Deep Reinforcement Learning Based Resource Allocation Approach for Wireless Networks Considering Network Slicing Paradigm

Hudson Henrique de Souza Lopes, Flávio Geraldo Coelho Rocha and Flávio Henrique Teles Vieira

***Abstract*—In this paper, we present an approach for resource scheduling in wireless networks based on the Network Slicing (NS) paradigm using Double Deep Q-Network (DDQN) Reinforcement Learning (RL) algorithm. More specifically, we propose a joint power and Scheduling Block (SB) allocation algorithm for networks with NS. The reinforcement learning algorithm applied to the resource allocation problem is formulated using state transitions regarding the system dynamics. We also present an algorithm, namely Network Slicing based on Reinforcement Learning (NSRL) that combines the proposed reinforcement learning based resource allocation with an approach based on reservation and sharing of resources among the slices where each RL agent acts in one slice. Simulations are carried out considering User Equipments (UEs) within a small cell coverage area - (Small Cells) with different Modulation and Coding Schemes (MCS) standardized by the 3rd Generation Partnership Project (3GPP) based on a simplified NS scenario with fifth generation (5G) wireless network characteristics. In the simulations, two slices are considered for the UEs: one considering Ultra-reliable and Low Latency Communications (URLLC) and other related to enhanced Mobile Broadband (eMBB) services. Simulation results show that the NSRL algorithm efficiently allocates power and SBs, outperforming other algorithms in the literature.**

***Index Terms*—Resource Allocation, Network Slicing, Reinforcement Learning.**

I. Introduction

THE fifth generation (5G) of wireless communication systems emerged to meet several requirements raised from the advent of sophisticated applications such as: Vehicle to Everything (V2X); Internet of Things (IoT) and industrial automation control. These applications generate a large amount of data. In this context, the use of Machine Learning (ML) algorithms, a branch of Artificial Intelligence (AI), yields to make appropriate predictions based on learning and pattern recognition from the data provided by the 5G network. In fact, such ML algorithms can learn from experience through interaction with the environment.

The International Telecommunication Union (ITU) has established characteristics for three types of services of 5G networks [1]:

1) Ultra-Reliable and Low Latency Communications (URLLC), or in other words, communication with an imperceptible delay and a high Probability of Successful Packet Transmission (PSPT), such as that required, for example, for the successful execution of remote surgery;
2) enhanced Mobile Broadband (eMBB), applications that require a high transmission rate for adequate performance, for example, cloud gaming with virtual and augmented reality;
3) massive Machine Type Communications (mMTC), that is, a large number of connected devices as in smart cities, where sensors are placed everywhere to collect data.

These demands for the different types of services imply significant changes in the architecture and infrastructure of telecommunication networks and to accomplish these changes, Network Slicing (NS) is considered a 5G candidate technology to meet these diverse requirements [2]. Managing multiple network elements is a complex problem to solve using traditional methods. Therefore, Reinforcement Learning (RL) based algorithms in conjunction with the NS paradigm have been a promising solution for next-generation wireless networks as pointed out in [3], [4] and [5]. However, resource allocation using NS is a highly complicated problem that existing traditional approaches cannot effectively and efficiently solve due to the following characteristics [6]:

- Problem of optimization through precise mathematical models: with the increasing complexity, scale, and diversity of network services, the constraints of the physical environment and the service requirements have become complex (e.g., latency, Service Level Agreement (SLA), and security), consequently increasing the difficulty of obtaining a closed-form mathematical expression for radio resource allocation in modern wireless networks.
- Traditional approaches do not adapt to episodic uncertainty: they are exhibited as hidden structures in networks due to the lack of knowledge and subsequent ability to explore and learn from the environment.

In this context, Deep Reinforcement Learning (DRL) algorithms comprising a branch of AI that uses Deep Neural Networks (DNNs) have become attractive for exploring and learning from the environment without assuming knowledge of models. The idea of integrating DRL with NS in future wireless network designs has recently emerged and has been a promising solution for next-generation wireless networks [3], [4] and [5].

Hudson Henrique de Souza Lopes, Flávio Geraldo Coelho Rocha and Flávio Henrique Teles Vieira are with the Department of Electrical, Computer and Mechanical Engineering, Federal University of Goiás, Goiânia, GO, Brazil (e-mail:hudson_lopes@ufg.br, flaviogcr@ufg.br, flavio_vieira@ufg.br).

The authors would like to thank Fundação de Amparo à Pesquisa no Estado de Goiás (FAPEG) and to the Centro de Excelência em Inteligência Artificial (CEIA) for their support in the development of the research.

Digital Object Identifier: 10.14209/jcis.2023.4

In this paper, we propose an approach based on Double Deep Q-Network (DDQN) Reinforcement Learning algorithm to solve the resource allocation problem in wireless networks considering NS. More specifically, we propose a joint power and Scheduling Block (SB) allocation approach considering the scenario with two slices, one for eMBB services and the other for URLLC services. The proposed approach is based on a heuristic algorithm that uses reservation and resource sharing to meet the Quality of Service (QoS) criteria in each slice. In order to apply and to evaluate the performance of the proposed approach, a communication scenario is considered in the computer simulations that consists of two network slices, i.e. eMBB and URLLC, where a reward function corresponding to the reinforcement learning algorithm was established for each slice and real data traffic from 5G networks are used. In the proposed approach, it is also assumed that the relation between the PSPT and the signal transmission power for each Channel Quality Indicator (CQI) can be represented by sigmoidal curves for each CQI, so nonlinear curve fitting is performed using an iterative method based on the Levenberg-Marquardt (LM), a nonlinear least-squares method that can estimate the parameters of a sigmoidal function.

The main contributions of this paper are summarized as follows:

- Proposal of using sigmoidal functions to represent the PSPT in terms of the Base Station (BS) power allocated to the User Equipments (UEs) for different CQIs according with the latest standard of the Third Generation Partnership Project (3GPP) [7].
- Formulation of the NS-based resource allocation problem considering real data traffic from 5G networks and assignment of Adaptive Modulation and Coding (AMC) schemes.
- Proposal of reward functions for the eMBB and URLLC slices, for which the learning process of each agent occurs without prior knowledge of the system statistics.
- Proposal of an approach to allocate power and SBs via DDQN based Reinforcement Learning strategy to solve the resource allocation problem in wireless networks considering NS, namely Network Slicing based on Reinforcement Learning (NSRL) algorithm, capable of supporting the dynamism and elasticity of end-to-end communications.

The remainder of this paper is structured as follows: section II reports the main works related to this paper; section III presents the model for the considered wireless communication system; section IV describes the method used to calculate the values of the parameters of sigmoidal utility functions that represent the PSPT as a function of signal transmission power; section V formulates the resource allocation problem considering a scenario with two slices, eMBB and URLLC, where each slice has its reward function. More specifically, in this section, we propose an algorithm for joint power and SBs allocation using reinforcement learning; section VI describes the main concepts of network slicing and its main advantages for modern wireless networks. In this same section, we present an algorithm, namely NSRL that combines the proposed reinforcement learning based resource allocation with an approach based on reservation and sharing of resources among the slices where each RL agent acts in one slice; section VII discusses the results obtained for each agent operating in each slice and evaluates the QoS parameters obtained for the wireless communication system. Finally, section VIII summarizes the obtained conclusions.

II. Related Work

Taking into account the challenges related to resource allocation in modern wireless networks, such as those involving 5G and beyond 5G (B5G), in the last decade many research projects have been developed that achieved promising results. In [8], the authors study the radio resource allocation problem involving energy efficiency optimization with specific QoS requirements for multiservice scenarios. In the present paper, we go further by using reinforcement learning to perform resource allocation in a network slicing scenario.

In [3], the authors deal with the problem in the resource allocation for network slicing scenarios using an approach based on the Markovian Decision Process (MDP) and DRL. The system model uses synthetic and real 4G workload data and the resources that are intended to be allocated are bandwidth and Virtual Machines (VM). However, these authors do not consider power allocation to UEs, a factor that directly influences the quality of the communication channel. In [4], the authors propose a new method for resource allocation for NS that integrates DRL and the Alternating Direction Method of Multipliers (ADMM). Likewise in [3], the authors in [4] also do not take into account the problem of efficient power allocation to UEs.

In [1], the transmission is in the uplink direction and the optimization problem is formulated as an MDP with infinite horizon and average reward. Based on the equivalent Bellman equation, the optimal power policy suffers from the problem widely known as the curse of dimensionality. Due to the high dimensionality of the state space. To solve this problem, the approximate dynamic programming method is used to simplify the optimization problem. In [9], the authors investigated the problem of maximizing the system throughput subject to user satisfaction ratio constraints in a multiservice scenario and proposed a new decentralized radio resource allocation mechanism employing multi-agent deep reinforcement learning. However, approaches where learning-based techniques are jointly responsible for power and bandwidth allocation are not analyzed.

In [10], the authors propose Deep Q-Learning (DQL) based resource allocation strategies in Industrial Wireless Nodes (IWNs) using URLLC services. The dynamic resource allocation of IWN are discussed based on the interference level, reliability, latency, and data rate. Variability is experienced in the individual IWN in terms of resource allocation due to rewards being dependent on actions of other IWNs. The impact of parameters intrinsic to DQL shows that this variability is controlled for lower values of learning rate and discount factor. Nevertheless, the authors did not consider an environment with eMBB services in [10].

In [5], the authors propose a network slicing scenario where each slice uses an intelligent agent that competes for limited radio resources and takes actions autonomously using Q-Learning as the algorithm for resource allocation. This approach intends to jointly optimize the performance of URLLC and eMBB services. In this paper, we also address URLLC and eMBB service performances, but exploring a DRL technique for resource allocation between slices.

In [11], the authors formulate the resource allocation problem as a Constrained Markov Decision Process and solve it using constrained reinforcement learning and assume user traffic patterns and mobility are unknown to the slicing algorithms, the algorithm explores and learns from the network without knowing those prior knowledge. This is one of the reasons why learning-based approaches, which incorporate exploration, perform better than the traditional methods based only on observed states. However, the work presented in [11] also does not consider reliability requirement of packet transmission in URLLC services.

The coexistence of URLLC and eMBB services using the same radio resource leads to a challenging resource allocation problem that is not easy to solve due to the trade-off between latency, reliability, and spectral efficiency. The main objective of this paper is to solve in a flexible and efficient way, the radio resource allocation problem, improving the desired QoS parameters for URLLC and eMBB services. To achieve this goal, we propose to use sigmoidal function modeling for each CQI standardized by 3GPP to perform power allocation among UEs in the same slice and we also propose a hybrid algorithm that uses RL agents for joint allocation of power and SB in each slice.

III. System Model

The proposed wireless communication system model is in the downlink direction and consists of a Small Cell with a BS in its center. We considered a scenario with two slices, each one of them with respect to the eMBB and URLLC use cases, as shown in Fig. 1. The resource allocation is performed at each Transmission Time Interval (TTI) and the position of each UE in the cell varies along the time, resulting in variable CQIs. Moreover, it is assumed that the initial distribution of UEs in each slice and the mode in which they move in the coverage area follow uniform probability distributions.

In time domain, the duration of the downlink frame of the Orthogonal Frequency Division Multiplexing (OFDM) signal is 10 ms. These frames are divided into 10 sub-frames, each one of them representing a TTI of 1 ms.

The bandwidth B is divided into S sub-bands indexed by f = $\{1, 2, ..., S\}$ and the time domain is divided into time intervals indexed by t = $\{1, 2, ..., O\}$ with a duration of 0.5 ms. A Resource Block (RB) is the minimum resource assignment, which consists of 7 OFDM symbols in case of normal Cyclic Prefix (CP), while 6 OFDM symbols in case of extended CP over a time interval of 0.5 ms, and 12 consecutive subcarriers (for a bandwidth of 180 kHz) [12]. Our implementation of 5G system takes into account the 4G LTE infrastructure in a non-standalone (NSA) mode, in such system, RBs are always scheduled in pairs, thus called Scheduling Block (SB), with a duration of 1 ms as shown in Fig. 2.

Fig. 1. Mobile Communication System Model.

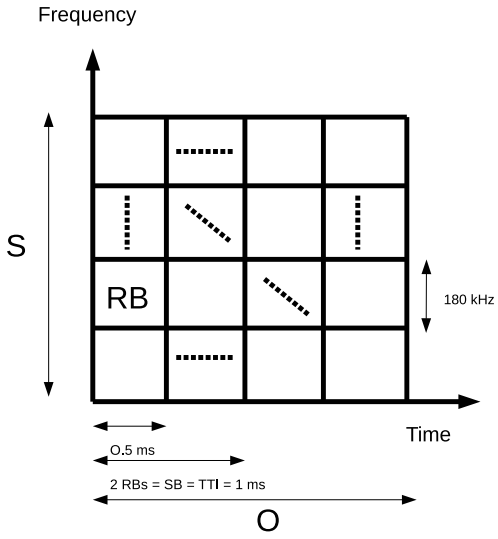

Fig. 2. Radio resources in time and frequency domains

A. Channel Quality Indicator (CQI)

The CQI is the feedback that the UE sends to BS to indicate the data rate that can be supported by the downlink channel, it is calculated at BS based on the estimated channel parameters and Signal-to-Noise Ratio (SNR). It is reported via Media Access Control (MAC) layer signaling in a transmission over Physical Sidelink Shared Channel (PSSCH) for this purpose [13]. The BS selects an appropriate modulation and coding scheme based on the CQI values as shown in Table I.

Release 12, known as LTE-Broadcast (LTE-B), was the first release to support high-order modulation schemes up to 256-QAM (QAM-Quadrature Amplitude Modulation) [7]. In this work, all simulations are carried out using Table I, that considers modulation schemes up to 256-QAM. High-order

TABLE I
CHANNEL QUALITY INDICATOR (CQI) [7]

Index	Modulation	Code Rate	Efficiency	Threshold SNR
1	QPSK	0.07	0.15	-6.92
2	QPSK	0.18	0.38	-5.01
3	QPSK	0.43	0.87	-3.09
4	16-QAM	0.36	1.47	1.18
5	16-QAM	0.47	1.91	0.73
6	16-QAM	0.60	2.40	2.64
7	64-QAM	0.45	2.73	4.56
8	64-QAM	0.55	3.32	6.47
9	64-QAM	0.65	3.90	8.39
10	64-QAM	0.70	4.52	10.30
11	64-QAM	0.85	5.11	12.21
12	256-QAM	0.69	5.55	14.13
13	256-QAM	0.77	6.22	16.04
14	256-QAM	0.86	6.90	17.96
15	256-QAM	0.92	7.40	19.87

modulation schemes require higher SNR so that they can guarantee a minimum PSPT [14].

Table I also shows the threshold SNR values for each CQI considering a linear function (1) proposed in [15], used to map SNR to the CQI using Exponential Effective SNR Mapping (EESM). The SNRs obtained by Equation (1) below are in (dB) and are considered to provide the threshold values for finding the ranges of each CQI.

$$CQI = 0.52 \times SNR + 4.61. \tag{1}$$

B. Signal Transmission Power and SNR

In this work, we consider the scenario of the communication system in [16] that relates the transmission power as a function of the SNR values according to Eq. (2). Similar approaches are made in [17] and [14].

$$SNR(p_i) = \frac{N_i G_i p_i}{G_i \theta [\sum_{i=1}^{J}(p_i) - p_i] + I_i} = \frac{N_i p_i}{\theta(P_{bs} - p_i) + A_i}, \tag{2}$$

where the parameters of this system are:

- p_i is the power allocated to the i-th UE;
- G_i is the path gain from a BS to a UE;
- N_i is a constant (e.g., processing gain);
- I_i is the background noise and inter cell interference to the i-th UE;
- A_i is the "goodness" of the transmission environment of the i-th UE, which is defined as $A_i = \frac{I_i}{G_i}$;
- θ is the orthogonality factor;
- P_{bs} is the total BS power.

Similarly as performed in [16], the parameters were set as: P_{bs} = 10 W, θ = 1, N_i = 16 and A_i = 0.7407. A packet size of 1024 bytes with channel coding was considered, the packet size is based on the IEC-61850 standard for energy distribution of medium and high voltage [18].

IV. SIGMOIDAL FUNCTION FOR PSPT

We assume that the Probability of Successful Packet Transmission (PSPT) in a wireless network can be represented by a sigmoidal function of its power allocation. Fig. 3 shows an example of a collection of sigmoidal functions that indicate the PSPT distribution in function of the transmission power. The curves were plotted for several CQI values, according to Table I.

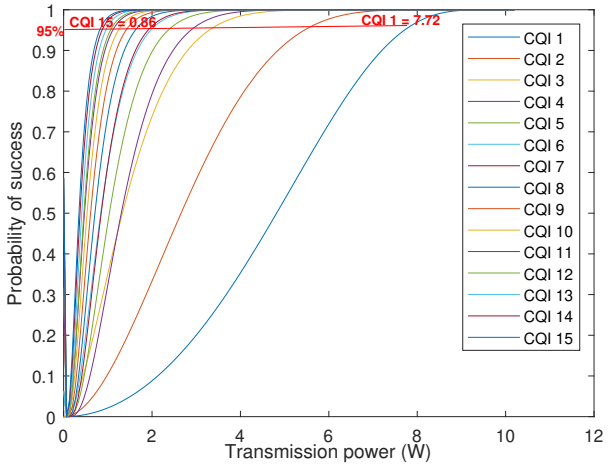

Fig. 3. Probability of successful packet transmission

Once the bit error probability equation depends on a numerical integral, in this work, the PSPT is modeled by a sigmoidal utility function with a normalization rule, represented by (3). In a similar manner to the works presented in [14], [16] and [19], in this approach, the utility function is related to the QoS of a user and can be used to control efficiency and fairness in resource allocation.

$$U_j(p) = \left(1 + \frac{1}{e^{a_j.b_j}}\right)\left(\frac{1}{1 + e^{-a_j(p-b_j)}} - \frac{1}{1 + e^{a_j.b_j}}\right). \tag{3}$$

We make the following assumptions: $U_j(0)$ = 0 and $U_j(\infty)$ = 1. The Levenberg-Marquardt (LM) algorithm is used in this work to estimate the parameter values of the sigmoidal function (3) fitting to the real curves via minimization of the Mean Squared Error (MSE) for each CQI j given by Eq. (4).

$$MSE_j = \frac{1}{k}\sum_{n=1}^{k}(R_j(p_n, m))^2, \tag{4}$$

where m = $[a_j, b_j]$, k is the number of points and $R_j(p, m)$ is the vector of residues, that is, the difference between the expected value and the estimated by Eq. (5).

$$R_j(p_k, m) = F_j(p_k) - U_j(p_k, m), \tag{5}$$

where $F_j(p_k)$ is the PSPT as a function of allocation power. The goal of applying the LM method is to obtain the values of a_j and b_j that minimize the MSE given by Eq. (4) for each CQI j. The LM method is summarized in (6) and (7) [20].

$$(J_k^T(m)J_k(m) + \mu_k \times II)\delta_k = -J_k^T(m)R_j(p_k, m), \tag{6}$$

$$m = m + \delta_k, \tag{7}$$

where J_k is the Jacobian matrix of U_j applied on m, II is the identity matrix, δ_k is the displacement vector and μ_k

is the damping parameter, the strategy for updating of μ_k is described in [21].

Fig. 4 shows the comparison of the parameterized curves with the real curves. The values obtained via the LM algorithm for the parameters of the 15 utility functions are shown in Table II. At certain points, convergence occurs faster. But in general, a low MSE in the order of 10^{-4} was obtained, as shown in Table II.

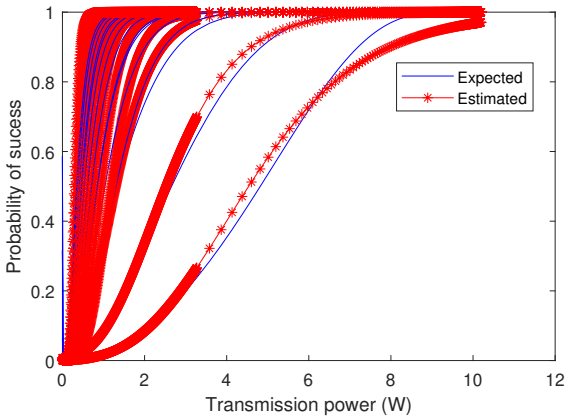

Fig. 4. PSPT parameterized by the utility function using the Levenberg-Marquardt algorithm.

TABLE II
VALUES FOR THE A AND B PARAMETERS OF THE UTILITY FUNCTIONS FOR EACH CQI.

CQI	a	b	MSE
1	0.57	2.79	3.46×10^{-4}
2	0.95	1.43	5.28×10^{-4}
3	1.75	0.68	5.22×10^{-4}
4	2.62	0.84	1.38×10^{-3}
5	3.32	0.66	1.17×10^{-3}
6	4.11	0.53	1.00×10^{-3}
7	4.75	0.55	1.43×10^{-3}
8	5.73	0.45	1.24×10^{-3}
9	6.71	0.39	1.10×10^{-3}
10	7.77	0.34	1.00×10^{-3}
11	8.79	0.30	9.18×10^{-4}
12	9.77	0.29	1.04×10^{-3}
13	10.97	0.26	9.66×10^{-4}
14	12.22	0.23	9.01×10^{-4}
15	13.13	0.22	8.62×10^{-4}

V. RESOURCE ALLOCATION BASED ON REINFORCEMENT LEARNING

In this section, we propose the application of reinforcement learning to solve the resource allocation problem in wireless networks considering network slicing concept. In this reinforcement learning approach, it is considered that the system states are represented by the number of packets that are waiting for transmission in the buffer $q(t)$, and the action given by the tuple: SB and PSPT $U(.)$. The $U(.)$ function relates the PSPT to the power allocated to the UEs, while the SBs are related to the bandwidth that is allocated to the UEs. We also propose two reward functions, one for each network slice, whose objectives are: slice URLLC maximizes the PSPT $U(.)$ and minimizes the average delay $\frac{q(t)}{\lambda}$, where λ is the parameter of the Poisson distribution; slice eMBB maximizes the throughput given by: $max(0, d(t) - q(t))$, where $d(t)$ is the channel capacity and $q(t)$ is the amount of packets queued to be transmitted.

In this work, the main elements for applying RL to the resource allocation problem are described as follows:

System state: the number of packets waiting to be transmitted in the t-th subframe is taken as the system state $s(t)$, i.e. the packets that are in the buffer:

$$s(t) = \{q(t)\}. \tag{8}$$

An R-sized buffer stores a maximum of R packets for all UEs. The system states are obtained while taking into account $(R + 1)$ possible states, including zero. Hence, the buffer state changes with the arrival or departure of packets at each subframe.

Control policy: the control action is the joint power (given by the PSPT $U(.)$ function) and SBs allocation to the UEs according to the queue size. Therefore, the control action on the t-th subframe is given by the tuple:

$$a(t) =< U(p), sb >, \tag{9}$$

where sb is the amount of scheduling blocks in each slice, and can assume discrete values in the range 1 to 100 for a bandwidth of 20 MHz. For the scheduling of sb between UEs of the same slice the *Round-Robin* (RR) algorithm was used

The RR algorithm consists of assigning the resources in equal parts in a circular order, handling the whole process without priorities. Therefore, this scheduler tends to perform a homogeneous distribution of the available resources and it is characterized by presenting a good fairness index among the UEs.

For power allocation the function $U(p)$ was used, so the power allocation for each UE can be written as:

$$\arg\max_{p_i} U_i(p_i) \quad \forall i = 1, 2, ..., Z, \tag{10}$$

where Z is the total number of UEs and $U_i(p_i)$ is discretized to two decimal digits $\in$ (0.0, 0.99).

State transition: given the system state, CQIs, and the control action of the t-th subframe, the state transition is represented by the dynamic queuing equation given by:

$$q(t+1) = max\{0, q(t) - d(t)\} + c(t), \tag{11}$$

where $q(t+1)$ is the new state of the system, $c(t)$ is the number of packets arriving in the t-th subframe. It is assumed that $c(t)$ follows the Poisson distribution with parameter λ. Thus, there are on average λ packets in a subframe. Packets of 5G network real data traffic collected from a large Irish mobile operator described in [22] are used. This dataset corresponds

to video streaming application in a mobile network. $d(t)$ is the channel capacity and represents the quantity of packets that are transmitted in the t-th subframe and is given by:

$$d(t) = \left(\frac{\sum_{i=1}^{Z} n_i \times c_i(SNR(p_i))}{B} \right), \tag{12}$$

where n_i is the amount of the Scheduling Block (SB) that are allocated to the i-th UE. B is the amount of bits in a packet. As in [12], the bit-rate for each SB can be rewritten as:

$$c_i(SNR(p_i)) = L \times \sigma \times SE(SNR(p_i)), \tag{13}$$

where L is the bandwidth for each SB, σ is the transmission time interval of each SB, and $SE(.)$ is the spectral efficiency of the selected CQI using Adaptive Modulation and Coding (AMC), according to Table I.

Reward function: According to Little's Law the average packet delay is given by [23]:

$$W = \frac{\bar{Q}}{\lambda} = \lim_{T \to \infty} E\left(\frac{1}{T} \sum_{t=1}^{T} \frac{q(t)}{\lambda} \right), \tag{14}$$

where $\bar{Q}$ is the average number of packets waiting to be transmitted. In this paper we consider the scenario where we have two slices with the URLLC and eMBB services, for the allocation of resources between the slices we propose to use one agent in each slice for that it learns based on a reward function. Therefore, in URLLC slice the following reward function is proposed:

$$r(t) = \sum_{i=1}^{Z} U_i(p_i) - \alpha \times W, \tag{15}$$

where α is the weight on the transmission average delay. Generally in reinforcement learning algorithms, the main objective is to maximize the reward function. Since the average delay has a negative value in the equation its value is forced down by a weight and the PSPT is maximized, according to the QoS of URLLC services.

For the eMBB slice, the main objective is to maximize throughput. Therefore, the queue size is penalized so that the channel capacity provides a higher throughput of packets that are in the buffer. Hence, the proposed reward function is given by:

$$r(t) = d(t) - \beta \times q(t), \tag{16}$$

where β is the weight on the amount of packets that are in the buffer.

A. Value-Based Reinforcement Learning Algorithms

Value-based algorithms are used to estimate the agent's value function. This value function is then used to implicitly and greedily obtain an optimal policy. There are two types of value-based function: the value function $V^{\pi}(s)$ and the state-action function $Q(s(t), a(t))$. Both represent the expected cumulative discontinuous reward received when taking an action $a(t)$ in state $s(t)$ for the value function or the pair ($s(t)$, $a(t)$) for the state-action function). These functions are very important since they represent the link between the mathematical formulation of the MDP and Reinforcement Learning. In MDP, given an action, we have the action-value function, which depends on both the state and the action just taken. The MDP based agent provides an expected return under a state and an action. If the agent acts according to a policy π, we denote it as $Q^{\pi}(s(t), a(t))$. The main aim of MDP is to obtain the optimal policy π^* (i.e., map the states to optimize the actions for maximizing the expected return), which is given by [24]:

$$\pi^* = \arg\max_{a(t)} E[\sum_{t=1}^{T} \gamma r_t(s(t), \pi(s(t)))], \tag{17}$$

$$\pi^*(s) = \arg\max_{a} Q^{\pi}(s(t), a(t)). \tag{18}$$

In RL, Q-Learning is the most widely used algorithm to approach MDPs [24]. It obtains optimal values of the Q-function by iteratively updating rule using the Bellman equation

$$Q_{t+1}(s,a) = Q_t(s,a) + \omega[r_t(s(t), a(t)) + \gamma \max_{a(t+1)} Q_t(s(t+1), a) - Q_t(s(t), a)], \tag{19}$$

where ω is the learning rate, γ is the discount factor $\in$ (0,1) and $r_t(s(t), a(t))$ is the reward of taking action $a(t)$ in state $s(t)$.

B. Double Deep Q-Network (DDQN)

The Q-Learning algorithm is based on building a table for the values of the Q function. Due to this reason, when the state space and the action space become large as in the cases commonly encountered in the resource management problems of modern wireless systems to obtain the optimal policy could be extremely time consuming.

In order to solve this problem, Deep Q-Network (DQN), which inherits the advantages of the Q-Learning and Deep Learning (DL) techniques are commonly used. The main idea is to replace the table of Q-Learning algorithms with a DNN that approximates the Q-values.

The DNN is also called a universal approximation function and is denoted by $Q(s(t), a(t)|\Theta)$, where Θ represents the parameters or weights of the DNN. To increase the stability of the DQN, another neural network is used, called the target Q network, whose weights $\Theta^{'}$ will be periodically updated to follow those of the main Q neural network [24].

The DQN algorithm is iteratively optimized by updating the Θ weights of its DNN to minimize the following Bellman loss function:

$$L(\Theta_t) = E_{s(t),a(t),r_t,s(t+1)}[r_t(s(t), a(t)) + \gamma \max_{a(t+1)} Q(s(t+1), a(t+1)|\Theta^{'}) - Q(s(t), a(t)|\Theta)]^2, \tag{20}$$

where $\Theta^{'}$ are the weights of the target Q network.

The DQN algorithm tends to overestimate Q-values, which can degrade the training process and lead to sub optimal policies. The overestimation results from the positive bias caused by the maximal operation employed in the Bellman equation. Specifically, the root cause is that the same training transitions are used in the selection and evaluation of an action [24]. As a solution to this problem, we propose to use the Double DQN (DDQN) technique, where two functions are employed for the Q-value, one to select the best action and the other to evaluate the best action. The action selection is still based on the Θ weights, while the second $\Theta^{'}$ weights are used to evaluate the value of this policy as shown in Fig. 5. Therefore, as in conventional Q-Learning, the value of the policy is still estimated based on the current Q-value. The parameters of weights $\Theta^{'}$ are updated by Θ [24].

The DDQN Algorithm uses the following modified Bellman loss function to update its weights:

$$L(\Theta_t) = E_{s(t),a(t),r_t,s(t+1)}[r_t(s(t),a(t)) + \gamma Q(s(t+1), \arg\max_{a(t+1)} Q(s(t+1),a(t+1)|\Theta),\Theta^{'}) - Q(s(t),a(t)|\Theta)]^2. \quad (21)$$

The DDQN architecture approaches three key techniques to improve convergence in learning [25]:

- Experience replay: Experience samples generated by the agents are stored in an experience replay buffer, and then a mini-batch is randomly sampled from the buffer to train the neural network. This strategy breaks the correlation between the training samples.
- Target Q-network: As shown in Fig. 5, the main network is trained using the loss function from Eq. (21), while a target network is run to generate experience samples for training purposes. The main Q-network and the target Q-network have the same architecture but with two different sets of weights: Θ and $\Theta^{'}$, respectively. The weights of the target network are updated periodically every τ step with the same weights as the main network.
- Decoupling in action selection and evaluation: The target Q-network is used to generate the Q-values that will be used to calculate the loss during training, while the main Q-network is used to select which is the best action to take to the next state. By decoupling action selection from evaluation, the risk of overestimating Q-values can be greatly reduced. In other words, in case the main Q-network overestimates the action, the target Q-network would generate an appropriate value.

Algorithm 1 illustrates the training process in more detail.

VI. Network Slicing with Reinforcement Learning

Network Slicing is an innovation in the 5G network architecture that plays an important role for the next generations, such as those B5G and sixth generation (6G). It allows multiple virtual networks to coexist independently and isolated

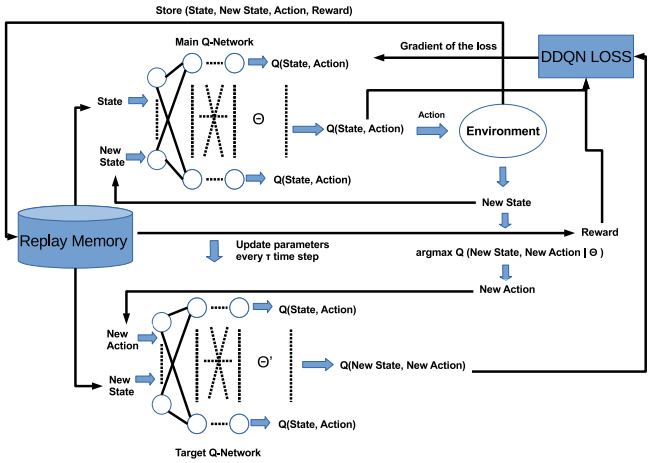

Fig. 5. Double DQN architecture

Algorithm 1: DDQN Training Process

```
probability of choosing an action ϵ = 0.1;
Initialize the weights of the main Q-network Θ = random;
Initialize the target Q-network weights Θ' = Θ;
for t from 1 to t_max do
    if mod(t,τ) == 0 then
        Updates the weights of the target network Θ' = Θ;
    end
    Exploration condition based on greedy policy;
    if rand(0,1) < ϵ then
        Sample a random action a(t);
    else
        a(t) = arg max_{a(t)} Q(s(t), a(t)|Θ);
    end
    Apply a(t) to the environment and observe r(t) and s(t+1);
    Add recent experience <s(t),a(t),r(t),s(t+1)> in the replay buffer;
    Sample a mini-batch from the replay buffer;
    Updates Θ according to the loss function Eq.(21);
end
```

in the same physical network infrastructure. Software Defined Networking (SDN) and Network Functions Virtualization (NFV) are key technologies for implementing NS and accommodating new services with very different requirements on the same infrastructure. The slices are established through a set of logic controllers and resources provided by the SDN controller. In addition, with SDN, NS allows any telecommunication company to share the resources of its network infrastructure with different operators, groups of User Equipments (UEs), or demand profiles.

Some important advantages of the NS concept are [1]:

- guarantee of a SLA for each type of service;
- provision of differentiated services; that is, NS increases

the adaptability and flexibility of the network management since the slices can be created and modified as required;
- provide support to multi-users based on the use of the same physical infrastructure by several Mobile Network Operators (MNOs).

The main factors resulting in the rapid adoption of network virtualization are low cost in resource sharing and high network utilization [26]. For 5G and B5G, the virtualized networks should become more common in practice in order to optimize infrastructure utilization. In this context, future virtualized networks demand new management mechanisms that provide efficiency in resource allocation.

In [27], the authors proposed an solution for slice-based resource allocation that firstly schedules a portion of the resources using a reservation-based approach and then allocates the remaining resources using a sharing-based approach. In this paper, we propose a solution to the resource allocation problem that combines the advantages of both mentioned approaches. For such a purpose, we propose the **Algorithm 2** (NSRL - *Network Slicing based on Reinforcement Learning*), a joint approach of resource reservation and sharing strategies. The proposed NSRL algorithm uses the RL agents to learn to allocate power and SBs efficiently according to the reinforcement learning algorithms described in section V. More specifically, to provide the high-reliability requirement of URLLC services it is necessary to make an efficient control of the power allocated by the BS. In the same way, to provide high throughput for eMBB services, it is necessary to make efficient control of the SB that are allocated as a consequence, the bandwidth allocated by the BS.

Algorithm 2 is a heuristic that aims to use the reservation-based approach and RL agents to learn through reward functions to efficiently allocate power and SBs in each slice. The agents compete for the limited network resources in order to enhance the desired QoS parameters of the URLLC and eMBB service. Let $\Omega_1 = \{1, 2, ..., K\}$ and $\Omega_2 = \{1, 2, ..., H\}$ be the UEs associated with the eMBB and URLLC services respectively. We propose the following strategies to be applied to the NSRL **Algorithm 2**:

- It is desirable that the network slice related to the URLLC service enhances the desired QoS parameters (i.e. latency and reliability) or at least provides adequate values to the UEs. To this aim, a fraction of the total base station power P_{bs} is reserved to this slice and referred in the algorithm as P_{urllc}. The percentage of the reservation is defined by the variable $percent$. In case the reserved power has not been fully allocated by the RL agent, this remaining power is made available to the eMBB UEs using the sharing-based approach;
- Similarly, to provide high throughput value for the eMBB services, it is necessary to effectively control the bandwidth allocated by the BS. The scheduling blocks (SBs) reserved for the eMBB UEs are represented by the variable sb_{embb} and the remaining unallocated SBs by the RL agent are made available to the URLLC UEs.

The proposed NSRL algorithm is based on the integration of a heuristic with reinforcement learning agents. Although we propose to use DDQN, different RL agents such as that based on Q-Learning can be used in the NSRL. The scenario considered in this work consists of two agents acting in each slice (URLLC and eMBB) simultaneously, where each slice possesses a corresponding reward function. Equations (15) and (16), that represent $r(t)$ in **Algorithm 1**, are integrated into the reinforcement learning agent training process for driving the actions taken by the agent in a state for URLLC and eMBB slice, respectively.

In this paper, we compare the performance of the NSRL algorithm with two DDQN agents in each slice (URLLC and eMBB) simultaneously with two Q-Learning agents acting on both slices. However, the Q-Learning agent has limitations when applied to radio resource management in modern wireless networks. Q-Learning is applicable in problems with low dimensionality of both the state space and actions, which makes it unscalable. Moreover, it is applicable only with discrete state space, unlike DDQN which uses a neural network that does not depend on discrete state space. In this work, a buffer with size of four packets was used to be able to quantize the state space. The action space was discretized into a combination of 100 values for reliability (probability of successful packet transmission) and 100 values for SB. Thus, considering the empty state of the buffer, the state and action space for each slice for Q-Learning algorithm will be $100 \times 100 \times 5$ = 50000. In this work, we assume a scenario with two slices. Therefore, the number of possible combinations for resource allocation between the two slices will be $10000 \times 10000 = 10^8$.

Given the limited bandwidth and transmission power resources of the BS, each agent acting in a slice will compete for resources to optimize its own goal, which can lead to a conflict. The agent in each slice chooses the action that has the highest Q value for a given state s while respecting the limited resource available. Thus, according to line 10 of **Algorithm 2** the agent chooses the action considering the constraints represented by Eqs. (23), (24) and (25) in the slice URLLC.

$$\arg\max_{a(t)} \quad Q(s(t), a(t)|\Theta) \tag{22}$$

$$\text{Subject to} \quad U_{urllc} \leq U(urllc_p^i), \forall i \in \Omega_2 \tag{23}$$

$$\sum_{i=1}^{H} urllc_p^i \leq P_{bs} - P_{embb}, \tag{24}$$

$$sb_{urllc} \leq sb_{bs} - sb_{embb}. \tag{25}$$

In the eMBB slice, according to line 18 of **Algorithm 2** the agent chooses the action considering the constraints represented by Eqs. (27), (28) and (29).

$$\arg\max_{a(t)} \quad Q(s(t), a(t)|\Theta) \tag{26}$$

$$\text{Subject to} \quad U_{embb} \leq U(embb_p^i), \forall i \in \Omega_1 \tag{27}$$

$$\sum_{i=1}^{K} embb_p^i \leq P_{bs} - P_{urllc}, \tag{28}$$

$$sb_{embb} \leq sb_{bs} - sb_{urllc}. \tag{29}$$

After defining the resources in each slice represented by the tuple $(U(.), sb)$, allocation of resources can be carried out. For power allocation the sigmoidal modeling described in section IV is used. Moreover, the power allocated to the UE i in slice eMBB is represented by the variable $embb_p^i$. Similarly, the power allocated to the UE i in slice URLLC is represented by $urllc_p^i$. For allocation of SBs, the widely known *Round-Robin* algorithm is used for making a homogeneous allocation of the resources. In terms of notation, the SBs allocated to the UE i in the eMBB slice is represented by the variable $embb_{sb}^i$. Similarly, the SBs allocated to the UE i in the URLLC slice is represented by $urllc_{sb}^i$.

Algorithm 2: NSRL

Input: $percent$, P_{bs}, sb_{bs}.
Initialize:
Resource reservation in each slice;
$P_{urllc} = percent \times P_{bs}$;
$P_{embb} = P_{bs} - P_{urllc}$;
$sb_{embb} = percent \times sb_{bs}$;
$sb_{urllc} = sb_{bs} - sb_{embb}$;
s = Real data traffic in [22];
repeat
  The agent takes action using Eq. (22);
  for *i from 1 to H* **do**
    Allocate power to the UE URLLC;
    $\arg\max_{urllc_p^i} (U(urllc_p^i)) = U_{urllc}$;
    Calculate the SNR_{urllc}^i using the power P_{urllc}^i in (2);
    Calculate the bit-rate for each SB for each UE URLLC using AMC in (13);
    $urllc_{sb}^i$ = Allocate the sb_{urllc} for the UE URLLC using the *Round-Robin* algorithm;
  end
  The agent takes action using Eq. (26);
  for *i from 1 to K* **do**
    Allocate power to the UE eMBB;
    $\arg\max_{embb_p^i} (U(embb_p^i)) = U_{embb}$;
    Calculate the SNR_{embb}^i using the power P_{embb}^i in (2);
    Calculate the bit-rate for each SB for each UE eMBB using AMC in (13);
    $embb_{sb}^i$ = Allocate the sb_{embb} for the UE eMBB using the *Round-Robin* algorithm;
  end
  $P_{sum} = sum(urllc_p)$;
  $sb_{sum} = sum(embb_{sb})$;
  $P_{embb} = P_{bs} - P_{sum}$;
  $sb_{urllc} = sb_{bs} - sb_{sum}$;
  s = Calculate the new state using Eq. (11);
until *Simulation Time*;

VII. Results

In this work, simulations were carried out using the following software and hardware configurations: Matlab software version R2021, Intel Core i5-1035G1 processor 1.00 GHz; 8 GB of RAM without a dedicated video card. The DDQN neural network was implemented with 4 fully connected hidden layers with 64 neurons and Leaky ReLU activation function. The hyperparameters were configured with a learning rate of 0.1 (ω), random action choice chance (ϵ) of 0.1, discount factor (γ) of 0.9, period for updating target Q-network weights (τ) every 50 steps, mini-batch size of 256 and the replay buffer size of 10000. For the Q-Learning algorithm, the same values were used for the ω, ϵ and γ hyperparameters. The rest of the full set of parameters for the simulations is given in Table III.

TABLE III
Simulation parameters

Parameters	Values
BS power (P_{bs})	10 W
Maximum buffer size (R)	4 packets
Small cell radius	300 m
TTI	1 ms
Mobility model	UEs are uniformly distributed in a small cell in each TTI
Number of OFDM symbols per TTI	7
Number of sub-carriers per SB	12
Each sub-carrier length	15 KHz
Each SB bandwidth	180 KHz
Bandwidth	20 MHz
SBs per TTI	100
Arrival and departure of UEs in each slice	Uniform distribution
Weight on average delay (α)	50
Weight on the buffer size (β)	10
5G real data traffic	[22]
Package arrival rate (λ)	Possoin distribution with rate 3
Number of bits in a packet (B)	8192
Training iterations	10000

The simulation time is 10000 TTI and the scenario consists of arrival and departure of UEs in each network slice and their mobility over the coverage area occurs randomly using a uniform distribution each 1000 TTI, i.e., it is assumed that there is no change in the scenario for each slice during 1000 TTI, for the total simulation time 10 changes are made. When considering a simulation time of 10000 TTI and number of rounds greater than 10, there was no significant change in the results obtained with the considered algorithms.

The average number of packets of real 5G data traffic arriving at each TTI is computed and used in the simulations as the parameter λ of the Poisson distribution. The data traffic used in the URLLC and eMBB slices is available in [22]. This dataset is based on the Netflix video stream with the driving mobility standard collected from the 5G mobile network. In our simulations, we also assume a finite buffer traffic model. With this model, the number of UEs in the cell varies with time. In this work, we limit the arrival and departure rate of UEs to 10, because there are temporary periods when an accumulation of UEs with poor channel quality occurs. Due to its simplicity, this type of traffic model has been widely adopted in OFDM-based simulations [28].

Figs. 6, 7, 8, 9, 10 and 11 show the performance of the NSRL algorithm in allocating resources to each agent (Q-Learning and DDQN) in each slice (eMBB and URLLC). A comparison is also performed with other algorithms in the

literature such as the Distributed Algorithm (DA) proposed in [14], an algorithm based on the proportional fairness technique that has as main objective to allocate more resource to the UEs with worse channel quality, and the Equal Sharing (ES) power algorithm described in [12], which allocates the same power to all UEs. That is, it consists of a simple share of the BS power. All the power allocation algorithms considered in this work are combined with the Round-Robin (RR) algorithm for SBs allocation.

In the simulations, 50% of the power resources and SBs in each slice are reserved when considering the DA and ES algorithms. That is, in this case, a power value equal to 5 W and 50 SBs are reserved for each slice. For the proposed NSRL algorithm, the methodology used consists in performing a minimum resource reservation, and if all the reserved resource is not used, share it with the other slice. Due to the direct relation of the PSPT with the signal transmission power, we consider a reservation of 75% of the BS power for the URLLC slice and a minimum reservation of 25% of the power for the eMBB slice. Similarly, 75% of the SBs were reserved for the eMBB slice to achieve a high throughput value and a minimum reservation of 25% of the SBs for the URLLC slice to provide the delay requirements. These resource reservation percentage values were obtained after performing several simulations in an exhaustive way.

In general, in the URLLC slice more BS power is consumed due to the need to maximize the PSPT. The presence of PSPT in the reward function of the URLLC slice makes the RL agents that act in the URLLC slice choose the best actions that provide a PSPT higher than 90% for the UEs, generating an almost constant PSPT curve when analyzed against the UEs arrival and departure rate, as shown in Fig. 6. It can be observed that the ES and DA algorithms do not effectuate an efficient power control between slices. Fig. 6 shows that they tend to decrease the PSPT values as the UEs rate increases. It can also be noticed that the agents acting in the eMBB slice do not provide good PSPT values due to the low power allocated to the UEs.

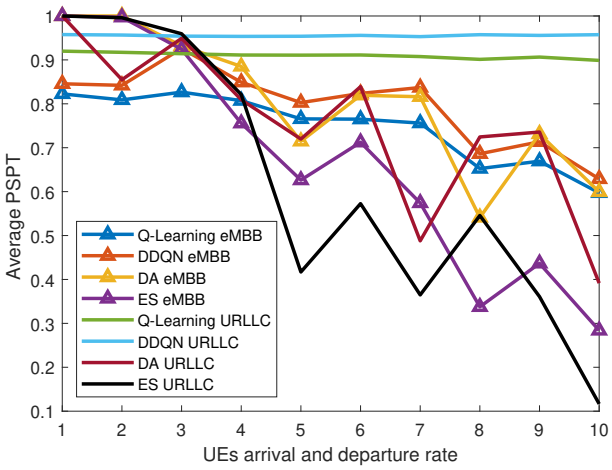

Fig. 6. Average PSPT in each slice

Since the agents in the URLLC slice allocate power to devices in order to attain better spectral efficiency using AMC scheme, consequently, high throughput of the SBs is provided for latency reduction. Fig. 7 shows that the NSRL algorithm with the DDQN agent in the URLLC slice presents the lowest average packet transmission delay among the considered algorithms even with the increase of the number of UEs.

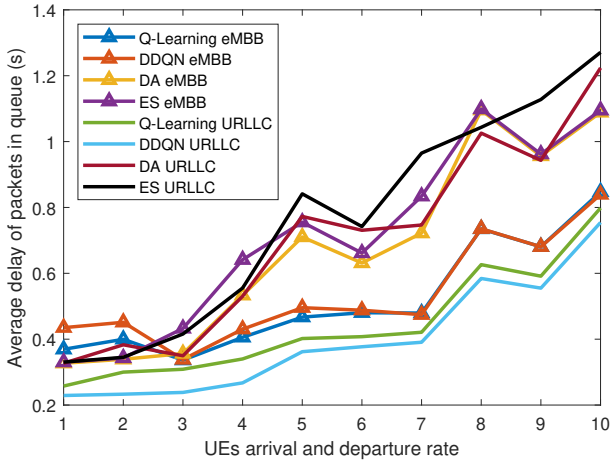

Fig. 7. Average packet transmission delay in each slice

With the increasing rate of UEs, the arrival of packets beyond the buffer capacity increases, generating packet loss. From Fig. 8, we observe a higher packet loss in the eMBB slice compared to the URLLC slice, as it does not maintain the buffer empty to perform the reduction in packet loss. The NSRL algorithm with the DDQN agent in the URLLC slice yields the lowest average loss rate in terms of the number of UEs in the network among the considered algorithms.

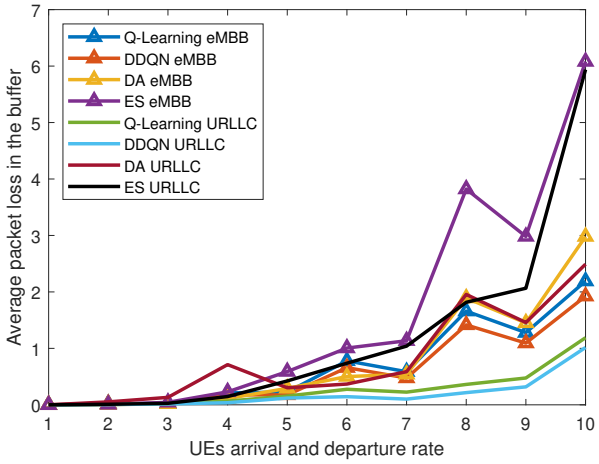

Fig. 8. Average packet loss in the buffer

The NSRL algorithm uses the RL agents to learn to allocate power and SBs efficiently to enhance the desired QoS parameters in each slice. Fig. 9 confirms that the NSRL algorithm using the DDQN agent in the eMBB slice presents a higher throughput in relation to the other algorithms. In this case, the adequate performance in the eMBB slice compared to the

URLLC slice is due to a higher allocation of SBs to enhance throughput for this slice.

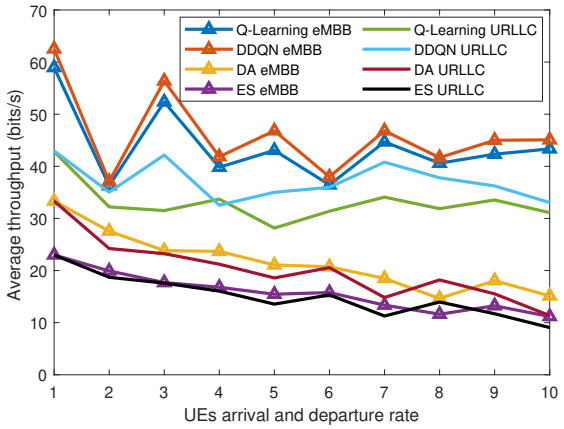

Fig. 9. Average throughput in packet transmission in each slice

Once energy efficiency (EE) has emerged as a key performance indicator for future 5G networks, a shift towards communications optimized not only in terms of throughput but in terms of EE has been receiving great attention. EE can be improved using different strategies such as network planning and development, energy harvesting and radio resource allocation [29]. The EE metric considered in this work is the ratio of the average throughput rate to the average total transmission power consumed. As shown in Fig. 10, the NSRL algorithm with the DDQN agent shows the best result in terms of energy efficiency.

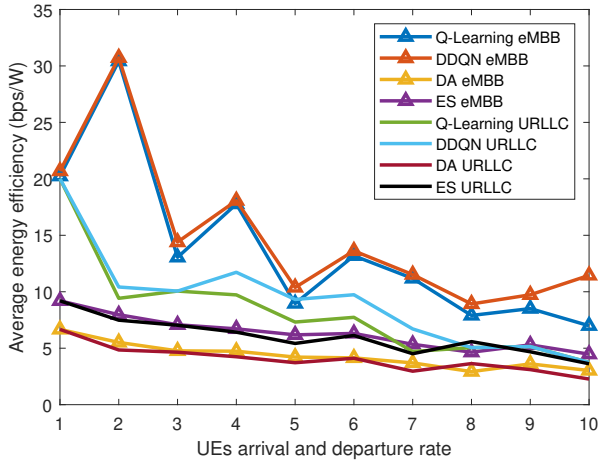

Fig. 10. Average energy efficiency in packet transmission in each slice

Analyzing the processing time as shown in Fig. 11, by increasing the number of UEs in the network, the processing time of the DA algorithm is greatly increased due to the number of iterations to find the best estimate for the Lagrange multiplier. This characteristic of the DA algorithm can make it unfeasible to be applied in a real scenario, whereas the proposed DDQN based NSRL algorithm manages to have a much lower processing time than the DA algorithm.

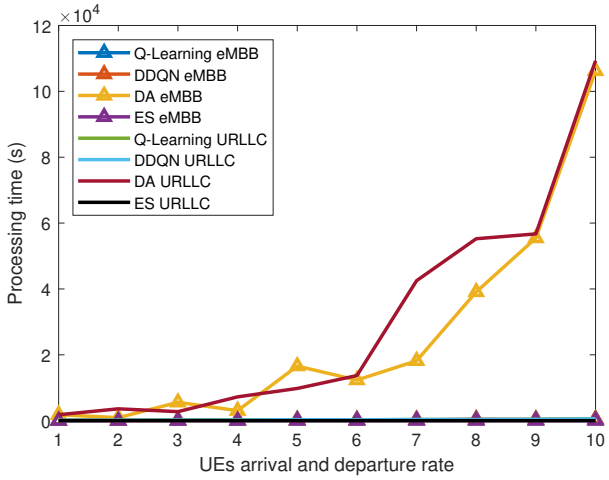

Fig. 11. Processing time of the resource allocation algorithms

DDQN and Q-Learning present different results for resource allocation considering URLLC and eMBB slices. This behavior occurs due to their different reward functions. It can be observed that the NSRL algorithm (with DDQN and Q-Learning agents) makes the system provide better specific QoS performance parameter according to the reward function related to each slice. That is, a higher throughput is obtained for the eMBB slice than for the URLLC slice, and a lower delay and higher PSPT for the URLLC slice than for the eMBB slice when NSRL is applied to the network.

VIII. Conclusion

In this work, it is assumed that the relationship between the PSPT and the signal transmission power are represented by sigmodal functions considering the CQIs standardized by 3GPP. Since, the PSPT calculation depends on a numerical integral which makes the optimization problem proposed in [14] intractable, it is proposed to perform a mathematical modeling using a robust implementation of the Levenberg Marquardt nonlinear curve fitting method. The sigmoidal function approximation presented a considerably low MSE of the order of 10^{-4}.

Next, we presented an effective solution for resource allocation in slices of wireless networks with 5G characteristics. To this end, a NS scenario with two slices URLLC and eMBB is proposed to be analyzed considering packets of a real 5G network collected from a large Irish mobile operator [22]. The resource allocation is modeled as a stochastic optimization problem, with state transitions without assuming prior knowledge of the system statistics. Besides, reward functions are proposed to each slice in the network.

For the joint allocation of power and SBs, the NSRL algorithm was proposed, a hybrid heuristic that combines the reinforcement learning techniques with the approach based on reservation and sharing of resources among the slices where each RL agent acts in one slice. The performances of the NSRL algorithm with the agent Q-Learning and DDQN (proposed approach) were compared with two other algorithms present in the literature, i.e, DA presented in [14] and ES

in [12]. The DA algorithm has a very high processing time, which makes it impractical in a real-time system and the ES algorithm presents a not desirable behavior of decreasing the PSPT values for the slices as the number of UEs in the network increases. The NSRL algorithm with the DDQN agent provided the best results in terms of the QoS parameters of the URLLC and eMBB services, confirming that DDQN based on a deep neural network performs better than the Q-Learning algorithm that is based on tabulated value search.

Deep Reinforcement Learning methods are considered promising techniques for resource allocation in cognitive wireless networks. Due to its great learning potential, it obtains interesting results monitoring the environment, without the a priori knowledge of the system statistics. The simulation results presented in this paper confirm the effectiveness of the proposed DDQN based NSRL algorithm in learning to allocate resources to UEs considering different slices. Finally, in future works, we intend to adapt the NSRL algorithm in order to consider more slices to the network.

REFERENCES

[1] Song, F., Li, J., Ma, C., Zhang, Y., Shi, L., Jayakody, D. N. K., et al. (2020). Dynamic Virtual Resource Allocation for 5G and Beyond Network Slicing, *IEEE Open Journal of Vehicular Technology*, 1, 215-226.

[2] Feng, L., Zi, Y., Li, W., Zhou, F., Yu, P., Kadoch, M., et al. (2020). Dynamic Resource Allocation With RAN Slicing and Scheduling for uRLLC and eMBB Hybrid Services, *IEEE Access*, 8, 34538-34551.

[3] J. Koo, V. B. Mendiratta, M. R. Rahman and A. Walid, Deep Reinforcement Learning for Network Slicing with Heterogeneous Resource Requirements and Time Varying Traffic Dynamics, *2019 15th International Conference on Network and Service Management (CNSM)*, 2019, pp. 1-5, doi: 10.23919/CNSM46954.2019.9012702.

[4] Q. Liu, T. Han, N. Zhang and Y. Wang, DeepSlicing: Deep Reinforcement Learning Assisted Resource Allocation for Network Slicing,*GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1-6, doi: 10.1109/GLOBECOM42002.2020.9322106.

[5] H. Zhou, M. Elsayed and M. Erol-Kantarci, RAN Resource Slicing in 5G Using Multi-Agent Correlated Q-Learning,*2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications*

[6] Y. Liu, J. Ding and X. Liu, A Constrained Reinforcement Learning Based Approach for Network Slicing, *IEEE 28th International Conference on Network Protocols (ICNP)*, 2020, pp. 1-6, doi: 10.1109/ICNP49622.2020.925937

[7] 3rd Generation Partnership Project (3GPP). (2015). Technical Specification (TS) 36.213 Evolved Universal Terrestrial Radio Access (E-UTRA. *Physical layer procedures*, Version 12.7.0.

[8] Fernandes Mauricio, W. V., Marques Lima, F. R., Abrão T., Ferreira Maciel, et al (2019). Resource Allocation for Energy Efficiency and QoS Provisioning. *Journal of Communication and Information Systems*, 34(1), 224-238. https://doi.org/10.14209/jcis.2019.24

[9] Saraiva, J., Braga Junior, I., Monteiro, V., Lima, F. R., Maciel, T., Freitas Junior, W., and Cavalcanti, F. R. (2020). Deep Reinforcement Learning for QoS-Constrained Resource Allocation in Multiservice Networks. *Journal of Communication and Information Systems*, 35(1), 66-76. https://doi.org/10.14209/jcis.2020.7

[10] Bhardwaj, Sanjay; Ginanjar, Rizki Rivai; Kim, Dong-Seong: Deep Q-learning based resource allocation in industrial wireless networks for URLLC,*IET Communications*, 2020, 14, (6), p. 1022-1027, DOI: 10.1049/iet-com.2019.1211

[11] Y. Liu, J. Ding and X. Liu, Resource Allocation Method for Network Slicing Using Constrained Reinforcement Learning, *2021 IFIP Networking Conference (IFIP Networking)*, 2021, pp. 1-3, doi: 10.23919/IFIPNetworking52078.2021.9472202.ications (PIMRC).

[12] Korrai, P. K., Lagunas, E., Sharma, S. K., Chatzinotas, S., Ottersten, B., et al. (2019). Slicing Based Resource Allocation for Multiplexing of eMBB and URLLC Services in 5G Wireless Networks, *2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 1-5.

[13] 3rd Generation Partnership Project (3GPP). (2020). Technical Specification (TS) 21.916 Evolved Universal Terrestrial Radio Access (E-UTRA. *Release 16 Description*, Version 1.0.0.

[14] Abdel-Hadi, A., Khawar, A., & Clancy, C. T. (2015). Optimal downlink power allocation in cellular networks, *Physical Communication*, 17, 1-14.

[15] Li, X., Fang, Q., and Shi, L. (2011). A effective SINR link to system mapping method for CQI feedback in TD-LTE system, *2011 IEEE 2nd International Conference on Computing, Control and Industrial Engineering*, 2, 208-211.

[16] Lee, J., Mazumdar, R. R., & Shroff, N. B. (2005). Downlink power allocation for multi-class wireless systems, *IEEE/ACM Transactions on Networking*, 13(4), 854-867.

[17] Lee, J., & Kwon, J. (2009). Utility-Based Power Allocation for Multiclass Wireless Systems, *IEEE Transactions on Vehicular Technology*, 58(7), 3813-3819.

[18] T. J. Wong and N. Das. (2014). Modelling and analysis of IEC 61850 for end-to-end delay characteristics with various packet sizes in modern power substation systems, *5th Brunei International Conference on Engineering and Technology (BICET 2014)*, pp. 1-6, doi: 10.1049/cp.2014.1073.

[19] Wang, Y., Abdel-Hadi, A., & Clancy, C. T. (2016). Optimal power allocation for LTE users with different modulations, *2016 Annual IEEE Systems Conference (SysCon)*, 1-5.

[20] Moré, J. J. (1977). The Levenberg-Marquardt algorithm: Implementation and theory, *Numerical Analysis*, Springer Berlin Heidelberg, Berlin, Heidelberg, 105-116.

[21] Nielsen, H. B. (1999). Damping parameter in Marquardt's method.

[22] Darijo Raca, Dylan Leahy, Cormac J. Sreenan, and Jason J. Quinlan. (2020). Beyond throughput, the next generation: a 5G dataset with channel and context metrics. *In Proceedings of the 11th ACM Multimedia Systems Conference (MMSys '20)*. Association for Computing Machinery, New York, NY, USA, 303–308. URL:http://cs1dev.ucc.ie/misl/5Gframework/5G-production-dataset.zip. DOI:https://doi.org/10.1145/3339825.3394938.

[23] Kleinrock L. Queueing Systems. Volume 1: Theory. 1st ed. New York: WileyInterscience; 1975.

[24] Alwarafy A, Abdallah M, Ciftler BS, Al-Fuqaha A, Hamdi M. (2021). Deep Reinforcement Learning for Radio Resource Allocation and Management in Next Generation Heterogeneous Wireless Networks: A Survey. *Institute of Electrical and Electronics Engineers (IEEE)*; May 28; Available from: http://dx.doi.org/10.36227/techrxiv.14672643.v1

[25] B. Gu, X. Zhang, Z. Lin and M. Alazab. (2021). Deep Multiagent Reinforcement-Learning-Based Resource Allocation for Internet of Controllable Things, *IEEE Internet of Things Journal*, vol. 8, no. 5, pp. 3066-3074, 1 March1, 2021, doi: 10.1109/JIOT.2020.3023111.

[26] Alfoudi, A. S. D., Newaz, S. H. S., Otebolaku, A., Lee, G. M., Pereira, R., et al. (2019). An Efficient Resource Management Mechanism for Network Slicing in a LTE Network, *IEEE Access*, 7, 89441-89457.

[27] Banchs, A., de Veciana, G., Sciancalepore, V. and Costa-Perez, X., et al. (2020) Resource Allocation for Network Slicing in Mobile Networks, *in IEEE Access* , 8, 214696-214706.

[28] Ameigeiras, P., Wang, Y., Navarro-Ortiz, J. et al. Traffic models impact on OFDMA scheduling design. J Wireless Com Network 2012, 61 (2012). https://doi.org/10.1186/1687-1499-2012-61

[29] Buzzi, S. I, C. Klein, T. E. Poor, H. V. et al. (2016), A Survey of Energy-Efficient Techniques for 5G Networks and Challenges Ahead, *in IEEE Journal on Selected Areas in Communications*, vol. 34, no. 4, pp. 697-709, April 2016, doi: 10.1109/JSAC.2016.2550338.

Hudson Henrique de Souza Lopes obtained his B.Sc degree in computer engineering and M.Sc. degree in Electrical and Computer Engineering from the Federal University of Goiás (UFG), in 2016 and 2021, respectively. He is currently a doctorate degree student in Electrical and Computer Engineering and Information Technology Technician in UFG. His research interests include artificial intelligence and resource allocation algorithms in wireless networks for QoS guarantees in network slicing scenarios with multiple services, slices, resources and users.

Flávio Geraldo Coelho Rocha received his B.Sc. degree in Electrical Engineering, the M.Sc. degree in Electrical and Computer Engineering and the doctorate degree in Computer Science from Federal University of Goiás (UFG) in 2009, 2011 and 2016, respectively. He is currently Professor of the Electrical, Mechanical and Computer (EMC) School of Engineering of UFG. He acts in the following research areas: Operational Research and Numerical Analysis Applied to Engineering Problems, Network Traffic Modeling and Control, Computers Network Management and Wireless Networks.

Flávio Henrique Teles Vieira received his B.Sc. degree in Electrical Engineering from the Federal University of Goiás (UFG) in 2000 and the M.Sc. degree in Electrical and Computer Engineering from UFG in 2002 and the doctorate degree in Electrical Engineering at State University of Campinas (FEEC-UNICAMP) in 2006. He is currently Professor of the Electrical, Mechanical and Computer (EMC) School of Engineering of Universidade Federal de Goiás (UFG). He acts in the following research areas: Communication Systems, Computational Intelligence and Artificial Intelligence Applied to Power and Communication Systems.