2023 19th International Conference on the Design of Reliable Communication Networks (DRCN) | 978-1-6654-7598-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/DRCN57075.2023.10108389

Learning to Fulfill the User Demands in 5G-enabled Wireless Networks through Power Allocation: a Reinforcement Learning approach

Anastasios Giannopoulos
Department of Ports Management and Shipping
National and Kapodistrian University of Athens
Psachna, Greece
angianno@uoa.gr

Sotirios Spantideas
Department of Ports Management and Shipping
National and Kapodistrian University of Athens
Psachna, Greece
sospanti@uoa.gr

Nikolaos Nomikos
Department of Ports Management and Shipping
National and Kapodistrian University of Athens
Psachna, Greece
nomikosn@pms.uoa.gr

Alexandros Kalafatelis
Department of Ports Management and Shipping
National and Kapodistrian University of Athens
Psachna, Greece
kalafatelis.alexander@gmail.com

Panagiotis Trakadas
Department of Ports Management and Shipping
National and Kapodistrian University of Athens
Psachna, Greece
ptrakadas@uoa.gr

Abstract— **The goal of the study presented in this paper is to evaluate the performance of a proposed Reinforcement Learning (RL) power allocation algorithm. The algorithm follows a demand-driven power adjustment approach aiming at maximizing the number of users inside a coverage area that experience the requested throughput to accommodate their needs. In this context, different Quality of Service (QoS) classes, corresponding to different throughput demands, have been taken into account in various simulation scenarios. Considering a realistic network configuration, the performance of the RL algorithm is tested under strict user demands. The results suggest that the proposed modeling of the RL parameters, namely the state space and the rewarding system, is promising when the network controller attempts to fulfill the user requests by regulating the power of base stations. Based on comparative simulations, even for strict demands requested by multiple users (2.5 – 5 Mbps), the proposed scheme achieves a performance rate of about 96%.**

Keywords—wireless networks, reinforcement learning, power control, Q-learning, resource allocation, radio resource management

I. Introduction

Driven by the rapid evolution of wireless communication systems, spanning from novel schemes on the radio channel, such as the concept of Non-Orthogonal Multiple Access (NOMA) [1]-[2], to the introduction of software-defined, virtualized network services [3]-[4], the fifth-generation (5G) networks have gained great interest to enable previously-unseen capabilities to support services in several vertical domains [5]-[9] and become the driving force for innovative business models [10]. According to 5G expectations [11], there are three main types of services that have to be realized in the future networks, namely the massive machine-type communication (mMTC), enhanced mobile broadband (eMBB) and ultra-reliable low-latency communication (URLLC) services. All 5G types of communication will exhibit strict QoS with network connectivity requirements even for cell-edge users and under severe interference [11].

Thus, in 5G networks, billions of wireless devices are provisioned to be interconnected, communicating in a fast, heterogeneous and reliable manner. The considerable increase in the spatial density of the network architecture raises, in turn, significant challenges in the radio resource management (RRM) entity. In general, network densification results in lower probability of finding uncovered areas and/or users, but, contradictorily, increases the probability of severe interference. Therefore, power adjustment of both macro- and small-cell radio units (RUs) has to be effectively re-addressed as an immediate consequence of the network densification. Solving an RRM problem is not a trivial task, considering the multidimensional space that affects the solution convergence. Typical approaches in managing the network resources include intelligent power configuration of RUs, allocation of channels to users, cell selection policies, frequency reuse management schemes, rotation of (even Multiple Input Multiple Output - MIMO) antenna beams, and so on [12]. Towards this direction, exclusively heuristic solutions have been already replaced with automated approaches aiming to encapsulate both learning and hardcoded rules during the search of an optimal configuration instance.

The tremendous progress in both computing power and artificial intelligence (AI) algorithms have placed the solution of non-convex optimization problems more closely to automated processes, rather than ruled-based, brute-force approaches. This is mainly attributed to the ability of machine learning (ML) to solve optimally (or sub-optimally) extremely complex problems, without requiring detailed knowledge about all the problem parameters [13]. With reinforcement learning (RL), a specific branch of AI, it is possible to find the optimal strategy according to which an agent will achieve an objective after interacting with the environment. As opposed to deep learning (DL), RL agents do not require training data to learn from, instead they can capture knowledge following a trial-and-error approach [14].

Several studies have recently used RL methods in resource allocation problems, showing that usually RL outperforms the previously known, rule-based search algorithms [12], [13]. Specifically, the authors in [15] suggested a joint user association and resource allocation (UARA) scheme for the downlink of heterogeneous networks in order to maximize the long-term utility of the network under QoS constraints. In addition, a power allocation RL strategy has been proposed to mitigate the network power consumption in cloud radio access networks (RANs), while maintaining the user demands [16]. Other studies [17] – [19] proposed similar RL frameworks aiming at maximizing the total network throughput by only adjusting the power of BSs. Finally, the authors in [20] describe an RL algorithm for dynamic spectrum access in multi-cell wireless systems.

In this paper, an urban coverage area is considered where the problem of configuring the power of RUs in order to ensure that users requesting different services within the cells are fulfilled in an optimal manner is formulated. To this end, we establish an RL framework which, given a demand vector of different service classes, corresponding to different services required (i.e. voice, video, etc.), attempts to maximize the number of fulfilled users by only adjusting the power vectors of RUs. This approach ensures that the users requesting a service inside a coverage area experience at least the requested throughput, according to the service class associated with it. We compared two user association rules, as well as several baseline power control methods were contrasted. In these comparisons, both iterative heuristic algorithms and deterministic schemes were considered.

The main contributions of this work include:

(i) Presentation of a different approach in the determination of the state space acknowledged to the RL agent. This modification allows the agent (i.e. the controller entity) to have a three-fold information (i.e. quality of the received signal, the ID of the associated BS and the ID of the associated channel) about the telecommunication environment, thus offering more flexibility to both the user association scheme and the learning process.

(ii) Proposition of an alternative approach regarding the power allocation objective targeting at ensuring that the user demands are fulfilled. As opposed to the existing RL schemes focusing on maximizing the total network-wide throughput, the proposed demand-driven power allocation algorithm defines a rewarding system that takes into account only the user requests.

The rest of the paper is organized as follows: in Section II, the system model and the problem formulation are presented, followed by the interference model, the mathematical background and proposed RL algorithm outline. In Section III, the simulation setup and the corresponding results are illustrated, along with the relevant discussion about the outcomes. Finally, Section IV concludes the paper.

II. System Model and Problem Formulation

A. Network Model

A cellular network consisting of a set of M RUs ($m = 1,2,\dots,M$) is considered. A sectorization vector K with elements k_i ($i = 1,2,\dots,M$) is introduced to represent the number of sectors of each RU. The system is controlled by a centralized cognitive controller, which attempts to efficiently adjust the transmit power of RUs. Each RU m may choose to transmit from a set of available channels N. All channels ($n = 1,2,\dots,N$) have the same bandwidth B. Hence, $B_1 = B_2 = \cdots = B_N = B$. The proposed approach aims at optimally adjusting a specific power level p, which is selected by a set of available power levels $\{l = 1,2,\dots,L\}$, for each RU. When a RU m transmits over channel n with a power level l, the transmitted power is notated as $P_{m,n} = p(l)$. Furthermore, a power constraint is established, in order to ensure a maximum total power available to each RU, i.e. $\sum_{n=1}^{N} P_{m,n} \leq P_{thres}^{m}, \forall m$. The users located in the network area are grouped according to the RU from which are served. Hence, the total number of users U ($u \in \{1,2,\dots,U\}$) is grouped into different-sized sets U_m, each one including the indices of the users that are associated with RU m. Moreover, user u requests a service s from a set of available service classes S. Each service corresponds to a particular throughput demand in order to ensure efficient QoS.

Apart from the aforementioned, a demand vector D, with respective elements d_u ($u = 1,2,\dots,U$), is adapted to notify the requested service class of user $u \in U$. To reflect practical service classes, we indicatively consider three available QoS levels, namely 0.1 Mbps (e.g. for VoIP), 2.5 Mbps (e.g. for video call) and 5 Mbps (e.g. for HD streaming). For instance, $d_2 = 3$ denotes that user 2 requests service class 3 (i.e. 5 Mbps). Finally, we define an allocation matrix with respective elements $a_{u,m} = 1$, when user u is associated with BS m (or 0 otherwise). We assume that each user is only connected to a single RU, whereas each RU can be associated with multiple users.

B. Interference Model

In wireless environments, each user receives not only the signal from the associated RU, but also the accumulated interference signals from other operating RUs. In this paper, the inter-RU interference is taken into account. Since user $u \in U$ is served by RU $m \in M$ in channel $n \in N$, the signal-to-interference-plus-noise ratio (SINR) is given by:

$$SINR_{u}^{m,n} = \frac{P_{m,n} \cdot G_{m,n,u}}{\left(\sum_{m' \neq m}^{M} P_{m',n} \cdot G_{m',n,u}\right) + N_0} \tag{1}$$

where $G_{m,n,u}$ denotes the channel gain from RU m to user u over channel n, and N_0 denotes the noise power at the receiver level. Specifically, the numerator in the above equation represents the received power of user u from the associated RU m, whereas the denominator reflects the unwanted power received by the surrounding RUs transmitting over the same channel plus the noise contribution. Note that, the channel gain highly depends on the propagation model and the distance between the BSs and users' location, and it is expressed as follows [17]:

$$G_{m,n,u} = \left|H_{m,n,u}\right|^2 \cdot 10^{(PL_u+X)/10} \tag{2}$$

where $H_{m,n,u}$ is the Rayleigh fading gain of user $u \in U$ from RU $m \in M$ over channel $n \in N$, X is the log-normal shadowing, and PL_u is the path-loss of user $u \in U$.

The achievable data rate of a particular link is characterized according to the respective SINR status. The computation of the reachable transmission rate experienced by user $u \in U$ that is served by RU $m \in M$ and channel $n \in N$, is based on the Shannon formula, as follows:

$$R_u^{m,n} = B \cdot log\,(1 + \beta \cdot SINR_u^{m,n}) \tag{3}$$

where β is a constant that depends on the Bit Error Rate (BER) threshold ($\beta = 1$ for $BER = 10^{-6}$).

C. Problem Formulation

In this section, we address the problem of determining M power vectors of RUs in order to maximize the number of users that experience adequate throughput according to the required QoS class. A binary variable F_u is defined in order to indicate whether a user $u \in U$ experiences at least the requested level of throughput (F_u =1, if $R_u^{m,n} \geq d_u$), or the selected power configuration failed to provide the requested user QoS (F_u =0) Thus, the *optimization problem (P)* may be defined as follows:

$$\begin{aligned} &max \sum_{u=1}^{U} F_u \\ s.t.\text{: } (C1)\ &\sum_{m=1}^{M} a_{u,m} = 1\,, \forall u \in U, \\ (C2)\ &\sum_{n=1}^{N} P_{m,n} \leq P_{thres}^{m}\,, \forall m \in m \end{aligned} \tag{4}$$

D. The Q- Learning Framework

In general, RL is a widely-used branch of the ML that relies on a goal-oriented algorithm in order to achieve a particular objective through trial-and-error approach. Given a detailed description of an environment, RL agents aim at becoming near-optimal predictors about which actions will yield the best rewards, starting with no prior knowledge. During the learning process, the agent keeps track of its past experiences by continuously filling-and-updating the Q-table. The rules governing the use of the Q-table rely on the general framework of the Q-learning process described below.

We consider an RL agent that interacts with an environment by taking actions and receiving rewards. The environment can be in a particular state s from a set of available states (state space S). The RL agent is able to perform an action a, given by a set of available actions (action space A), and receives a reward r from the environment, immediately after taking action a. Q-learning allows the agent to predict the "quality" of being in state s_t and performing the action a_t, based on the following Bellman equation:

$$\begin{aligned} Q_t(s_t, a_t) = (1-\alpha) \cdot Q_{t-1}(s_t, a_t) + \alpha \\ \cdot (r(s_t, a_t) + \gamma \\ \cdot \max_{a'}\{Q(s_{t+1}, a')\}) \end{aligned} \tag{5}$$

The aforementioned formula comprises the update rule of the Q-table at time t and implies that the new Q-value depends on both the previous Q-value for a given state-action pair (first term), while the second term consists of the immediate reward ($r(s_t, a_t)$) and the optimal future reward ($\gamma \cdot \max_{a'}\{Q(s_{t+1}, a')\}$). Specifically, the learning rate $\alpha \in [0,1]$ is used to balance between old and new Q-values, while the discount factor $\gamma \in [0,1]$ accounts for future rewards.

In principle, Q-values take into account that certain actions may place the agent either in an advantageous or disadvantageous situation, which will have a long-term effect. Ideally, the agent gradually gathers experience about the beneficial actions and finally gains sufficient knowledge about the environment, meaning that the temporal difference (TD) between the learned value $r(s_t, a_t) + \gamma \cdot \max_{a'}\{Q(s_{t+1}, a')\}$ and the old value $Q_{t-1}(s_t, a_t)$ is minimized, i.e. for a trained agent (after a sufficient number of interactions with the environment), the TD function may be expressed as:

$$\begin{aligned} TD_t(s_t, a_t) = \Big(r_t(s_t, a_t) + \gamma \\ \cdot \max_{a'}\{Q(s_{t+1}, a')\}\Big) \\ - Q_{t-1}(s_t, a_t) \approx 0, \end{aligned} \tag{6}$$

Finally, the trained Q-table will act as a consultant of the agent in order to guide its action selection policy that leads to the optimal accumulated future rewards.

In the proposed RL framework, a central network entity (controller, see Fig. 1) monitors the telecommunication environment and intelligently allocates power vectors to RUs with the goal of solving the optimization problem *P*. Notably, the crucial part of the learning process relies on the fact that the controller has no prior knowledge of the environment and the impact of its actions. The only way to extract knowledge from the unknown environment is by taking random actions and gradually exploit those having beneficial (past) outcomes. In order to clearly describe the RL framework, it is essential to define (i) the possible states that the agent can potentially visit,

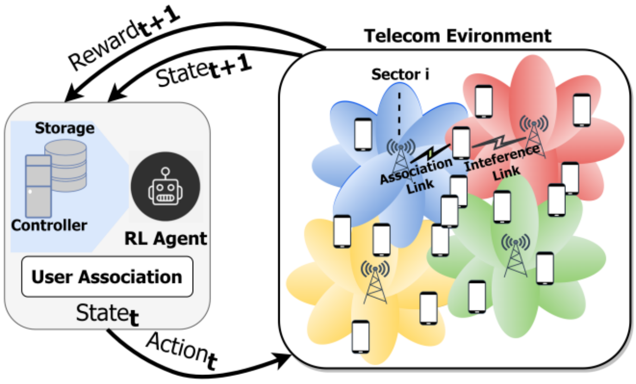

Fig. 1. The RL cycle over the 5G-enabled telecommunication environment.

(ii) the possible actions that the agent can perform from a given state, and (iii) the rewarding system according to which the environment responds to the agent's actions.

State space: It is a function that describes the telecom environment, transforming the action taken in the previous step into a reward and a new set of actions. In the proposed algorithm, the state space includes the channel quality indicator (CQI) for each user, followed by the associated RU and associated channel IDs, i.e. $S = \{S_1, \dots, S_t, \dots, S_T\}$ where at a given time t the system is at state $S_t = [(CQI_1, BS_1, CH_1), \dots, (CQI_U, BS_U, CH_U)]$. This means that the controller, before taking an action, knows a triplet for each user u, namely the received signal quality CQI_u experienced by associating user u to RU_u and channel CH_u. Noteworthy, the CQI values are computed directly from the received signal strength indicators (RSSI) according to LTE specifications (CQI range 1-15) [21].

Action space: The controller performs a sequence of actions $\{A_1, \dots, A_t, \dots, A_T\}$ during an episode, i.e. a complete series of agent-environment interactions, beginning from the initial state and terminating in the final state. At a given step t, a specific power vector for each BS is selected, i.e. $A_t = [(P_{1,1}, \dots, P_{1,N}), \dots, (P_{M,1}, \dots, P_{M,N})]$, where $P_{m,n}$ is the allocated power to channel n of RU m. Power levels are selected from the set of available power levels

Reward system: The action taken by the agent results into a different system state, thus leading to different CQIs and association configuration. The reward returned at time t is defined as follows:

$$r_t(S_t, A_t) = \begin{cases} FU_t - FU_{t-1}, & if\ FU_t > FU_{t-1} \\ 0 \quad , & otherwise \end{cases} \qquad (7)$$

where FU_t is the number of fulfilled users (the experienced throughput satisfies their demands, i.e. $R_u \geq d_u$) at time t. For instance, if the previous power configuration resulted to $FU_{t-1} = 5$ fulfilled users, and the current action increased this number into $FU_t = 8$, the returned reward would be $r_t = 3$. Intuitively, this rewarding system guides the agent to gradually prefer those power configurations that maximize the total number of fulfilled users.

Action selection policy: In each episode, the agent selects either a (random) explorative action or an exploitative action (based on the Q-table). For the action selection strategy, we used the *ε-greedy* method, according to which the agent passes smoothly-over-time from an exclusively exploration phase to an exclusively exploitation phase. The ε decaying rule was selected to be linear, starting from 1 and ending to 0 for the first half of the episodes.

E. Algorithm

The proposed RL algorithm is composed of five steps:

Step 1: The cognitive controller associates each user with a specific BS-channel pair based on the user association scheme. For comparison purposes, we considered two different user association rules:

(i) the user u is associated with the nearest RU based on the Euclidean distance between them (*MinDist* association):

$$BS_u = argmin_m \left\{ \sqrt{(x_m - x_u)^2 + (y_m - y_u)^2} \right\}, \forall m \in M \quad (8)$$

where (x, y) denote the position coordinates. Moreover, the associated channel of user u can be expressed:

$$CH_u = argmax_n \{ R_u^{BS_u, n} \}, \forall n \in N \qquad (9)$$

(ii) the user u is associated with the RU that provides the best-quality signal (*MaxThrough* association). This association rule can be described:

$$BS_u, CH_u = argmax_{m,n} \{ R_u^{m,n} \}, \forall m \in M, n \in N \qquad (10)$$

Step 2: A power vector is allocated to each RU, depending on whether the algorithm operates in exploration (a random power vector is selected) or exploitation (the power vector with the maximum Q-value is selected) phase.

Step 3: The environment informs the RL agent about the next state and the immediate reward of the action taken. This procedure includes the re-calculation of the throughput vectors of each user and the comparison between the allocated throughput at the current and the previous state.

Step 4: The controller updates the policy by replacing the previously known Q-value for that state-action pair with the new Q-value according to the Q-learning formula.

Step 5: The agent reduces the value of ε to get closer to the exploitation stage and repeats steps 1-4 until convergence.

III. Simulation Results

In this section, we illustrate the simulation results of the proposed RL algorithm implemented in Python 3.8. We consider four different simulation scenarios (5, 10, 15 and 20 users) according to the number of users inside the network area (1500m × 1500m). Without loss of generality, four RUs were placed in a square topology with a minimum inter-RU distance of 1000m. To validate the proposed RL scheme in extreme interference conditions, we considered 2 channels for each RU. Three sectors with a spacing of 120 degrees were adapted to

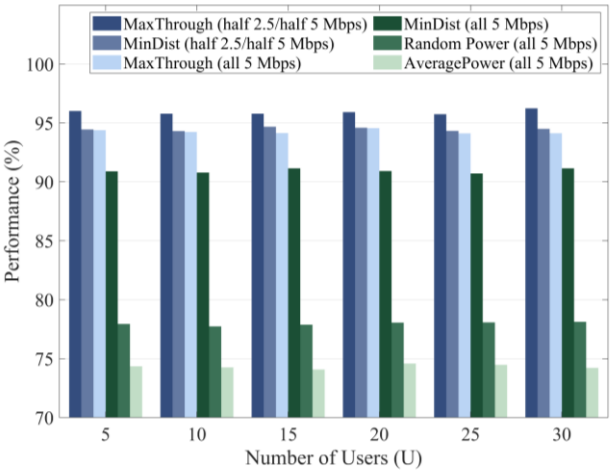

Fig. 2. Performance of the RL algorithm vs the number of users

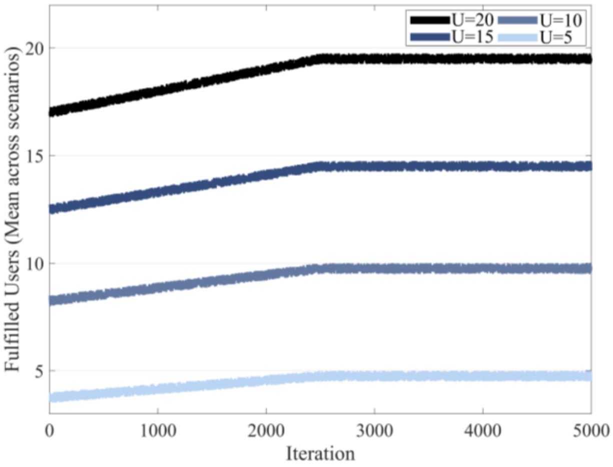

Fig. 3. Learning curves for the worst-case demand setting (all users request services of 5Mbps) averaged across 1000 different positioning scenarios.

each RU. Each channel operated at a specific power level selected by the set of available powers of {6.4, 9.6, 12.8, 16, 19.2} W. The receiver noise was set equal to $-174dBm/Hz$ and the bandwidth of each channel was set at $B = 2.88MHz$ (16 subcarriers of 180 kHz). The path loss part of the channel gain was computed according to the model specified in [21]-[22].

The antenna system of each RU was selected to have three beamforms, each one characterized by a half power beamwidth (HPBW) of 70 degrees and a minimum gain of -35dB. We calculated the performance of each simulation setting as the ratio between the total number of users satisfied and total number of the optimal solution for 1000 different user positioning scenarios. In each positioning scenario, each user was randomly placed within the network area. Specifically, the performance for a simulation setting with K users is given by:

$$P_K\ (\%) = \frac{\sum_{i=1}^{1000} k_i}{K \cdot 1000} \times 100, \qquad (11)$$

where k_i is the number of users fulfilled by the RL algorithm in positioning scenario i.

A. Association scheme selection versus the number of users

For comparison purposes, the two different association rules were contrasted (*MinDist* vs. *MaxThrough*). After extended simulations, the Q-learning hyperparameters were fine-tuned at α=0.1 (learning rate) and γ=0.95 (discount factor). In order to validate the performance of the proposed algorithm, namely 5, 10, 15, 20, 25 and 30 users are sequentially assumed to request services. Furthermore, the proposed scheme is evaluated for different service types, namely (i) a case where all users request QoS class 3, and (ii) some users request QoS class 2 and the rest require QoS class 3. The proposed RL algorithm achieves a 100% performance rate for any number of users, when they request service class by all users is 2 (or less).

As readily observed by Fig. 2, the maximum throughput association rule outperforms the minimum distance association rule for the same demand vectors. Evidently, as the services required by the users become more stringent, the performance of the proposed RL scheme slightly decreases, but even in the extreme case when 20 users request service of 5 Mbps, the proposed scheme achieves a performance rate of 91.51%. Additionally, when the users require a mix of different services, the algorithm achieves a satisfaction percentage of 96% in the case of 20 users. The enhanced performance of the *MaxThrough*, as compared with the *MinDist* association scheme, confirms that the nearest RU is not always the best server. As expected, in such interference environments, it could be beneficial to associate users with the best-signal RUs, although they may be located further away. In Fig.2, two baseline methods are also illustrated: (i) a fixed power allocation policy which allocates the average power level (i.e. 12.8 W) to each BS (*AveragePower* method), and (ii) a greedy power allocation policy which assigns random power vectors (*RandomPower* method). We confirmed that the proposed RL algorithm exhibits greater performance in comparison to both baseline methods.

B. Learning Course

Fig. 3 depicts the learning curves (average time-course over the 1000 different user positioning scenarios) of the proposed RL algorithm as a function of the training steps. In the beginning of each power allocation episode, RUs are set to transmit with the minimum power levels and the agent explores for the appropriate actions (power vectors) to fulfill the uncovered users. According to the rewards received throughout the exploration period, the agent constantly increases the collected rewards since it gradually prefers past-and-beneficial moves, as readily shown in Fig. 3.

The nearly optimal performance of the algorithm in several positioning scenarios was expected (making the average rewards not converging to the optimal solution, see Fig. 3); however, there are various configurations of the users'

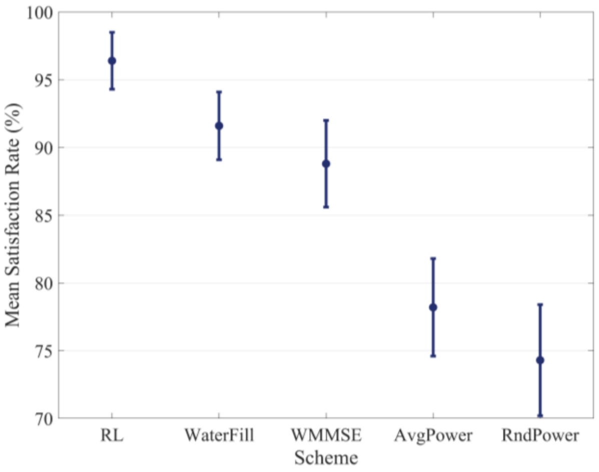

Fig. 4. Mean Satisfaction Rate (%) computed as the averaged rate of fulfilled users across 100 evaluation scenarios for the 5 different schemes. Error lines represent the standard error of the means.

positions in which it is not feasible to cover all the users' demands. For instance, the associated BS cannot achieve the requested throughput by a single user even if it operates at the highest power level due to limitations of the antenna system (the user is located at a minimum of the radiation pattern). In such cases, a further increase of the power of the associated RU would potentially result into enhanced interferences for the already covered users. Other possible solutions include either the placement of supplementary RUs inside the network area or the utilization of additional channels. However, there is a trade-off between the benefits of introducing additional network resources (densification, additional channels, etc.) and the degradation of the overall interference.

C. Comparative performance for varying user demands

This section presents comparative results of the RL performance against widely-used iterative heuristic power control schemes, namely the Water-filling algorithm [23] and the Weighted Minimum Mean Squared Error (WMMSE) [24]. Two deterministic rule-based schemes were also considered, including fixed average power allocation (all RUs constantly transmit with the average power level, so as to reach a balanced trade-off between interference and satisfaction level) and the random power allocation. The number of users was set to $|U| = 30$. Simulations included 100 different runs of the algorithms, whereas the final evaluation metric was derived by computing the average user satisfaction rate across runs (i.e. percentage Mean Satisfaction Rate). In each run (series of episodes until convergence), user demands were time-varying and selected uniformly for the set of demands.

The results are shown in Fig. 4. RL scheme demonstrates the optimal satisfaction rate, outperforming the rest of the baselines. This is attributed to the fact that iterative schemes (Water-filling and WMMSE algorithms) directly maximize the sum-rate of the total users. This means that the results of both schemes may result in higher network-wide throughput, however exhibiting over- or under-satisfaction in particular users. Contrary to the attributes of Water-filling and WMMSE algorithms (they assign high power levels to the channels showing good channel conditions so as to take advantage of the total network throughput), RL objective is directly targeted to the satisfaction status of the users, regardless of the network sum-rate utility.

D. Work extensions and limitations

It is worth noting that, although the proposed RL framework was applied to LTE-like network configurations in our simulations, it can be easily adapted in several types of network realizations by modifying the operating band, the channel model (urban channels with shadowing, diffraction due to obstacles), the inter-RU distances, the power levels and the antenna related parameters (directivity, sectorization, MIMO beams). However, the problem formulation is agnostic to the channel model and the operating frequency band that are considered, allowing proper channel condition and/or band adoptions. Furthermore, an immediate straight-forward extension of the proposed algorithm will be the introduction of a neural network (deep reinforcement learning) in the agent's side, allowing to consider larger state-action spaces. Other extensions would be to allow continuous values in the action space by adopting the principles of Deep Deterministic Policy Gradient (DDPG) with Actor-Critic Models.

IV. Conclusion

In the present work, an RL power allocation algorithm is proposed, which efficiently adjusts the transmit power of RUs in order to ensure that the maximum number of users requesting service is accommodated. Instead of maximizing the total network throughput, the present study proposed a demand-driven power adjustment approach aiming at optimally fulfilling the users inside a coverage area. To validate the proposed scheme, 4 RUs located inside a service area of were considered. The proposed algorithm was tested for different user deployments and varying number of users, and achieved very high-performance ratios, even in extreme demand scenarios, namely 91.51% in the case of 20 users requesting 5 Mbps each, whereas it achieved 100% performance rate for 20 users, each requiring 2.5 Mbps. Several modifications and extensions of the proposed scheme can easily take place, ranging from different network topologies (macro-, femto- and pico-cell architectures), channel models, channelization schemes, designing constraints, to alternative RL implementation, such as deep Q-network, allowing to consider larger state-action spaces.

ACKNOWLEDGEMENT

This work has been partially supported by the project "Towards a functional continuum operating system (ICOS)" funded by European Commission (Project code/Grant Number: 101070177; Call for proposal: HORIZON-CL4-2021-DATA-01; Funded under: HE | HORIZON-RIA\HORIZON-AG).

V. REFERENCES

[1] Nomikos, N., et al. (2020). A UAV-based moving 5G RAN for massive connectivity of mobile users and IoT devices. *Vehicular Communications*, *25*, 100250.

[2] Nomikos, N., Michailidis, E. T., Trakadas, P., Vouyioukas, D., Zahariadis, T., & Krikidis, I. (2019). Flex-NOMA: exploiting buffer-aided relay selection for massive connectivity in the 5G uplink. *IEEE Access*, *7*, 88743-88755.

[3] Trakadas, P., et al. (2020). Comparison of management and orchestration solutions for the 5G era. *Journal of Sensor and Actuator Networks*, *9*(1), 4.

[4] Peuster, M., et al. (2019). Introducing automated verification and validation for virtualized network functions and services. *IEEE Communications Magazine*, *57*(5), 96-102.

[5] Zafeiropoulos, A., Fotopoulou, E., Peuster, M., Schneider, S., Gouvas, P., Behnke, D., ... & Karl, H. (2020, June). Benchmarking and profiling 5G verticals' applications: an industrial IoT use case. In *2020 6th IEEE Conference on Network Softwarization (NetSoft)* (pp. 310-318). IEEE.

[6] Trakadas, P., et al. (2019). Hybrid clouds for data-intensive, 5G-enabled IoT applications: An overview, key issues and relevant architecture. *Sensors*, *19*(16), 3591.

[7] Rizou, S., et al. (2020). Programmable edge-to-cloud virtualization for 5G media industry: The 5G-media approach. In *Artificial Intelligence Applications and Innovations. AIAI 2020 IFIP WG 12.5 International Workshops: MHDW 2020 and 5G-PINE 2020, Neos Marmaras, Greece, June 5–7, 2020, Proceedings 16* (pp. 95-104). Springer International Publishing.

[8] Alvarez, F., et al. (2019). An edge-to-cloud virtualized multimedia service platform for 5G networks. *IEEE Transactions on Broadcasting*, *65*(2), 369-380.

[9] Alemany, P., et al. (2019, November). Network slicing over a packet/optical network for vertical applications applied to multimedia real-time communications. In *2019 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)* (pp. 1-2). IEEE.

[10] Caruso, G., Nucci, F., Gordo, O. P., Rizou, S., Magen, J., Agapiou, G., & Trakadas, P. (2019, September). Embedding 5G solutions enabling new business scenarios in Media and Entertainment Industry. In *2019 IEEE 2nd 5G World Forum (5GWF)* (pp. 460-464). IEEE.

[11] Andrews, J. G., Buzzi, S., Choi, W., Hanly, S. V., Lozano, A., Soong, A. C., & Zhang, J. C. (2014). What will 5G be?. *IEEE Journal on selected areas in communications*, *32*(6), 1065-1082.

[12] Zhang, C., Patras, P., & Haddadi, H. (2019). Deep learning in mobile and wireless networking: A survey. *IEEE Communications Surveys & Tutorials*, *21*(3), 2224-2287.

[13] Morocho-Cayamcela, M. E., Lee, H., & Lim, W. (2019). Machine learning for 5G/B5G mobile and wireless communications: Potential, limitations, and future directions. *IEEE Access*, *7*, 137184-137206.

[14] Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

[15] Zhao, N., Liang, Y. C., Niyato, D., Pei, Y., Wu, M., & Jiang, Y. (2019). Deep reinforcement learning for user association and resource allocation in heterogeneous cellular networks. *IEEE Transactions on Wireless Communications*, *18*(11), 5141-5152.

[16] A Giannopoulos, et al, “Supporting Intelligence in Disaggregated Open Radio Access Networks: Architectural Principles, AI/ML Workflow, and Use Cases”, IEEE Access, 10, 39580-39595, 2022.

[17] Karamplias, T., Spantideas, S. T., Giannopoulos, A. E., Gkonis, P., Kapsalis, N., & Trakadas, P. (2022, June). Towards Closed-Loop Automation in 5G Open RAN: Coupling an Open-Source Simulator with XApps. In *2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)* (pp. 232-237). IEEE.

[18] Giannopoulos, A., Spantideas, S., Kapsalis, N., Gkonis, P., Karkazis, P., Sarakis, L., Trakadas, P., & Capsalis C. "WIP: Demand-driven power allocation in wireless networks with deep Q-learning." In 2021 IEEE 22nd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), pp. 248-251. IEEE, 2021.

[19] Giannopoulos, A., Spantideas, S., Tsinos, C., & Trakadas, P. (2021, June). Power control in 5G heterogeneous cells considering user demands using deep reinforcement learning. In *IFIP International Conference on Artificial Intelligence Applications and Innovations* (pp. 95-105). Springer, Cham.

[20] Naparstek, O., & Cohen, K. (2017, December). Deep multi-user reinforcement learning for dynamic spectrum access in multichannel wireless networks. In *GLOBECOM 2017-2017 IEEE Global Communications Conference* (pp. 1-7). IEEE.

[21] “LTE Evolved Universal Terrestrial Radio Access (E-Utra): Physical Layer Procedures (3gpp ts 36.213 version 8.8.0 release 8),” ETSU TS 136 213 V8.8.0 Technical Specification, 2009-10.

[22] TSGR, E. (2011). Lte: Evolved universal terrestrial radio access (e-utra). *Multiplexing and channel coding (3GPP TS 36.212 version 10.3. 0 Release 10) ETSI TS*, *136*(212), V10.

[23] Qi, Q., Minturn, A., & Yang, Y. (2012, May). An efficient water-filling algorithm for power allocation in OFDM-based cognitive radio systems. In *2012 International Conference on Systems and Informatics (ICSAI2012)* (pp. 2069-2073). IEEE.

[24] Shi, Q., Razaviyayn, M., Luo, Z. Q., & He, C. (2011). An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel. *IEEE Transactions on Signal Processing*, 59(9), 4331-4340.