exploration phase is finished. First, we analyze the performance with respect to the minimum SE, which is shown in Fig. 5(a). As we can see, the Naive solution presents the worst performance. This occurs because the Naive solution does not perform a dynamic power allocation, which affects the channel estimation and data transmission, mainly of those users with worse channel conditions. The Mai solution, in its turn, conducts a pilot power allocation to minimize the largest normalized MSEs among users and, next, applies a data power control to maximize the fairness among users, which justifies its good performances compared to the Naive solution. Interestingly, we observe that the DRL - Sum-SE presents a slight gain compared to the Mai solution for the simulated scenarios. Finally, the GP - Max-Min and DRL - Max-Min solutions present the best performances, which shows that performing JPDPC as well as optimizing the receive filter coefficients can improve the system performance in terms of minimum SE. Considering the 95%-likely point, the GP - Max-Min and DRL - Max-Min solutions are able to increase the minimum SE by 4 times compared to the Mai solution and 12 times with respect to the Naive solution.

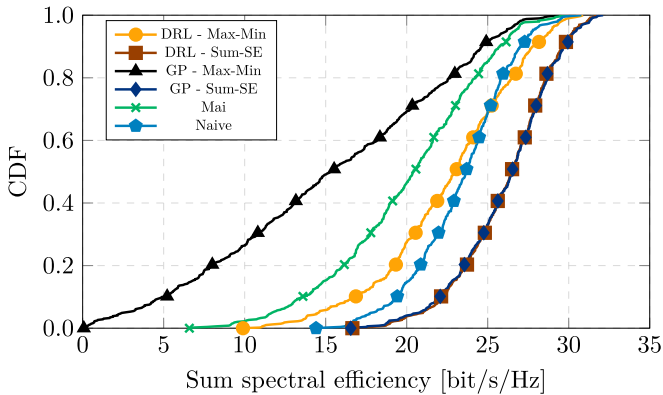
Fig. 5(b) presents the CDF of sum-SE. It can be observed that the GP - Max-Min presents the worst performance, which occurs because it focuses on users with worse channel conditions to



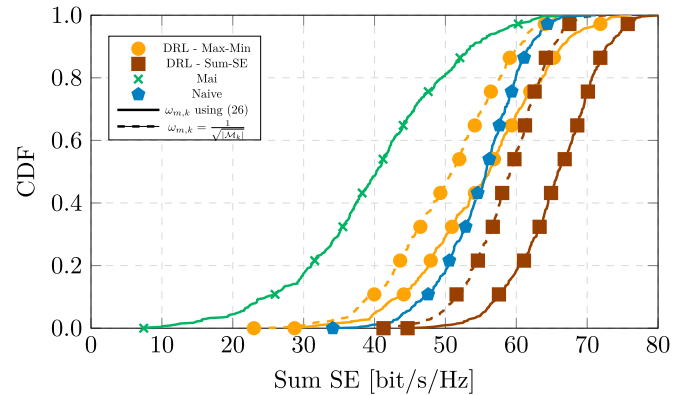
(a)



(a)



(b)



(b)

Fig. 5. CDF of the utility function with $\{M, K, N, |\mathcal{M}_k|, \tau_p\} = \{100, 8, 8, 3, 4\}$. (a) Max-min SE. (b) Max-sum SE.

Fig. 6. CDF of the utility function with $\{M, K, N, |\mathcal{M}_k|, \tau_p\} = \{100, 20, 8, 10, 10\}$. (a) Max-min SE. (b) Max-sum SE.

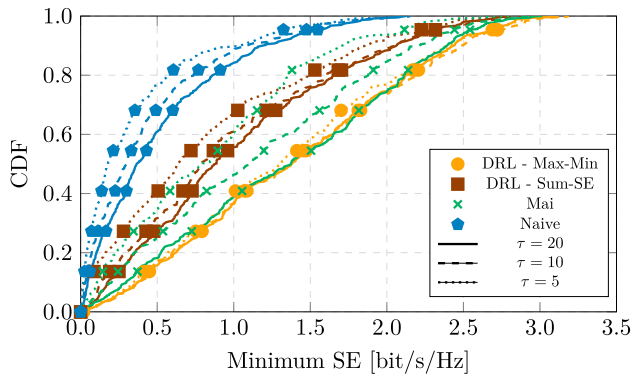
increase the minimum SE, thus decreasing the sum SE. The same reasoning is valid for the Mai solution. On the other hand, the GP - Sum-SE and DRL - Sum-SE have the best performance in terms of sum-SE. This is expected because they aim at maximizing the sum-SE. Compared to the Naive solution, for example, the 95%-likely of the sum-SE presents a gain of almost 17%.

In summary, we note a significantly increased system performance in terms of minimum SE and sum SE when optimizing the pilot and data powers jointly with the receive filter coefficients. Moreover, the DRL-based decentralized solutions present very close performances compared to the centralized solutions using GP and SCA. Although there is no guarantee that Algorithm 1 can yield an optimal solution, it is the best solution available that we know. Thus, these results validate the effectiveness of DRL-based decentralized solutions.

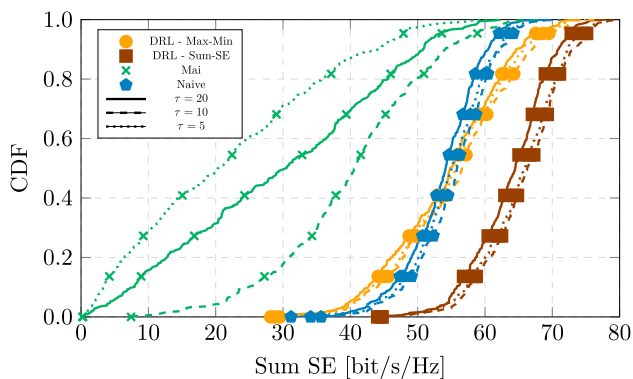
However, Algorithm 1 fails to work on large-scale problems due to its high computational complexity. Then, to further evaluate the performance of the DRL-based decentralized solutions, in the next numerical simulations, we compare the DRL - Max-Min and DRL - Sum-SE solutions with the Mai and Naive solutions using a larger scale network.

First, we analyze the impact of optimizing the receive filter coefficients, which can be seen in Fig. 6. Thus, we consider two alternative solutions in which only the powers are computed

based on DRL and the receive filter coefficients, $\{\omega_{m,k}\}_{\forall(m,k)}$, are set equal to $1/\sqrt{|\mathcal{M}_k|}$, which is equivalent to the solution in [17]. As we can see, the DRL - Max-Min and DRL - Sum-SE solutions present the best performance in terms of minimum SE and sum SE, respectively, which shows that the DRL-based decentralized solutions can outperform the benchmarking solution even in large-scale scenarios. When increasing the scenario, we observe that the DRL - Max-Min solution presents a good performance in terms of sum SE. Indeed, it is able to overcome the Naive solution for values above the 40th-percentile. Also, it can be seen that the optimization of the receive filter coefficients has different impacts on the utility function to be optimized. In fact, in Fig 6(a) we note that the minimum SE presents similar results for both cases, i.e., with or without receive filter coefficients optimization. The reason is that the max-min SE problem focuses on the users with the worst channel condition, which have low freedom to improve their SE. Then, after performing the JPDPC, the gains of optimizing the receive filter coefficients are practically inexistent. However, as the channel conditions of users improve, we observe that optimizing the receive filter coefficients brings benefits in terms of SE. That is more evident when we focus on the sum SE objective function. As we can see in Fig. 6(b) there is a significant increase in the sum SE when the receive filter coefficients are jointly optimized with



(a)

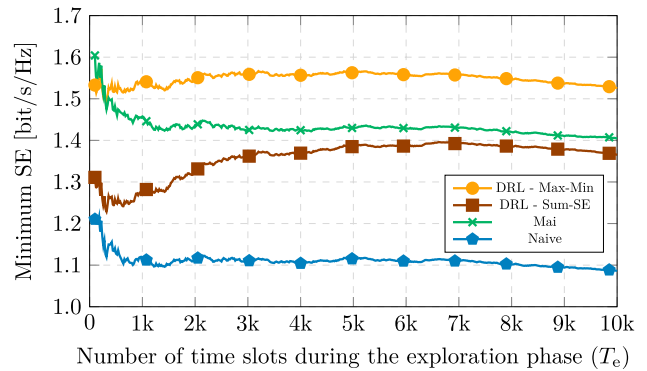


(b)

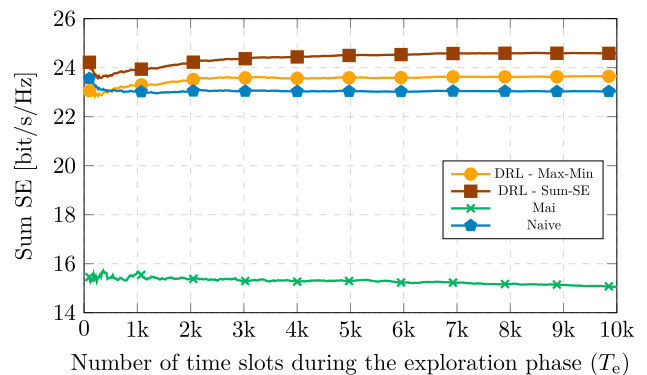
Fig. 7. CDF of the utility function with different τ_p and $\{M, K, N, |\mathcal{M}_k|, \tau_s\} = \{100, 20, 8, 10, 1\}$. (a) Max-min SE. (b) Max-sum SE.

the pilot-and-data powers. Indeed, the whole CDF is shifted to the right by almost 7 b/s/Hz when the optimization of the receive filter coefficients is performed, which represents a gain of almost 15% compared to case in which the receive filter coefficients are not optimized.

Last but not least, Fig. 7 analyzes the performance of the proposed solutions considering different τ_p (i.e., different levels of pilot contamination). It can be observed that the minimum SE increases as τ_p increases. This occurs because the pilot contamination decreases, benefiting users in worse channel conditions, consequently, the minimum SE tends to increase. On the other hand, when analyzing the sum SE we note that the performance of all solutions decreases when τ_p is equal to 20. The reason behind it is that the number of samples available for data transmission decreases. Moreover, comparing the performance of all solutions we have that the DRL - Max-Min and DRL - Sum-SE solutions present the best performance for all values, indicating that the DRL-based decentralized solutions are also able to manage different levels of pilots contamination. Another important aspect to be mentioned is that the DRL-Max-Min solution increases the gain compared to the Mai solution in terms of minimum SE as τ_p decreases. This occurs because, differently from the Mai solution, it performs a JPDPC, allowing an enhanced management of the power resources and optimizes the receive filter coefficients, which are assumed to be fixed



(a)



(b)

Fig. 8. Exploration Phase with user mobility, user speed of 3 km/h and $\{M, K, N, |\mathcal{M}_k|, \tau_s\} = \{100, 8, 8, 3, 4\}$. (a) Max-min SE. (b) Max sum-SE.

in the Mai solution. Hence, the DRL - Max-Min is able to deal with the pilot contamination more efficiently than the Mai solution.

C. Extension to Scenarios With User Mobility

The previous case focused on stationary scenarios. However, the proposed solution can also be extended to scenarios with user mobility. Note that the status information does not directly depend on the large-scale fading coefficients. In other words, each agent observes only different levels of interference based on the actions of each other. Thus, in order to obtain a good performance, the proposed method must be trained assuming different user positions and, consequently, different interference levels. The user mobility model is based on [49, Annex A], in which the user location should be updated every 20 time slots. Moreover, the results are an average of ten initial positions obtained randomly.

In Figs. 8 and 9 we show the performance of the proposed method when user mobility is considered. Specifically, Fig. 8 shows the minimum SE and sum SE in the exploration phase along the time with user speed equal to 3 km/h. For a better visualization, each point is an average of the previous time slots. That said, note that the DRL-based algorithms are able to learn to take good actions since the proposed methods outperform the state-of-art algorithms in a few time slots and the gap compared to those algorithms increases over time. Thus, we have that

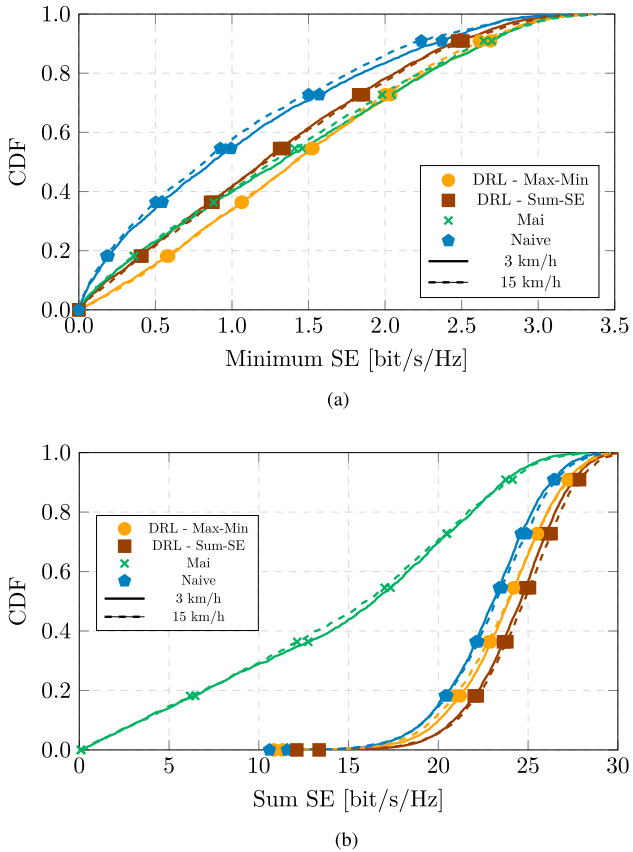


Fig. 9. Execution Phase with user mobility and $\{M, K, N, |\mathcal{M}_k|, \tau_s\} = \{100, 8, 8, 3, 4\}$. (a) Max-min SE. (b) Max sum-SE.

the proposed method can also learn to perform good actions even when user mobility is considered, improving the system performance in terms of minimum SE and sum SE compared to the benchmarking algorithms.

In Fig. 8 the agents are always training the neural network weights. Although this can be advantageous since agents are constantly learning, this can be computationally costly. Thus, the system operator can stop the training at any time and only actions are then taken from the neural networks. We denote this phase as the execution phase. The idea is to show that the proposed method can generalize the training for other users' positions and, consequently, different interference levels. Therefore, we dedicate 5,000 time slots for the execution phase in addition to the 10,000 time slots used for training. In Fig. 9 we present the performance of the proposed solutions considering different user speeds during the execution phase. As we can see the proposed DRL-based solutions are also able to outperform benchmarking algorithms during the execution phase. This result shows that the proposed DRL-based solutions can generalize the training performed during the exploration phase in scenarios with user mobility.

It is worth mentioning that even though the proposed solutions have achieved interesting results in scenarios with mobility, we believe that the performance of these solutions can be further improved, e.g., by considering some information related to user

positions in the state of the proposed DRL-based solutions, which is left for future works.

VII. CONCLUSION

In this paper, we investigated the JPDC and receive filter coefficients design in the uplink of cell-free systems. Specifically, two different objectives were considered, namely: 1) max-min SE and 2) max-sum SE. The formulated problems were verified to be non-convex and very difficult to be optimally solved. They were then reformulated and iteratively solved up to a local optimal solution by using SCA and GP. More importantly, decentralized solutions were proposed based on the multiple agents DRL. Signaling aspects for practical implementation of the decentralized solution were also provided. The numerical results showed that DRL-based decentralized solutions for JPDC in cell-free systems is feasible and can perform close to centralized solutions. Moreover, the decentralized solution outperformed the benchmarking algorithms in terms of minimum SE and sum SE for different scenarios and showed to be more efficient in dealing with pilot contamination. Finally, as perspectives for further studies we indicate the development of solutions that take into account quality-of-service requirements and extensions of the proposed framework considering high mobility scenarios.

REFERENCES

- [1] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, "Cell-free massive MIMO: Uniformly great service for everyone," in *Proc. IEEE 16th Int. Workshop Signal Process. Adv. Wireless Commun.*, 2015, pp. 201–205.
- [2] S.-H. Park, O. Simeone, O. Sahin, and S. S. Shitz, "Fronthaul compression for cloud radio access networks: Signal processing advances inspired by network information theory," *IEEE Signal Process. Mag.*, vol. 31, no. 6, pp. 69–79, Nov. 2014.
- [3] E. Björnson and L. Sanguinetti, "Making cell-free massive MIMO competitive with MMSE processing and centralized implementation," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 77–90, Jan. 2020.
- [4] E. Björnson, E. G. Larsson, and M. Debbah, "Massive MIMO for maximal spectral efficiency: How many users and pilots should be allocated?," *IEEE Trans. Wireless Commun.*, vol. 15, no. 2, pp. 1293–1308, Feb. 2016.
- [5] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, "Cell-free massive MIMO versus small cells," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1834–1850, Mar. 2017.
- [6] ö. T. Demir, E. Björnson, and L. Sanguinetti, "Foundations of user-centric cell-free massive MIMO," *Found. Trends Signal Process.*, vol. 14, no. 3/4, pp. 162–472, 2021.
- [7] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Commun. Surv. Tuts.*, vol. 21, no. 3, pp. 2224–2287, Mar. 2019.
- [8] N. C. Luong et al., "Applications of deep reinforcement learning in communications and networking: A survey," *IEEE Commun. Surv. Tuts.*, vol. 21, no. 4, pp. 3133–3174, Oct.-Dec. 2019.
- [9] I. M. Braga, E. d. O. Cavalcante, G. Fodor, Y. C. B. Silva, C. F. M. e Silva, and W. C. Freitas, "User scheduling based on multi-agent deep Q-learning for robust beamforming in multicell MISO systems," *IEEE Commun. Lett.*, vol. 24, no. 12, pp. 2809–2813, Dec. 2020.
- [10] Y. Zhang, W.-P. Zhu, and J. Ouyang, "Energy efficient pilot and data power allocation in multi-cell multi-user massive MIMO communication systems," in *Proc. IEEE 85th Veh. Technol. Conf. (VTC-Fall)*, 2016, pp. 1–5.
- [11] H. V. Cheng, E. Björnson, and E. G. Larsson, "Optimal pilot and payload power control in single-cell massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 65, no. 9, pp. 2363–2378, May 2017.
- [12] P. Zhao, G. Fodor, G. Dán, and M. Telek, "A game theoretic approach to uplink pilot and data power control in multi-cell multi-user MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 8707–8720, Sep. 2019.

- [13] T. H. Nguyen, T. K. Nguyen, H. D. Han, and V. D. Nguyen, "Optimal power control and load balancing for uplink cell-free multi-user massive MIMO," *IEEE Access*, vol. 6, pp. 14462–14473, 2018.
- [14] H. Q. Ngo, L.-N. Tran, T. Q. Duong, M. Matthaiou, and E. G. Larsson, "On the total energy efficiency of cell-free massive MIMO," *IEEE Trans. Green Commun. Netw.*, vol. 2, no. 1, pp. 25–39, Nov. 2018.
- [15] T. C. Mai, H. Q. Ngo, M. Egan, and T. Q. Duong, "Pilot power control for cell-free massive MIMO," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 11264–11268, Aug. 2018.
- [16] H. Masoumi and M. J. Emadi, "Joint pilot and data power control in cell-free massive MIMO system," in *Proc. IEEE 5th Int. Conf. mmWave Terahertz Technol.*, 2018, pp. 34–37.
- [17] I. M. Braga, R. P. Antonoli, G. Fodor, Y. C. B. Silva, and W. C. Freitas, "Joint pilot and data power control optimization in the uplink of user-centric cell-free systems," *IEEE Commun. Lett.*, vol. 26, no. 2, pp. 399–403, Feb. 2022.
- [18] E. Nayebe, A. Ashikhmin, T. L. Marzetta, and B. D. Rao, "Performance of cell-free massive MIMO systems with MMSE and LSFD receivers," in *Proc. IEEE 50th Asilomar Conf. Signals, Syst. Comput.*, 2016, pp. 203–207.
- [19] M. Bashar, K. Cumanan, A. G. Burr, M. Debbah, and H. Q. Ngo, "On the uplink max–min SINR of cell-free massive MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2021–2036, Jan. 2019.
- [20] M. Farooq, H. Q. Ngo, and L. Nam Tran, "A low-complexity approach for max-min fairness in uplink cell-free massive MIMO," in *Proc. IEEE 93rd Veh. Technol. Conf.*, 2021, pp. 1–6.
- [21] C. D'Andrea, A. Zappone, S. Buzzi, and M. Debbah, "Uplink power control in cell-free massive MIMO via deep learning," in *Proc. IEEE Internat. Workshop Comput. Adv. Multi-Sensor Adaptive Process.*, 2019, pp. 554–558.
- [22] N. Rajapaksha, K. B. S. Manosha, N. Rajatheva, and M. Latva-aho, "Deep learning-based power control for cell-free massive MIMO networks," in *Proc. IEEE Int. Conf. Commun.*, 2021, pp. 1–7.
- [23] R. Nikbakht, A. Jonsson, and A. Lozano, "Unsupervised-learning power control for cell-free wireless systems," in *Proc. IEEE Annu. Internat. Symp. Pers., Indoor Mobile Radio Commun.*, 2019, pp. 1–5.
- [24] W. Li, W. Ni, H. Tian, and M. Hua, "Deep reinforcement learning for energy-efficient beamforming design in cell-free networks," in *Proc. IEEE Wireless Commun. Netw. Conf. Workshops*, 2021, pp. 1–6.
- [25] Y. Al-Eryani, M. Akrouf, and E. Hossain, "Multiple access in cell-free networks: Outage performance, dynamic clustering, and deep reinforcement learning-based design," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 4, pp. 1028–1042, Apr. 2021.
- [26] F. Fredj, Y. Al-Eryani, S. Maghsudi, M. Akrouf, and E. Hossain, "Distributed uplink beamforming in cell-free networks using deep reinforcement learning," 2020, *arXiv:2006.15138*.
- [27] R. Wang, M. Shen, Y. He, and X. Liu, "Performance of cell-free massive MIMO with joint user clustering and access point selection," *IEEE Access*, vol. 9, pp. 40860–40870, 2021.
- [28] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
- [29] Z.-Q. Luo and S. Zhang, "Dynamic spectrum management: Complexity and duality," *IEEE J. Sel. Areas Commun.*, vol. 2, no. 1, pp. 57–73, Feb. 2008.
- [30] S. Boyd, S.-J. Kim, L. Vandenberghe, and A. Hassibi, "A tutorial on geometric programming," *Optim. Eng.*, vol. 8, no. 1, pp. 67–127, Apr. 2007.
- [31] M. Chiang, C. W. Tan, D. P. Palomar, D. O'Neill, and D. Julian, "Power control by geometric programming," *IEEE Trans. Wireless Commun.*, vol. 6, no. 7, pp. 2640–2651, Jul. 2007.
- [32] E. D. Andersen, B. Jensen, J. Jensen, R. Sandvik, and U. Worsøe, "MOSEK version 6," Tech. Rep. C2009C3, Oct. 2009. [Online]. Available: <https://docs.mosek.com/whitepapers/mosek6.pdf>
- [33] S. Diamond and S. Boyd, "CVXPY: A python-embedded modeling language for convex optimization," *J. Mach. Learn. Res.*, vol. 17, no. 83, pp. 1–5, 2016.
- [34] B. R. Marks and G. P. Wright, "Technical note a general inner approximation algorithm for nonconvex mathematical programs," *Oper. Res.*, vol. 26, no. 4, pp. 525–683, Aug. 1978.
- [35] I. M. Braga, R. P. Antonoli, G. Fodor, Y. C. B. Silva, C. F. M. e Silva, and W. C. Freitas, "Joint resource allocation and transceiver design for sum-rate maximization under latency constraints in multicell MU-MIMO systems," *IEEE Trans. Commun.*, vol. 69, no. 7, pp. 4569–4584, Jul. 2021.
- [36] E. Dahlman, S. Parkvall, and J. Skold, *5G NR: The Next Generation Wireless Access Technology*, 2nd ed. Cambridge, MA, USA: Academic Press, Oct. 2020.
- [37] E. Dahlman, S. Parkvall, and J. Skold, *4G, LTE-Advanced Pro and the Road to 5G*, 3rd ed. Cambridge, MA, USA: Academic Press, 2016.
- [38] H. Asplund et al., *Advanced Antenna Systems for 5G Network Deployments: Bridging the Gap Between Theory and Practice*. Cambridge, MA, USA: Academic Press, 2020.
- [39] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.
- [40] C. Zhong, Z. Lu, M. C. Gursoy, and S. Velipasalar, "A deep actor-critic reinforcement learning framework for dynamic multichannel access," *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 4, pp. 1125–1139, Dec. 2019.
- [41] E. Björnson, J. Hoydis, and L. Sanguinetti, "Massive MIMO networks: Spectral, energy, and hardware efficiency," *Found. Trends Signal Process.*, vol. 11, no. 3/4, pp. 154–655, Nov. 2017.
- [42] P.-W. Chou, D. Maturana, and S. Scherer, "Improving stochastic policy gradients in continuous control with deep reinforcement learning using the beta distribution," in *Proc. 34th Int. Conf. Mach. Learning. PMLR*, 2017, pp. 834–843.
- [43] H.-S. Lee, J.-Y. Kim, and J.-W. Lee, "Resource allocation in wireless networks with deep reinforcement learning: A circumstance-independent approach," *IEEE Syst. J.*, vol. 14, no. 2, pp. 2589–2592, Jun. 2020.
- [44] J. Saraiva et al., "Deep reinforcement learning for QoS-constrained resource allocation in multiservice networks," *J. Commun. Inf. Syst.*, vol. 35, no. 1, pp. 66–76, Apr. 2020.
- [45] A. Tolli et al., "Distributed coordinated transmission with forward-backward training for 5G radio access," *IEEE Commun. Mag.*, vol. 57, no. 1, pp. 58–64, Jan. 2019.
- [46] Z. Wang, E. K. Tameh, and A. R. Nix, "Joint shadowing process in urban peer-to-peer radio channels," *IEEE Trans. Veh. Technol.*, vol. 57, no. 1, pp. 52–64, Jan. 2008.
- [47] M. Abadi et al., "TensorFlow: A system for large-scale machine learning," in *Proc. USENIX Symp. Operating Syst. Des. Implementation*, 2016, pp. 265–283.
- [48] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–15.
- [49] 3rd Generation Partnership Project (3GPP), Study on LTE-based V2X services," Release 14 v14.0.0, 3GPP, Sophia Antipolis, France, Tech. Rep. 36.885, Jun. 2016.



Iran Mesquita Braga Jr. received the B.Sc. degree in computer engineering and the M.Sc. degree in electrical and computer engineering, from the Federal University of Ceará (UFC), Sobral, Brazil, in 2017 and 2019, respectively. He is currently working toward the D.Sc. degree in telecommunications engineering with UFC, Fortaleza, Brazil and, since 2018, he has been a Researcher with Wireless Telecom Research Group, UFC, participating of projects in a technical and scientific cooperation with Ericsson Research. His research interests include radio resource management, machine-learning techniques, numerical optimization and multiuser/multiantenna communications.



Roberto Pinto Antonoli received the B.Sc. degree in teleinformatics engineering (*magna cum laude*) from the Federal University of Ceará (UFC), Fortaleza, Brazil, in 2016, and the M.Sc. and Ph.D. degrees in teleinformatics engineering also from UFC in 2017 and 2020, respectively. He currently holds a Post-doc. position with the Wireless Telecom Research Group (GTEL), UFC, where he works on projects in technical and scientific cooperation with Ericsson Research. He is also a Software Developer with Instituto Atlântico, Fortaleza, Brazil. In 2018/2019, he was a Visiting Researcher with Ericsson Research, Sweden. His research interests include 5G wireless communication networks with multiple radio access technologies and multi-connectivity, and also scheduling algorithms for QoS provision.



Gábor Fodor (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the Budapest University of Technology and Economics, Budapest, Hungary, in 1998, and the D.Sc. degree from the Hungarian Academy of Sciences (doctor of MTA), Budapest, Hungary, in 2019. He is currently a Master Researcher with Ericsson Research and a docent and an Adjunct Professor with the KTH Royal Institute of Technology, Stockholm, Sweden. He has authored or coauthored more than 100 refereed journal and conference papers, seven book chapters and more than 100 European and U.S. granted patents. He was the corecipient of the IEEE Communications Society Stephen O. Rice prize in 2018 and the Best Student Conference Paper Award by the IEEE Sweden VT/COM/IT Chapter in 2018. Dr. Fodor is currently the Chair of the IEEE Communications Society Emerging Technology Initiative on Full Duplex Communications. Between 2017 and 2020 was also a Member of the board of the IEEE Sweden joint Communications, Information Theory and Vehicle Technology chapter. He is currently the Editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, a Guest Editor for the *IEEE Communications Magazine* (Special Issue on Terahertz communications) and a Guest Editor for the IEEE WIRELESS COMMUNICATIONS (Special Issue on full-duplex communications).



Yuri C. B. Silva received the B.Sc. and M.Sc. degrees in electrical engineering from the Federal University of Ceará, Fortaleza, Brazil, in 2002 and 2004, respectively, and the Dr.-Ing. degree in electrical engineering from the Technische Universität Darmstadt, Darmstadt, Germany, in 2008. From 2001 to 2004 he was with the Wireless Telecom Research Group (GTEL), Fortaleza, Brazil. In 2003 he was a Visiting Researcher at Ericsson Research, Stockholm, Sweden. From 2005 to 2008, he was with the Communications Engineering Laboratory of the Technische Universität Darmstadt. He is currently an Associate Professor with the Federal University of Ceará and Researcher at GTEL. He also holds a productivity fellowship in technological development and innovation from CNPq. His main research interests include wireless communications systems, multi-antenna processing, interference management, multicast services, and cooperative communications.



Walter C. Freitas Jr. received the B.S. and M.S. degrees in electrical engineering from the Federal University of Ceará, Fortaleza, Brazil, and the Ph.D. degree in teleinformatic engineering from the Federal University of Ceará, in 2006. During his studies, he was supported by the Brazilian Agency FUNCAP and Ericsson. During Q3 of 2015 up to Q2 of 2016, he was a Postdoc Researcher with I3S/CNRS Laboratory, from the University of Nice, Sophia Antipolis, France. During 2005 Walter Freitas Jr. was a Senior Research of the Nokia Technology Institute. He is currently an Assistant Professor with the Department of Teleinformatics Engineering of the Federal University of Ceará and Researcher of Wireless Telecom Research Group one of the most important research groups in telecommunication in Brazil. His main research interests include features development to improve the performance of the wireless communication systems, interference avoidance tools, multilinear algebra, and tensor-based signal processing applied to communications.