Scheduling for Massive MIMO With Hybrid Precoding Using Contextual Multi-Armed Bandits

Weskley V. F. Mauricio, Tarcisio Ferreira Maciel, Anja Klein, *Member, IEEE*,
and Francisco Rafael Marques Lima, *Senior Member, IEEE*

***Abstract*—In this work we study different scheduling problems in the downlink of a Frequency Division Duplex multiuser wireless system that employs a hybrid precoding antenna architecture for massive Multiple Input Multiple Output. In this context, we propose a scheduling framework using Reinforcement Learning (RL) tools, namely Contextual Multi-Armed Bandits (CMAB), that can dynamically adapt themselves to solve three scheduling problems, which are: i) Maximum Throughput (MT); ii) Maximum Throughput with Fairness Guarantees (MTFG), and; iii) Maximum Throughput with QoS Guarantees (MTQG), which are well-known relevant problems. Before performing scheduling itself, we exploit statistical Channel State Information (CSI) to create clusters of spatially compatible User Equipmentss (UEss). This structure, combined with the usage of Zero-Forcing precoding, allows us to reduce the scheduler complexity by considering each cluster as an independent virtual RL scheduling agent. Next, we apply a new learning-based scheduler aiming to optimize the desired system performance metric. Moreover, only scheduled UEss need to feed back instantaneous equivalent CSI, which also reduces the signaling overhead of the proposal. The superiority of the proposed framework is demonstrated through numerical simulations in comparison with reference solutions.**

***Index Terms*—Multi-Armed Bandits, Reinforcement Learning, Scheduling, Massive MIMO.**

I. Introduction

IN THIS section, we visit some aspects of massive Multiple Input Multiple Output (MIMO), hybrid precoding, resource allocation and clustering in 5th Generation (5G) systems that motivate our proposed scheduling solutions, as well as we describe the basic concepts of CMAB within the RL field on top of which our proposals are built. Lastly, we show how the article is organized.

Nowadays, the potential gains of massive MIMO turn it into a key technology for 5G systems [1], [2]. Indeed, using a massive number of antennas at the Base Station (BS) allows to schedule multiple UEss in the same time slot and frequency resource, thus increasing spatial reuse and spectral efficiency. However, for massive MIMO the usage of the classical fully-digital precoding is impractical, since it needs as many Radio Frequency (RF) chains as antennas resulting into unsustainable costs and power consumption. One way to counteract these disadvantages is to split precoding into analog and digital domains, an approach also known as hybrid precoding. Its main idea is to reduce the number of RF chains by using a base band digital precoding stage of low dimension followed by a high dimension analog precoding stage implemented with phase shifters only [3].

The massive MIMO with hybrid precoding architecture brings several challenges, such as the joint optimization of analog and digital precoding, which leads to a non-convex optimization problem. Moreover, only phase shifters are used to implement the analog precoder, which imposes additional constraints in the precoding design. Also, scheduling in massive MIMO systems is challenging due to the inherent non-orthogonality among UEss, which can lead to high Multi-User (MU) interference and affects the system performance negatively. Finally, the number of scheduling possibilities grows rapidly with the number of UEss and available RF chains [4], turning it into a combinatorial problem. This way, the use of traditional rule-based algorithms is challenging in schedulers with such large dimensionality [5].

Another difficulty in scheduling in massive MIMO systems is how to perform it efficiently only with partial CSI, since UEss often have to be scheduled without complete knowledge of their full instantaneous channel at the BS. In this context, RL tools appear as a very suitable solution to this problem due to their capability of operating under the lack of information and still achieving good performances in the long run [6]. Also, RL is very suitable for scheduling solutions since it aims to make an agent learn how to behave in an environment to optimize a predetermined objective, such as the system performance metrics considered by the scheduler, e.g., throughput, fairness or Quality of Service (QoS). Herein, we highlight the particular case of RL solutions known as Contextual Multi-Armed Bandits (CMAB) [7]. In CMAB, the learning agent observes some side

Manuscript received December 27, 2021; revised March 3, 2022; accepted March 27, 2022. Date of publication April 12, 2022; date of current version July 18, 2022. The work of Francisco Rafael Marques Lima was supported by the FUNCAP (edital BPI) under Grant BP4-0172-00245.01.00/20. This work was supported in part by FUNCAP, CAPES, and CNPq, in part by CAPES under Projects PROBRAL 88887.144009/2017-00 and PRINT 88887.311965/2018-00, in part by DAAD with funds from the Federal Ministry of Education and Research (BMBF) and performed in the context of the DFG Collaborative Research Center (CRC) 1053 MAKI - sub-projects C1 and B3, and in part by the BMBF Project Open6GHub under Grant 16KISK014. The review of this article was coordinated by Dr. M. Shahgir Ahmed. *(Corresponding author: Weskley V. F. Mauricio.)*

Weskley V. F. Mauricio is with Wireless Telecomunnications Research Group (GTEL), Federal University of Ceará CEP 60416200, Brazil, and also with Communications Engineering Lab, CEP 64283 Darmstadt, TU, Germany (e-mail: weskleyvfm@gmail.com).

Tarcisio Ferreira Maciel and Francisco Rafael Marques Lima are with Wireless Telecomunnications Research Group (GTEL), Federal University of Ceará, Brazil (e-mail: maciel@gtel.ufc.br; rafaelm@gtel.ufc.br).

Anja Klein is with Communications Engineering Lab, Darmstadt, TU, Germany (e-mail: a.klein@nt.tu-darmstadt.de).

Digital Object Identifier 10.1109/TVT.2022.3166654

information (e.g., outdated CSI) called context, and, based on that, chooses an action (e.g., schedules certain UEss) obtaining a reward (e.g., throughput) from it. Then, the expected reward, called action value, can be calculated by averaging the obtained rewards of that action over time. Therefore, at each iteration, each decision is made based on the current context and current action values aiming at maximizing the expected cumulative (long run) reward. CMAB solutions have to obtain a balance between increasing the information about the action values of different actions (exploration) and selecting the actions with higher action values (exploitation) to reach this goal. The introduction of multi-agents in the model was shown to fit very well to solve wireless communication problems, such as interference coordination [8], [9]. Furthermore, there are many variations of the CMAB model that make this solution capable of being applied in several scenarios of wireless network systems [10].

Another recurrent issue in Frequency Division Duplex (FDD) massive MIMO systems is the signaling overhead involved to obtain the CSI at the transmitter for the large number of employed antennas. Motivated by this, in [11] a two-stage precoding exploiting the spatial correlation among UEss channels, named Joint Spatial Division and Multiplexing (JSDM), was proposed to reduce the amount of required CSI. This work became the basis for several studies in this area, such as [4], [12]–[17].

The remainder of this article is organized as follows: in Section II, we present the state of the art and our main contributions. In Section III, we describe our adopted system model. In Section IV, the statistical CSI and used clustering algorithm employed in our proposal are described. Then, in Section V the analog and digital precoding design is presented. Afterwards, in Section VI we firstly show how clustering and Zero-Forcing (ZF) digital precoding can be used to reduce the scheduling search space of our proposal. In the same section, we propose the three learning based schedulers of our framework. Section VII shows the numerical results of the proposed solutions. Finally, Section VIII presents our main conclusions.

II. Related Works and Main Contributions

In this section, we present some related works about scheduling, as well as we describe afterwards our main contributions and our proposed scheduling framework.

Scheduling with QoS guarantees is investigated in references [18]–[20]. In reference [18], the authors study the problem of throughput maximization guaranteeing the QoS requirements considering multiple services. The proposed scheduler, termed Joint Satisfaction Maximization (JSM), utilizes derivatives of a sigmoidal function that are dynamically adapted to protect the most prioritized service satisfying the UEss' QoS requirements. In reference [19], the authors propose a new low complex scheduling method based on graph theory aiming at maximizing the throughput considering QoS requirements in an Orthogonal Frequency Division Multiple Access (OFDMA) wireless network. In reference [20], the authors propose a scheduling with the objective of balancing energy-efficiency and fairness among UEss considering QoS requirements in an OFDMA system. Therein, the scheduler is divided into two parts: the first part schedules the UEss aiming at achieving fairness among UEss and the QoS requirements, whereas the second part employs a power allocation algorithm to achieve the maximum energy efficiency with the already scheduled UEss. Despite their relevant contributions, the aforementioned works [18]–[20] do not consider a MIMO scenario, which includes the challenge of managing spatial resources besides the already considered time and frequency resources.

As will be discussed in the sequel, several works propose scheduling solutions for massive MIMO systems considering fully digital precoding. In reference [11], the authors propose a solution that first creates clusters of UEss with similar spatial channel covariance (statistical CSI) using K-means algorithm [21]. Then, this statistical CSI of the UEss in a cluster is used to create an outer precoder that nearly suppresses the inter-cluster interference. Afterwards, UEss are suitably (e.g., by the scheduler) polled by the BS for their equivalent instantaneous CSI, which takes into account the UEss' channels and the cluster outer precoder. This equivalent instantaneous CSI has a smaller dimension than the full instantaneous channel (implying less signaling) and it is used to create an inner precoder that suppresses the UEss intra-cluster interference. In reference [12], a graph theory-based clustering and scheduling method is proposed to maximize the system throughput while guaranteeing fairness. This same objective of maximizing the system throughput while guaranteeing fairness has also been studied by [13] and [14]. In reference [13] the authors propose an RL based scheduling solution, where each UEs is considered as an autonomous agent that makes its own Resource Block (RB) allocation. In reference [14] the authors propose a greedy scheduling that selects the UEss based on their channel gains. In reference [22], the authors propose a cooperative scheduling for a massive MIMO scenario aiming at reducing the impact of pilot contamination. The developed scheduler also deals with three objectives: maximizing the system data rate, maximizing the Jain's fairness index, and guaranteeing a balanced trade-off between them. In reference [23], the authors propose a scheduling strategy aiming at maximizing the energy efficiency guaranteeing minimum data rate to the UEss in a multi-cell massive MIMO scenario. The authors propose a new beamforming technique that serves different sets of UEss in small time-slot fractions, therefore, different sets of UEss are served in an entire time-slot. Although the proposals in [11]–[14], [22], [23] have their own merits, the assumption of fully-digital precoding is hard to hold in practice when massive MIMO is considered as previously explained.

In references [4], [15]–[17], [24], [25], new scheduling schemes to massive MIMO are proposed using hybrid precoding. In reference [4], the authors propose a scheduler that uses only statistical CSI to select the UEss aiming at maximizing the system throughput. It schedules the UEss using a parameter that controls a trade-off between channel gain and spatial channel correlation. The authors in [15] propose a new scheduling method based on matrix vectorization. The scheduler vectorizes the channel matrix and creates groups of UEss based on Pearson's correlation coefficient. Afterwards, a set of UEss is selected from existing groups aiming at maximizing the throughput. In [16] the authors also proposed a scheduler based only on statistical CSI, however, their objective is to maximize the throughput while guaranteeing the fairness among UEss. The problem is formulated based on Lyapunov-drift optimization, which models the UEss priority based on their transmission history creating virtual queues. Afterwards, a low-complexity greedy algorithm is proposed to obtain near-optimal performance. In reference [17], the authors analyzed two scheduling strategies based on statistical CSI aiming at maximizing the throughput in a scenario where the UEss are moving at high speeds. The scheduling strategies are semi-orthogonal user selection and a greedy algorithm. Since the schedulers are based on statistical CSI, the same scheduled UEss are served in subsequent Transmission Time Intervals (TTIs) without rescheduling while the UEss are moving. Simulation results showed that rescheduling is necessary, otherwise throughput drops over time. In reference [24], the authors propose a joint antenna selection and UEs scheduling aiming at maximizing the system data rate and achieving energy efficiency using limited number of RF chains. To solve this problem, the authors modeled UEs scheduling and antenna selection using learning-based stochastic gradient descent method. In reference [25], the authors study a centralized architecture to improve the energy efficiency and to maximize the system data rate for a multi-cell massive MIMO scenario. The authors divide the UEss in clusters, where each cluster has its central processing unit. These clusters exchange limited information among themselves in order to serve the scheduled UEss and select the BSs that will serve each of them aiming at maximizing the system data rate. Despite their contributions, the aforementioned works [4], [15]–[17], [24], [25] do not take into account the mandatory features of modern wireless networks, such as QoS provisioning and support of multiple services. These features impose additional constraints on the optimization problem as well as more challenges since the UEss have different QoS demands and channel quality states. In reference [26], the authors study scheduling in massive MIMO considering hybrid beamforming using optimization tools aiming at maximizing the throughput with QoS guarantees. However, the work in [26] does not study fairness among UEss and does not consider the impact of scheduling along consecutive slots.

Since obtaining CSI is one of the principal issues of massive MIMO, the CSI usage by schedulers should be taken into account. It is important to notice that most of the presented works assume that instantaneous CSI is always available in the scheduler part, such as references [12], [13], [15], [18]–[20] The aforementioned works that use only the statistical information as main scheduler parameter are [4], [16] and [17]. The advantage of using statistical CSI is that it reduces the signaling overhead since the estimation can be done without dedicated pilots and the statistical CSI variation speed is much lower than that of the instantaneous CSI [16].

RL-based scheduling has gained popularity and has been applied in different contexts with different objectives [27]–[29]. In reference [27], the authors propose a combinatorial Multi-Armed Bandit (MAB) based scheduler to solve the joint mode selection and resource allocation in a device-to-device system. They reduced the action space and improved the algorithm learning speed dividing the problem into two stages, which reduces the complexity of the problem. The first stage is responsible for scheduling only the cellular UEss and aims at maximizing the throughput. The second stage is responsible for scheduling the device-to-device pairs aiming at maximizing the system throughput. In reference [28], the authors propose an actor-critic RL-based scheduling aiming at maximizing the fairness among UEss maintaining QoS in Long Term Evolution (LTE) systems. In reference [29], the authors propose an RL and neural networks based framework aiming at guaranteeing the QoS requirements for different types of services in OFDMA ultra-reliable low-latency communications. They evaluate the framework performance over different RL algorithms. The aforementioned RL based schedulers [27]–[29], as many other RL-based schedulers, model their problem, environment, action, state, and reward space, taking into account their predetermined scenario and assumptions. Thus, their modeling does not fit on our problem. Therefore, we consider that a novel RL based scheduler is needed to address this complex scenario containing a massive number of antennas, hybrid beamforming, different objectives, and services. Table I shows the main aspects covered by the presented articles in our literature reviews.

In summary, none of the presented works in this literature review has jointly considered the following features: massive MIMO systems with hybrid precoding architecture, different objectives (throughput maximization, throughput maximization considering fairness among UEss, and throughput maximization considering QoS requirements), RL-based scheduler, multiple services and a scheduler that does not require instantaneous CSI. In this article, we do consider all these aspects. Therefore, the main contributions of this work are:

- A framework that leverages RL in order to schedule UEss in a massive MIMO scenario with hybrid precoding is proposed that has the following advantages:
  - The framework is based on virtual learning agents, where the BS is the physical entity, and the UEs clusters are the logical entities.

    It learns from past experience about the spatial compatibility of the different UEss inside a cluster and how to select the UEss to achieve a given objective.
  - To perform scheduling, virtual agents do not require instantaneous CSI of UEss for the compatibility check, but instead the system sum data rate (reward) of UEss scheduled together in the past, thus reducing the signaling overhead (feed back costs).

    Note that the system sum data rate considered in this paper is equal to the sum of data rates achieved by the scheduled UEss in the current TTI.

    Furthermore, the instantaneous CSI is only reported for the scheduled UEss reducing even more the signaling overhead.
  - The framework is capable of supporting UEss with different data rate requirements (multiple services).
- The proposed framework supports three different objectives, which can dynamically configure/adapt the objective to consider throughput maximization, fairness guarantees, or balance between throughput maximization and QoS provisioning.

TABLE I
SUMMARY OF COVERED FEATURES BY PAPER

Paper	Features						
-	Throughput	Fairness	QoS	Massive MIMO	Hybrid Precoder	RL	Imperfect CSI
Our paper	✓	✓	✓	✓	✓	✓	✓
[18]	✓		✓				
[19]			✓				
[20]		✓	✓				✓
[11], [12], [14]	✓	✓	✓	✓			✓
[13]	✓	✓		✓		✓	✓
[22]	✓	✓	✓	✓			✓
[23]			✓	✓			✓
[4], [15], [24], [25]	✓			✓	✓		✓
[16], [17]	✓	✓		✓	✓		✓
[26]	✓		✓	✓	✓		✓
[27]	✓					✓	
[28], [29]	✓		✓			✓	

- Performance evaluation and comparison of the proposed framework in a massive MIMO scenario with hybrid precoding against several reference solutions from the literature is conducted.

III. System Model

In this section, we describe the scenario considered in our work. In the sequel, we present the hybrid precoding and received signal followed by how the Signal to Interference-plus-Noise Ratio (SINR) and data rate are calculated.

We consider the downlink (DL) of a single-cell FDD massive MIMO system based on OFDMA. The system is formed by a BS equipped with a Uniform Planar Array (UPA) with $N \gg 1$ transmit antennas serving J omnidirectional single-antenna UEss. We consider that the smallest allocable resource unit, termed RB, has N_{symb} consecutive Orthogonal Frequency Division Multiplexing (OFDM) symbols and N_{sc} adjacent OFDMA subcarriers. Moreover, we also consider that the channel remains nearly constant within an RB, i.e., during one TTI. Therefore, for a given RB, by $\boldsymbol{H} \in \mathbb{C}^{J\times N}$ we denote the channel matrix between the BS and all UEss, whose coefficients are taken considering the middle subcarrier and first OFDM symbol of the RB. Notice that the channel between the BS and UEs j corresponds to the j-th row of $\boldsymbol{H}$ and that at this point we omit the index for the RB. Later, when referring to channel and precoding matrices, these will be indexed to a specific RB whenever necessary. Also, in this paper, the considered channels are correlated.

At each TTI, the BS can schedule K out of J UEss to receive data at the same RB using Space Division Multiplexing. Since we consider hybrid precoding [30], the information vector $\boldsymbol{x} \in \mathbb{C}^{K\times 1}$ is transmitted using a hybrid precoding matrix $\boldsymbol{W} \in \mathbb{C}^{N\times K}$. Moreover, the product of the analog precoder $\boldsymbol{W}_{\text{RF}} \in \mathbb{C}^{N\times K}$ and digital precoder $\boldsymbol{W}_{\text{BB}} \in \mathbb{C}^{K\times K}$ composes the hybrid precoding matrix $\boldsymbol{W}$.

Using hybrid precoding, we can create a reduced equivalent channel $\boldsymbol{H}_{\text{eq}} = \tilde{\boldsymbol{H}}\boldsymbol{W}_{\text{RF}} \in \mathbb{C}^{K\times K}$ to be used as the effective DL channel, where $\tilde{\boldsymbol{H}} \in \mathbb{C}^{K\times N}$ is the channel matrix of the K scheduled UEss composed by taking from $\boldsymbol{H}$ only the rows associated to the scheduled UEss[11]. An advantage of this approach is that in general $K \ll N$[31], so that $\boldsymbol{H}_{\text{eq}}$ has a much lower dimension than $\tilde{\boldsymbol{H}}$.

In this paper, we consider as available information only the statistical CSI and the equivalent channel of the scheduled UEss. The statistical CSI varies only at a slow speed and, therefore, reduces the frequency at which CSI estimation is required. The equivalent channel of the scheduled UEss has a much lower dimension than the entire channel matrix. Although, in practical scenarios, the CSI is never perfectly available (even the statistical one), as discussed in [4, 16, 17, 26], our approach of assuming statistical CSI and a reduced-dimension equivalent channel, gets closer to a practical scenario

In order to have an Equal Power Allocation (EPA) among RBs, we consider that $\|\boldsymbol{W}_{\text{RF}}\boldsymbol{W}_{\text{BB}}\sqrt{\boldsymbol{P}}\|_{\text{F}}^2 = P_{\text{RB}}$ for each RB, where $\boldsymbol{P} \in \mathbb{R}_+^{K\times K}$ is a diagonal power matrix whose elements on its diagonal are the power values allocated to each UEs k scheduled on the RB and P_{RB} is the power available to each RB. Later, in Section V, we will describe the adopted design for the analog precoder $\boldsymbol{W}_{\text{RF}}$ and digital precoder $\boldsymbol{W}_{\text{BB}}$. Furthermore, since we are using EPA, the power will be allocated equally between the UEss, independent of the UEs path loss.

Now, for a given RB, we can express the receive signal vector $\boldsymbol{y} \in \mathbb{C}^{K\times 1}$ of the K scheduled UEss as

$$\boldsymbol{y} = \tilde{\boldsymbol{H}}\boldsymbol{W}\sqrt{\boldsymbol{P}}\boldsymbol{x} + \boldsymbol{z}, \tag{1}$$

where $\boldsymbol{z} \in \mathbb{C}^{K\times 1}$ is an additive Gaussian noise vector whose elements are Independent and Identically Distributed (IID) as $\mathcal{CN}(\boldsymbol{0}, \sigma^2\boldsymbol{I}_K)$ and $\boldsymbol{I}_K$ is a $K \times K$ identity matrix, and σ^2 is the average noise power.

Then, using 1, we define $\boldsymbol{A} = [a]_{i,j} = \tilde{\boldsymbol{H}}\boldsymbol{W}\sqrt{\boldsymbol{P}} \in \mathbb{C}^{K\times K}$ to express the average SINR perceived by the k-th scheduled UEs as

$$\gamma_k = \frac{|a_{k,k}{}^2}{\sigma|^2 + \sum\limits_{j\neq k}^{K} |a_{k,j}|^2}. \tag{2}$$

Using 2, we apply Shannon's formula to calculate data rate r_k of the UEs k as

$$r_k = N_{\text{sc}}N_{\text{symb}} \min\{\log_2(1+\gamma_k), 8\} \text{ bits/TTI}. \tag{3}$$

where $\min\{x, 8\}$ refers to our modulation order upper bound using 256-Quadrature Amplitude Modulation (QAM), which is the highest modulation order supported by 5G New Radio (NR) systems [32].

We utilize Quasi Deterministic Radio Channel Generator (QuaDRiGa) [33], which is a generic channel model that simulates different wireless communication environments considering the same method for generating channel coefficients. This channel model supports scenario transitions, time-evolving channel, and variable UEss speeds. In this paper, we used both Urban Micro (UMi) Line Of Sight (LOS) and Non-Line Of Sight (NLOS) scenarios and some of the parameters to generate these environments are small fading, shadow fading, path loss, number of clusters, delay, and angular spread, and so on. References [33], [34] provide a detailed description of all these parameters.

IV. Statistical CSI and Clustering Algorithm

In this section, we describe how the spatial covariance matrix and its eigendecomposition are considered in this work. The information they convey corresponds to the statistical CSI used in our study by the K-means algorithm to cluster the UEss, as also described later in this section. Note that, this is not a new method but the method we used is explained to make the paper self-contained.

As adopted in [2], [4], [11], we assume that UEss are split into clusters based on statistical CSI only. As we will show in Section V, the clustering information is fundamental to the analog precoder design. The use of statistical CSI is highly beneficial in FDD systems since this information varies only at a slow speed and, therefore, reduces the frequency at which CSI estimation is required.

Herein, the covariance matrix is given as in [4], [11] by

$$\boldsymbol{R}_j = \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{h}_{t,j}^{\mathrm{H}} \boldsymbol{h}_{t,j}, \tag{4}$$

where $\boldsymbol{h}_{t,j} \in \mathbb{C}^{1\times N}$ is the channel of the j-th UEs (j-th row of $\boldsymbol{H}$) in TTI t and T is the number of TTIs considered to average (and hence approximate) the channel covariance matrix, which we can decompose as

$$\boldsymbol{R}_j = \boldsymbol{U}_j \boldsymbol{\Lambda}_j \boldsymbol{U}_j^{\mathrm{H}}, \tag{5}$$

where $\boldsymbol{U}_j \in \mathbb{C}^{N\times N}$ and $\boldsymbol{\Lambda}_j \in \mathbb{R}^{N\times N}$ contain the eigenvectors and eigenvalues of $\boldsymbol{R}_j$, respectively.

In the following, we describe the K-means clustering algorithm, which partitions the J UEss into C clusters [21]. Note that, although we use K-means, we are not limited to it and other clustering methods could be employed such as agglomerative clustering [2], fuzzy c-means [35] or K-medoids [36]. Herein, the K-means algorithm uses the dominant eigenvector $\boldsymbol{u}_{j,1}$ of each UEs j from the covariance matrix $\boldsymbol{U}_j$ of 5 as input. Firstly, the algorithm randomly chooses C UEss and considers their dominant eigenvectors as the initial cluster's centroids. After that, the UEss are associated to the cluster that minimizes the Euclidean distance between their dominant eigenvectors, i.e., to the nearest cluster. Then, the mean of the dominant eigenvectors from the UEss belonging to each cluster determines the cluster's new (updated) centroid. The K-means algorithm repeats this process until there is no change in the UEs-to-cluster association, or a maximum number of iterations is reached. For more details on the K-means, please refer to [21]. A pseudo-code for K-means is presented in 1.

Algorithm 1: K-means algorithm.

```
Choose randomly the dominant eigenvector of $C$ UEss as the initial clusters' centroids;
while UEss-to-cluster assignments and clusters' centroids do not converge or the maximum number of iterations is not reached do
    Assign UEss to the closest cluster;
    Update clusters' centroids;
    Increment the iteration counter;
end while
```

V. Analog and Digital Precoding Design

In this section, we describe how the analog and digital precoding considered in our hybrid precoding scheme are determined. Note that, these are not new methods but the methods we used are explained to make the paper self-contained. In general, only phase shifters are used to implement the analog precoders. Herein, the analog precoders $\boldsymbol{w}_{\mathrm{RF},k}$ are defined in the scheduling part using the phases from the dominant eigenvectors $\boldsymbol{u}_{k,1}$ of the scheduled UEss k. This way, we define the analog precoder of the K scheduled UEss as

$$\boldsymbol{W}_{\mathrm{RF}} = \frac{1}{\sqrt{N}}\begin{bmatrix} \mathrm{e}^{i\angle \boldsymbol{u}_{1,1}} & \mathrm{e}^{i\angle \boldsymbol{u}_{2,1}} & \dots & \mathrm{e}^{i\angle \boldsymbol{u}_{K,1}} \end{bmatrix}, \tag{6}$$

where $\angle \boldsymbol{u}_{k,1}$ extracts the phases of the eigenvector of the scheduled UEs k, and $i = \sqrt{-1}$. For more details on analog precoder design, please refer to [30].

Since the signal transmitted from the BS serving the UEss belonging to cluster x produces negligible interference at the UEss in cluster y [4], [11].

Besides dealing with the inter-cluster interference, we also need to suppress the intra-cluster interference when multiple UEss of a same cluster are scheduled. To deal with intra-cluster interference, we use a ZF digital precoder [37] defined as

$$\boldsymbol{W}_{\mathrm{BB}} = \frac{\boldsymbol{H}_{\mathrm{eq}}^{\mathrm{H}}\left(\boldsymbol{H}_{\mathrm{eq}}\boldsymbol{H}_{\mathrm{eq}}^{\mathrm{H}}\right)^{-1}}{\|\boldsymbol{H}_{\mathrm{eq}}^{\mathrm{H}}(\boldsymbol{H}_{\mathrm{eq}}\boldsymbol{H}_{\mathrm{eq}}^{\mathrm{H}})^{-1}\|_F}. \tag{7}$$

Moreover, we are going to see in the next section that the characteristics of the ZF precoder can be exploited to reduce the problem complexity.

VI. Proposed Scheduling

In this section, we propose three solutions based on CMAB theory aiming at maximizing throughput, fairness, and assuring QoS requirements, respectively, whose action space is reduced by taking advantage of the interference mitigation properties of ZF precoding and clustering described in the previous sections.

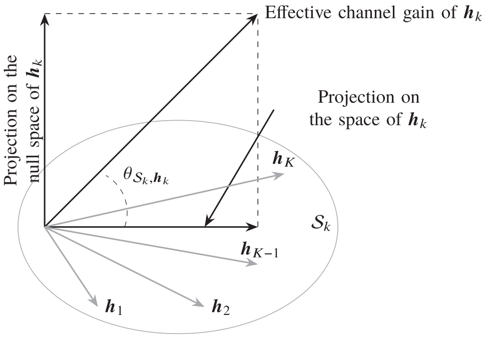

Effective channel gain of $\boldsymbol{h}_k$
Projection on the null space of $\boldsymbol{h}_k$
Projection on the space of $\boldsymbol{h}_k$
$\theta_{\mathcal{S}_k,\boldsymbol{h}_k}$
$\boldsymbol{h}_K$
$\mathcal{S}_k$
$\boldsymbol{h}_{K-1}$
$\boldsymbol{h}_1$
$\boldsymbol{h}_2$

Fig. 1. Null space projection and space projection of channel $\boldsymbol{h}_i$ [37].

A. Action Space Reduction

We assume that the BS is the physical learning agent responsible for maximizing the reward α by scheduling K UEss. The reward α is defined as the current system data rate, which is the sum of data rates achieved by the K scheduled UEss. The reward definition is the same for all three proposed scheduling algorithms. The three proposed scheduling algorithms differ by the use of context information, as will be seen later. Furthermore, we assume that an action consists of selecting K UEss that will be scheduled by the BS. Then, the number A of possible actions in the action space $\mathcal{A}$ is given by

$$A = \binom{J}{K}, \tag{8}$$

which increases combinatorially with J and K, making the action set size rapidly get impractical.

In the sequel, we describe an assumption that is used to drastically reduce the number of possible compositions of scheduled UEss in a given TTI. For a group of UEss scheduled in the same resource, ZF precoding sends the signal of a served UEs in the joint null space projection of the other scheduled UEss. Therefore, the channel correlation among UEss directly impacts the channel gain of each UEs after ZF precoding [37]. Fig. 1 illustrates this behavior, where $\mathcal{S}_k = \text{Span}(H_k)$, and $\theta_{\mathcal{S}_k,\boldsymbol{h}_k}$ is the angle between the channel vectors $\boldsymbol{h}_k$ and $\mathcal{S}_k$. Using $\theta_{\mathcal{S}_k,\boldsymbol{h}_k}$, the channel correlation coefficient between UEs k and the subspace $\mathcal{S}_k$ is given by $\cos(\theta_{\mathcal{S}_k,\boldsymbol{h}_k})$, with the channels becoming more uncorrelated as $\theta_{\mathcal{S}_k,\boldsymbol{h}_k}$ approaches $\frac{\pi}{2}$, which increases the effective channel gains after ZF precoding [37]. The channel $\boldsymbol{h}_k$ can be decomposed into two projections: the null space projection and space projection, where the null space projection is the zero-forcing direction, the space projection is the signal degradation of the user k caused by channel correlation and the effective channel gain is the squared magnitude of the null space projection [37].

As previously mentioned, UEss belonging to different clusters are supposed to have low-correlated channels (i.e., $\cos(\theta_{\boldsymbol{s}_k,\boldsymbol{h}_k})$ tends to be close to zero for UEss of different clusters). Then, the interference from the signals transmitted from the BS serving UEss belonging to different clusters becomes negligible [4]. The channel correlation among UEs channels of different clusters will be analyzed in Section VII.

We mentioned that the number of possible actions defined in 8 grows combinatorially, which is impractical. Therefore, to reduce it, we consider each cluster as a logical virtual agent by using the previous assumption that the signal transmitted from the BS serving UEss in cluster x produces negligible interference at the UEss in cluster y. Each virtual agent is responsible for performing an action, i.e., scheduling the UEss belonging to its own cluster. Therefore, we can define for each virtual agent c the set of actions $\mathcal{A}_c$, whose size A_c is given by

$$A_c = \binom{J_c}{K_c}, \tag{9}$$

where J_c and K_c are the total number of UEss and the number of scheduled UEss of cluster c, respectively. Since $\sum_{c=1}^{C} A_c \ll A$, the action space is drastically reduced. Therefore, we assume that there are virtual distributed units that have the BS as central unit that manages the them, processes the data, and takes the actions. Therefore, the objective of virtual agents is to reduce the search space by considering an action subset A_c of the entire action set A and to select the best action from subset A_c that will later define the scheduled UEss by the BS to transmit data.

The virtual RL agent strategy is a suboptimal strategy since instead of assuming a huge action set A we work with a much smaller action subset A_c. This way, by considering that we are using the action subset A_c, we are actually working with a reduced space of solutions to the considered problem. However, a similar strategy was adopted by [26] to address scheduling problems using optimization tools in a massive MIMO scenario with hybrid beamforming. In [26] the authors also use clustering and reduce the search space by solving the problem separately for each cluster, therefore, the authors also consider as scheduling search space only a smaller subset of the total possible combinations. Also, the authors prove by simulation, considering the reduced search space strategy (solving the problem separately by cluster) against the entire search space strategy, that this strategy brings a huge gain in complexity at a small cost in performance. Therefore, this analysis also is true to our scenario since the virtual agent are defined as clusters and each virtual agent determines its scheduled UEss.

B. Maximum Throughput Solution

In the sequel, we describe the MT learning algorithm as a CMAB problem. Note that the term throughput refers to the throughput obtained in the long run, i.e., from the first TTI until the current TTI. Consequently, we need to estimate action values which are used to make the action selection decision. The action value of an action is defined as the mean received reward when that action is selected. This way, the incremental average updating method is used to define the action value vector $\boldsymbol{d}_c \in \mathbb{R}_+^{A_c \times 1}$ as [7]

$$\boldsymbol{d}_c(a_c) = \boldsymbol{d}_c(a_c) + \frac{1}{\boldsymbol{n}_c(a_c)}(\alpha - \boldsymbol{d}_c(a_c)), \tag{10}$$

where a_c is a given action, $\boldsymbol{n}_c \in \mathbb{Z}_+^{A_c \times 1}$ is the vector containing the number of times that a given action was selected, and $\boldsymbol{f}(g)$ refers to the element g from vector $\boldsymbol{f}$. For example, $\boldsymbol{n}_c(a_c)$ refers to the number of times in cluster c that the virtual agent selected the action a that belongs to cluster c. Since the three proposed scheduling algorithms use the same reward, the same estimation of action values will be used by the next presented schedulers. The framework containing this solution and the others will be presented later in 2.

C. Maximum Throughput With Fairness Guarantees Solution

Note that, since the reward was defined as the system data rate, the scheduler presented in Section VI-B does not require any additional information. This is not the case of the next presented schedulers, they require more information about the system to achieve their objectives. Therefore, a context information is going to be used by the next proposed schedulers and its definition as well as its usage are going to be presented in this section and in Section VI-D.

The MTFG algorithm proposed here is modeled as a CMAB problem and aims to maximize the system throughput with fairness guarantees. Since we are working with FDD massive MIMO system, obtaining the instantaneous CSI is impractical. Therefore, the outdated CSI, which we consider available and was used before by the clustering in Section IV, is the considered context information used by our MTFG algorithm, which is used to obtain the UEss' priority for being scheduled. Note that, we consider that the UEss average throughput is the sum of the throughput values obtained by each UEs divided by the number of UEss. The UEs priority for being scheduled is a value between 0 and 1 that reflects the distance of the UEs throughput in relation to the UEss average throughput, i.e., a UEs throughput smaller than the UEss average throughput result in a UEs scheduling priority close to 1, and a UEs throughput greater than the UEss' average throughput results in a UEs scheduling priority close to 0. In the following, we describe how the UEss' scheduling priority is modeled to achieve maximum fairness and throughput. We use a priority function in which the UEs scheduling priority decreases rapidly when its throughput approaches or exceeds its target (sigmoidal function). Therefore, similarly to [18] we propose to use the mentioned function

$$P_j(v_j) = \frac{1}{1 + \mathrm{e}^{-\delta(\frac{v_j}{v^{\mathrm{avg}}} - 1)}}, \tag{11}$$

where $\delta > 0$ controls the function shape, v^{avg} is the sum of the throughput values obtained by the UEss over the number of UEss, and v_j is the throughput of the j-th UEs. Note that, we normalize v_j by v^{avg} to represent the throughput of v_j as a portion of the UEss' average throughput, i.e., between 0 and 2. $P_j(v_j)$ is a decreasing function of the UEss throughput with a controllable shaping parameter δ and centered at v^{avg}, as shown in Fig. 2. As in [18], we used $\delta = -9.1912$ to obtain the shape shown in Fig. 2. The idea is to prioritize the scheduling of the UEs with the lowest throughput in order to improve the system fairness.

Therefore, we can define the mean scheduling priority $q_c(a_c)$ of UEss which is contained in the action a_c belonging to cluster c as

$$q_c(a_c) = \frac{1}{K_c} \sum_{j \in \mathcal{U}_c(a_c)} P(v_j), \tag{12}$$

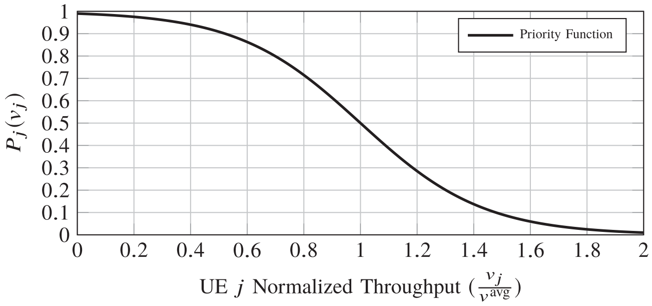


Fig. 2. UEs prioritization function.

where $\mathcal{U}_c(a_c)$ is the group of UEss which is contained in the action a_c in the cluster c. The UEss scheduling priorities and the action values will be jointly used to determine which actions are going to be selected in a given TTI. Therefore, we can define $\boldsymbol{q}_c \in \mathbb{R}_+^{A_c \times 1}$ as mean UEss scheduling priority vector of each action in a cluster c.

D. Maximum Throughput With QoS Guarantees

In the following, we describe how the UEss' scheduling priority is modeled aiming at maximizing the system throughput with QoS requirements. For the proposed scheduler in this section, we consider that there are different UEs throughput requirements. Therefore, we replace the variable v^{avg} that defines the center of our function 11 by v_j^{req}, which is the required throughput of UEs j. In this work, a UEs is considered satisfied when it achieves a target throughput. With this in mind, we need a scheduling priority function capable of being adaptable to achieve a higher throughput while guaranteeing QoS requirements. Therefore, the UEs's scheduling priority depends on its current throughput, so the algorithm chooses the action with the best trade-off between the QoS requirements of UEss and the system throughput.

This can be achieved using the function defined in 11 and controlling its shape through the variable δ. Therefore, as it can be seen, depending on the $P_j(v_j)$ shape and current throughput, an unsatisfied UEs ($v_j < v_j^{req}$) can have a priority to be scheduled between 0.5 and 1 while a satisfied UEs can have a scheduling priority between 0 and 0.5. In this work, we consider the system satisfaction as the ratio between the number of satisfied UEss and the total number of UEss. Thus, the shape of $P_j(v_j)$ is associated with the system satisfaction, where [18] obtained good results for $\delta = -9.1912$. Following a similar approach as [18], we created 21 shapes for $P(\cdot)$ based on the value of δ, which are shown in Fig. 3 for $s \in \{-10, -9, \ldots, 9, 10\}$ and $\delta = 9.1912 \times 2^s$, where s determines the shape of the function P_j.

E. Proposed Framework

The framework containing the MT, MTFG, and MTQG solutions is presented in 2. Note that, since the all proposed schedulers use the same reward (system throughput), we can create a jointly framework even if they have different goals. Therefore, the solutions MTFG and MTQG can reach their desired goals by combining the reward with 12. Also, for the

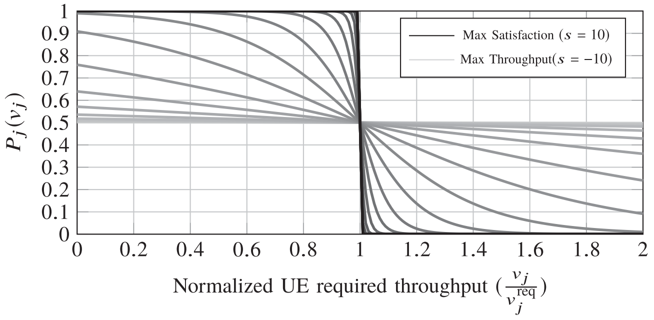


Fig. 3. UEs prioritization function.

MTQG solution, the changes in 11 need to be considered, as explained in Section VI-D.

This way, for each TTI and virtual agent, depending on the system operator objective (MT, MTFG or MTQG), our framework selects its actions based on the ϵ-greedy policy, which has two distinct phases: exploration (with probability ϵ) and exploitation (with probability $1-\epsilon$). The ϵ value decays linearly at each TTI until it reaches the desired value, and this strategy is known in literature as ϵ-decaying method [38]. For more details on the ϵ-greedy strategy as well the ϵ-decaying method, please refer to [38]. In the exploration phase, a random action is selected from $\mathcal{A}_c$. An exploration phase is needed to get information about the unexplored actions or actions that have not yet been selected so often. In the exploitation phase, if the system provider selects to operate using the MT scheduling, the virtual agent selects an action that maximizes the value of $\boldsymbol{d}_c$. Otherwise, if the system provider selects to operate using the MTFG or MTQG scheduling, the virtual agent computes its weights using 12 following the definitions of Section VI-C and Section VI-D, respectively.

After that, the virtual agent selects the action that maximizes the trade-off between the UEs's throughput and their scheduling priorities, which corresponds to the maximum value in the Hadamard (element-wise) product $\boldsymbol{q}_c \odot \boldsymbol{d}_c$. Then, the BS schedules the UEss chosen by the virtual agents. These scheduled UEss use the analog precoder to feed back their equivalent channel, which is used as CSI to calculate their digital precoders. Afterwards, the system data rate is calculated by the BS using 3.

In the next step, the system data rate associated with the action selected by each virtual agent, as well as the number of times that these actions were selected, are stored in $\boldsymbol{d}_c$ and $\boldsymbol{n}_c$, respectively. Moreover, if the framework is operating using the MTQG scheduling, the shape of the weight function needs to be managed. Therefore, let us introduce the variables μ and λ. μ is a value between 0 and 1 that refers to the percentage pf satisfied UEss required by the system operator, i.e., it is the system satisfaction desired by the system operator. λ is a value between 0 and 1 that refers to the throughput security threshold that the worst UEs needs to have before the system starts to concern about the system satisfaction, i.e., the threshold that triggers the mechanism that changes the scheduling priority shape. The variables μ, λ and s determine the shape of the function P_j, which starts at a predetermined shape $s = 10$ (maximum QoS priority) and may change as the system evolves. The change criteria of those parameters are shown afterward.

Therefore, if the system satisfaction is above μ and the worst satisfied UEs has its throughput λ greater than its target, the s value is reduced by 1 so as to make P_j in 11 approach a straight/flat line (light gray). The μ and λ values are decided by the network operator, e.g., based on the environment and past experience. When the function assumes a flat line shape, the UEss have the same scheduling priority of approximately 0.5, leading the scheduler to select UEss following a maximum throughput policy. Otherwise, the s value increases in 1 as to make the shape $P_j(v_j)$ in 11 approach a step function (dark gray). When the function assumes the step function shape, the unsatisfied and over-satisfied UEss have scheduling priorities of almost 1 and 0, respectively, leading the scheduler to select UEss following a maximum satisfaction policy.

Hence, each virtual agent learns over time its best groups of UEss to maximize system throughput, a knowledge that is combined with the context information on prioritizing certain UEss to scheduler improving QoS provisioning. 2 presents the proposed framework containing the MT, MTFG and MTQG solutions.

In the following, we present the computational complexity analysis of the proposed framework for a given TTI. The proposed framework, independently of the desired objective, is mainly dominated by two operations: finding the best actions and computing the action values. The variable a_c represents the selected action for cluster c. Thus, to find the best action a_c, we can consider the use of a sorting algorithm to sort the vector used to obtain a_c in lines 9, 12, and 15 of 2. Therefore, if we consider the well known sorting algorithm called merge sort, the worst-case computational complexity for the sorting procedure is O$(A_c \log A_c)$, where A_c is the number of possible actions for a given cluster c. To compute the actions values, only a sum is required which gives us a complexity of O(1). These two procedures are repeated for each cluster, i.e., C times, where C is the number of clusters. Therefore, the worst-case computational complexity of our framework is $O(C \cdot A_c \log A_c)$. So, the proposed MT, MTFG and MTQG have the same worst-case complexity of O$(C \cdot A_c \log A_c)$.

F. Signaling Overhead Reduction

In our work, at each TTI, the BS firstly schedules the UEss using one of our proposed algorithms presented in Section VI. Secondly, the scheduled UEss feed back their CSI using the analog precoders 6 and eigenvectors of equivalent channel 5 presented in Section V. Finally, the digital precoder 7 is computed following Section V, and the data is sent to the scheduled UEss using 3.

In general, most works in the literature (e.g., [2], [11], [15], [39]) consider that all UEss in the system feed back their instantaneous equivalent CSI before scheduling providing information that helps the scheduler. Our scheme, however, needs only the equivalent instantaneous CSIs of the K scheduled UEss. Therefore, it requires less signaling than those previous works. Fig. 4 shows the main steps of all proposed algorithms.

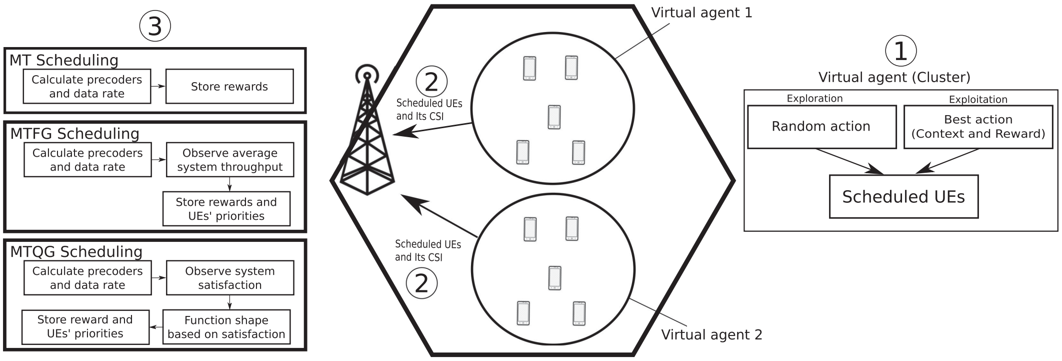

3
MT Scheduling
Calculate precoders and data rate
Store rewards
MTFG Scheduling
Calculate precoders and data rate
Observe average system throughput
Store rewards and UEs' priorities
MTQG Scheduling
Calculate precoders and data rate
Observe system satisfaction
Store reward and UEs' priorities
Function shape based on satisfaction
2
Scheduled UEs and Its CSI
Scheduled UEs and Its CSI
2
Virtual agent 1
Virtual agent 2
1
Virtual agent (Cluster)
Exploration
Exploitation
Random action
Best action (Context and Reward)
Scheduled UEs

Fig. 4. Illustration of the main steps of the proposed scheduling algorithms. Firstly, the virtual agents select the UEss to be scheduled by the BS based on the exploration or exploitation strategy. Secondly, the selected UEss are scheduled by the BS and feed back their equivalent instantaneous CSIs. Finally, depending on the desired system performance the MT, MTFG or MTQG scheduling is executed.

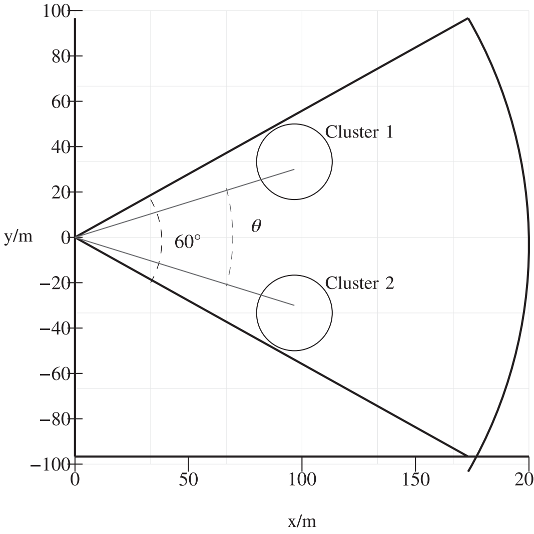


Fig. 5. Scenario considering 2 hotspots with a determined angle θ between their centers.

VII. Numerical Results

In this section, we describe our main assumptions as well as the scenario considered in our study. Afterwards, we compare the performance of the proposed algorithms against that of some baseline solutions previously proposed in the literature.

The scenario considered herein is shown in Fig. 5, where a massive MIMO BS at coordinate $(0, 0)$ is equipped with an 8×8 UPA ($N = 64$), where x and y axis indicate the distances from the BS to the other elements of the scenario. The BS serves 20 UEss which are uniformly distributed inside two hotspots, each with a radius of 15 m, located inside a 60° cell sector with 200 m of radius. The centers of the hotspots are 100 m away from the BS and 30° apart. Moreover, as in [15], [40], we consider at most 10% of the number of transmit antennas as the quantity of available RF chains.

We adopt the QuaDRiGa UMi LOS/NLOS channel model [41] and assume the BS power to be evenly divided among 125 RBs. Note that, we are going to evaluate two scenarios: 1) pure LOS scenario and 2) mixed LOS/NLOS scenario. The motivation to consider the mixed LOS/NLOS scenario is to show show that our proposed solution can support to a certain degree of NLOS channels. Also, in this work, we consider one of the 125 RBs as representative, and for the others, the same holds. The most relevant parameters used in our simulations are shown in Table II.

Herein, we consider two services with UEss requiring different throughput to be satisfied. For simplicity, we consider that each UEs is using only one service. In order to differentiate the services, we consider that UEss utilizing service 2 require 200 kbps more than the ones utilizing service 1. Furthermore, we consider that the BS requires a system satisfaction $\mu = 90\%$ which we assume as the minimum acceptable satisfaction rate in our system. Note that these different service requirements are going to be used to evaluate the MTQG solution. The value of λ is chosen to make the framework adapt itself before the system satisfaction falls bellow the required system satisfaction μ. Therefore, we consider the value of $\lambda = 120\%$ of the worst satisfied UEs throughput to trigger the change of shape conditions.

In order to get the system into a typical normal long-run condition for the learning algorithms, we consider a warm-up phase. Note that this warm-up phase is considered only to the learning schedulers since they perform decisions based on past experiences. In this phase, we consider that the UEs's throughput requirement starts fulfilled and the ϵ values decay linearly from 100% to 5% over time. The motivation of having more exploration at the start is to avoid getting stuck into a local optimum by acquiring more information about the action space. In our paper, the warm-up phase has a duration of 100 TTIs.

Afterward, we are going to support the considered assumption of the interference from the signals transmitted from the BS serving UEss belonging to different clusters becomes negligible. In Fig. 6 we applied the K-means clustering algorithm to observe the behavior of the channel correlation among the UEss' channels belonging to different clusters varying the angle θ where the center of the hotspots are disposed. Therefore, we performed 100 Monte Carlo simulations considering 20 UEss and 2 hot spots

Algorithm 2: Proposed Framework.

```
Input: $\mathcal{A}_c$, $C$ and $T$
Initialize: $\boldsymbol{q}_c = \mathbf{0}_{A_c \times 1}$ and $\boldsymbol{d}_c = \mathbf{0}_{A_c \times 1}$, $\forall c$
Initialize: set of scheduled UEs $\mathcal{S} = \emptyset$
Initialize: counter vectors $\boldsymbol{n}_c = \mathbf{0}_{A_c \times 1}$ of each action $\forall c$
Initialize: shape control variable $s = 10$                ▷ Max. priority.
for Each TTI do
    for $c = 1$ to $C$ do
        if operator system decides for MT scheduling then
            $a_c \leftarrow \begin{cases} \underbrace{\text{action that maximizes } (\boldsymbol{d}_c)}_{\text{Exploitation}}, & \text{probability } 1-\epsilon \\ \underbrace{\text{random action from } \mathcal{A}_c}_{\text{Exploration}}, & \text{probability } \epsilon \end{cases}$
        else if operator systems decides for MTFG scheduling then
            Calculate the vector of weights $\boldsymbol{q}_c$ using (12) following Section VI-C
            $a_c \leftarrow \begin{cases} \underbrace{\text{action that maximizes } (\boldsymbol{q}_c \odot \boldsymbol{d}_c)}_{\text{Exploitation}}, & \text{probability } 1-\epsilon \\ \underbrace{\text{random action from } \mathcal{A}_c}_{\text{Exploration}}, & \text{probability } \epsilon \end{cases}$
        else if operator systems decides for MTQG scheduling then
            Calculate the vector of weights $\boldsymbol{q}_c$ using (12) following Section VI-D
            $a_c \leftarrow \begin{cases} \underbrace{\text{action that maximizes } (\boldsymbol{q}_c \odot \boldsymbol{d}_c)}_{\text{Exploitation}}, & \text{probability } 1-\epsilon \\ \underbrace{\text{random action from } \mathcal{A}_c}_{\text{Exploration}}, & \text{probability } \epsilon \end{cases}$
        end if
        $\mathcal{S} \leftarrow \mathcal{S} \cup \mathcal{A}_c(a_c)$                ▷ Schedule the UEs.
    end for
    Scheduled UEs $\mathcal{S}$ feed back their CSI
    Compute hybrid (analog and digital) precoder using (6) and (7)
    $d \leftarrow$ sum of scheduled UEs data rate using (3)                ▷ Reward.
    for $c = 1$ to $C$ do
        $\boldsymbol{n}_c(a_c) \leftarrow \boldsymbol{n}_c(a_c) + 1$   ▷ Number of times that $a_c$ was chosen.
        $\boldsymbol{d}_c(a_c) \leftarrow \boldsymbol{d}_c(a_c) + \frac{1}{\boldsymbol{n}_c(a_c)}(\alpha - \boldsymbol{d}_c(a_c))$       ▷ Action values.
    end for
    if Operator systems decides for MTQG scheduling then
        if System satisfaction $\geq \mu$ and the smaller satisfied UE throughput $\geq \lambda$ its throughput requirement then
            if $s \geq -10$ then
                $s = s - 1$                ▷ Prioritize more the throughput.
            end if
        else
            if $s \leq 10$ then
                $s = s + 1$                ▷ Prioritize more the satisfaction.
            end if
        end if
    end if
end for
```

TABLE II
SIMULATION PARAMETERS

Parameter	**Value**
System bandwidth	100 MHz
System carrier frequency	28 GHz
Number of subcarriers per RB	12
Subcarrier spacing	60 kHz
TTI duration	0.25 ms
Number of OFDM symbols per TTI	14
Total transmit power	35 dBm
Noise figure	9 dB
Noise spectral density	-174 dBm/Hz
Shadowing standard deviation	3.1 dB
Cell radius	200 m
UEs Speed	3 and 60 km/h
Number of UEs	20
Number of clusters	2
Number of UEs per cluster	10
Number of UEs selected per cluster	2
Number of simulation rounds	100
Simulation duration	1 s

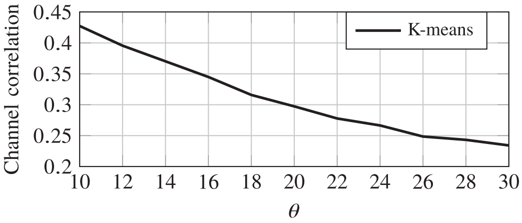


Fig. 6. Channel correlation for UEss from two distinct clusters for different values of θ.

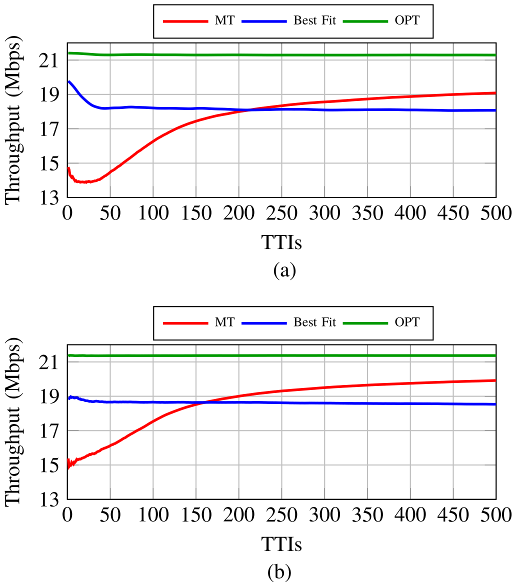
with a radius of 15 m located at a distance of 100 m from the BS.

Applying K-means for varying angle θ (see Fig. 5), one can see that it clusters UEss with low correlated channels even for very close hotspots, with the correlation being inversely proportional to the angular distance between the hotspots. Thus, as mentioned before, our assumption that the interference among clusters are negligible becomes increasingly true as the distance between clusters increases.

A. Maximum Throughput Evaluation

In this section, we compare the proposed MT scheduler with the solution proposed in [4] (Best Fit) – which employs a Best Fit algorithm to solve the scheduling problem –, and with the optimum solution (OPT) – which know the equivalent instantaneous CSI to calculate all precoders and uses brute force enumerating all possible solutions choosing the best one. The Best Fit scheduler in [4] has a parameter β that establishes the trade-off between spatial correlation and channel gain aiming at maximizing the system throughput. This trade-off is used by the algorithm to schedule the UEss sequentially aiming at maximizing the system throughput. Due to space limitations, we are omitting more details of the comparison algorithms. Therefore, for more details on the (Best Fit) algorithm, please refer to [4]. Notice that, due to the combinatorial size of the problem, the OPT solution has impractical computational complexity. Also notice that all the simulated algorithms in this work use hybrid precoding and perform clustering before scheduling and, consequently, select only K_c UEss per cluster. Therefore, the interference among clusters are supposed to be negligible for all simulated algorithms, as explained in Section VI-A.

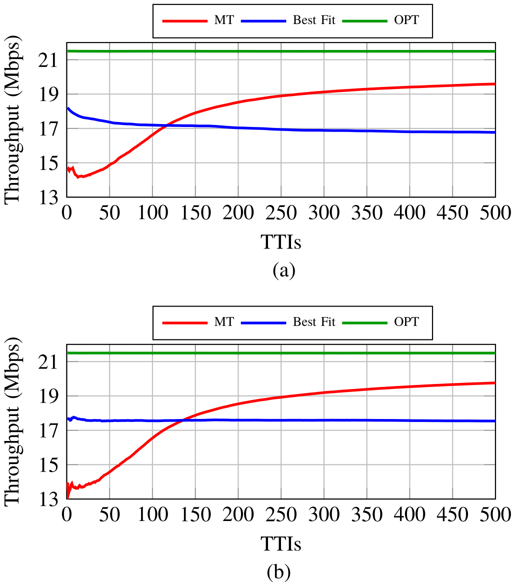
In Fig. 7, MT, Best Fit, and OPT schedulers are compared in terms of system throughput for the two different UEs speeds 3 km/h and 60 km/h in a pure LOS scenario. The motivation to analyze scenarios with different speeds is to evaluate if there is a loss in the learning algorithm when the scenario gets more dynamic. Note that the optimal value for β parameter in the Best Fit algorithm can change with the UEs distribution, so the system needs to find and adjust its value. Therefore, we consider that the optimal β is known by the system. As our proposed MT scheduler learns from the past, it is not sensitive to the UEs distribution, different to the Best Fit solution. Another drawback of



(a)

(b)

Fig. 7. System throughput over the TTIs for pure LOS scenario. (a) UEss speed 3 km/h. (b) UEss speed 60 km/h.



(a)

(b)

Fig. 8 System throughput over the TTIs for mixed LOS/NLOS scenario considering. (a) UEss speed 3 km/h. (b) UEss speed 60 km/h.

the Best Fit scheduler in [4] is that it uses the covariance matrix to schedule the UEss. However, this information becomes outdated over time and needs to be estimated with a certain periodicity. In our simulations the same statistical CSI is used during the entire simulation. Note that these issues do not affect our proposed MT since the UEss are scheduled based on past experiences, which avoids the need for a trade-off control parameter and the CSI dependence.

Focusing on the relative performance among algorithms, we can see that the MT scheduler needs only approximately 250 TTIs to outperform the solution in [4]. Furthermore, Figs. 7(a) and 7(b) show that the OPT solution outperforms by approximately 10% and 16% the throughtput of MT and Best Fit schedulers, respectively. Finally, Fig. 7(b) shows that the MT scheduler is robust to high mobility scenarios, obtaining almost the same performance as shown in Fig. 7(a).

In Fig. 8, our proposed MT scheduler is compared with Best Fit and OPT schedulers in terms of system throughput for the two different UEs speeds 3 km/h and 60 km/h in a mixed LOS/NLOS scenario. We can see that the MT solution still has the best trade-off between complexity and performance, even when considering a challenging scenario where some UEss have NLOS channels. Also, the Best Fit solution has a loss in its performance compared to the LOS scenario, while the MT has a similar performance than obtained in a LOS scenario. Therefore, focusing on the relative performance among algorithms, Figs. 8(a) and 8(b) show that the OPT solution outperforms the throughput of MT and Best Fit schedulers by approximately 10% and 23%, respectively. Moreover, Fig. 8(b) shows that the MT scheduler is robust to high mobility scenarios, obtaining almost the same performance as shown in Fig. 8(a).

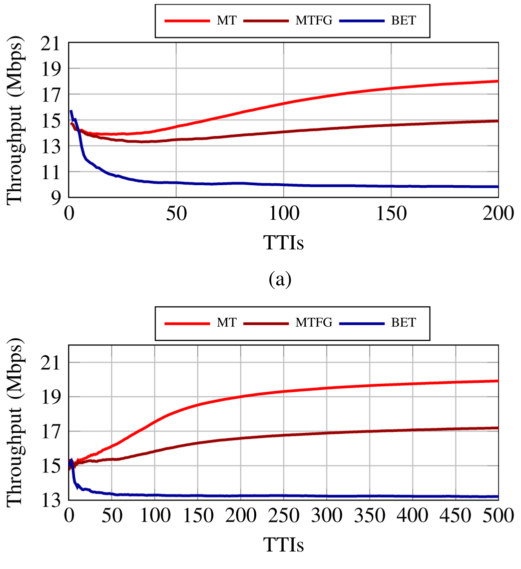
TABLE III
COMPLEXITY OF DATA RATE MAXIMIZATION SCHEDULERS

Scheduler	**Complexity**
Proposed MT	O$(C \cdot A_c \log A_c)$
Best fit [4]	O$(C \sum_{i=0}^{\frac{K}{C}-1} (J_c - 1))$
OPT (brute force)	O$(\binom{J}{K})$

Table III shows the worst-case computational complexity for the algorithms considered (base-line and proposed) in the paper to solve the throughput maximization problem. Considering that J_c is the number of UEss belonging to a given cluster, the computational complexity calculation of the MT algorithm was presented in Section VI-E, while the Best Fit algorithm was calculated in [4] giving a worst-case complexity of O$(C \sum_{i=0}^{\frac{K}{C}-1} (J_c - 1))$. Since the optimal solution was obtained by the brute force method and considering J and K as the number of the UEss in the system and the number of scheduled UEss, respectively, the optimal solution, namely OPT, needs to go through all the possible actions. Therefore, its worst-case computational complexity grows combinatorially with the number of UEss. Therefore, we can see that the MT scheduler presents a good trade off between performance and complexity.

B. Maximum Thoughtput With Fairness Guarantees Evaluation

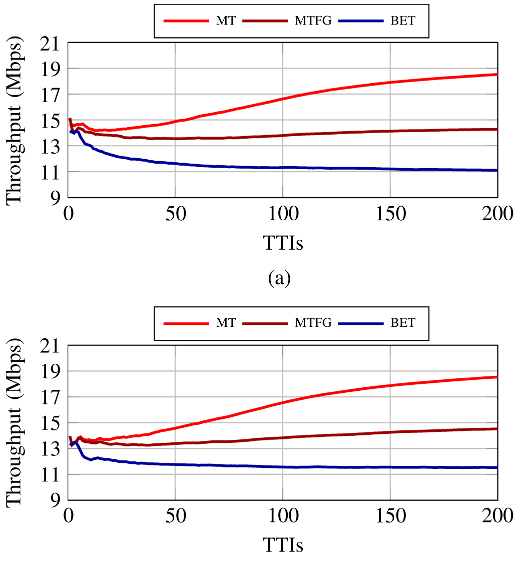
In this section, we compare the proposed MTFG scheduler with the MT solution, which does not take into account any context information, and with the Blind Equal Throughput (BET) [42], which uses the past average throughput as metric to schedule UEss. The BET scheduling stores the past average throughput and uses it as a metric to calculate the UEss scheduling priorities, providing fairness among UEss regardless of their



(a)

(b)

Fig. 9. System throughput over the TTIs for pure LOS scenario. (a) UEss speed 3 km/h. (b) UEss speed 60 km/h.



(a)

(b)

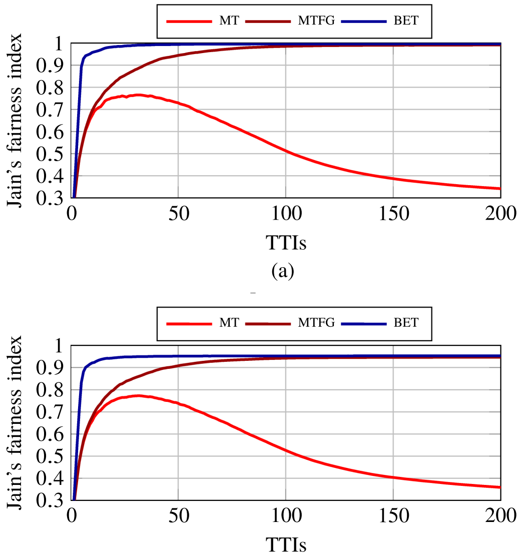
Fig. 10 System throughput over the TTIs for mixed LOS/NLOS scenario. (a) UEss speed 3 km/h. (b) UEss speed 60 km/h.

channel conditions [42]. For more details on the BET algorithm, please refer to [42].

In Fig. 9, MT, MTFG, and BET schedulers are compared in terms of system throughput for the two different UEs speeds 3 km/h and 60 km/h in a pure LOS scenario. As we can see, the MT scheduler achieves the best performance in terms of throughput. However, it will be seen afterwards that seeking only for maximum throughput negatively affects the fairness among UEss. The MTFG scheduler solves this problem by giving priority to the UEss with lowest throughput, which is done through the utility function 11. Therefore, actions with smaller actions values will be selected more often aiming at increasing the fairness among UEss, which negatively impacts the system throughput. Moreover, the BET scheduler has the worst performance due to it search for fairness, without concerning about the system throughput achieved.

Focusing on the relative performance among schedulers in terms of throughput, we can see in Figs. 9(a) and 9(b) that the MT scheduler outperforms the MTFG and BET by approximately 22% and 72%, respectively. Moreover, Fig. 9(b) shows that the proposed scheduler can maintain its throughput performance even for higher mobility.

In Fig. 10, we compare our proposed MTFG scheduler with our proposed MT and BET schedulers in terms of system throughput for the two different UEs speeds 3 km/h and 60 km/h in a mixed in LOS/NLOS scenario. As we can see, the considered schedulers maintain a similar behavior in terms of throughput for the considered pure LOS and mixed LOS/NLOS scenarios. Focusing on the relative performance among schedulers in terms of throughput, in Fig. 8 we can see that the MT scheduler outperforms the MTFG and BET by approximately 34% and 81%, respectively. Moreover, Fig. 9(b) shows that the proposed MTFG scheduler can maintain its throughput performance even for higher mobility.

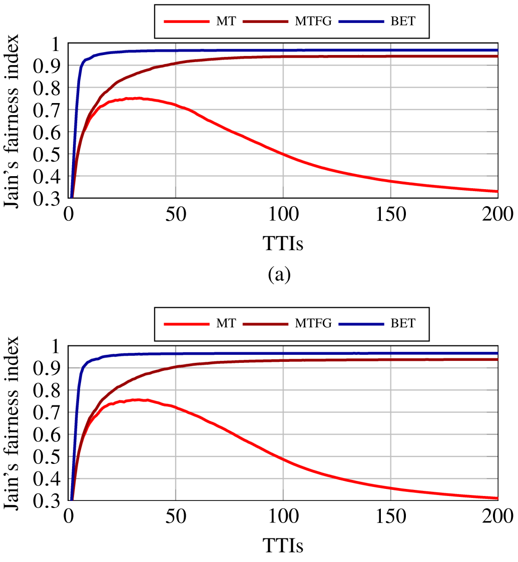


(a)

(b)

Fig. 11. Jain's fairness over the TTIs for pure LOS scenario. (a) UEss speed 3 km/h. (b) UEss speed 60 km/h.

In Fig. 11, MT, MTFG, and BET schedulers are compared in terms of Jain's fairness index for the two different UEs speeds 3 km/h and 60 km/h in a pure LOS scenario. Note that, the mathematical expression for Jain's fairness index is $\frac{(\sum_J^{i=1} v_i)^2}{J\cdot\sum_J^{i=1} v_i^2}$, for more details on Jain's fairness index, please refer to [43]. As we can see, the performance of the schedulers in terms of fairness is the opposite of the one presented in Fig. 9, as expected. The MT achieves the worst performance since it is concerned only about the system throughput. Moreover, in



(a)

(b)

Fig. 12 Jain's fairness over the TTIs for mixed LOS/NLOS scenario. (a) UEss speed 3 km/h. (b) UEss speed 60 km/h.

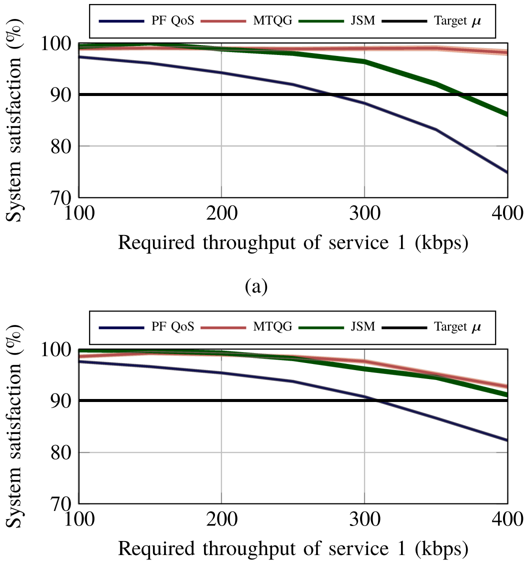
TABLE IV
COMPLEXITY OF THROUGHPUT MAXIMIZATION WITH FAIRNESS GUARANTEES SCHEDULERS

Scheduler	Complexity
Proposed MT	$O(C \cdot A_c \log A_c)$
Proposed MTFG	$O(C \cdot A_c \log A_c)$
BET [42]	$O(C \cdot J_c)$

Figs. 11(a) and 11(b) the MTFG needs approximately 100 TTIs to achieve the same performance of the BET solution. Note that the MTFG scheduler achieves higher fairness among UEss at a price of a relatively small loss in throughput, thus offering a good fairness-throughput trade-off. Moreover, Fig. 10(b) shows that the proposed MTFG can maintain its performance even for higher mobility.

In Fig. 12, we compare our proposed MTFG scheduler with our proposed MT and BET schedulers in terms of Jain's fairness index for the two different UEs speeds 3 km/h and 60 km/h in a mixed LOS/NLOS scenario. As we can see, the considered schedulers maintain a similar behavior in terms of Jain's fairness index for the considered pure LOS and mixed LOS/NLOS scenarios. This way, we can see that the MTFG solution still has the best trade-off between throughput and fairness, even when considering a more challenging scenario where some UEss have NLOS channels. Moreover, Fig. 12(b) shows that the proposed scheduler can maintain its performance even for higher mobility.

Table IV shows the worst-case computational complexity for the algorithms considered (base-line and proposed) in the paper to solve the throughput maximization with fairness guarantees problem. The computational complexity of the MT and MTFG algorithms are the same since they use the same framework. The BET algorithm is a low complexity scheduler that stores the past average throughput achieved by each UEs using it as a metric [42]. Therefore, the BET scheduler needs to calculate its metric for each UEs before scheduling. Despite the low complexity of the involved algorithms, we can see by the throughput and fairness results presented in the Figures of Section VII-B that the MTFG scheduler is the one that showed the best trade-off between performance and complexity.



(a)

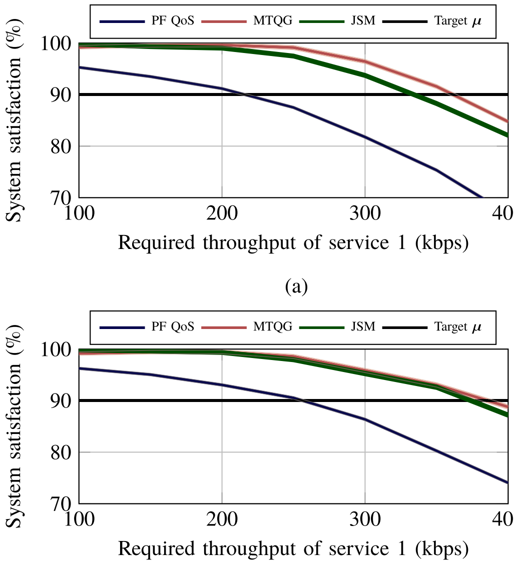
(b)

Fig. 13. System satisfaction versus the required throughput of service 1 for pure LOS scenario. (a) UEss speed 3 km/h. (b) UEss speed 60 km/h.

C. Maximum Throughput With QoS Guarantees Evaluation

In this section, we compare the proposed MTQG scheduler with the QoS-aware Proportional Fair QoS (PF QoS) scheduler [42] and an adaptation of JSM scheduler proposed in [18]. The Proportional Fair (PF) QoS is similar to the traditional PF scheduling. However, it works with two sets of UEss: i) the priority set with UEss that do not meet their QoS requirements, which we assigned the highest priorities and; ii) the low priority set with the rest of the UEss (currently satisfied UEs). The JSM scheduler uses two policies based on the derivatives of the sigmoidal to obtain the UEss' priority. For more details on the PF QoS and JSM algorithms, please refer to [18], [42], respectively. Moreover, the instantaneous CSI is used by the PF QoS and JSM schedulers to estimate the data rate, which is the information utilized to schedule the UEss. Therefore, for the sake of fairness in comparisons among different solutions, we use in PF QoS and JSM schedulers dominant eigenvalues and eigenvectors to estimate the data rate, since this is the same CSI employed by our proposed framework.

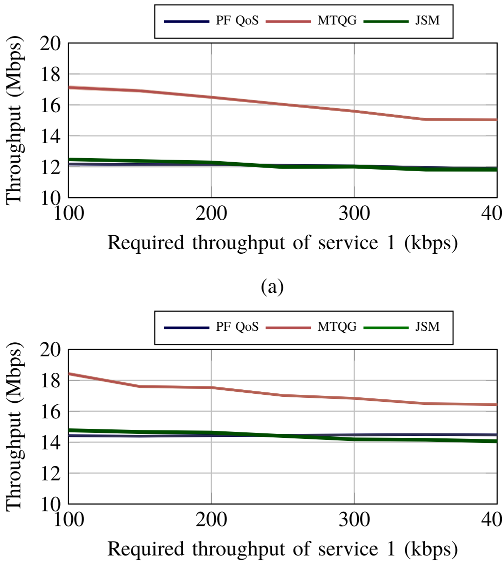
In Fig. 13, the system satisfation for MTQG, PF QoS, and JSM schedulers is shown for increasing values of the required throughput of service 1 in a pure LOS scenario. As we can see, the PF QoS is the scheduler that achieves the worst performance for both UEs speeds. This happens since the simplicity of the scheduler and the inaccurate CSI available, which decreases the performance compared to other schedulers. In Fig. 13(a), we can see that the MTQG achieves the best performance compared to



(a)

(b)

Fig. 14 System satisfaction versus the required throughput of service 1 for mixed LOS/NLOS scenario. (a) UEss speed 3 km/h. (b) UEss speed 60 km/h.



(a)

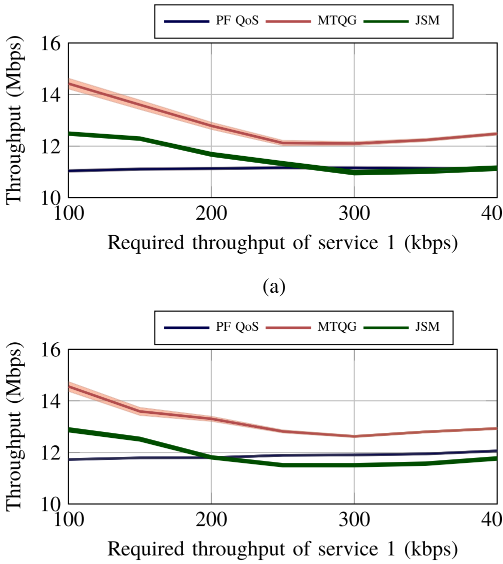
(b)

Fig. 15. System throughput versus required throughput of service 1 for pure LOS scenario. (a) UEss speed 3 km/h. (b) UEss speed 60 km/h.

the other schedulers. However, the increase in the UEss' speed makes the MTQG and JSM schedulers achieve the same performance in Fig. 13(b). This happens due to the quick change of the channel state caused by the higher UEss' speed, which makes the best scheduling compositions change more often, leading to a more challenging scenario to be learned by the MTQG. Also, the performance loss of the baseline algorithms occurs because they do not take into account any information about the interference among scheduled UEss, increasing the probability of scheduling UEss with correlated channels in the same RB. On the other hand, MTQG learns about channel correlation through rewards and uses the same CSI as the baseline schedulers.

In Fig. 14, the system satisfaction for MTQG, PF QoS, and JSM schedulers is shown for increasing values of the required throughput of service 1 in a mixed LOS/NLOS scenario. As we can see, the considered schedulers maintain a similar behavior in terms of system satisfaction for the considered pure LOS and mixed LOS/NLOS scenarios. Focusing in the relative performance among schedulers, the BET solution is the best one, however, it will come at a cost of reduced system throughput, as we are going to see in the next result. The MTFG solutions achieves a good QoS rate after the TTI 50 and getting even closer when the TTI increases. The MT algorithm is the worst one since its focus is only in the system throughput maximization.

In Fig. 15, MTQG, PF QoS, and JSM schedulers are compared in terms of system throughput for different values of required throughput for service 1 in a pure LOS scenario. We recall that the required throughput of service 2 is 200 kbps higher than that of service 1. As it can be seen, the system throughput decreases as the required throughput increases, so that there is a trade-off between satisfaction and system throughput. Moreover, the MTQG scheduler achieves the highest system throughput independently of the required throughput. The BET and JSM schedulers can maintain almost the same throughput. However, in Fig. 13(a) their satisfaction decrease drastically compared to that of MTQG. In Fig. 13(b), the BET has a low performance in terms of satisfaction compared to the JSM and MTQG solutions. Also, the MTQG scheduler provides a gain in system throughput of up to 41% compared to PF QoS and JSM algorithms. Moreover, Fig. 15(b) shows that the proposed algorithm can maintain its performance even for higher mobility.



(a)

(b)

Fig. 16 System throughput versus required throughput of service 1 for mixed LOS/NLOS scenario. (a) UEss speed 3 km/h. (b) UEss speed 60 km/h.

In Fig. 16, MTQG, PF QoS, and JSM schedulers are compared in terms of system throughput for different values of required

TABLE V
COMPLEXITY OF THROUGHPUT MAXIMIZATION WITH QOS GUARANTEES SCHEDULERS

Scheduler	Complexity
Proposed MTQG	$O(C \cdot A_c \log A_c)$
JSM [18] [18]	$O(C \cdot J_c)$
PF QoS [42]	$O(C \cdot J_c)$

throughput for service 1 in a mixed LOS/NLOS scenario. As we can see, the considered schedulers present a loss in system throughput performance in relation to the pure LOS scenario. This happens due to the worst UEss have poor channels in the mixed LOS/NLOS scenario which impact their achieved throughput. Focusing on the relative performance among schedulers in terms of system throughput, the MTQG scheduler provides a performance gain of up to 16% compared to PF QoS and JSM algorithms. Moreover, Fig. 16(b) shows that the proposed algorithm can maintain its performance even for higher mobility. Table V shows the worst-case computational complexity for the considered algorithms to solve the throughput maximization problem with QoS guarantees. The computational complexity of the MTQG algorithm is the same as the one calculated for MT and MTFG, while the computational complexity of the JSM algorithm can be found in [18]. Since the PF QoS scheduler works similarly to the BET algorithm, it only needs to calculate its metric for each UEs before scheduling them. Even though all algorithms present low computational complexity, we can see by the previous results presented in the Figures of Section VII-C that the MTQG scheduler is the one that shows the best trade-off between performance and complexity.

VIII. CONCLUSION

In this paper, we proposed and evaluated a framework of Radio Resource Allocation (RRA) for hybrid precoding massive MIMO communication systems using RL tools. In order to deal with the combinatorial search space of the scheduler, we created clusters of UEss with correlated statistical channels and proposed a new strategy that considers each cluster as a virtual learning agent. We considered that an action is the selection of UEss to be scheduled. Therefore, the consideration of virtual agents leads to a practicable number of possible actions. Also, the virtual agent selects the UEss belonging to its own cluster aiming at maximizing a pre-determined objective. Therefore, the BS is responsible to know the selected UEss by the virtual agents and schedule it.

We solved three RRA problems in our framework, which are maximization of throughput, maximization of throughput with fairness guarantees, and maximization of throughput with QoS guarantees. The proposed framework dynamically adapt itself to solve the RRA problem desired by the operator. Also, the proposed solutions to those problems utilize the CMAB tool. This tool achieves good performance even working with limited/scarce information, which is one of the challenges of massive MIMO. Therefore, we utilized only the statistical CSI to schedule the UEss, which reduces the signaling overhead. Also, since we proposed a CMAB framework that learns by trial and error, we reduced even more the signaling overhead by considering that only the scheduled UEss have to feed back their equivalent instantaneous CSI. Moreover, the precoders are calculated only to the scheduled UEss, which avoids the computation for every scheduling possibility. Simulation results showed that the reference algorithms are outperformed by the proposed solutions in low and high mobility scenarios. Also, the results show that the learning algorithm is robust even when the scenario gets more dynamic.

Moreover, we see some interesting research questions for future work, such as:

- Multicell interference - the use of MIMO and higher frequencies reduce the impact of multi-cell interference.
  The massive MIMO technology employs narrow beams reducing multi-cell interference by spatial filtering.
  However, the simulation of a multicell scenario in this paper was unpractical since we considered many features such as time evolution, UEss moving at different speeds, optimal solutions, and so on.
  Therefore, the scope of our paper is to study the different scheduling objectives using contextual bandits algorithms, leaving the challenges of multi-cell interference as the future perspective of the study.
- Consideration of pure NLOS case - since the NLOS channel does not have one dominant path, our framework is not designed for the pure NLOS case since we are using clustering and analog precoders based on the dominant eigenvectors of the UEs statistical channels.
  To address the pure NLOS case, we should consider the following modifications: another clustering algorithm or a modification in the K-means algorithm should be considered, and a different analog precoder design that considers not only the dominant eigenvector but an average of the strongest ones.
- Imperfect CSI feedback - note that, to be fair, in our paper we are considering that all algorithms are using the same CSI. Since the consideration of imperfect CSI feedback will impact negatively in the performance of all algorithms, it is expected that the same relative performance behavior maintains.
  However, further analysis is needed to confirm this behavior.

REFERENCES

[1] B. Wang, F. Gao, S. Jin, H. Lin, and G. Y. Li, "Spatial- and frequency-wideband effects in millimeter-wave massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 66, no. 13, pp. 3393–3406, Jul. 2018.

[2] X. Sun, X. Gao, G. Y. Li, and W. Han, "Agglomerative user clustering and cluster scheduling for FDD massive MIMO systems," *IEEE Access*, vol. 7, pp. 86 522–86 533, 2019.

[3] R. W. Heath, N. González-Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, "An overview of signal processing techniques for millimeter wave MIMO systems," *IEEE J. Sel. Top. Signal Process.*, vol. 10, no. 3, pp. 436–453, Apr. 2016.

[4] W. V. F. Mauricio, D. C. Araujo, F. H. C. Neto, F. R. M. Lima, and T. F. Maciel, "A low complexity solution for resource allocation and SDMA grouping in massive MIMO systems," in *Proc. 15th Int. Symp. Wireless Commun. Syst.*, 2018, pp. 1–6.

[5] F. D. Calabrese, L. Wang, E. Ghadimi, G. Peters, L. Hanzo, and P. Soldati, “Learning radio resource management in RANs: Framework, opportunities, and challenges,” *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 138–145, Sep. 2018.

[6] F. Hussain, S. A. Hassan, R. Hussain, and E. Hossain, “Machine learning for resource management in cellular and IoT networks: Potentials, current solutions, and open challenges,” *IEEE Commun. Surv. Tut.*, vol. 22, no. 2, pp. 1251–1275, Oct.–Dec. 2019.

[7] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 1998.

[8] M. Simsek, M. Bennis, and I. Guvenc, “Learning based frequency- and time-domain inter-cell interference coordination in HetNets,” *IEEE Trans. Veh. Technol.*, vol. 64, no. 10, pp. 4589–4602, Oct. 2015.

[9] S. Jiang, Y. Chang, and K. Fukawa, “Distributed inter-cell interference coordination for small cell wireless communications: A multi-agent deep Q-learning approach,” in *Proc. Int. Conf. Comput. Inf. Telecommun. Syst.*, 2020, pp. 1–5.

[10] S. Maghsudi and E. Hossain, “Multi-armed bandits with application to 5G small cells,” *IEEE Wireless Commun.*, vol. 23, no. 3, pp. 64–73, Jun. 2016.

[11] J. Nam, A. Adhikary, J.-Y. Ahn, and G. Caire, “Joint spatial division and multiplexing: Opportunistic beamforming, user grouping and simplified downlink scheduling,” *IEEE J. Sel. Top. Signal Process.*, vol. 8, no. 5, pp. 876–890, Oct. 2014.

[12] A. Maatouk, S. E. Hajri, M. Assaad, and H. Sari, “On optimal scheduling for joint spatial division and multiplexing approach in FDD massive MIMO,” *IEEE Trans. Signal Process.*, vol. 67, no. 4, pp. 1006–1021, Feb. 2019.

[13] G. Bu and J. Jiang, “Reinforcement learning-based user scheduling and resource allocation for massive MU-MIMO system,” in *Proc. IEEE/CIC Int. Conf. Commun. China*, 2019, pp. 641–646.

[14] R. Chataut and R. Akl, “Channel gain based user scheduling for 5G massive MIMO systems,” in *Proc. IEEE 16th Int. Conf. Smart Cities: Improving Qual. Life ICT IoT AI*, 2019, pp. 49–53.

[15] H. Xu, T. Zhao, S. Zhu, D. Lv, and J. Zhao, “Agglomerative group scheduling for MmWave massive MIMO under hybrid beamforming architecture,” in *Proc. IEEE 18th Int. Conf. Commun. Technol.*, 2018, pp. 347–351.

[16] Z. Jiang, S. Chen, S. Zhou, and Z. Niu, “Joint user scheduling and beam selection optimization for beam-based massive MIMO downlinks,” *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2190–2204, Apr. 2018.

[17] N. W. Moe Thet, T. Baykas, and M. K. Ozdemir, “Performance analysis of user scheduling in massive MIMO with fast moving users,” in *Proc. IEEE 30th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun.*, 2019, pp. 1–6.

[18] R. P. Antonioli, E. B. Rodrigues, T. F. Maciel, D. A. Sousa, and F. R. P. Cavalcanti, “Adaptive resource allocation framework for user satisfaction maximization in multi-service wireless networks,” *Telecommunication Syst.*, vol. 68, no. 2, pp. 259–275, Jun. 2018. [Online]. Available: https://doi.org/10.1007/s11235-017-0391-3

[19] F. Zhao, W. Ma, M. Zhou, and C. Zhang, “A graph-based QoS-Aware resource management scheme for OFDMA femtocell networks,” *IEEE Access*, vol. 6, pp. 1870–1881, 2018.

[20] J. Wang, Y. Zhang, H. Hui, and N. Zhang, “QoS-Aware proportional fair energy-efficient resource allocation with imperfect CSI in downlink OFDMA systems,” in *Proc. IEEE 26th Annu. Int. Symp. Personal, Indoor, Mobile Radio Commun.*, 2015, pp. 1116–1120.

[21] T. Y. Young and T. W. Calvert, *Classification, Estimation and Pattern Recognition*. New York, NY, USA: American Elsevier, 1974.

[22] Y.-K. Hua, W. Chang, and S.-L. Su, “Cooperative scheduling for pilot reuse in massive MIMO systems,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 12 857–12 869, Nov. 2020.

[23] L. D. Nguyen, H. D. Tuan, T. Q. Duong, H. V. Poor, and L. Hanzo, “Energy-efficient multi-cell massive MIMO subject to minimum user-rate constraints,” *IEEE Trans. Commun.*, vol. 69, no. 2, pp. 914–928, Feb. 2021.

[24] M. Guo and M. C. Gursoy, “Statistical learning based joint antenna selection and user scheduling for single-cell massive MIMO systems,” *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 1, pp. 471–483, Mar. 2021.

[25] X. Li, X. Zhang, Y. Zhou, and L. Hanzo, “Optimal massive-MIMO-aided clustered base-station coordination,” *IEEE Trans. Veh. Technol.*, vol. 70, no. 3, pp. 2699–2712, Mar. 2021.

[26] W. V. F. Maurício, D. C. Araújo, T. F. Maciel, and F. R. M. Lima, “A framework for radio resource allocation and SDMA grouping in massive MIMO systems,” *IEEE Access*, vol. 9, pp. 61 680–61696, 2021.

[27] A. Ortiz, A. Asadi, M. Engelhardt, A. Klein, and M. Hollick, “CBMoS: Combinatorial bandit learning for mode selection and resource allocation in D2D systems,” *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2225–2238, Oct. 2019.

[28] P. K. Tathe and M. Sharma, “Dynamic actor-critic: Reinforcement learning based radio resource scheduling for LTE-Advanced,” in *Proc. 4th Int. Conf. Comput. Commun. Control Automat.*, 2018, pp. 1–4.

[29] I. Comşa *et al.*, “Towards 5G: A reinforcement learning-based scheduling solution for data traffic management,” *IEEE Trans. Netw. Service Manag.*, vol. 15, no. 4, pp. 1661–1675, Dec. 2018.

[30] D. C. Araújo, E. Karipidis, A. L. F. de Almeida, and J. C. M. Mota, “Hybrid beamforming design with finite-resolution phase-shifters for frequency selective massive MIMO channels,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2017, pp. 6498–6502.

[31] M. A. Albreem, A. H. A. Habbash, A. M. Abu-Hudrouss, and S. S. Ikki, “Overview of precoding techniques for massive MIMO,” *IEEE Access*, vol. 9, pp. 60 764–60801, 2021.

[32] V. Kumar and N. B. Mehta, “Modeling and analysis of differential CQI feedback in 4G/5G OFDM cellular systems,” *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2361–2373, Apr. 2019.

[33] S. Jaeckel, L. Raschkowski, K. Borner, L. Thiele, F. Burkhardt, and E. Eberlein, “QuaDRiGa - quasi deterministic radio channel generator, user manual and documentations,” Fraunhofer Heinrich Hertz Institute, Tech. Rep. v2.6.1, 2021.

[34] 3GPP, “Study on channel model for frequencies from 0.5 to 100 GHz,” 3rd Generation Partnership Project (3GPP), Technical Specification, version 14.3.0, 2018. V.15.0.0. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3173

[35] H. Costa, D. Araújo, and T. Maciel, “Hybrid beamforming design bbased on unsupervised mmachine learning for millimeter wave systems,” *Int. J. Commun. Syst.*, vol. 33, Mar. 2020, doi: 10.1002/dac.4276.

[36] Y. Xu, G. Yue, and S. Mao, “User grouping for massive MIMO in FDD systems: New design methods and analysis,” *IEEE Access*, vol. 2, pp. 947–959, 2014.

[37] E. Castaneda, A. Silva, A. Gameiro, and M. Kountouris, “An overview on resource allocation techniques for multi-user MIMO systems,” *IEEE Commun. Surv. Tut.*, vol. 19, no. 1, pp. 239–284, Jan.–Mar. 2017.

[38] O. Caelen and G. Bontempi, “Improving the exploration strategy in bandit algorithms,” in *Learning and Intelligent Optimization*, V. Maniezzo, R. Battiti, and J.-P. Watson, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 56–68.

[39] A. Destounis and M. Maso, “Adaptive clustering and CSI acquisition for FDD massive MIMO systems with two-level precoding,” in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2016, pp. 1–6.

[40] Z. Wang, M. Li, Q. Liu, and A. L. Swindlehurst, “Hybrid precoder and combiner design with low-resolution phase shifters in mmWave MIMO systems,” *IEEE J. Sel. Top. Signal Process.*, vol. 12, no. 2, pp. 256–269, May 2018.

[41] S. Jaeckel, L. Raschkowski, K. Börner, and L. Thiele, “Quadriga: A 3-D multi-cell channel model with time evolution for enabling virtual field trials,” *IEEE Trans. Antennas Propag.*, vol. 62, no. 6, pp. 3242–3256, Jun. 2014.

[42] F. Capozzi, G. Piro, L. A. Grieco, G. Boggia, and P. Camarda, “Downlink packet scheduling in LTE cellular networks: Key design issues and a survey,” *IEEE Commun. Surv. Tut.*, vol. 15, no. 2, pp. 678–700, Apr.–Jun. 2013.

[43] R. Jain, *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling.* Hoboken, NJ, USA: Wiley, 1991. [Online]. Available: https://www.bibsonomy.org/bibtex/2a8b45ddf325d187cbaa68bf0c4db96bf/chsivic

Weskley V. F. Mauricio received the D.Sc. degree in telecommunications engineering from the Federal University of Ceará, Fortaleza, Brazil, in 2021. From 2017 to 2018, he has actively participated in projects in technical and scientific cooperation with Wireless Telecom Research Group (GTEL), UFC, and Ericsson Research, and from 2019 to 2020, he was a Ph.D. Guest Student with Technische Universität Darmstadt, Darmstadt, Germany. He is currently a Senior R&D Engineer with Protocol Team, Samsung Instituto de Desenvolvimento para Amazônia. His research interests include radio resource management, QoS, hybrid beamforming, reinforcement learning, and massive MIMO technology.

Tarcisio Ferreira Maciel received the B.Sc. and M.Sc. degrees in electrical engineering from the Federal University of Ceará (UFC), Fortaleza, Brazil, in 2002 and 2004, respectively, and the Dr.-Ing. degree in electrical engineering from Technische Universität Darmstadt (TUD), Darmstadt, Germany, in 2008. Since 2001, he has been actively participating in several projects in a technical and scientific cooperation with Wireless Telecom Research Group (GTEL), UFC, and Ericsson Research. From 2005 to 2008, he was a Research Assistant with the Communications Engineering Laboratory, TUD. Since 2008, he has been a Member of the Postgraduation Program in teleinformatics enginnering with UFC. In 2009, he was a Professor of computer engineering with UFC-Sobral. Since 2010, he has been a Professor with the Center of Technology, UFC. His research interests include radio resource management, numerical optimization, and multiuser or multiantenna communications.

Anja Klein (Member, IEEE) is currently a Full Professor with Technische Universität Darmstadt, Darmstadt, Germany, heading the Communications Engineering Laboratory. She has authored more than 300 peer-reviewed papers and has contributed to 12 books. Her main research interests include mobile radio, including interference management, cross-layer design, relaying, and multihop, computation offloading, smart caching, and energy harvesting. In 1999, she was named the Inventor of the Year by Siemens AG. She is a Member of Verband Deutscher Elektrotechniker - Informationstechnische Gesellschaft.

Francisco Rafael Marques Lima (Senior Member, IEEE) received the B.Sc. degree (Hons.) in electrical engineering, and the M.Sc. and D.Sc. degrees in telecommunications engineering from the Federal University of Ceará, Fortaleza, Brazil, in 2005, 2008, and 2012, respectively. In 2008, he has been in an internship with Ericsson Research, Luleå, Sweden, where he studied scheduling algorithms for LTE system. Since 2010, he has been a Professor with the Department of Computer Engineering, Federal University of Ceará. He is also a Researcher with the Wireless Telecom Research Group (GTEL), Fortaleza, Brazil, where he works at projects in cooperation with Ericsson Research. He has authored or coauthored several conference papers and journal articles, and patents in the wireless telecommunications field. His research interests include radio resource allocation algorithms for QoS guarantees in scenarios with multiple services, resources, antennas, and users.