AVD643 : Innovative Design Project

Scheduling for Minimizing Uplink Delay in 5G Systems

Submitted by

Sudarsanan A K
Student ID : SC21M063

Under the guidance of

Dr. Vineeth B S,
Asst.Prof., Department of Avionic, IIST.

Indian Institute of Space science and Technology,
Valiamala, Trivandrum
May 24, 2022

Contents

List of Figures

1 Introduction

1.1 Motivation for the Problem

The main objective of the radio resource scheduling (RRS) task is to dynamically allocate spectrum resources to user equipment (UEs) with an optimal trade-off between some key performance indicators (KPIs), like spectrum efficiency (SE), fairness, delay, etc [2]. RRS systems are currently powered by off-the-shelf specified tools. This comprises, for example, proportional fairness (PF). best CQI and round robin (RR) variations. This plan has shown to be successful, but it will be tough to sustain in the future [1].
From [1], current radio resource scheduling tools are designed to always follow a single policy. The 5G system, on the other hand, will use a variety of policies to adapt to network configuration and traffic dynamics. Model-based and model-free ways to learn (or select) the appropriate policy under current network conditions can be provided by artificial intelligence (AI). One of the most common approaches is to use recent AI developments such as deep reinforcement learning (DRL). Applying RL based methods in this specific problem by modelling it to fit the RL framework may produce better results. The large dimensional state space containing the context of the UEs (CQI, bufferstatus, HARQs, served data rates etc) and the scheduler and the large dimensional action space consisting of the RBG bit map makes the radio resource scheduling a good candidate problem that is approachable in RL perspective.

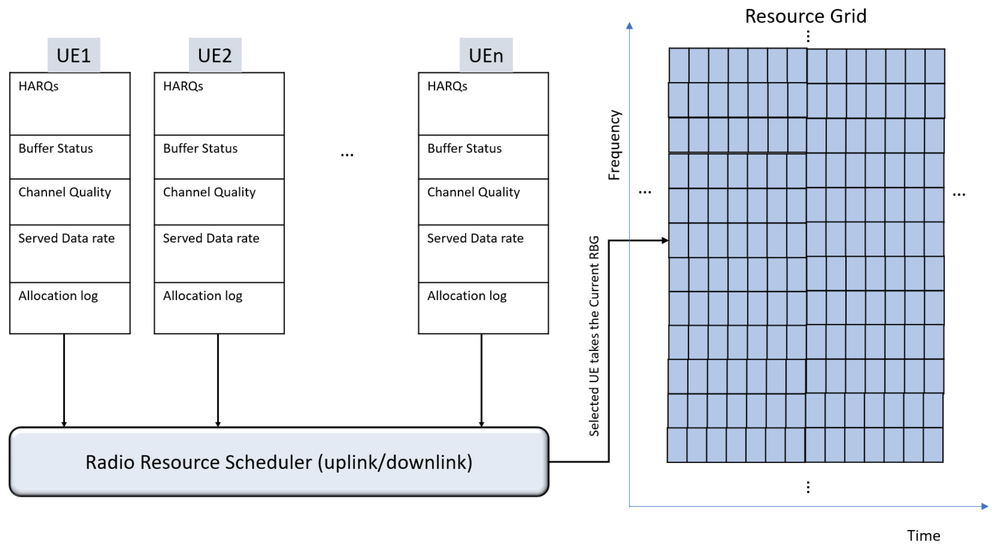

Figure 1: Radio resource scheduler

As indicated in figure 1, the gNB is able to collect information from the competing UEs which includes channel feedback information, buffer status, HARQs, and allocation log. Additional information can also be obtained for selecting the UE to which RBG would be assigned (among the eligible ones). The radio resource scheduler runs at the gNB at every (or k^{th}) slot and uses this information to share the available RBGs between active UEs. Therefore, the problem is to filling the resource grid by deciding which UE will win the current RBG in the current slot.

1.2 Literature Survey

[1] Describes a fine-grained centralized scheduling strategy for the radio resource allocation problem. They design the state-space as a combination of eligibility, data rate, and fairness. The action space is selection of a UE among the competing eligible UEs. Reward is designed as variant of a discounted best CQI function. In [2] the notion of minislots are described and used to serve for the uRLLC QoS use case. They introduce a "Superposition/puncturing" framework for multiplexing eMBB and URLLC in 5G cellular system. [3] Uses a 1D convolution layer (conv1D) and softmax activation in output layer to construct a deep RL agent for radio resource scheduling. The cell throughput is considered as the reward. [3] Develops a dynamic RRM scheduler able to select, at each TTI, appropriate scheduling rules according to the momentary network conditions and QoS requirements. The obtained results show significant gains in terms of both delay and PDR satisfaction. A RL algorithm is used to learn non-linear functions that approximate the scheduling rule decision at each TTI based on the instantaneous scheduler state. To evaluate the performance of different RL algorithms, five RL algorithms were selected and implemented. Their performance was tested in terms of variable window size, traffic type, objective and dynamic network conditions. In order to reduce the scheduler state space and speed-up the learning procedure when refining the NNs weights. In this paper, the focus of the compression procedure is on packet delay and PDR as Key Performance Indicators (KPIs).

For the simulation of the 5G scheduling, we require a scheduling simulator which provides the environment. 5G-LENA Simulator[4] is a GPLv2 New Radio (NR) network simulator, designed as a pluggable module to ns-3. The simulator is the natural evolution of LENA, the LTE/EPC Network Simulator, the development started from the mmWave module, and it incorporates fundamental PHY-MAC NR features aligned with NR Release 15 TS 38.300. MATLAB [5] provides 5G Toolbox™ system-level simulations model provides standard-compliant functions and reference examples for the modeling, simulation, and verification of 5G New Radio (NR) communication systems. The simulations operate across a protocol stack that includes physical (PHY), medium access control (MAC), radio link control (RLC), and application layers. Tetcos Netsim[6] is an end-to-end, full stack, packet level network simulator and emulator. It provides network engineers with a technology development environment for protocol modeling, network RD and military communications. The behavior and performance of new protocols and devices can be investigated in an virtual network within NetSim at significantly lower cost and in less time than with hardware prototypes.

2 Background

2.1 5G Scheduling

2.1.1 The 5G stack and functionalities

The functions of 5G layer 1 i.e. PHYSICAL (PHY) Layer[7] include Error detection on the transport channel and indication to higher layers , FEC encoding/decoding of the transport channel , Hybrid ARQ soft-combining , Rate matching of the coded transport channel to physical channels , Mapping of the coded transport channel onto physical channels , Power weighting of physical channels , Modulation and demodulation of physical channels , Frequency and time synchronisation , Radio characteristics measurements and indication to higher layers , Multiple Input Multiple Output (MIMO) antenna processing , Transmit Diversity (TX diversity) , Digital and Analog Beamforming and RF processing

The functions of 5G layer 2 i.e. MAC sublayer are Beam management , Random access procedure , Mapping between logical channels and transport channels , Concatenation of multiple MAC

SDUs belonging to one logical channel into transport block (TB), Multiplexing/demultiplexing of 5G-MAC SDUs belonging to one or different logical channels into/from transport blocks (TB) delivered to/from the physical layer on transport channels , Scheduling information reporting , Error correction through HARQ , Priority handling between logical channels of one UE , Priority handling between UEs by means of dynamic scheduling , Transport format selection and Padding
The functions of 5G layer 2 i.e. RLC sublayer include, Transfer of upper layer PDUs , Error Correction through ARQ (only for AM data transfer) , Reordering of 5G-RLC data PDUs (only for UM and AM data transfer) Duplicate detection (only for UM and AM data transfer) , Protocol error detection (only for AM data transfer) , 5G-RLC SDU discard (only for UM and AM data transfer) , Segmentation (only for UM and AM data transfer) , Resegmentation (only for AM data transfer) and 5G-RLC re-establishment.
The functions of 5G layer 3 i.e. RRC Layer are Broadcasting of system informations to NAS and AS. , Establishment, maintenance and release of RRC connection. , Security including key management , Establishment, configuration, maintenance and release of point-point radio bearers. , Mobility functions along wth cell addition and cell release , UE measurement reporting, control of UE reporting, UE based mobility and NAS direct message transfer to/from NAS from/to UE

2.1.2 PHY layer and numerology

In the frequency domain, resources are grouped in units of 12 subcarriers, such that one unit of 12 subcarriers for a duration of one slot is called a resource block (RB). A collection of RBs can be grouped to form a resource block group (RBG), which is the smallest scheduling unit. The smallest unit of a resource is a resource element (RE) that consists of one subcarrier for a duration of one OFDM symbol. The time domain is divided into frames, each frame is divided again into 10 subframes. The number of slots (aka TTI) in each subframe, the duration of the slot, and the bandwidth of RB is given in figure 2 [1]

Index	SCS (kHz)	# slots	slot duration (ms)	RB bandwidth (kHz)
0	15	1	1	180
1	30	2	0.5	360
2	60	4	0.25	720
3	120	8	0.125	1440
4	240	16	0.0625	2880

Figure 2: Numerology settings in 5G

Each subframe consists of 2 slots. Each slot can have either 14 (normal CP) or 12 (extended CP) OFDM symbols. All the subcarrier spacing options have 14 OFDM symbols. For $\mu = 0$ there is 1 slot per subframe , for $\mu = 1$ there are 2 slots per subframe, for $\mu = 2$ there are 4 slots per subframe and so on. Number of slots per frame is ten times the number of slots per subframe. Hence for $\mu = 2$, there are 40 slots per frame. The UEs in the system compete for the available resources, according to their traffic use case. A UE is eligible for scheduling if it has data in the buffer and is not retransmitting in the current slot. i.e., if it is not associated with a HARQ process in progress. Only those UEs that are eligible (active) will be considered to compete for the available RBGs. Scheduling a transmission can be prioritized depending on the traffic use case. Subcarrier spacing(SCS) in 5G can be $15*2^{\mu}$, where μ=0, 1, 2, 3 or 4 is a system numerology. Such numerology gives the scheduler provision to assign resources with varying time intervals depending on the services and use cases.
The frame structure, resource grid and basic terminologies are illustrated in figure 3

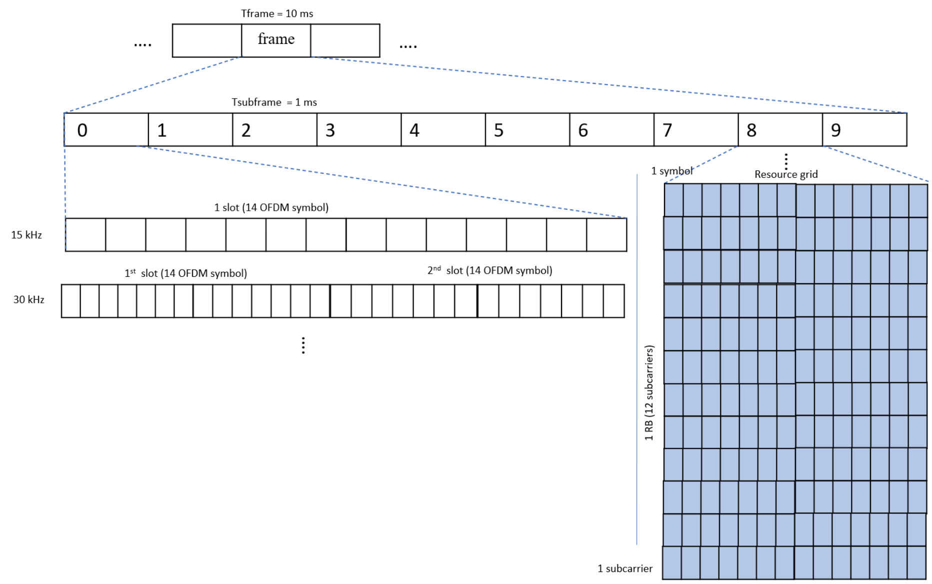

Figure 3: 5G NR frame structure, resource grid and basic terminologies

2.2 5G Scheduler Implementation in MATLAB^{TM}

The 5G scheduler simulator models the following.[5]

- Slot-based and symbol-based DL and UL scheduling.
- Non-contiguous allocation of frequency-domain resources in terms of resource block groups (RBGs).
- Configurable subcarrier spacing (SCS) resulting in different slot durations.
- Asynchronous adaptive hybrid automatic repeat request (HARQ) mechanism in UL and DL.
- Multiple logical channels (LCHs) to support different kind of applications.

The control packets required are assumed to be sent out of band without the need of resources for transmission. The control packets are UL assignment, DL assignment, buffer status report (BSR), and PDSCH feedback. To model PHY, a probability-based passthrough physical layer (PHY) without any signal processing is used. The demodulation reference signal (DM-RS) is not modeled. However, one symbol is kept unused for it in the PUSCH and PDSCH assignments. Simulation is run slot by slot. In each slot, these operations are executed:

- Run the MAC and PHY layers of gNB.
- Run the MAC and PHY layers of UEs.
- Layer specific logging and visualization.
- Advance the timer for the nodes. Every 1 ms it also sends trigger to application and RLC layers. Application layer and RLC layer execute their scheduled operations based on 1 ms timer trigger.

2.3 Reinforcement Learning [1], [2]

Reinforcement Learning (RL) is a principled mathematical framework for experience driven autonomous learning. RL develops a policy that will maximise the expected return with trial and error. In value based methods, we formulate return G_t (accumulated discounted reward) and the value function as the expected return $V(s)$

$$G_t = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k r(s_{t+k}, a_{t+k}) | s_0 = s_t\right] \tag{1}$$

where the discount factor $\gamma < 1$

$$V(s) = \mathbb{E}\left[G_t | s_t = s\right] \tag{2}$$

It is now basically an optimisation problem of maximizing the value function. Quality (Q)-function is similar to the value function except that the first action is not chosen according the control policy.

$$Q(s,a) = \mathbb{E}\left[G_t | s_t = s, a_t = a\right] \tag{3}$$

The Bellman's optimality equation for $Q^*(s)$ is given as

$$Q^*(s,a) = r(s,a) + \gamma \sum_{s' \in \mathcal{S}} p_{ss'}(a) \max_{a'} Q^*(s', a') \tag{4}$$

The optimal policy $\pi^*(s)$, therefore, can be obtained as

$$\pi^*(s) = \arg\max_{a \in \mathcal{A}} Q^*(s,a), \quad \forall s \in \mathcal{S} \tag{5}$$

In bootstrapping the value function estimation is done with dynamic programming concepts (SARSA - On policy, Q-learning -off policy) and policy evaluation with TD errors. In SARSA based policy, the updation equation is given by

$$Q^\pi(S_t, A_t) = Q^\pi(S_t, A_t) + \alpha[R_t + \gamma Q^\pi(S_{t+1}, A_t) - Q^\pi(S_t, A_t)] \tag{6}$$

where the policy π is an ϵ - greedy policy.

In sampling based approach, we have Monte Carlo methods to estimate expected return based on averaging over multiple rollouts of policy. Policy-based methods / policy search can either be gradient-based or gradient-free. Model-free RL can directly learn value function from the interactions with the environment whereas model-based RL Can simulate transitions using the learned model.

3 A RL based Scheduler for 5G Uplink

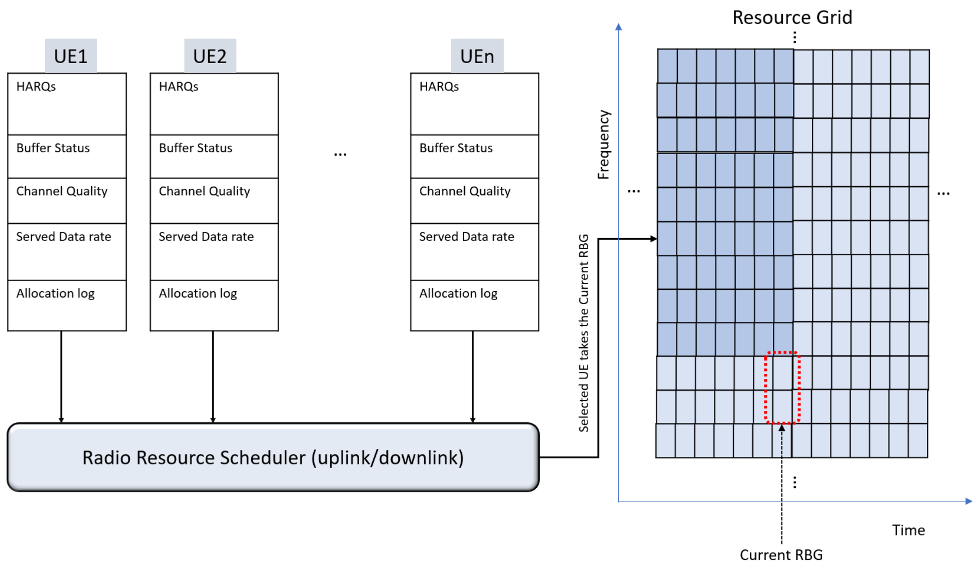

Figure 4: Radio resource scheduler

3.1 Queueing Model and the Slot based Scheduling Problem

As shown in Figure 4, a set of UEs in the system compete for the available resources. The gNB collects the context which includes the information of channel feedback, buffer status, HARQs, and allocation log. The radio resource scheduler runs at the gNB at every (or kth) slot and makes scheduling decision to share the available RBGs between active UEs, based on the collected information. The slot based scheduling approach views the problem of resource allocation as the filling of the resource grid by deciding to which UE the current RBG in the current slot can be assigned so that we have an optimal trade-off between the KPIs. However, not all users can be considered for scheduling at every RBGs. Only those UEs that are eligible (active) will be considered and allowed to compete for the RBGs under consideration in the slot. The eligibility of a UE is decided based on the non-emptiness in buffer and not retransmitting status in the current slot, i.e., if it is not associated with a HARQ process in progress.

The implemented policy for scheduling based on the backpressure weights the eligible UEs with the product of their buffer status and the achievable data rate and selects the UE with maximum weight in the current slot. The intuition behind this policy is that we should try to serve the UE with maximum queue length and which also has a good rate so that the queue length would come down from a large value. The time average BSR of UEs and the average BSR across UEs as an indication of the delay experienced by the UEs in the uplink is considered as the performance metric. The backpressure based policy helped to understand that the buffer status and the achievable data rates are good indicators to define the state of the UEs.

3.2 A Simplified Slot based RL Agent

A slot-wise RL based scheduler implementation hs been carried out. The scheduler makes the decisions sequentially in both frequency and time. i.e., for each RGB in each slot. To model the problem of scheduling to fit into the framework of RL, we describe the state, action and reward in following manner, keeping in mind, the results obtained from the backpressure based policy implementation.

State description : Both the BSR and the achievable data rate of each UE(quantized) combinedly models the state-space.

Actions : The selection of a UE among the eligible ones to assign in a slot.

$$\textbf{Reward} = -\sum_{i=1}^{NumUEs} BSR_i$$

We have implemented the SARSA based policy as described in equation (6) . The pseudocode for the same is described below.

```
# SARSA based Policy

#Initialize the parameters
# finding out the current state
State space ← bufferStatus , achievable data rate of all the eligible UEs
Actions  ←  which UE to assign in a slot
Reward = -Sum_{i}(BSR_i)
Quantize the state space to reduce complexity
while i < length(UEs)
    index = find(eligibleUE == i, 1)
    if index is not empty
       current state  ←  bufferStatus and achievable data rate of UE indexed by index
    end
    i = i+1
end

 #Implementation of epsilon greedy policy
if rand() < epsilon
   select  UE randomly from the eligible ones
else
     select the UE which maximizes the Q function corresponding  to the current state
end

# Q function updation - SARSA
Q function(last state) = Q function(last state)  + ...
alpha*(last reward + gamma*Q function(current state) - Q function(last state) )
laststate = current_state
last action = selectedUE
lastreward = -sum(obj.weight.*current_state
```

3.3 Comparison Studies

The resource grid allocation allocation for the UEs by the SARSA based scheduler is given in figure 5.

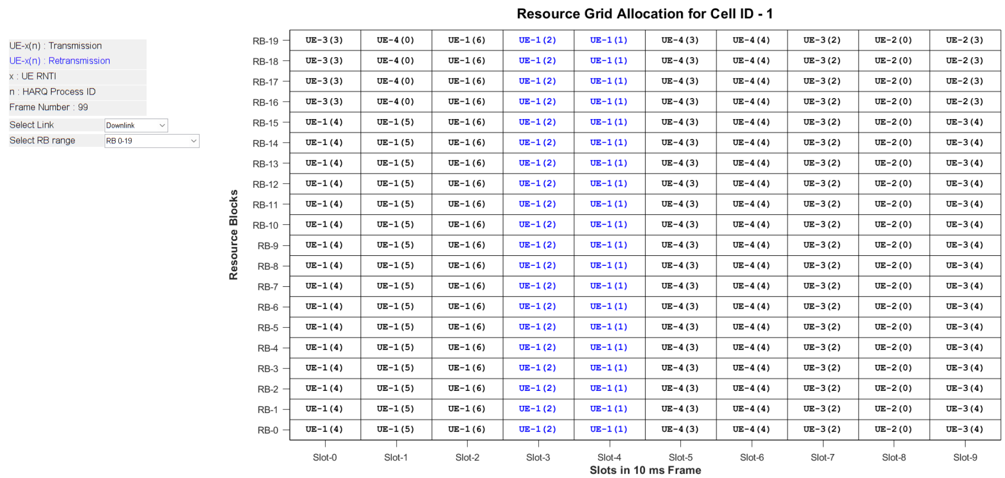

Figure 5: Resource allocation in SARSA based policy

The buffer status plots of the four UEs in under different scheduling algorithms are given in figure 6, 7, 8 & 9

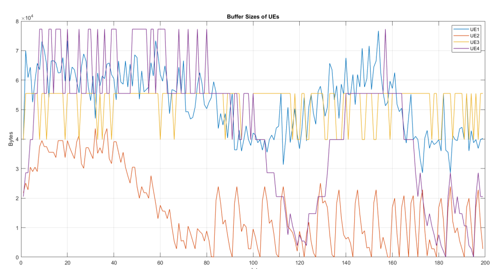

Figure 6: Buffer status of UEs under Round-Robin policy

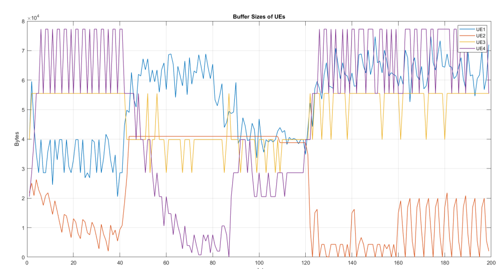

Figure 7: Buffer status of UEs under Best CQI policy

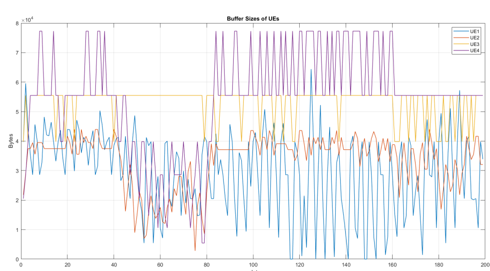

Figure 8: Buffer status of UEs under backpressure based policy

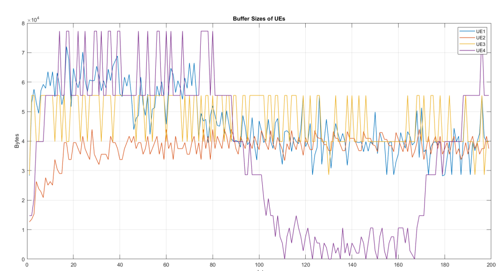

Figure 9: Buffer status of UEs under SARSA based policy

The comparison of performance of different scheduling algorithms in therms of the time averaged BSR of UEs and the averaged BSR across UEs is given in figure 10.

Performance Metric		RR	BestCQI	Back Pressure	RL SARSA (alpha = 0.1)	RL SARSA (alpha = 0.05)
Time Average BSR of UEs	UE 1	52651.43719	52621.09548	29302.37186	59273.07538	47041.88945
	UE2	16598.8191	21744.9196	33440.91457	12142.44724	37584.14573
	UE3	52563.08543	48165.62312	52956.45226	52663.96482	47695.48744
	UE4	45444.86935	49845.55276	54557.80402	60127.23618	37179.00503
Average BSR across UEs		**4.18E+04**	4.31E+04	**4.26E+04**	4.61E+04	**4.24E+04**

Figure 10: Performance metric comparison

From the comparison chart, it is evident that the best performance in terms of the minimum average BSR across UEs is achieved for round-robin algorithm (4.18E+04 Bytes). The second best performance is for the SARSA based policy with the parameter $\alpha = 0.05$ (4.24E+04 Bytes) and the backpressure based policy stands in the third place. The parameter tuning has not been carried out for the SARSA based policy. We are expecting better result after performing the same, which is part of future work.

4 Conclusion and Future Work

We have limited the number of UEs in the simulation as 4 considering the simulation time constraint. Extending the simulation with more number of UEs to model more realistic scenario is intended for future work. Also including 5G Toolbox™ PHY layer instead of the probabilistic pass-through modelling for the physical channel will be helpful in modelling more realistic traffic / link conditions. Simulations with large simulation times will be helpfull in analysing the performance of the RL algorithms in an evolution basis as time progress. Using backpressure's Q as the initial value for the RL algorithm would improve the learning rate and performance. This is also a part of future work. Parameter tuning and detailed simulations are required.

References

[1] Faroq Al-Tam, Noélia Correia, and Jonathan Rodriguez. Learn to schedule (leasch): A deep reinforcement learning approach for radio resource scheduling in the 5g mac layer. *IEEE Access*, 8:108088–108101, 2020.

[2] Arjun Anand, Gustavo de Veciana, and Sanjay Shakkottai. Joint scheduling of urllc and embb traffic in 5g wireless networks. *IEEE/ACM Transactions on Networking*, 28(2):477–490, 2020.

[3] Ioan-Sorin Comşa, Sijing Zhang, Mehmet Emin Aydin, Pierre Kuonen, Yao Lu, Ramona Trestian, and Gheorghiţă Ghinea. Towards 5g: A reinforcement learning-based scheduling solution for data traffic management. *IEEE Transactions on Network and Service Management*, 15(4):1661–1675, 2018.

[4] About MONET group. 5g-lena simulator. Available at `https://https://5g-lena.cttc.es/#:~:text=The\%20simulator\%20is\%20the\%20natural,simulator\%20is\%20dated\%20February\%202019.`

[5] Mathworks. Nr fdd scheduling performance evaluation. Available at `https://in.mathworks.com/help/5g/ug/nr-fdd-scheduling-performance-evaluation.html`.

[6] Satishkumar Mudaliar. 5g nr numerology – subcarrier spcaing (scs). Available at `https://www.techplayon.com/5g-nr-numerology-subcarrier-spcaing-scs/` (2020/09/20).

[7] RF Wireless World. 5g protocol stack — 5g layer 1, 5g layer 2, 5g layer 3. Available at `https://www.rfwireless-world.com/Terminology/5G-Protocol-Stack-Layer-1-Layer-2-and-Layer-3.html`.