EAI Endorsed Transactions
on Wireless Spectrum

Research Article EAI.EU

Analysis of RAN Slicing for Cellular V2X and Mobile Broadband Services Based on Reinforcement Learning

Haider Daami R. Albonda* and J. Pérez-Romero

Universitat Politècnica de Catalunya (UPC), Barcelona, Spain

Abstract

Radio Access Network (RAN) slicing is one of the key enablers to provide the design flexibility and enable 5G system to support heterogeneous services over a common platform (i.e., by creating a customized slice for each service). In this regard, this paper provides an analysis of a Reinforcement Learning (RL)-based RAN slicing strategy for a heterogeneous network with two generic services of 5G, namely enhanced mobile broadband (eMBB) and vehicle-to-everything (V2X). In particular, this paper investigates the RAN slicing by evaluating the proposed scheme under different algorithm configurations (i.e., number of actions of RL) and parameters in order to analyze the performance in terms of metrics such as RL convergence time and to demonstrate the capability of the algorithm to perform an efficient allocation of resources among slices. In addition, this study compares the results obtained by the proposed solution to those obtained with a Proportional Scheme.

Keywords: Vehicle-to-everything (V2X), reinforcement learning, network slicing, RAN slicing.

Received on 31 October 2019, accepted on 12 March 2020, published on 25 March 2020

doi: 10.4108/eai.13-7-2018.163841

*Corresponding author. Email:haider.albonda@tsc.upc.edu

1. Introduction

The new fifth generation (5G) of mobile networks will support a wide variety of services and applications over a shared network infrastructure [1]. 5G capabilities need to adapt to the wide and dynamic variations of 5G requirements for a particular situation, as the 5G capabilities required to provide services depend on each particular use case. In this respect, 5G is expected to provide a great variety of services ranging over three generic types: a) enhanced Mobile Broad Band (eMBB), which focuses on services that require high data rate requirements, b) massive machine-type communications (mMTC), which support a massive number of static or dynamic machine communications, which are only intermittently active, and c) ultra-reliable and low-latency communications (URLLC), which focus on applications requiring very low latency and high reliability, such as mission critical communications or autonomous driving and Vehicle-to-Everything (V2X) [2-4].

These services may have very different requirements in terms of 5G system functionality (e.g. priority, charging, policy control, security, and mobility) and key performance indicators (e.g. latency, mobility, availability, reliability and data rates). Besides, the stringent requirements in expected performance (e.g. peak rates above 10 Gb/s, latencies below 1 ms with 10^{-5} reliability, 500 km/h mobility target) cannot always be achieved with a common network setting (e.g. optimizing

the network for low latency with high reliability could come at the expenses of reduced spectral efficiency) [5].
Given these aspects, 5G networks should provide solutions that enable the creation of logically isolated network partitions across the shared physical network infrastructure, referred to as *network slices*, to be created on top of a common shared physical infrastructure. Each network slice can be used to serve a particular service category (e.g., applications with different functional requirements) through the use of specific control plane (CP) and/or user plane (UP) functions [6]. Through the network slicing, 5G networks will be able to integrate the abovementioned multiple services with various performance requirements into a single physical network infrastructure. The Third-Generation Partnership Project (3GPP) has identified network slicing as one of the key technologies to achieve the aforementioned objectives in future 5G networks [7].
Focusing on the Radio Access Network (RAN) part of a network slice, due to the shared nature of the radio channel, an important problem in RAN slicing is to make an efficient distribution of the scarce radio resources among the different RAN slices [8]. For this reason, an efficient RAN slicing strategy for 5G systems needs to face different challenges and bring some benefits, especially in terms of network capacity utilization, reducing the network traffic congestion, avoiding the outage of service due to the lack of resources, and ensuring a high Quality of Service (QoS), e.g., in terms of data latency and transmission rate. For this reason, network slicing has received increasing attention in the literature, where different algorithms have been proposed, e.g. for slice admission control, slice scaling, etc. Specifically, a two-layer scheduler for an efficient and low complexity RAN slicing approach is proposed in [9]. The proposed solution showed that different trade-offs between isolation and efficiency could be achieved by setting some parameters in the utility function. The authors in [10] proposed a low complexity heuristic algorithm and slicing for joint admission control in virtual wireless networks. In turn, the deployment of function decomposition and network slicing as a tool to improve the Evolved Packet Core (EPC) is presented in [11].
Meanwhile, the use of machine learning-based network control and management strategies has also gained attention in the context of mobile networks due to its promising performance. They have been used by different works for managing the split of the available radio resources among different slices based on reinforcement learning (RL) [12-17], Markov Decision Process (MDP) [18-19], game theory [20], and multi-armed bandit [21].

In particular, reference [12] proposed a novel radio resource slicing framework for 5G networks with haptic communications based on virtualization of radio resources. The authors adopted a reinforcement learning (RL) approach for dynamic radio resource slicing in a flexible way, while accounting for the utility requirements of different vertical applications. An RL strategy proposed in [13] for intelligently scaling up/down slices according to traffic patterns of mobile users. The authors have investigated some typical resource management solutions based on Deep RL (DRL), including radio resource slicing and priority-based core network slicing. A slice admission strategy based on RL is proposed in [14], where slices for different services are virtualized over the same RAN infrastructure. Reference [15] provided RL-based radio resource scheduling policy for 5G radio access network (RAN) to maximize the probability of meeting Quality-of-Service (QoS) requirements (i.e., throughput, PLR and delay). Inspired by the success of Enhanced DRL in solving complex control problems, [16] introduced a new DRL-based framework for allocating energy-efficient resources in cloud RANs. Specifically, the authors defined the state space, action space and reward function. They have also applied a deep neural network (DNN) to approximate the value of the work function, and formally formulated the resource allocation problem as a convex improvement problem. Reference [17] proposed an RL-based strategy for scheduling resources in a multi-tenant network with mobile and cloud service providers (SPs) based on a network policy designed to maximize the efficiency of infrastructure resources.

A model for orchestrating network slices based on the service requirements and available resources is introduced in [18]. The authors proposed a Markov decision process framework to formulate and determine the optimal policy that manages cross-slice admission control and resource allocation for the 5G networks. Similarly, an adaptive algorithm for virtual resource allocation based on constrained Markov decision process is proposed in [19]. A network slicing scheme based on game theory for managing the split of the available radio resources in a RAN among different slice types is proposed in [20] to maximize utility of radio resources. In [21], an Online network slicing solution based on multi-armed bandit mathematical model to maximizes network slicing multiplexing gains and achieving the accommodation of network slice requests in the system with an aggregated level of demands above the available capacity is proposed.

Although the above works have proposed different approaches for RAN slicing, none of them has dealt with scenarios including slices for supporting Vehicle-to-Vehicle (V2V) communications, which constitute the focus of this paper. In this respect, in our previous work [22], we proposed a novel strategy based on offline Q-learning and softmax decision-making to determine the adequate split of resources between the different slices while accounting for their utility requirements and the dynamic changes in the traffic load.
Starting from the outcome of this off-line Q-learning algorithm, it was complemented with a low-complexity heuristic approach in [23] for fine tuning the resource assignment and achieving further improvements in terms of network performance.

In this paper, we extend our previous works [22, 23] by investigating the RAN slicing strategy under different algorithm configurations (i.e., number of actions of RL) and different algorithm parameters in order to demonstrate its capability to perform an efficient allocation of resources among slices in terms of network metrics such as resource utilization, latency, network traffic load, achievable throughput, and to analyze the impact on algorithm-related metrics such as convergence time.

The rest of the paper is organized as follows. Section 2 presents the system model and the problem formulation. Section 3 provides the proposed solution for RAN slicing. Section 4 presents the performance evaluation followed by the conclusions in Section 5.

2. System Model and Problem Formulation

2.1 System Model

The considered scenario assumes a cellular Next Generation Radio Access Network (NG-RAN) with a gNodeB (gNB) [24] composed by a single cell. A roadside unit (RSU) supporting V2X communications is attached to the gNB. A set of eMBB cellular users (CUs) numbered as m=1,...,M are distributed randomly around the gNB and a flow of several independent vehicles move along a straight highway, as illustrated in the right part of Fig.1. The highway segment is divided into sub-segments (clusters) by sectioning the road into smaller zones according to the length of the road. It is assumed that each vehicle includes a User Equipment (UE) that enables communication with the UEs in the rest of vehicles in the same cluster. Clusters are numbered as j=1,...,C, and the vehicles in the j-th cluster are numbered as i=1,...,$V(j)$.

The vehicles in the highway are assumed to enter the cell coverage following a Poisson process with arrival rate λ_a. The association between clusters and vehicles is managed and maintained by the RSU based on different metrics (e.g. position, direction, speed and link quality) through a periodic exchange of status information.

Regarding the V2X services, this paper assumes V2V communication between vehicles. They can be performed either in cellular or in sidelink mode. In cellular mode each UE communicates with each other through the Uu interface in a two-hops transmission via the gNB while in sidelink mode, direct V2V communications can be established over the PC5 interface. We assume that, when sidelink transmissions are utilized, every member vehicle can multicast the V2V messages directly to multiple member vehicles of the same cluster $1 \leq i \leq V(j)$ using one-to-many technology. The decision on when to use cellular or sidelink mode is done based on [25].

To simultaneously support the eMBB and the V2X services, the network is logically divided into two network slices, namely RAN_slice_ID=1 for V2X and RAN_slice_ID=2 for eMBB. The whole cell bandwidth is organized in Resource Blocks (RBs) of bandwidth B. Let denote as N_{UL} the number of RBs in the UpLink (UL) and N_{DL} the number of RBs in the DownLink (DL). The RAN slicing process should distribute the UL and DL RBs among the two slices. For this purpose, let denote $\alpha_{s,UL}$ and $\alpha_{s,DL}$ as the fraction of UL and DL resources, respectively, for the RAN_slice_ID=s with s=1,2. Regarding sidelink communications, and since the support for sidelink has not been yet specified for 5G in current 3GPP release 15, this paper assumes the same approach as in current LTE-V2X system, in which the SL RBs are part of the total RBs of the UL. For this reason, the slice ratio $\alpha_{s,UL}$ is divided into two slice ratios, namely $\tilde{\alpha}_{s,UL}$, which corresponds to the fraction of UL RBs that are used for uplink transmissions, and, $\alpha_{s,SL}$, which corresponds to the fraction of UL RBs used to support sidelink transmissions.

Each vehicle is assumed to generate packets randomly with rate λ_v packets/s according to Poisson arrival model. The length of the messages is S_m. When the vehicles operate in sidelink mode, the messages are transmitted using the SL resources allocated to the slice. Instead, when the vehicles operate in cellular mode, the messages are transmitted using the UL and DL resources. The average number of required RBs from V2X users of RAN_slice_ID= 1 per Transmission Time Interval (TTI) in UL, DL and SL, denoted respectively as $\Gamma_{1,UL}$, $\Gamma_{1,DL}$, $\Gamma_{1,SL}$ can be estimated as follows:

$$\Gamma_{1,x} = \frac{\sum_{t=1}^{T}\sum_{j=1}^{C}\sum_{i=1}^{V(j)} m(j,i,t)\cdot S_m}{T\cdot SP_{eff,x}\cdot B\cdot F_d} \qquad (1)$$

where x denotes the type of link, i.e. $x \in$ {UL,DL,SL}, $m(j,i,t)$ is the number of transmitted messages by the vehicles of the j-th cluster in the t-th TTI and $Sp_{eff,x}$ is the spectral efficiency in the x link, F_d is the TTI duration, which is 0.1 ms and T is the number of TTIs that defines the time window used to compute the average.

Regarding the eMBB service, the average number of required RBs for eMBB users of RAN_slice_ID=2 in UL and DL in order to support a certain bit rate R_b is denoted as $\Gamma_{2,UL}$, $\Gamma_{2,DL}$, respectively, and can be statistically estimated as follows:

$$\Gamma_{2,x} = \frac{\sum_{t=1}^{T}\sum_{m=1}^{M} \rho_x(m,t)}{T} \qquad (2)$$

where x denotes the type of link, and $\rho_x(m,t)$ is the number of required RBs by the m-th user in the link x and in the t-

th TTI in order to get the required bit rate R_b. It is given by $\rho_x(m,t)=R_b/(SP_{eff,x}\cdot B)$. The values $\Gamma_{2,UL}$, $\Gamma_{2,DL}$ are computed within a time window T TTIs. Note also that $\Gamma_{2,SL}=0$, since the eMBB slice does not generate sidelink traffic.

2.2 Problem Formulation for RAN Slicing

The focus of this paper is to determine the optimum slicing ratios $\alpha_{S,UL}$, $\alpha_{S,DL}$ in order to maximize the overall resource utilization under the constraints of satisfying the resource requirements for the users of the two considered slices.

The total utilization of UL resources U_{UL} is given by the aggregate of the required RBs in the UL and SL for each slice, provided that the aggregate of a given slice s does not exceed the total amount of resources allocated by the RAN slicing to this slice, i.e. $\alpha_{s,UL}\cdot N_{UL}$. Otherwise, the utilization of slice s will be limited to $\alpha_{s,UL}\cdot N_{UL}$ and the slice will experience outage. Correspondingly, the optimization problem for the uplink is defined as the maximization of the UL resource utilization subject to ensuring an outage probability lower than a maximum tolerable limit p_{out}. This is formally expressed as:

$$\max_{\alpha_{s,UL}} U_{UL} = \max_{\alpha_{s,UL}} \sum_s \min\left(\Gamma_{s,SL} + \Gamma_{s,UL}, \alpha_{s,UL}\cdot N_{UL}\right) \quad (3)$$

$$\text{s.t.} \quad \Pr\left\lfloor \Gamma_{s,SL} + \Gamma_{s,UL} \geq \alpha_{s,UL}\cdot N_{UL} \right\rfloor < p_{out} \quad s=1,2 \quad (3a)$$

$$\sum_s \alpha_{s,UL} = 1 \quad (3b)$$

Following similar considerations, the optimization problem to maximize the resource utilization U_{DL} in the DL subject to ensuring a maximum outage probability is given by:

$$\max_{\alpha_{s,DL}} U_{DL} = \max_{\alpha_{s,DL}} \sum_s \min\left(\Gamma_{s,DL}, \alpha_{s,DL}\cdot N_{DL}\right) \quad (4)$$

$$\text{s.t.} \quad \Pr\left\lfloor \Gamma_{s,DL} \geq \alpha_{s,DL}\cdot N_{DL} \right\rfloor < p_{out} \quad s=1,2 \quad (4a)$$

$$\sum_s \alpha_{s,DL} = 1 \quad (4b)$$

3. Reinforcement Learning-based RAN Slicing Solution

The problems in (3) and (4) with their constraints are nonlinear optimization problems. Such an optimization problem is generally hard to solve. The complexity of solving this problem is high for a network of realistic size

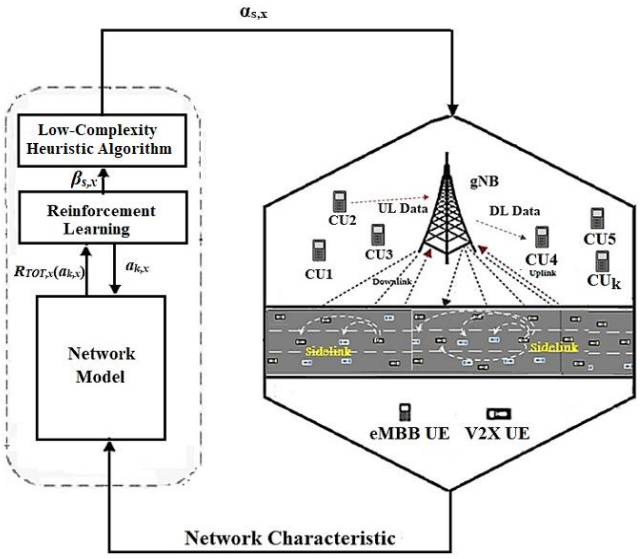

Figure 1. RAN Slicing Strategy.

with fast varying traffic conditions. For this reason, the use of an offline reinforcement learning approach to solve the problem in a more practical way is considered here, following previous works [22,23].

The general approach is depicted in Fig.1. Specifically, a slicing controller is responsible for determining a first estimation of the slicing ratios, denoted as $\beta_{s,UL}$, $\beta_{s,DL}$, for each slice by executing the RL algorithm. It is assumed that two separate RL algorithms are executed for the UL and the DL to determine respectively $\beta_{s,UL}$, $\beta_{s,DL}$. In the general operation of RL, the optimum solutions are found based on dynamically interacting with the environment based on trying different actions $a_{k,x}$ (i.e. different slicing ratios) selected from a set of possible actions numbered as $k=1,...,A_x$, where $x \in \{UL, DL\}$.

As a result of the selected action, the RL process gets a reward $R_{TOT,x}(a_{k,x})$ that measures how good or bad the result of the action has been in terms of the desired optimization target. Based on this reward, the RL algorithm adjusts the decision making process to progressively learn the actions that lead to highest reward. The action selection is done by balancing the trade-off between exploitation (i.e. try actions with high reward) and exploration (i.e. try actions that have not been used before in order to learn from them). In case this interaction with the environment was done in an on-line way, i.e. by configuring the slicing ratios on the real network and then measuring the obtained performance, this could lead to serious performance degradation since, during the exploration process, wrong or unevaluated decisions could be made at certain points of time due to the exploration, and affecting all the UEs of a given slice. To avoid this problem, this paper considers an off-line RL, in which the slicing controller interacts with a network model that simulates the behavior of the network and allows testing the performance of the different actions

in order to learn the optimum one prior to configuring it in the real network. The network model is based on a characterization of the network in terms of traffic generation, propagation modelling, etc.

The considered RL algorithm in this paper is based on the proposed scheme in [22], which enables an exploration-exploitation traversing all possible actions in long-term. In turn, the reward is defined in accordance with the optimization problem, which intends to maximize the resource utilization subject to the outage probability constraint. The details about the reward function and the detailed operation of the Q-learning algorithm are presented in the following.

Once the first estimation of the slicing ratios $\beta_{s,UL}, \beta_{s,DL}$ has been obtained, a low complexity heuristic scheme is applied in order to perform a fine tuning and get the final slicing ratios $\alpha_{s,UL}, \alpha_{s,DL}$.

3.1 Reward Computation

The reward function should reflect the ability of the taken action to fulfill the targets of the optimization problems (3) and (4). Based on this, and for a given action $a_{k,x}$ with associated slicing ratios $\alpha_{s,x}(k)$ the reward is computed as function of the normalized resource utilization $\Psi_{s,x}(a_{k,x})$ of slice s in link $x \in \{UL,DL\}$ defined as the ratio of used resources to the total allocated resources by the corresponding action. For the case of the V2X slice (s=1), it is defined as:

$$\Psi_{1,UL}\left(a_{k,\mathrm{UL}}\right) = \frac{\Gamma_{1,UL}+\Gamma_{1,SL}}{\alpha_{1,UL}(k)\cdot N_{UL}} \tag{5}$$

$$\Psi_{1,DL}\left(a_{k,\mathrm{DL}}\right) = \frac{\Gamma_{1,DL}}{\alpha_{1,DL}(k)\cdot N_{DL}} \tag{6}$$

In turn, for the case of eMBB slice (s=2), it is defined as:

$$\Psi_{2,UL}\left(a_{k,\mathrm{UL}}\right) = \frac{\Gamma_{2,UL}}{\alpha_{2,UL}(k)\cdot N_{UL}} \tag{7}$$

$$\Psi_{2,DL}\left(a_{k,\mathrm{DL}}\right) = \frac{\Gamma_{2,DL}}{\alpha_{2,DL}(k)\, N_{DL}} \tag{8}$$

Based on these expressions, the reward $R_{s,x}(a_{k,x})$ for the slice s in link $x \in \{UL,DL\}$ as a result of action $a_{k,x}$ is defined as

$$R_{s,x}\left(a_{k,x}\right) = \begin{cases} e^{\Psi_{s,x}\left(a_{k,x}\right)} & \Psi_{s,x}\left(a_{k,x}\right) \le 1 \\ 1/\Psi_{s,x}\left(a_{k,x}\right) & otherwise \end{cases} \tag{9}$$

In (9), whenever $\Psi_{s,x}(a_{k,x})$ is a value between 0 and 1, the reward function will increase exponentially to its peak at $\Psi_{s,x}(a_{k,x})$=1. Therefore, the actions that lead to higher value of $\Psi_{s,x}(a_{k,x})$ (i.e. higher utilization) provide larger rewards and therefore this allows approaching the optimization target of (3) and (4). In contrast, if the value of $\Psi_{s,x}(a_{k,x}) > 1$, it means that the slice s will be in outage and
thus the reward decreases to take into consideration constraints (3a)(4a). Consequently, the formulation of the reward function per slice in (9) takes into account the constraints of the optimization problem. In addition, since the total reward has to account for the effect of the action on all the considered slices s=1,..,S, it is defined in general as the geometric mean of the per-slice rewards, that is:

$$R_{TOT,x}\left(a_{k,x}\right) = \left(\prod_{s=1}^{S} R_{s,x}\left(a_{k,x}\right)\right)^{\frac{1}{s}} \tag{10}$$

3.2 Q-learning and low complexity heuristic algorithm

The ultimate target of the Q-learning scheme at the slicing controller is to find the optimal action (i.e. the optimal slicing ratios for a given link $x \in \{UL,DL\}$) that maximizes the expected long-term reward to each slice. To achieve this, the Q-learning interacts with the network model over discrete time-steps of fixed duration and estimates the reward of the chosen action. Based on the reward, the slice controller keeps a record of its experience when taking an action $a_{k,x}$ and stores the action-value function (also referred to as the Q-value) in $Q_x(a_{k,x})$. Every time step, the $Q_{UL}(a_{k,UL})$ and $Q_{DL}(a_{k,DL})$ values are updated following a single-state Q-learning approach with a null discount rate [26] as follows:

$$Q_x(a_{k,x}) \leftarrow (1-\alpha)\, Q_x(a_{k,x}) + \alpha \,.\, R_{TOT,x}(a_{k,x}) \tag{11}$$

where $\alpha \in (0, 1)$ is the learning rate, and $R_{TOT,x}(a_{k,x})$ is the total reward accounting for both V2X and eMBB slices after executing an action $a_{k,x}$. At initialization, i.e. when action $a_{k,x}$ has never been used in the past, $Q_x(a_{k,x})$ is initialized to an arbitrary value.

The selection of the different actions based on the $Q_x(a_{k,x})$ is made based on the softmax policy [26], in which the different actions are chosen probabilistically. Specifically, the probability $P_x(a_{k,x})$ of selecting action $a_{k,x}$, k=1,...,A_x, is defined as

$$P_x\left(a_{k,x}\right) = \frac{e^{Q_x\left(a_{k,x}\right)/\tau}}{\sum_{j=1}^{A_x} e^{Q_x\left(a_{j,x}\right)/\tau}} \tag{12}$$

where τ is a positive integer called temperature parameter that controls the selection probability. With high value of τ, the action probabilities become nearly equal. However, low value of τ causes a greater difference in selection probabilities for actions with different Q-values. Softmax decision making allows an efficient trade-off between exploration and exploitation, i.e. selecting with high probability those actions that have yield high reward, but also keeping a certain probability of exploring new actions, which can yield better decisions in the future. The pseudo-code of the proposed RL-based RAN slicing algorithm is summarized in Algorithm 1.

Once the offline RL algorithm has converged, i.e. the selection probability of one of the actions is higher than 99.99%, a heuristic algorithm based on [23] will take the results of the RL algorithm as input and adjust the initial slicing ratios $\beta_{s,UL}$ and $\beta_{s,DL}$ chosen by the RL in order to determine the final optimized values $\alpha_{S,UL}$ and $\alpha_{S,DL}$, as illustrated in Fig.1.

The idea of this fine tuning is that, based on the actual RB demands of each slice and the slicing ratios $\beta_{s,UL}$, $\beta_{s,DL}$ the algorithm assesses if one of the two slices s has more resources than actually required in the link $x \in \{UL,DL\}$, i.e. $\Psi_{s,x}(a_{x\_sel}) < 1$, and at the same time the other slice s' has less resources than required, i.e. $\Psi_{s',x}(a_{x\_sel}) > 1$. If this is the case, the slice s leaves some extra capacity $\Delta C_{s,x}$ that can be transferred to the other slice s'. Specifically, the extra capacity is defined as:

Algorithm 1: RAN slicing algorithm based on RL

1. **Inputs:** N_{UL}, N_{DL}: Number of RBs in UL and DL. S: number of slices, Set of actions $a_{k,x}$ for link $x \in \{UL,DL\}$
2. **Initialization of Learning**: $t \leftarrow 0$, $Q_x(a_{k,x}) = 0$, $k=1,...,A_x$, $x \in \{UL,DL\}$
3. **Iteration**
4. **While** learning period is active do
5. **for** each link $x \in \{UL,DL\}$
6. Apply softmax and compute $P_x(a_{k,x})$ for each action $a_{k,x}$ according to (12);
7. Generate a uniformly distributed random number $u \in \{0,1\}$
8. Select an action $a_{k,x}$ based on u and probabilities $P_x(a_{k,x})$
9. Apply the selected action to the network and evaluate $\Psi_{s,x}(a_{k,x})$ based on (5)-(8).
10. **If** $\Psi_{s,x}(a_{k,x}) \leq 1$ then
11. $R_{s,x}(a_{k,x}) = e^{\Psi_{s,x}(a_{k,x})}$
12. **else**
13. $R_{s,x}(a_{k,x}) = 1/\Psi_{s,x}(a_{k,x})$
14. **End**
15. **Compute** $R_{TOT,x}(a_{k,x})$ based on equation (10)
16. **Update** $Q_x(a_{k,x})$ based on equation (11)
17. **End**
18. **End**

$$\Delta C_{s,x} = \left(1 - \Psi_{s,x}\left(a_{x\_sel}\right)\right). \omega \qquad (18)$$

where the configuration parameter ω is a scalar in the range [0,1] used to leave some margin capacity to cope with the variations of the RBs consumption.

4. Performance Evaluation

In this section, we evaluate the performance of the RAN slicing strategy through system level simulations performed in MATLAB.

4.1 Simulation Setup

Our simulation model is based on a single-cell hexagonal layout configured with a gNB. The model considers vehicular UEs communicating through cellular mode (uplink / downlink) and via sidelink (direct V2V) and use slice (RAN_slice_ID=1) and eMBB UEs operating in cellular mode (uplink / downlink) and using slice (RAN_slice_ID=2) based on the assumptions described in section 2. Note that the slice ratio $\alpha_{1,UL} \cdot N_{UL}$ is divided into two ratios ($\bar{\alpha}_{1,UL}$= 65 % of $\alpha_{1,UL} \cdot N_{UL}$ RBs for V2X users in sidelink and $\alpha_{1,SL}$ =35 % of $\alpha_{1,UL} \cdot N_{UL}$ RBs for V2X service in uplink direction).

The traffic generation associated to each eMBB UE at a random position assumes that services generate sessions following a Poisson process with rate λ_e, required bit rate R_b=1 Mb/s and average session duration of 120 s. The gNB supports a cell with a channel organized in 200 RBs composed by 12 subcarriers with subcarrier separation Δf=30 kHz, which corresponds to one of the 5G NR numerologies defined in [27].

The actions specify the fraction of resources for V2X and eMBB slices and they are defined such that action $\beta_{k,x}$ corresponds to $\beta_{1,x}(k) = k/N$ and $\beta_{2,x}(k) = (1-k/N)$ for $k=1,...,N$, and $x \in \{UL, DL\}$, where N is the number of actions. The simulation time is measured in units referred to as "time steps" that determine when the different simulation events occur. In the considered simulation, there is a set of possible actions numbered as $k=1,...,A_x$. For each action taken from this set of actions, the proposed RL dynamically interacts with a network model that simulates the behavior of the network and estimates the reward of the chosen action according to equation (14). Based on the reward, the RL algorithm keeps a record of its experience when taking an action $a_{k,x}$ and stores the Q-value in $Q_x(a_{k,x})$. Every time step, the $Q_{UL}(a_{k,UL})$ and $Q_{DL}(a_{k,DL})$ values are updated based on equation (15). Then, after multiple times of learning, RL selects the most appropriate action (i.e., the selection probability of one of the actions is higher than 99.99%). Once the RL algorithm has converged, the slicing ratios $\beta_{s,x}$ associated to this action are passed to the low-complexity heuristic algorithm which in turn fine tunes

the initial slicing ratios $\beta_{s,UL}$, $\beta_{s,DL}$ chosen by the RL based on the resource requirements for each slice.
Different simulations will be executed for different values of N in order to assess the impact of the number of actions. All relevant simulation parameters are summarized in Table 1.

Table 1. Simulation parameters

Parameter	Values
General parameters	
Cell radius	500m
Number of RBs per cell	N_{UL}=N_{DL}=200 RBs
Frequency	2.6 GHz
Path loss model	The path loss and the LOS probability for cellular mode are modeled as in [28]. In sidelink mode, all V2V links are modeled based on freeway case (WINNER+B1) with hexagonal layout [ITU-R] [29].
Spectral efficiency model to map SINR.	Model in section A.1 of [30]. The maximum spectral efficiency is 8.8 b/s/Hz.
Shadowing standard deviation	3 dB in LOS and 4 dB in NLOS.
height of the gNB	10m
Base station antenna gain	5 dB
TTI duration (F_d)	1ms
Time window T	3s
V2X parameters	
Length of the highway	1Km
Number of lanes	3 in one direction (one is considered in the freeway)
Lane width	4 m
Number of clusters	4
Size of cluster	250m
Vehicular UE height	1.5m
vehicle speed	80 Km/h
Vehicle arrival rate λ_a	1 UE/s
Packet arrival rate λ_v	1 packets/s
Message size (S_m)	300 bytes
eMBB parameters	
UE arrival rate λ_m	1 UE/s
UE height	1.5m
Average session generation rate λ_e	Varied from 0.2 to 1.2 sessions/s
R_b	1 Mb/s
Average session duration	120 s
RAN slicing algorithm parameters	
Learning rate α	0.1
ω	{0.25, 0.55, 0.85}
Temperature parameter τ	0.1
Actions of the RL algorithm	N= {10,15,20,25}

The presented evaluation results intend to assess and illustrate the performance of the proposed solutions in terms of network capacity, throughput, and outage probability when considering different configurations of the algorithm in relation to the number of actions N of the complexity heuristic approach.
In addition, and as a reference for comparison, we assume a RAN slicing strategy denoted as "Proportional Scheme", in which the ratio of RBs for each slice is proportional to its total traffic rate (in Mb/s). Similarly, comparison will also be presented against the case in which the algorithm includes only the Q-learning but not the heuristic approach.

4.2 Impact of the number of Actions on the performance

Fig. 2 presents the aggregate RB utilization (i.e. the number of used RBs normalized to the number of total available RBs) for both V2X and eMBB slices in the uplink (including both sidelink and uplink traffic), as a function of the number of actions N. It is worth mentioning that, although the aggregate of slicing ratios for V2X and eMBB slices will be 100%, this does not mean that the aggregate of resource utilization should be necessarily 100%, because the utilization measures the actual RBs that are occupied in accordance with the existing traffic. Therefore, it is possible that, at a certain point of time, one slice does not consume all the allocated RBs. The figure illustrates the behavior of the proposed solution with different values of ω and of the reference scheme. From the presented results, we notice that as the number of actions increases, the proposed solution with all the assumed values of ω maintains high resource utilization compared with the reference scheme. The reason for this is that, as the number of actions increases, there will be a greater chance of obtaining actions that lead to a higher value of $\Psi_{s,x}(a_{k,x})$ (i.e., higher utilization) and provide larger rewards. Therefore, this allows better approaching the optimization target.

Fig. 3 presents the time for convergence, as a function of the number of actions. It is measured as the number of simulated time steps of 0.1s in the execution of the off-line RL until reaching convergence. We can clearly observe from the Fig. 3 that as the number of actions increases, the convergence time grows gradually (i.e., in the analyzed results this effect is particularly observed when the number of Actions increases beyond 15) because when the number of actions increases, the system needs to explore more actions (i.e. try more actions that have not been used before in order to learn from them) before finding the most appropriate one. Thus, this leads to a noticeable increase in convergence time.
Looking at Fig. 2 and Fig. 3, a trade-off is found between resource utilization and convergence time. In particular, when increasing the number of actions, the proposed algorithm improves the resource utilization but with a

longer convergence time. For example, when the number of actions is 20, the RAN slicing strategy with offline RL followed by the low-complexity heuristic algorithm with ω = 0.85 reaches a utilization of around 0.92 of the resources and the time needed to converge is about 18000 time steps. Then, when increasing to 25 actions the utilization is improved up to 0.95, which corresponds to a relative gain of 3%. However, the convergence time increases up to 23000 time steps, representing an increase of 27%. Therefore, the slight improvement in utilization when increasing from 20 to 25 actions does not compensate for the degradation in convergence time.

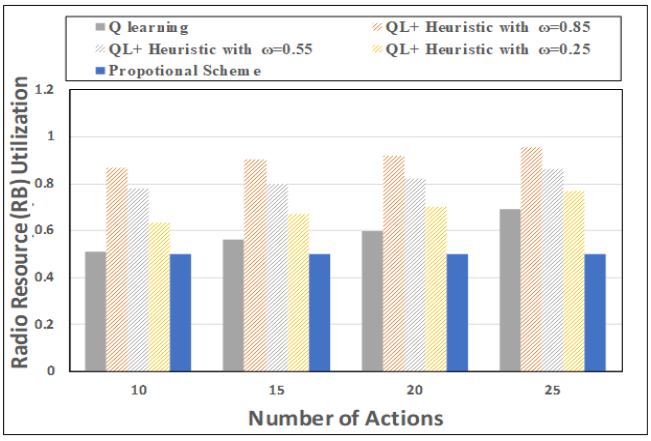

Figure 2. Uplink RB utilization as a function of the Number of Actions.

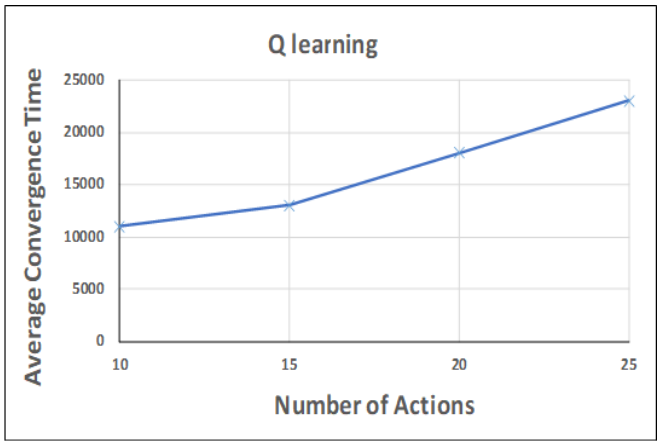

Figure 3. Convergence time as a function of Number of Actions.

4.3 Network Performance Metrics

In this subsection, the performance of the RAN slicing strategy is compared with the reference scheme in terms of the obtained RB utilization, throughput, outage probability, and latency.

Fig.4 plots the obtained RB utilization for UL, as a function of the eMBB session arrival rate (λ_e) when the number of actions is 20. Since SL and UL make use of the same set of RBs, the results included in Fig.4 refer to the total utilization by both links for V2X and eMBB slices.

From the presented results, we notice that the slicing strategy with both off-line RL and off-line RL followed by the low-complexity heuristic approach with all the assumed values of ω maintains higher resource utilization compared to the reference scheme for all the considered loads. This is due to the RL-based slicing solution that inherently tackles slice dynamics by selecting the most appropriate action. Further improvements are obtained by the offline RL followed by a low-complexity heuristic approach by checking the unused capacity left by each slice after selecting an action and use it to serve more traffic load in the other slice.

Besides, we can see from fig. 4 that, when increasing the value of ω, the system provides more resources and therefore leads to better utilization, as it is observed when comparing the results for ω equal to 0.85 against the results for other values of ω.

Regarding the quantitative comparison between strategies, the figure reflects that, for the RAN slicing strategy with offline RL followed by the low-complexity heuristic algorithm with ω = 0.85, the system utilizes around 94 % of radio resources in uplink when the eMBB session arrival rate is 0.8 sessions/s. In contrast, in case of the proposed scheme with only offline RL algorithm, the system utilizes around 60 % of radio resources in uplink. Finally, for the reference proportional approach, the utilization is only about 51 % in uplink (i.e. offline RL followed by the low-complexity heuristic algorithm with ω = 0.85 achieves a relative gain of 84%).

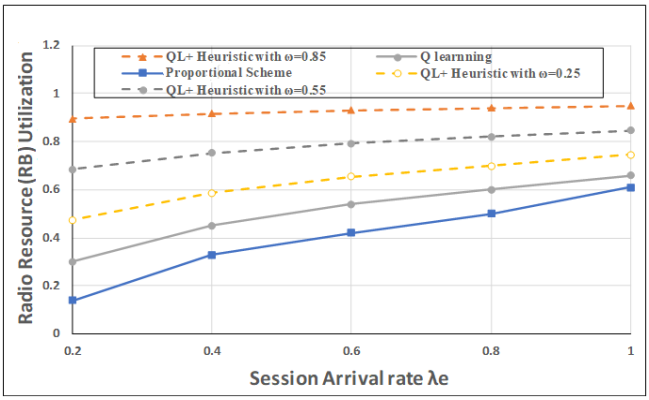

Figure 4. Uplink RB utilization as a function of the eMBB session generation rate λ_e (sessions/s).

Fig.5 presents the aggregate throughput delivered in Mbits/sec for both eMBB and V2X slices in the sidelink and uplink. The figure illustrates the behavior of the RAN slicing strategy and the proportional scheme. We can

observe that the off-line RL and off-line RL followed by the low-complexity heuristic approach outperform the reference scheme. Specifically, the RAN slicing strategy with offline RL followed by the low-complexity heuristic algorithm with ω = 0.85 achieves a throughput of 123 Mb/s when the eMBB session arrival rate is 0.8 sessions/s. In turn, the RAN slicing strategy with only off-line RL achieves a throughput of 90 Mb, and the reference proportional approach a throughput of only 81 Mb/s (i.e. offline RL followed by the low-complexity heuristic approach with ω = 0.85 achieves a relative gain of 51% with respect to the reference). The reason for this behavior is that, as the number of eMBB sessions increases, requiring more radio resources, the proposed off-line RL followed by the heuristic algorithm ensures more RBs and achieves higher radio resource utilization than the reference schemes. Therefore, these RBs can be used to transmit more data.

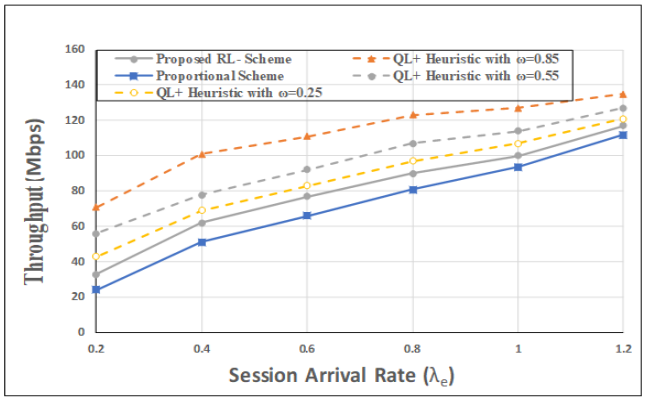

Figure 5. Aggregated throughput experienced by both slices in uplink as a function of the eMBB session generation rate λe (sessions/s).

In Fig.6, we investigate the probability of having outage (i.e. the probability that there are no sufficient RBs to serve all the transmission requests) at a certain point of time. As shown in the figure, increasing the traffic load leads to an increase in the outage probability of the services.

We notice from Fig.6 that, regardless of the considered scheme, the system does not experience outage when the eMBB session arrival rate λ_e is less than 1.2 sessions/s. This is due to the fact that, for this low load, the system has sufficient amount of RBs to serve the traffic. Then, when the load increases (i.e. session arrival rate increases) the system starts to face situations in which some RB limitations may occur. For this reason, it is for these loads when a more efficient slicing strategy is needed to properly distribute the RBs among the slices. Therefore, it is observed that the proposed approach based on Q-learning followed by heuristic algorithm is able to achieve a better outage probability. In particular, for the RAN slicing strategy with offline RL followed by the low-complexity heuristic algorithm with ω = 0.85 the probability of outage is around 14 % when the eMBB session generation rate λ_e is 1.6 sessions/s. In the case of the RAN slicing strategy with only off-line RL, the probability of outage is 28%. In turn, for the reference proportional approach, the probability of outage is 32 % (i.e. offline RL followed by the low-complexity heuristic approach with ω = 0.85 achieves a relative improvement of 56 % with respect to the reference).

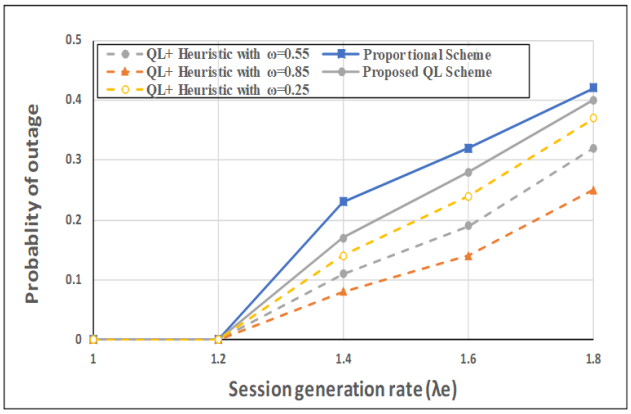

Figure 6. Outage probability as a function of the eMBB session generation rate λe (UEs/s).

Fig. 7 illustrates the average latency for V2V service as a function of the V2X UEs packet generation rate λ_v (packets/s). We clearly observe from Fig.7 that the delay is only 0.03s when packet generation rate $\lambda_v \leq$ 6vehicles/s, while there is a marked increase when the packet generation rate λ_v increases (i.e., when $\lambda_v \geq 6$). The reason for this increase is that for low loads (i.e., when $\lambda_v < 6$), when the system has sufficient radio resources regardless of the considered scheme, the latency is only due to the transmission delay. On the contrary, when the load increases, some situations of resource unavailability may arise, leading to increased queueing delay. In this case, the approach based on Q-learning followed by the heuristic algorithm, is able to better handle the load increase and lead to lower latency values than the other techniques.

From the presented results, we also notice that the approach proposed in this paper reduces the latency compared to the reference schemes. In case of the proposed strategy with offline RL followed by the low-complexity heuristic algorithm with ω = 0.85, when the vehicle arrival rate is 10 vehicles/s, the average latency is only around 0.12s, while.in case of the proposed scheme with only offline RL algorithm, the average latency is about 0.28s. In case of the reference with proportional approach, the latency is about 0.32s (i.e. offline RL followed by the low-complexity heuristic approach with ω = 0.85 achieves a relative gain of 62 %). The gains are achieved because the proposed approach makes a more efficient use of the available RBs. Thus it reduces the corresponding waiting time and the transmission delay.

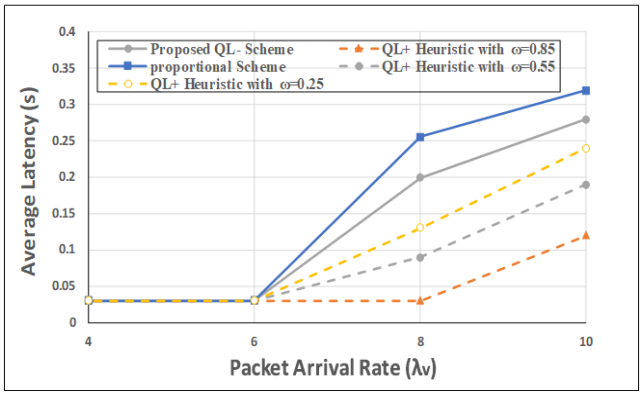

Figure 7. Average Latency as a function of the V2X UEs packet generation rate λ_V (packets/s).

5. Conclusions

In this paper, we have investigated the performance of a RAN slicing strategy for splitting the radio resources into multiple RAN slices to support V2X and eMBB services in uplink, downlink and sidelink (direct V2V) communications. The RAN slicing strategy is based on off-line RL followed by a low-complexity heuristic approach. This strategy has been compared against a reference scheme that makes an allocation of resources in proportion to the traffic rate of each slice. Extensive simulations were conducted to validate and analyze the performance of the RAN slicing strategy.

Simulation results show the capability of the RAN slicing strategy to allocate the resources efficiently and improve the network performance. From the presented results, we notice that the RAN slicing strategy with both off-line RL and off-line RL followed by the low-complexity heuristic approach maintains high resource utilization significantly, when the number of actions increases. The presented results also showed that further improvements are obtained when the configuration parameter ω of the low-complexity heuristic approach is increased. The proposed solution achieved better resource utilization, data rate, latency and outage probability with the value of ω equal to 0.85 compared to the proposed solution with other values of ω. Besides, our RAN slicing scheme outperforms the proportional scheme in terms of resource utilization, data rate, latency and outage probability for all the assumed values of the configuration parameter ω.

Future work includes the possibility of extending the evaluation of the algorithm for multi-cell scenarios. In this respect, since the algorithm is devised to work on a per-cell basis, this extension could be carried out just by having a separate slicing controller for each cell operating based on the specific traffic conditions of that cell. This would allow handling situations in which the traffic is not homogeneous in different cells.

Acknowledgements

This work was supported in part by the Spanish Research Council and FEDER Funds under SONAR 5G Grant with reference TEC2017-82651-R, and in part by the Baghdad University of Technology.

References

[1] NGMN Alliance. *Description of Network Slicing Concept*. Accessed: Apr. 5, 2019. [Online]. Available: https://www.ngmn.org/_leadmin/user_upload/160113_Network_Slicing_v1_0.pdf

[2] ITU-R, "ITU-R M.[IMT-2020.TECH PERF REQ] - Minimum Requirements Related to Technical Performance for IMT-2020 Radio Interface(s)," Report ITU-R M.2410-0, Nov. 2017.

[3] 3GPP, "Study on new radio (NR) access technology physical layer aspects," TR 38.802, Mar. 2017.

[4] *Description of Network Slicing Concept*, NGMN-Alliance, 2016, vol. 1.[Online]. Available: https://www.ngmn.org/_leadmin/user_upload/160113_Network_Slicing_v1_0.pdf

[5] R. Ferrús, O. Sallent, J. Pérez-Romero, and R.Agustí, `` On 5G Radio Access Network *Slicing: Radio Interface Protocol Features and Configuration,"* IEEE Communications Magazine, *Volume: 56* , Issue: 5 *,vol. 7*, pp. 184 - 192, *May, 2018*.

[6] *Network Slicing for 5G Networks and Services*, document, 5G Americas, Bellevue, WA, USA, Nov. 2016. Accessed: Apr. 5, 2019. [Online]. Available: http://www.5gamericas.org/_les/1414/8052/9095/5G_Americas_Network_Slicing_11.21_Final.pdf

[7] Management and orchestration; Concepts, use cases and requirements (Release 15), document 3GPP TS 28.530 V15.0.0, Sep. 2019.

[8] O. Sallent, J. Perez-Romero, R. Ferrus, R. Agusti, "On Radio Access Network Slicing From a Radio Resource Management Perspective", IEEE Wireless Communications, October, 2017, pp. 166-174.

[9] D. Marabissi, and R. Fantacci, `` Highly Flexible RAN Slicing Approach to Manage Isolation, Priority, Efficiency," *IEEE Access*, vol. 7, pp. 97130 _ 97142, Jul. 2019.

[10] H. M. Soliman and A. Leon-Garcia, ``QoS-aware frequency-space network slicing and admission control for virtual wireless networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1_6.

[11] M. R. Sama, X. An, Q. Wei, and S. Beker, ``Reshaping the mobile core network via function decomposition and network slicing for the 5G Era," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Apr. 2016, pp. 1_7.

[12] A. Aijaz, ``Hap-SliceR: A radio resource slicing framework for 5G networks with haptic communications," *IEEE Syst. J.*, vol. 12, no. 3, pp. 2285_2296, Sep. 2018.

[13] R. Li, Z. Zhao, and Qi Sun. (May. 2018). `` Deep reinforcement learning for network slicing." [Online]. Available: https://arxiv.org/abs/1805.06591.

[14] L. Tang, Q. Tan, Y. Shi, C.Wang, and Q. Chen, `` Reinforcement Learning for Slicing in a 5G Flexible RAN," IEEE Journal of Light wave Technology, vol. 37, no. 20 ,pp. 5161_5169, Oct., 2019.
[15] I. S. Comsa, A. De-Domenico, and D. Ktenas, "QoS-driven scheduling in 5G radio access networks—A reinforcement learning approach," in Proc. IEEE Global Commun. Conf., 2017, pp. 1–7.
[16] Z. Xu, Y.Wang, J. Tang, J.Wang, and M. C. Gursoy, "A deep reinforcement learning based framework for power-efficient resource allocation in cloud RANs," in Proc. IEEE Int. Conf. Commun., 2017, pp. 1–6.
[17] C. Natalino, M. R. Raza, A. Rostami, P. Ohlen, L. Wosinka, and P. Monti, "Machine learning aided resource orchestration in multi-tenant networks," in Proc. IEEE Photon. Summer Top. Meeting, Jul. 2018, doi: 10.1109/PHOSST.2018.8456735.
[18] D. T. Hoang, D. Niyato, P. Wang, A. de Domenico, and E. C. Strinati. (Dec. 2017). ``Optimal cross slice orchestration for 5G mobile services." [Online]. Available: https://arxiv.org/abs/1712.05912.
[19] L. Tang, Q. Tan, Y. Shi, C.Wang, and Q. Chen, ``Adaptive virtual resource allocation in 5G network slicing using constrained markov decision process," IEEE Access, vol. 6, pp. 61184_61195, Oct. 2018.
[20] P. Caballero, A. Banchs, G. de Veciana, and X. Costa-Pérez, ``Network slicing games: Enabling customization in multi-tenant networks," in *Proc. IEEE Conf. Comput. Commun.*, May 2017, pp. 1_9.
[21] V. Sciancalepore, L. Zanzi, X. Costa-Perez, and A. Capone. (Jan. 2018).``ONETS: Online network slice broker from theory to practice." [Online]. Available: https://arxiv.org/abs/1801.03484.
[22] Haider Daami R. Albonda, J. Pérez-Romero, "Reinforcement Learning-based Radio Access Network Slicing for a 5G System with Support for Cellular V2X"`. International Conference on Cognitive Radio Oriented Wireless Networks (CROWNCOM), Poznan, Poland. 2019.
[23] Haider D. Resin Albonda Jordi Pérez-Romero , "An Efficient RAN Slicing Strategy for a Heterogeneous Network With eMBB and V2X Services", IEEE Access, vol. 7, pp. 44771 - 44782, Mar. 2019.
[24] 3GPP TS 38.401 v15.2.0, "NG-RAN; Architecture description (Release 15)", June, 2018.
[25] Haider Daami R. Albonda, J. Pérez-Romero, ``An Efficient Mode Selection for improving Resource Utilization in Sidelink V2X Cellular Networks. IEEE (CAMAD) workshops. Barcelona, Spain, Sep. 2018.
[26] R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction. Cambridge, MA, USA: MIT Press, 1998.
[27] 3GPP TS 38.211 v15.2.0 "NR; Physical channels and modulation (Release 15)", June, 2018.
[28] 3GPP TR 36.942 v15.0.0, "Radio Frequency (RF) system scenarios", September, 2018.
[29] Report ITU-R M.2135 "Guidelines for evaluation of radio interface technologies for IMT-Advanced", 2009
[30] WINNER II Channel Models, D1.1.2 V1.2., available at http://www.cept.org/files/1050/documents/winner2%20%20final%20r report.pdf.