Reinforcement Learning-based Dynamic Resource Allocation For Grant-Free Access

Mariam Elsayem, Hatem Abou-zeid, *Member, IEEE*, Ali Afana and Sidney Givigi, *Senior Member, IEEE*

GLOBECOM 2022 - 2022 IEEE Global Communications Conference | 978-1-6654-3540-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/GLOBECOM48099.2022.10001586

***Abstract*—Cellular networks have evolved to deliver high-speed broadband services to support the requirements of IoT applications, which demand high speed, low latency, and massive capacity. A primary market goal is to provide support for ultra-reliable low latency communication (URLLC). URLLC requires sub-milliseconds-level latencies as defined by the third generation partnership project (3GPP). One of the promising technologies to achieve the aforementioned specifications is grant-free (GF) access for uplink resources. The GF scheme enables the user equipment (UE) to transmit data over pre-allocated resources which reduces communication latency. This paper proposes an intelligent Reinforcement Learning (RL) based allocator of grants trained via Deep Q-Learning. The experimental results show effect of the number of UEs in the network, and the percentage of unstable UEs on the speed of the RL agent's convergence.**

***Index Terms*—5G NR; Grant-free; URLLC; DQN.**

I. Introduction

5G wireless systems are expected to support new services, such as mission-critical Ultra-reliable and Low-latency Communication (URLLC) services [1]. Examples of IoT use cases that require URLLC include Factory Automation, Remote surgery, Smart Transportation, Tactile Internet, augmented/virtual reality, delivery drones, frequency and voltage automation in smart electricity grids and more [2]. However, enabling URLLC requires a shift in the wireless system design and innovative solutions that can be achieved through communication systems with flexible structures such as the 5G NR. Recent studies on resource allocation for URLLC have proposed grant-free (GF) scheduling rather than the traditional high latency Grant-Based (GB) scheduling used in 4G LTE (LTE) [3]. In the traditional uplink (UL) access procedure, every transmission is based on a scheduling request and a grant. This process relies on reliable control signaling between the user equipment (UE) and the base station (BS), which introduces excessive delays and results in overhead in the communication link. GF transmissions are shown to be a promising scheme for reducing the latency in the UL access by avoiding the scheduling request process. In GF, the grants are scheduled beforehand and sent to the UEs to be used once packets are ready for transmission.

M. Elsayem and S. Givigi are with School of Computing, Queen's University of Kingston, 557 Goodwin Hall, Kingston, ON, Canada 20mje1@queensu.ca, sidney.givigi@queensu.ca

H. Abou-zeid is with the Department of Electrical and Software Engineering, University of Calgary, ICT 402, 2500 University Drive NW, Calgary, AB T2N 1N4 hatem.abouzeid@ucalgary.ca

A. Afana is with the Ericsson, Ottawa, ON ali.afana@ericsson.com

To optimally apply GF, two objectives must be accomplished. The first is the selection of UEs to utilize GF access. The second is an optimal assignment of resources to ensure the stability of the network while minimizing resource wastage. The main obstacle to overcome is that the network environment is constantly changing and difficult to predict. This stems from the fact that UEs usually have a variety of traffic behaviors that change over time. Also, given that the UEs are expected to be mobile, their channel quality indicator (CQI) is also expected to vary. Thus, in order to achieve the aforementioned objectives, an intelligent-based resource allocation technique is needed to optimally assign resources to UEs based on the forecasted traffic and their CQIs. Such an algorithm needs to be adaptable to changes in the network and meet the needs of the selected UEs while considering other UEs sharing the same channel.

Reinforcement Learning (RL) is a type of Machine Learning that differs from other types of algorithms, such as supervised and unsupervised learning algorithms, by its interactive nature. As such, RL algorithms do not require large amounts of data to be provided beforehand to be trained. Rather, an RL agent trains on its own following an interacting paradigm to find which actions to take through trial and error [4], which is a merit for the GF scheduling problem. The main challenge is to design an adequate framework that will help the RL agent to understand the objectives that it needs to accomplish. In our case, that would be optimally selecting UEs for GF access and allocating the proper amount of resources given the environment to reduce the transmission latency while maintaining a fair distribution of resources. The paper is organized as follows. A brief literature review is presented in section II. The system model and problem formulation are presented in section III. Section IV discusses the proposed GF access scheme with DRL. Section V presents the simulation setup and the results. Finally, Section VI concludes the paper.

II. Literature Review

Previous efforts in studying the GF scheme can be categorized into two classes. First the study of GF scheduling using shared resources, and second, the study of GF scheduling using dedicated resources. In the case of aperiodic/sporadic traffic, resources are shared by groups of UEs to improve resource utilization. However, shared resources lead to collisions if two or more UEs simultaneously send over the same resources [5]. On the other hand, the dedicated resources scheme is beneficial when the traffic is periodic/deterministic

Authorized licensed use limited to: Beijing Information Science & Tech Univ. Downloaded on February 11,2023 at 07:07:18 UTC from IEEE Xplore. Restrictions apply.

and it is possible to anticipate the demand for resources. There have been some efforts to help analyze the impact of using GF over the traditional GB scheme. In the next subsections, we review some of the works presented in the literature for both GF with shared and dedicated resources.

A. Grant-Free with Shared Resources

One of the early attempts of GF proposes a blind re-transmission scheme over the shared resource to enhance the reliability of URLLC [6]. In this scheme, a group of UEs share a pool of resources, where UEs are provided with one or multiple opportunities to re-transmit their packets. The scheme uses interference cancellation receivers to reconstruct previously received re-transmission and subtract them from the new re-transmissions from the shared resources. Thus, it does not require extra signalling to ask UEs for re-transmission. In the proposed scheme, UEs transmit initial data over dedicated resources configured by the BS and, then, the UEs perform blind re-transmissions using a pool of shared resources. Once the initial packet is received at the BS, the data is decoded as well as the data sent over the shared resources. The BS then subtracts the initial decoded signals from the combined signal. Since the method [6] does not require control signalling to request re-transmissions, the communication latency is reduced. Moreover, the solution is more resource-efficient than re-transmission over dedicated resources, due to the fact that resources are shared between groups of UEs.

In [7], the authors propose a hybrid resource allocation scheme with both dedicated and shared resource allocation. The main objective is to determine the needed amount of dedicated and/or shared resources to satisfy the requirements of the UEs without wasting resources. The aim of adding shared resources is to improve resource utilization in case of a low traffic load. The solution is based on a probabilistic model to determine the number of resources and number of repetitions needed to achieve URLLC requirements.

Two solutions to enhance reliability and increase resource efficiency are proposed in [8]. The first solution is to perform re-transmission over shared resources, whereas the second one is GF transmission with an advanced receiver to resolve the overlapping introduced by the used GF non-orthogonal multiple access schemes. Similar to [7], re-transmissions are done over a pool of shared resources assigned by the BS. Each UE can randomly select a re-transmission resource from the pool of resources, then the BS decodes the initial transmission as well as the transmitted packets. Since the initial transmissions are done over dedicated resources the success probability of these transmissions is high, thus the chance of collisions over the shared resources is negligible.

Authors in [9] study the levels of reliability and latency that can be achieved by GF scheduling with K-repetition and shared resources in case of aperiodic or sporadic traffic. The authors have provided an analysis as a function of the number of UEs, the number of assigned resources, and the value of replicas K. Moreover, the impact of packet collisions and self-collision has been taken into account to study their effect on reliability. Through simulation and analytical expressions, the study has demonstrated that GF with shared resources and K-repetitions cannot meet the tight reliability and latency requirements of URLLC due to the collisions, as well as self-collisions, having a non-negligible impact on reliability.

A similar study was conducted to analyze the performance of the K-repetitions scheme with GF access [10], showing that this scheme, with optimized power control, number of K replicas, and number of sub-bands, can outperform GF HARQ. However, the queuing effect introduced by the K-replicas results in latency violation.

As mentioned earlier, uplink packet repetitions are one way to increase the reliability of an application. However, these repetitions need to happen within a given time interval to avoid two main issues [11]. The first one is mixing HARQ Ids of different HARQ processes, and the second is missing re-transmission opportunities and thus decreasing the reliability. A scheme of reserved shared resources between a group of UEs is then presented [11], enabling the UEs to transmit K replicas to satisfy the reliability and latency requirements. The size of each resource is optimized based on its position to balance reliability and resource consumption, where the size of the resource decreases from the first to the last resource in a given period.

B. Grant-Free with Dedicated Resources

For periodic traffic where allocations can be planned ahead to provide UEs with more transmission opportunities, dedicated allocation of resources is a better choice. In dedicated resource allocation, each resource is reserved to one and only one UE at any time [5].

A semi Grant-free solution where both GF and GB protocols are implemented in a single channel can be used [12]. In the Semi GF scheme, some users can transmit using configured grants while others can use GB protocol through the same channel, improving the connectivity and spectral efficiency by sharing the channel between grant-free users and grant-based users. Two contention control mechanisms are proposed: Open-Loop Contention Control and Distributed Contention Control. These mechanisms are required to control the number of GF admitted UEs to the channel and to ensure low levels of interference between GF and GB UEs. The analytical analysis demonstrated that Semi GF applied with both Open-Loop Contention Control and Distributed Contention Control mechanisms outperformed both GF and GB. Moreover, the results show that Semi GF with Open-Loop Contention Control is suitable when the UE with GB access is close to the BS and GF UEs are edge users, and Vice Versa for the Distributed Contention Control mechanisms.

Flexibility in the selection from configure-grants can be achieved [13] Since the resources are configured with different parameters, the UE can select the most optimal resource for transmission that satisfies the latency and reliability requirements needed for its application.

A priority-based GF scheme with dynamic slot allocation is proposed in [14]. The model grants dedicated slots to high

Authorized licensed use limited to: Beijing Information Science & Tech Univ. Downloaded on February 11,2023 at 07:07:18 UTC from IEEE Xplore. Restrictions apply.

priority traffic based on transmission status and traffic estimation, the remaining slots are then granted to UEs with low traffic priority. This work could be implemented to solve one of the major challenges discussed in the literature regarding multiplexing URLLC packets with other applications. In this case, URLLC transmissions are considered of high priority, yet other applications can utilize the remaining available resources. Another study that tackles the coexistence of multiple services is presented in [15]. The authors investigated the coexistence of GF with GB schemes and URLLC with non-URLLC services. Moreover, hybrid multiple access solutions leveraging Machine Learning (ML) is proposed to serve the coexistence of URLLC with other services.

III. System Model & Problem Statement

A. Grant-free System Model

The designed model considers a GF system of a set of m UEs, M, where $m = |M|$, served by one BS. The UEs are classified into two classes, stable and unstable. UEs with periodic traffic and consistent channel conditions are considered stable. Similarly, UEs with inconsistent traffic and channel conditions are considered unstable. We assume fixed packet distribution for both the UL and downlink (DL) throughout the simulation, where each entity sends a fixed size *L* packet with fixed periodicity *p*. Moreover, we assume that the UL channel suffers from free space path loss and fading. The communication between UEs and the BS, also known as the gNodeB (gNB), follows the following pattern. At the start of the communication, the gNB informs the set of selected UEs *B*, where $B \subseteq M$, of the resource configurations RB. Each stable UE i can directly use the assigned resources RB_i once a new packet is ready for transmission. Meanwhile, unstable UEs undergo the traditional scheduling scheme. The designed model restricts the usage of the GF configured resources to every other frame, therefore, GF configurations are enabled on even frames only, and in odd frames, UEs in the set *B* are disallowed from transmitting their data.

IV. Proposed Methodology

An RL agent is used to find the best GF allocations for the network. The training phase is executed offline to avoid causing any delays in the network. The goal of this phase is to train the agent to produce a policy for GF access. Next, during the execution phase, the agent will rely on the policy to calculate the system's configuration.

A. Proposed Artificial Neural Network Architecture

We propose a Deep RL (DRL) approach, the algorithm performs both the selection of UEs for GF access and assigns the needed number of RBs. Based on preliminary testing, the use of DQN provides less computational overhead compared to tabular Q-learning and allows for a larger number of UEs in the network. The neural network is implemented using the MATLAB RL Toolbox. As illustrated in Fig 1, the neural network has three fully connected layers each with 128 neurons. The input of the network has a size of m (which is the number of UEs in the system).

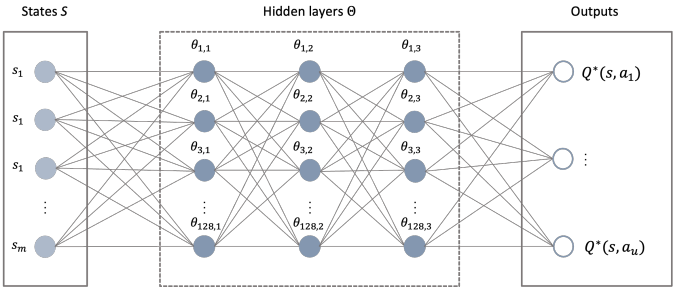

Fig. 1: Architecture of the deep Q-network used for the implementation of the algorithms with three hidden layers each of dimension 128. Here, $|S| = m$, and $|A| = u$

The proposed algorithm has the following parameters: an Experience Replay Memory of size 1000 and a mini-batch size of 64. The algorithm relies on the Epsilon-Greedy approach. An RL agent is defined by the tuple (S, A, T, r), where S is the set of states, A the set of actions, T an unknown transition function, and $r : S \times A \to \mathbb{R}$ a reward function.

- State space: The states consist of the buffer status of all m UEs connected to the BS. Since the buffer status is a continuous variable, one way to have countable states is to discretize the values into three states: State (1) is considered the most stable state, and state (3) is the least. The thresholds to define these states are configurable to the application. The state vector $s \in S$ is defined over the states of each UE:

$$s = \begin{bmatrix} S_{UE_1} & S_{UE_2} & \dots & S_{UE_m} \end{bmatrix}.$$

- Action space: An action is an array of size m. The value of the i^{th} index holds the number of assigned resource blocks (RB) to the i^{th} UE. RB_{UE_i} for the i^{th} UE can be in a range $0 \leq RB_{UE_i} \leq x$, where x will be defined by the user. Therefore, the action a is represented by

$$a = \begin{bmatrix} RB_{UE_1} & RB_{UE_2} & \dots & RB_{UE_m} \end{bmatrix}.$$

  There are a few constraints that limit the values of the action array. Any action that does not comply with the following conditions is invalid:
  - During a run, a maximum $n < m$ number of users can use GF access.
  - Each user could get a maximum of x grants.
  - The total number of assigned grants should not exceed k grants.

  The values of $n, x, k \in \mathbb{Z}^+$ are determined based on the application and the system requirements.
- Reward function: a weighted summation function $r(s, a)$ is used to calculate the reward of each configuration:

$$r(s,a) = -\sum_{i=1}^{m} w_1 \cdot S_{UE_i} + w_2 \cdot W_{UE_i} + w_3 \cdot E_{UE_i} \quad (1)$$

Authorized licensed use limited to: Beijing Information Science & Tech Univ. Downloaded on February 11,2023 at 07:07:18 UTC from IEEE Xplore. Restrictions apply.

where S_{UE_i}, W_{UE_i}, and E_{UE_i} are the buffer status, the number of wasted resources, and the number of transmission errors for the i^{th} UE, respectively. Additionally, w_1, w_2, and $w_3 \in \mathbb{R}$ can be used to weigh each variable based on the application requirements.

B. Training Algorithm

In DQN, one of the hyperparameters is epsilon (ϵ), which acts as a tradeoff between exploring the environment and exploiting the obtained knowledge [16]. To obtain the value of each state-action pair, the algorithm can either follow a greedy approach by selecting the action that yields the highest value in state s, or select a random action to explore new options.

$$a_t = \begin{cases} \arg\max_{a \in A} \ Q_t(s_t, a) & with\ probability\ 1-\varepsilon; \\ Random\ action\ a & with\ probability\ \varepsilon \end{cases}, \tag{2}$$

and update the Q-value based on equation (3) accordingly [17],

$$\begin{aligned} Q(s_t, a_t) \quad \leftarrow \quad & (1-\alpha) Q(s_t, a_t) + \\ & \alpha [r_{t+1}(s_t, a_t) + \gamma \max_{a \in A} Q(s_{t+1}, a)], \end{aligned} \tag{3}$$

where $0 \leq \alpha \leq 1$ is called the *learning rate* which decays over time. When α is 0, the value of the state-action pair never get updated, hence the agent learns nothing. Setting a high learning rate, on the contrary, speeds up the learning process.

As in the typical RL training phase, the agent first takes a random action, in other words, it randomly assigns resource blocks to the UEs, which propagates through the 5G network. Then, the agent receives feedback from the environment about the new state of the network as well as the resulted reward. The agent then learns from the consequences of its action and chooses the next action accordingly. The process is repeated until the agent reaches the best stable policy and concludes the training phase. A stable policy is achieved when the cumulative rewards stop changing.

Fig. 2 shows the link between RL and the 5G simulation during the training phase. The training starts by running a DQN algorithm where it outputs an action that encodes the allocation of resources process for all the UEs in the system. The 5G simulation module then receives the suggested resource allocation and simulates 6 frames (60ms), which results in an updated environment state. The state is returned to the agent to update the policy. These steps are repeated until policy convergence is achieved for the given scenario.

V. Simulation Setup & Results

In this section, we first introduce the system setup and describe the updates done in the Matlab 5G toolbox to test the solution. We also present the chosen parameters to simulate the 5G environment and the needed parameters to train the RL agent. Finally, we provide and discuss the experimental results of the proposed solution.

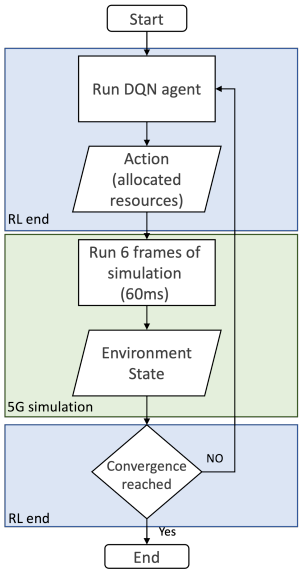

Fig. 2: Flowchart of the training phase

A. Experimental Setup

This section summarizes the edits done to the original 5G toolbox implementation to accurately simulate the environment needed to test the proposed solution. MATLAB offers a 5G toolbox that provides standard functions to configure, simulate, measure, and analyze the end-to-end 5G NR communications. The toolbox supports link-level simulation, and it can be customized to design and test prototypes. Some modifications to the existing functions were performed to allow for simulation of GF access, not implemented by default.

To integrate GF scheduling, functions that allocate both GB and GF grants were implemented in the scheduler module on the BS side. These functions allocate GF grants based on the RL agent's suggestions while using the traditional Proportional Fair scheduling technique for the GB scheme. In order to properly integrate both schemes, the GB and GF, the following algorithm was added to the toolbox:

- scheduleULResourcesSlot: This function runs in the scheduler of the gNB to assign UL resources for every slot of the 10ms frame in TDD mode. As described in Algorithm 1, the algorithm schedules both GF and GB UEs. The function takes $GRANTINPUT$ as an input which comprises of the following parameters: *slotNum* (the current slot number in the 10 ms frame whose UL resources are getting scheduled), *GFSlots* (the slots reserved for GF scheduling), *GfUEs* (he UEs selected by the RL agent for GF scheduling), *RNTI* (stands for Radio Network Temporary Identifier, a unique identifier assigned to the UEs), *startSymb* (start symbol of time-domain resources), *numSym* (number of symbols allotted in time-domain). For slots not reserved for GF scheduling, the algorithm runs the Proportional Fair technique to allocate resources with GB access. The algorithm

Authorized licensed use limited to: Beijing Information Science & Tech Univ. Downloaded on February 11,2023 at 07:07:18 UTC from IEEE Xplore. Restrictions apply.

prohibits UEs with GF access to use GB resources. Thus, GF UEs undergo a different scheduling process. This process is done in the first frame of the simulation, and the output of the algorithm is sent to the UEs with GF access. The UEs then save the received grants to be used in the assigned slots of every eligible frame.

Algorithm 1: scheduleULResourcesSlot

```
Input: GRANTINPUT
Output: UPLINKGRANTS
if slotNum ∉ GFSlots then
    if RNTI ∉ GfUES then
        Run traditional Proportional Fair scheduling
        using parameters in GRANTINPUT.
else
    for all UL TTIs do
        if RNTI require retransmission then
            Allocate RBs for retransmission
            if RBs ≠ 0 after retransmissions then
                Allocate remaining RBs for new
                transmissions
        else
            Allocate RBs for new transmissions
    Run Traditional GB
Update the next to-be-scheduled UL slot
Save all UL assignments to UPLINKGRANTS
```

The list of parameters used to simulate the system are summarized in Table I.

B. Experimental Results

This experiment aims to shed light on the two main 5G network-related parameters that directly affect the convergence of the RL model. These parameters are the number of UEs and the percentage of unstable UEs in the network.

Model convergence is reached when the cumulative rewards stop changing and the network reaches a stable state. Convergence depends on the number of UEs in the system. For example, for 10 UEs with 30% unstable UEs, the system takes around 400 episodes to converge, while when 40 UEs with the same percentage of unstable UEsare considered, the system only converges after 700 iterations. This is due to the fact that the size of the action space is directly proportional to the number of UEs in the network.

TABLE I: System parameters

Parameter	Value
Spectrum technique	TDD
Carrier and Bandwidth	$2.595GHz$ and $5MHz$
PHY numerology	$15kHz$ sub-carrier spacing
Number of resource blocks	25
Traffic model	On-0ff traffic pattern.
Number of uplink/ downlink slots in $10ms$ frame	7 Downlink slots 3 Uplink slots
Number of Users	40
Learning rate α	0.01
Discount factor γ	0.80
Maximum epsilon ε	1.00
Minimum epsilon ε	0.01
Decay rate of epsilon ε	0.0050
Reward function weights	$w_1 = w_2 = w_3 = 1$
Size of replay memory	10000
DQN Activation function	$ReLU(x) = max(0, x)$
# of hidden layers & neurons	3 layers & 128 neuron in a layer

One way to speed up convergence is to reduce the number of UEs included in the action space by eliminating the unstable UEs. This is a profitable decision when UEs with sporadic traffic, and/or unstable CQIs do not fit the characteristics required for GF access. An experiment was performed to show the effect of eliminating the unstable UEs on the speed of convergence. In the first part of the experiment, we focused on the number of UEs as a parameter, where it was increased gradually to see the effect it had on convergence. As shown in Fig. 3, the convergence time increases as the number of UEs increases. As the same figure shows, convergence time when eliminating unstable UEs significantly decreased as compared to the default setup.

In the second part of the experiment, we studied the effect of increasing the percentage of unstable UEs while keeping the total number of UEs in the network constant. The results in Fig. 4 show that by increasing the percentage of unstable UEs, the agent requires a longer time to converge. In order to better understand the given results, a stress test was performed to monitor the agent's behavior during training. Fig. 5 shows that the agent finds difficulty in reaching convergence while training with high percentages of unstable UEs, as it keeps on oscillating between various actions. This is due to the fact that in a highly unstable environment, the traffic and UEs' behavior is unpredictable, making it hard to learn. For the same experiment in Fig. 4, eliminating the unstable UEs for the same scenarios resulted in faster convergence. The figure illustrates that as the percentage of unstable UEs increases, the convergence time decreases. This is expected for two reasons, first, the reduction of the action space. Second, the RL agent now only trains with stable and predictable UEs.

VI. Conclusion

In this paper, an intelligent RL-based resource allocator was proposed for GF access. The agent was trained using Deep Q-Learning to optimally select UEs for GF access, and allocate the proper number of resources given to the environment to ensure low latency while minimizing resource wastage. A MATLAB 5G simulation toolbox was modified to simulate the proposed system. The results showed the effect of the number of UE in the network, and the percentage of unstable UEs on the RL convergence and that eliminating the unstable UEs improves the convergence time.

References

[1] H. Chen, R. Abbas, P. Cheng, M. Shirvanimoghaddam, W. Hardjawana, W. Bao, Y. Li, and B. Vucetic, "Ultra-reliable low latency cellular networks: Use cases, challenges and approaches," *IEEE Communications Magazine*, vol. 56, no. 12, pp. 119–125, 2018.

Authorized licensed use limited to: Beijing Information Science & Tech Univ. Downloaded on February 11,2023 at 07:07:18 UTC from IEEE Xplore. Restrictions apply.

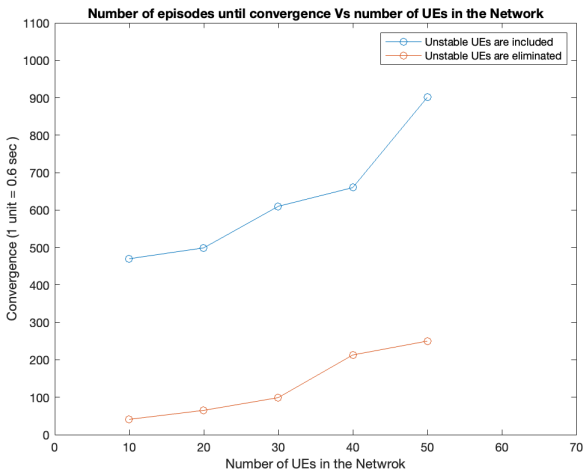

Fig. 3: Effect of increasing the number of UEs on the convergence time of the agent

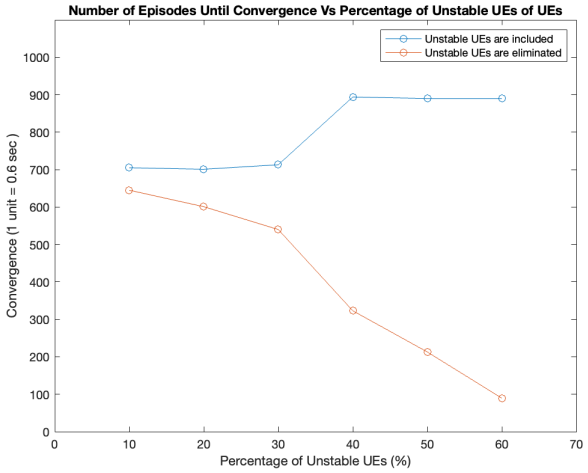

Fig. 4: Convergence time vs the percentage of unstable UEs in the network

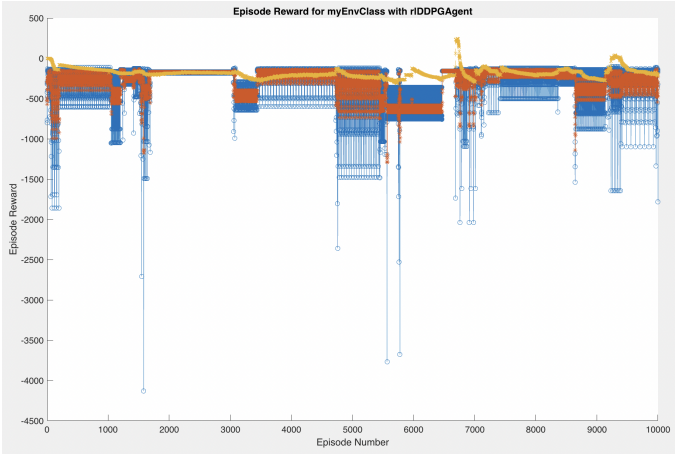

Fig. 5: A stress test showing the agent's convergence in a network with 40 UEs where the percentage of unstable UEs is 90%.

[2] M. Shafi, A. F. Molisch, P. J. Smith, T. Haustein, P. Zhu, P. De Silva, F. Tufvesson, A. Benjebbour, and G. Wunder, "5g: A tutorial overview of standards, trials, challenges, deployment, and practice," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 6, pp. 1201–1221, 2017.

[3] 3GPP, "Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Study on Latency Reduction Techniques for LTE," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 36.331, 04 2016, version 14.0.0.

[4] G. Neto, "From single-agent to multi-agent reinforcement learning: Foundational concepts and methods," 01 2005.

[5] 3GPP, "Study on scenarios and requirements for next generation access technologies," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.212, 04 2018, version 15.0.0. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2440

[6] R. Abreu, G. Berardinelli, T. Jacobsen, K. Pedersen, and P. Mogensen, "A blind retransmission scheme for ultra-reliable and low latency communications," in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, 2018, pp. 1–5.

[7] Z. Zhou, R. Ratasuk, N. Mangalvedhe, and A. Ghosh, "Resource allocation for uplink grant-free ultra-reliable and low latency communications," in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, 2018, pp. 1–5.

[8] N. H. Mahmood, R. Abreu, R. Böhnke, M. Schubert, G. Berardinelli, and T. H. Jacobsen, "Uplink grant-free access solutions for urllc services in 5g new radio," in *2019 16th International Symposium on Wireless Communication Systems (ISWCS)*, 2019, pp. 607–612.

[9] M. d. C. Lucas-Estañ, J. Gozalvez, and M. Sepulcre, "On the capacity of 5g nr grant-free scheduling with shared radio resources to support ultra-reliable and low-latency communications," *Sensors*, vol. 19, p. 3575, 08 2019.

[10] T. Jacobsen, R. Abreu, G. Berardinelli, K. Pedersen, I. Z. Kovacs, and P. Mogensen, "System level analysis of k-repetition for uplink grant-free urllc in 5g nr," in *European Wireless 2019; 25th European Wireless Conference*, 2019, pp. 1–5.

[11] T. Le, U. Salim, and F. Kaltenberger, "Optimal reserved resources to ensure the repetitions in ultra-reliable low-latency communication uplink grant-free transmission," in *2019 European Conference on Networks and Communications (EuCNC)*, 2019, pp. 554–558.

[12] Z. Ding, R. Schober, P. Fan, and H. V. Poor, "Simple semi-grant-free transmission strategies assisted by non-orthogonal multiple access," *IEEE Transactions on Communications*, vol. 67, no. 6, pp. 4464–4478, 2019.

[13] A. Gunturu, V. S. Tijoriwala, and A. K. R. Chavva, "Optimal configured grant selection method for nr rel-16 uplink urllc," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1–6.

[14] T. N. Weerasinghe, V. Casares-Giner, I. A. M. Balapuwaduge, and F. Y. Li, "Priority enabled grant-free access with dynamic slot allocation for heterogeneous mmtc traffic in 5g nr networks," *IEEE Transactions on Communications*, pp. 1–1, 2021.

[15] A. Azari, M. Ozger, and C. Cavdar, "Risk-aware resource allocation for urllc: Challenges and strategies with machine learning," *IEEE Communications Magazine*, vol. 57, no. 3, pp. 42–48, 2019.

[16] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3, pp. 279–292, May 1992. [Online]. Available: https://doi.org/10.1007/BF00992698

[17] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. The MIT Press, 2020.

Authorized licensed use limited to: Beijing Information Science & Tech Univ. Downloaded on February 11,2023 at 07:07:18 UTC from IEEE Xplore. Restrictions apply.