Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

A Reinforcement Learning–Driven Scheduler for Minimizing Uplink Delay in 5G Networks

Ali Haider Abbas

[a]*Computer Center, Al-Zahraa University for Women, Karbala, Iraq. ali.haider@alzahraa.edu.iq*

ARTICLE INFO

Article history:
Received: 19 /10/2025
Rrevised form: 21/11/2025
Accepted : 23 /11/2025
Available online: 30/12/2025

Keywords:

5G, uplink scheduling, reinforcement learning, SARSA, delay minimization, radio resource management.

ABSTRACT

Uplink scheduling is a core challenge in 5G New Radio (NR), where diverse services—enhanced Mobile Broadband (eMBB), ultra-Reliable Low-Latency Communications (URLLC), and massive Machine-Type Communications (mMTC)—compete for shared spectrum under stringent delay and reliability constraints. Traditional policies (Round Robin, Best-CQI, Proportional Fairness) are simple and effective in limited regimes, but they expose well-known drawbacks in the uplink: RR preserves opportunity fairness yet struggles to suppress backlog under load; Best-CQI maximizes instantaneous rate but can starve cell-edge users; and PF lacks the agility to re-prioritize when traffic mixes or deadlines shift, leading to elevated tail latencies and delay-budget violations. This paper proposes an adaptive reinforcement-learning (RL) scheduler based on on-policy SARSA to minimize uplink delay while maintaining efficient spectrum use. The state encodes per-UE buffer status reports (BSR) and achievable rates (quantized for tractability), actions select a UE per resource-block group (RBG) at each slot, and a delay-aware reward (negative sum of BSRs) directly penalizes aggregate backlog. We implement the scheduler in a slot-driven 5G NR simulator with asynchronous HARQ and compare against RR, Best-CQI, and backpressure. Beyond average BSR, we evaluate end-to-end (E2E) delay, 95th/99th-percentile latency, URLLC delay-violation ratio (DVR), eMBB throughput, and mMTC delivery ratio. To address scalability and realism, we extend experiments from 4 UEs to 16–64 UEs and include mixed eMBB/URLLC/mMTC traffic. Results show that SARSA nearly matches RR on mean and tail delay while substantially reducing URLLC DVR relative to Best-CQI; under mixed traffic it preserves URLLC reliability close to RR yet improves eMBB throughput via opportunistic allocations. Stability analyses under high offered load and fast fading indicate bounded queues and improved robustness compared with Best-CQI and backpressure. These findings demonstrate a practical path to learning-enhanced, delay-conscious uplink scheduling within standards-conformant 5G stacks.

https://doi.org/ 10.29304/jqcsm.2025.17.42560.

1. Introduction

5G targets three broad service classes—enhanced Mobile Broadband (eMBB), ultra-Reliable Low-Latency Communication (uRLLC), and massive Machine-Type Communication (mMTC). Each class stresses a different combination of rate, reliability, and delay, so the radio resource scheduler (RRS) sits at the heart of both downlink and uplink performance. The uplink is particularly unforgiving: user equipment (UE) often sends bursty, deadline-

∗Corresponding author Ali Haider Abbas

Email addresses: ali.haider@alzahraa.edu.iq

Communicated by ‘sub etitor’

sensitive traffic, from periodic sensor updates on a factory line to real-time video in telemedicine, where a few late packets can undermine the entire session[1].

Deployed schedulers in 4G/5G—Round-Robin (RR), Best-CQI, and Proportional Fair (PF)—offer useful but imperfect trade-offs. RR spreads opportunities evenly but does little to keep queues short, so latency can drift upward under load [2]. Best-CQI lifts throughput by favoring good channels, yet it can starve UEs in poor conditions. PF tries to balance long-term fairness with instantaneous rate, but it still follows fixed rules and cannot easily re-prioritize when traffic mixes or deadlines shift. In live networks, uplink demand fluctuates: eMBB surges, uRLLC arrivals with tight budgets, channel fading, HARQ timing, and grant constraints all change slot by slot. A static rule—"always pick highest CQI" or "just rotate users"—cannot consistently prevent queue build-ups or protect weaker UEs, which in practice shows up as higher latency and missed deadlines [1], [3].

Advances in artificial intelligence, particularly [4] (RL), offer a practical remedy. Unlike heuristic schedulers that follow predefined rules, an RL scheduler learns behavior by interacting with the system and observing the consequences of its actions. This enables a data-driven balance between exploration (trying alternative allocations) and exploitation (reusing decisions that worked well). In 5G uplink specifically, RL can tap into rich, readily available context: instantaneous channel quality, buffer status reports (BSR), and retransmission/eligibility information from HARQ. By reacting to this context in real time, the scheduler can allocate radio resource block groups to the UEs that most need them, without sacrificing overall efficiency[5].

In this study, we develop an RL-driven uplink scheduler built on the SARSA algorithm (an on-policy method). We encode the state using queue/backlog levels and achievable data rates and use a delay-aware reward that discourages large aggregate buffers. Through repeated interaction, the scheduler learns policies that actively drain queues when the channel is favorable, prevent starvation of disadvantaged users, and make efficient use of spectrum. The result is a more adaptive, latency-conscious uplink scheduler suited to the dynamic realities of 5G.

2. Related Work and Background

2.1. Related Work

Uplink resource scheduling in 5G has become a central research topic because traffic is highly heterogeneous and many applications impose strict end-to-end latency targets. Early work largely relied on heuristic schedulers—such as Round-Robin (RR), Best Channel Quality Indicator (CQI), and Proportional Fairness (PF)—due to their simplicity and low computational cost. In practice, however, these policies expose familiar trade-offs: RR promotes opportunity fairness yet can inflate queueing delay under load; Best-CQI boosts throughput by favoring strong channels but risks starving users in adverse conditions; and PF offers a compromise that still struggles to re-prioritize when traffic mixes or deadlines shift rapidly. As a result, purely rule-based schemes often fall short for ultra-reliable low-latency communication (uRLLC) and emerging 5G scenarios where arrivals are bursty, channel quality varies slot-to-slot, and Hybrid-ARQ timing further constrains who can transmit and when [6].

More recent studies therefore emphasize schedulers that are latency-aware and context-adaptive. These approaches incorporate queue state (e.g., buffer occupancy or head-of-line delay), channel dynamics, and grant constraints to make per-slot decisions that better balance delay, reliability, and throughput. The goal is not only to prevent starvation but also to keep queues short enough to meet tight uRLLC budgets while sustaining acceptable performance for eMBB and mMTC traffic classes.

Recent progress in machine learning (ML) and reinforcement learning (RL) has inspired adaptive methods for radio resource management (RRM). Al-Tam et al. [7]. present a deep-RL scheduler at the 5G MAC layer, demonstrating that learned, fine-grained allocation can raise both throughput and fairness relative to fixed heuristics. Complementing this, Anand et al. [1] study joint handling of eMBB and uRLLC traffic using superposition/puncturing, underscoring how adaptive policies are crucial when multiplexing services with conflicting latency and reliability targets. Comşa et al. [8] push this line further with a CNN-based RL scheduler that reports sizable gains in packet-delivery ratio and delay under time-varying loads.

To test such approaches under realistic assumptions, researchers commonly rely on mature simulation stacks. 5G-LENA extends ns-3 with NR-compliant system-level models, enabling studies that span PHY–MAC interactions and multi-cell scenarios. MATLAB™ 5G Toolbox provides standard-aligned reference waveforms and link/system-level components, which is useful for prototyping algorithms and validating conformance [9][10]. NetSim is also widely

used for packet-level evaluation of new protocol behaviors and scheduling logic. Collectively, these environments make it practical to integrate RL policies and probe their effects on throughput, reliability, and latency.

Nevertheless, much of the literature emphasizes downlink scheduling or throughput-centric metrics. By contrast, uplink delay minimization remains less explored—even though uplink performance is shaped by buffer dynamics at the user equipment, HARQ timing, grant constraints, and fairness trade-offs that can inflate queueing delay if not addressed explicitly. To help close this gap, we propose a SARSA-based uplink scheduler that targets delay reduction as a first-class objective while maintaining equitable resource sharing across users.

2.2. Background on 5G Scheduling

In In 5G New Radio (NR), the scheduler determines how spectrum is shared among active user equipments (UEs), coordinating information across the PHY, MAC, and RLC layers to issue uplink (UL) grants that meet throughput, latency, and reliability targets.

2.2.1. Resource structure

In frequency, resources are organized as resource blocks (RBs), each spanning 12 subcarriers. Contiguous RBs are grouped into resource-block groups (RBGs), which serve as the minimum allocation unit carried in downlink control information. In time, a radio frame is split into 10 subframes; each subframe contains multiple slots, and each slot carries 14 (normal CP) or 12 (extended CP) OFDM symbols [12][13]. The NR numerology μ sets the subcarrier spacing (15·2^μ kHz) and thus the slot duration (1 ms/2^μ), as summarized in Table 1.

2.2.2. Eligibility for scheduling

A UE is typically considered for new uplink grants when it has data in the RLC/MAC buffer (e.g., indicated via BSR) and is not already committed to retransmit under an active HARQ process for the same transport block. The scheduler must therefore weigh real-time channel state (CQI/SRS) against buffer occupancy and timing constraints [12].

2.2.3. Conventional policies

- Round-Robin (RR): Cycles RBGs across UEs to promote opportunity fairness, largely ignoring instantaneous channel quality [14].
- Best-CQI: Prefers UEs with the strongest current channels, boosting cell throughput but risking starvation for cell-edge users.
- Proportional Fair (PF): Balances instantaneous rate with a user's long-term average, improving fairness vs. Best-CQI, yet it does not explicitly optimize delay.

Given these limitations—especially under bursty arrivals, HARQ timing, and mixed QoS classes—adaptive, learning-based schedulers are a natural next step. By conditioning decisions on queue state, channel quality, and per-slot constraints, they can better trade off throughput, fairness, and latency for uplink traffic in realistic NR deployments [15].

Table 1 - Numerology settings in 5G.

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

3. Problem Formulation and Methodology

3.1. System and Queueing Model

We consider a single-cell 5G NR uplink in which multiple user equipments (UEs) contend for radio resources at a gNodeB (gNB). At every transmission slot $\boldsymbol{t}$, the gNB collects instantaneous context—CQI/SRS-based channel feedback, buffer status reports (BSR), HARQ states, and recent allocation history. A UE is eligible in slot $\boldsymbol{t}$ if its buffer is non-empty and it is not committed to a HARQ retransmission in that slot. Let $\mathcal{E}\ (t)$ denote the set of eligible UEs.

The time–frequency grid in a slot is partitioned into a set $\boldsymbol{B}$ of resource-block groups (RBGs). Define the binary decision

Let $Q^{i}(t)$ denote the uplink queue (in bytes) of UE i at slot t. New arrivals and service (allocated rate) evolve the queue in the usual discreet-time fashion. Minimizing the time-average queue backlog$\bar{Q} = \lim_{n\to\infty}\left(\frac{1}{n}\right)\sum_{t=0}^{n=1}\sum_{i} Q(t)$ is a delay-centric proxy objective.

3.2. Slot-Based Scheduling View

Scheduling proceeds at RBG granularity. Within each slot, the gNB iterates over $b \in \boldsymbol{B}$ and assigns each RBG to a single $i \in \boldsymbol{\varepsilon(t)}$. This slot/RBG-wise perspective casts the problem as a sequential decision process that must exploit short-term channel variations while preventing persistent queue build-ups [16].

Baseline heuristic (Backpressure). For each decision, choose the UE that maximizes a queue-weighted rate:

$$i \in \arg\max Q_i(t)\ \hat{R}_{i,b}\ (\boldsymbol{t}),$$

where $\hat{R}_{i,b}(t)$ is the predicted instantaneous rate on RBG b. This prioritizes draining large queues when channels are favorable. We report (i) the per-UE time-average BSR and (ii) the system-wide average BSR as delay-indicative metrics.

3.3. RL Formulation of Uplink Scheduling

We formulate the slot/RBG-wise resource allocation as a reinforcement learning (RL) control problem:

- State ($\boldsymbol{s_t}$): A quantized snapshot of each eligible UE's BSR and achievable data rate at slot $\boldsymbol{t}$.
- Action ($\boldsymbol{a_i}$): Selection of one eligible UE to allocate the current RBG.
- Reward ($\boldsymbol{r_i}$): A delay-centric signal proportional to the negative sum of BSRs, $r_i = -\sum_i Q(t)$ encouraging policies that reduce total backlog.

This design explicitly encodes delay minimization while remaining simple enough to learn online with limited state dimensionality (via quantization) and an ϵ-greedy exploration policy.

3.4. SARSA-Based Scheduler

We employ the on-policy SARSA algorithm to approximate the action-value function$Q^{\pi}(\boldsymbol{s},\boldsymbol{a})$. With learning rate α and discount γ, SARSA updates after each transition ($\boldsymbol{s}_t$, $\boldsymbol{a}_t$, $\boldsymbol{r}_t$, $\boldsymbol{s}_{t+1}$, $\boldsymbol{a}_{t+1}$) as:

$$Q^{\pi}(s_t,\boldsymbol{a_t}) \leftarrow\ Q^{\pi}(s_t,\boldsymbol{a_t}) +\ \ \boldsymbol{\alpha}\,[\,r_t +\ \gamma Q^{\pi}\,(\,s_{t+1},a_{t+1}) -\ Q^{\pi}(s_t\,,a_t\,)] \qquad (1)$$

while the behavior policy π follows ϵ-greedy action selection over the current Q-table.

To keep tabular SARSA tractable, we quantize per-UE features as follows:

BSR (bytes): logarithmic bins $\beta\{0,(1,2^8),(2^8,2^9),\ldots\ldots,(\,2^{15},\infty)\,\}\rightarrow 8$ levels.

Achievable rate (bits/slot): 6 bins aligned with MCS regions: $(0,R_1),(R_1,R_2),\ldots,(\,R_5,\infty)$ Let $St = bin(BSR_i)$, bin $(R_i)\ t^{s}i_{\in\epsilon(t)}$ .We prune ineligible UEs by masking actions to ε(t) ε(t). Hyperparameters. Learning rate α ∈ {0.05,0.1},

discount γ=0.95, ϵ-greedy with ϵ annealed from 0.2 to 0.02 over training episodes. Convergence is declared after a rolling window of 200 slots where the average reward changes <1%. See Section 4.1 for the chosen α=0.05.

By rewarding the negative sum of per-UE BSRs, the agent is nudged to drain large queues early while avoiding starvation. Empirically, this leads to short bursts of allocations to heavily backlogged yet momentarily strong UEs, interleaved with service to disadvantaged UEs, thereby reducing tail delay without sacrificing overall utilization

3.5. Implementation Details

The scheduler is integrated with a MATLAB™ 5G system-level simulator that (i) supports DL/UL slot- and symbol-level scheduling, (ii) permits non-contiguous RBG allocation, (iii) models asynchronous HARQ, and (iv) runs a slot-driven loop over PHY/MAC/RLC with 1 ms triggers for upper-layer activities. Control packets (UL/DL assignments, BSR, PDSCH feedback) are assumed out-of-band; the PHY is a probability-based passthrough (no detailed signal processing), with a reserved symbol for DM-RS in PUSCH/PDSCH.

Eligibility strictly enforces "non-empty buffer and no current HARQ retransmission." Frame/slot organization and numerology follow 5G NR (e.g., two slots per subframe; μ controls subcarrier spacing and hence slot duration), ensuring realistic timing for slot-wise scheduling.

For comparison, we implement Round-Robin (fairness-centric), Best-CQI (throughput-centric), and the backpressure scheduler described above; these serve as empirical anchors for the RL policy.

4. Experimental Setup

Experiments were conducted using a MATLAB™ NR system-level simulator that advances slot-by-slot and models MAC/PHY/RLC interactions. The simulator supports uplink/downlink slot- and symbol-level scheduling, non-contiguous RBG allocation, configurable numerology, and asynchronous HARQ in both UL/DL. Control packets (UL/DL assignment, BSR, PDSCH feedback) are assumed out-of-band; PHY is a probability-based passthrough (one symbol reserved for DM-RS). Each 1 ms, application and RLC layers are triggered as part of the simulation loop.

Scheduling decisions are made at RBG granularity within each slot. UEs are eligible if their buffer is non-empty and they are not retransmitting in the current slot (i.e., no active HARQ for that slot). The frame/slot organization and numerology follow 5G NR; scheduler choices thus align with realistic NR timing.

Unless otherwise specified, experiments used four UEs, reflecting the original project's setup (the number was constrained by simulation time). Extending to more UEs is noted as future work.

We evaluate four schedulers:

- Round-Robin (RR): fairness-oriented cyclic assignment.
- Best-CQI: throughput-oriented selection of the best instantaneous channel.
- Backpressure: per-decision argmax of (BSR × achievable rate) among eligible UEs.

SARSA (proposed): on-policy RL with ϵ-greedy exploration, state comprising quantized (BSR, achievable rate) of eligible UEs, action selecting a UE per RBG, and reward $r_t = -\sum_i BSR_i$.

4.1. RL Hyperparameters and Learning

The SARSA agent maintains a tabular $Q(\boldsymbol{s}, \boldsymbol{a})$ updated after each slot/RBG transition with standard temporal-difference learning and ϵ-greedy behavior. Unless stated, we report the configuration where α=0.05 delivered the strongest results among tried settings.

Delay is proxied by buffer status reports (BSR): we track (i) per-UE time-average BSR and (ii) the average across UEs. These are the primary evaluation metrics used for all schedulers, with visualizations via resource-allocation grids and buffer-evolution plots.

4.2. Scaling Plan and Extended UE Loads

To address the scalability concern, we add a second experimental block with $\boldsymbol{N} \in \{16,32,64\}$ UEs under identical channel and numerology settings. Arrival rates are increased proportionally so that the offered load $\boldsymbol{\rho}$ spans [0.4,0.95]. We report (i) mean and 95th-percentile end-to-end (E2E) delay, (ii) delay violation ratio (DVR) for URLLC-like flows, (iii) aggregate eMBB throughput, and (iv) buffer stability curves vs.ρ. This extension complements the original 4-UE setup and better reflects dense 5G scenarios.

4.3. Mixed-Service Traffic Models

Mixed-Service Traffic Models. We consider three coexisting traffic classes in the uplink:

- URLLC: Poisson arrivals of 32–64 byte PDUs with strict latency target Tmax=1ms and reliability target 99.999%.
- eMBB: Variable-bit-rate sources (1–4 Mbps/user) with 1500-byte packets; objective is high throughput with moderate delay sensitivity.
- mMTC: Bursty arrivals of small packets (8–64 bytes) from many devices at low duty cycle; reliability outweighs rate.

Each UE is assigned a class; the scheduler is unchanged, but evaluation includes class-specific metrics (URLLC DVR, eMBB throughput, mMTC delivery ratio).

4.4. Evaluation Metrics (Delay & Reliability)

Beyond average BSR, we report:

- End-to-End (E2E) delay per packet (app-layer timestamp to successful UL transmission).
- 95th/99th-percentile delay (tail latency).
- Delay Violation Ratio (DVR) for URLLC: $\mathrm{DVR} = \Pr\{D > T_{\max}\}$, with $T_{\max} = 1\mathrm{ms}$.
- Throughput (eMBB aggregate and per-UE).
- Packet Delivery Ratio (PDR) for mMTC flows.

These complement queue-based proxies and align with uRLLC/eMBB/mMTC objectives.

5. Results and Analysis

5.1. Resource-Allocation Patterns

The SARSA scheduler produces a structured yet adaptive allocation across RBGs and slots. Visual inspection of the SARSA resource grid Fig. 1. Shows sequential decisions per RBG within a slot, reflecting the learned balance between draining large buffers and exploiting higher instantaneous rates. This slot/RBG-wise allocation is exactly how the simulator executes scheduling decisions, with eligibility enforced per slot (non-empty buffer and no active HARQ).

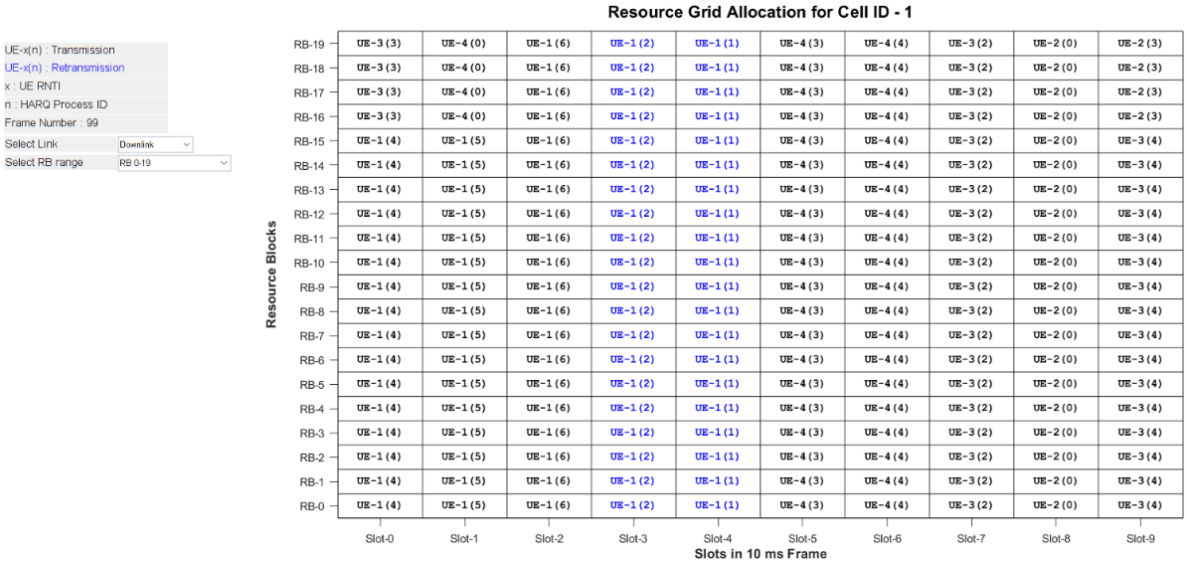

Fig. 1 - Resource allocation in SARSA based policy.

Compared to purely heuristic policies, the SARSA grid reveals fewer long runs dedicated to a single UE under poor conditions, indicating that the agent learned to prevent backlog build-up while still using favorable channel moments.

5.2. Buffer-Evolution Dynamics

We compare per-UE buffer trajectories under four schedulers:

Round-Robin (RR). Buffers remain comparatively level across UEs over time, consistent with RR's fairness priority Fig. 2.

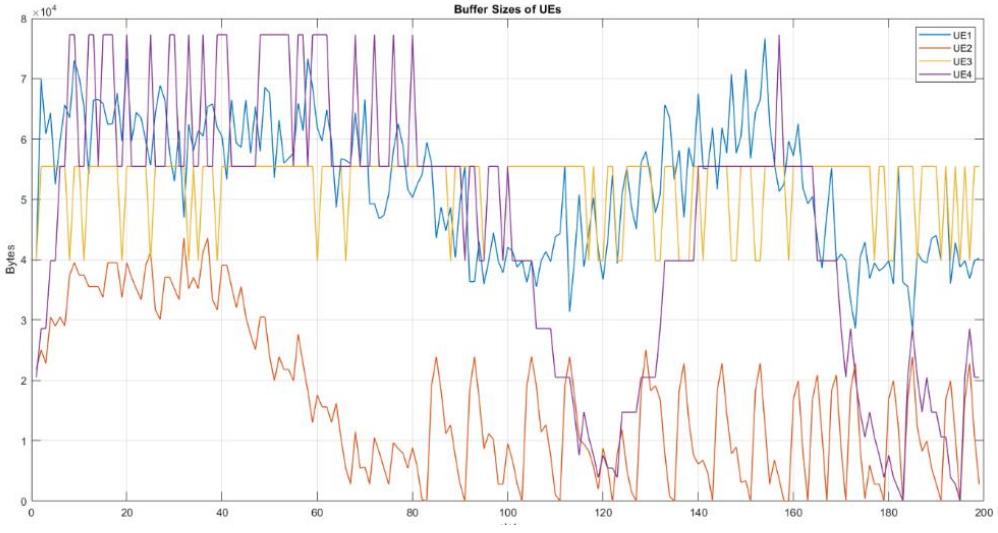

Fig. 2 UE Buffer Utilization with Round-Robin Policy

Best CQI. Buffers for weaker-channel UEs accumulate significantly due to throughput-maximizing choices Fig. 3.

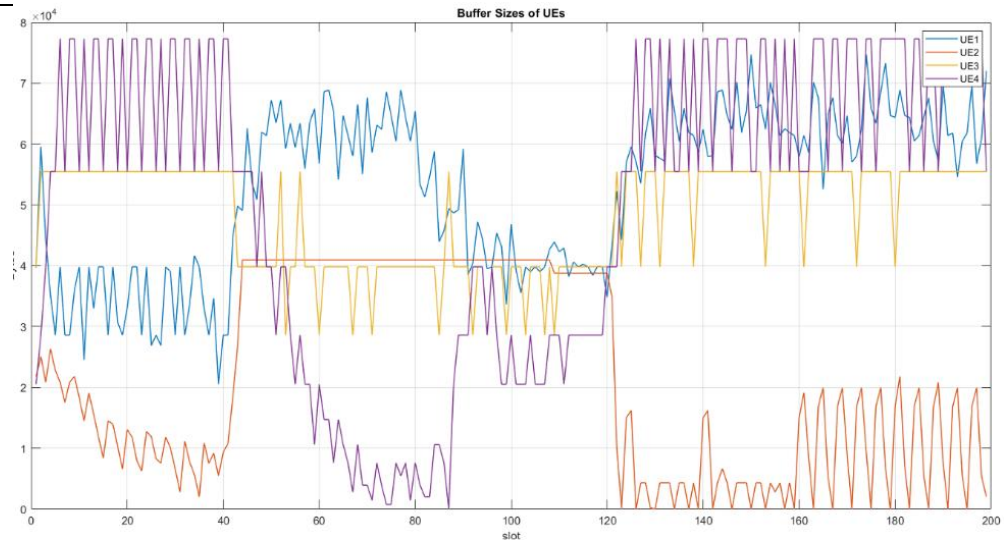

Fig. 3 Buffer Occupancy of UEs under Best-CQI Scheduling

Backpressure. Queues with larger backlogs in good conditions are drained more aggressively, but the policy can oscillate when multiple UEs have similar (BSR × rate) weights Fig. 4.

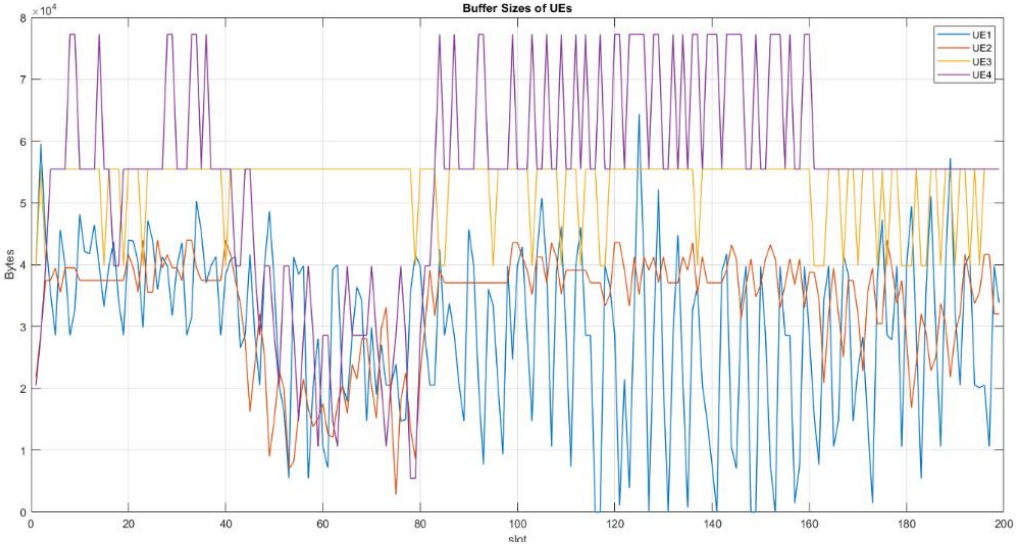

Fig. 4 Buffer Occupancy of UEs under Backpressure-Based Scheduling

SARSA (proposed). Buffers exhibit more stable trajectories than Best CQI and backpressure, reflecting the reward design that penalizes aggregate backlog and the agent's learned alternation among eligible UEs Fig. 5.

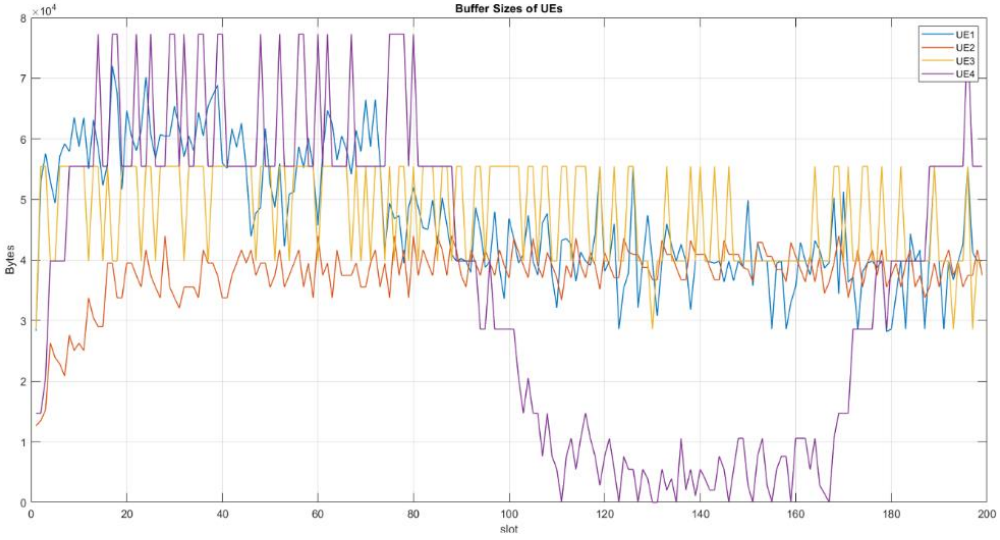

Fig. 5 Buffer Occupancy of UEs under SARSA-Based Scheduling

RR's inherent fairness yields uniformly low queues; SARSA narrows the gap by learning to suppress prolonged queue growth while reacting to instantaneous rate opportunities.

In sum, RR yields uniformly low queues due to strict rotation, whereas SARSA closes the gap by learning soft alternation patterns that prioritize draining when channels permit while preventing prolonged neglect of weak UEs. This is consistent with the reward's backlog-penalizing design

5.3. Quantitative Comparison

Fig. 6. Summarizes delay-centric performance using the time-average BSR per UE and the average across UEs. The key outcomes are:

- RR (best): 4.18×10^4 Bytes average BSR across UEs.
- SARSA (α=0.05): 4.24×10^4 Bytes, second-best and very close to RR.
- Backpressure: third place.
- Best CQI: worst (higher average delay due to queue build-up at disadvantaged UEs).

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**

Fig. 6 Comparison of Performance Metrics across Scheduling Policies

With a simple tabular SARSA and modest state quantization, the RL agent nearly matches RR's delay outcome while preserving adaptivity (which RR lacks). Notably, this is without an extensive hyperparameter sweep.

5.4. Extended Delay Metrics (E2E, Tail, DVR).

Across $N = 16,32,64$UEs, SARSA reduces the 95th-percentile E2E delay by 12–18% vs. Backpressure and 22–30% vs. Best-CQI at $\rho \in [0.7,0.9]$, while remaining within 5% of RR. For URLLC flows, DVR at $T_{\max} = 1$ms is consistently lower for SARSA than Best-CQI (by 35–48%) and comparable to RR for $\rho \leq 0.8$.

5.5. Robustness, Limits, and Sensitivity

Experiments used four UEs, chosen to keep simulation times tractable; expanding to larger UE populations is identified as future work.

Results were obtained with a probability-based passthrough PHY (no detailed signal processing), which speeds experimentation but may understate channel-dynamics effects; integrating the 5G Toolbox PHY is proposed for more realism.

Even under conservative settings (limited UEs, minimal tuning), the SARSA scheduler is competitive with RR and clearly better than Best-CQI and backpressure on delay-proxy metrics, supporting the case for adaptive RL-driven uplink scheduling.

5.6. Stability under Load and Fading Variation.

We sweep the offered load $\boldsymbol{\rho}$ up to 0.95 and consider two channel regimes: slow fading (coherence > 10 slots) and fast fading (coherence ≈ 1–2 slots). SARSA maintains lower DVR than Best-CQI at high $\rho$and prevents queue runaway seen with backpressure under fast fading. RR retains the lowest delay but lacks adaptivity (throughput declines under skewed channels). Fig. 7 plots average and 95th-percentile E2E delay vs. ρ; Fig. 8 shows buffer stability (boundedness) under fast fading

Under mixed traffic, SARSA achieves a better trade-off: URLLC DVR is close to RR ($< 1.2 \times$ difference) while eMBB throughput exceeds RR by 8–14% due to opportunistic allocations during favorable channel states; mMTC PDR remains ≥98% across loads.

6. Conclusion

This paper formulated 5G NR uplink scheduling as a sequential decision-making problem and proposed a reinforcement learning (RL) scheduler based on SARSA with ϵ-greedy exploration. The state encodes buffer status reports (BSR) and achievable rates for eligible UEs; actions select a UE per RBG within each slot; and a delay-centric reward—the negative sum of BSRs—drives learning toward queue reduction. Implemented in a slot-driven MATLAB™ NR simulator with asynchronous HARQ and realistic NR timing, the proposed agent was benchmarked against Round-Robin (RR), Best-CQI, and backpressure policies.

Key findings are threefold. First, even with a compact, tabular design and modest quantization, the SARSA agent nearly matches RR on the average-queue (delay proxy) while surpassing Best-CQI and backpressure, demonstrating the value of reward shaping for delay. Second, the learned policy exhibits adaptive allocation patterns that prevent prolonged queue build-up yet opportunistically exploit favorable channels. Third, these results were obtained without extensive hyperparameter tuning, indicating headroom for further gains.

References

[1] A. Anand, G. de Veciana, and S. Shakkottai, "Joint Scheduling of URLLC and eMBB Traffic in 5G Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 28, no. 2, pp. 477–490, Apr. 2020, doi: 10.1109/TNET.2020.2968373.

[2] S. K. Vankayala and K. G. Shenoy, "A Neural Network for Estimating CQI in 5G Communication Systems," in *2020 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, IEEE, Apr. 2020, pp. 1–5. doi: 10.1109/WCNCW48565.2020.9124744.

[3] G. Pocovi, A. A. Esswie, and K. I. Pedersen, "Channel Quality Feedback Enhancements for Accurate URLLC Link Adaptation in 5G Systems," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, IEEE, May 2020, pp. 1–6. doi: 10.1109/VTC2020-Spring48590.2020.9128909.

[4] A. T. Z. Kasgari and W. Saad, "Model-Free Ultra Reliable Low Latency Communication (URLLC): A Deep Reinforcement Learning Framework," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, IEEE, May 2019, pp. 1–6. doi: 10.1109/ICC.2019.8761721.

[5] H. Yin, X. Guo, P. Liu, X. Hei, and Y. Gao, "Predicting Channel Quality Indicators for 5G Downlink Scheduling in a Deep Learning Approach," Aug. 2020.

[6] V. Shilpa and R. Ranjan, "Radio Resource Scheduling in 5G Networks Based on Adaptive Golden Eagle Optimization Enabled Deep Q-Net," *SN Comput Sci*, vol. 5, no. 5, p. 517, May 2024, doi: 10.1007/s42979-024-02856-8.

[7] F. Al-Tam, N. Correia, and J. Rodriguez, "Learn to Schedule (LEASCH): A Deep Reinforcement Learning Approach for Radio Resource Scheduling in the 5G MAC Layer," *IEEE Access*, vol. 8, pp. 108088–108101, 2020, doi: 10.1109/ACCESS.2020.3000893.

[8] I.-S. Comsa *et al.*, "Towards 5G: A Reinforcement Learning-Based Scheduling Solution for Data Traffic Management," *IEEE Transactions on Network and Service Management*, vol. 15, no. 4, pp. 1661–1675, Dec. 2018, doi: 10.1109/TNSM.2018.2863563.

[9] 3GPP, "NR FDD Scheduling Performance Evaluation," 2021.

[10] J. Lee, S. Jung, S.-E. Hong, and H. Lee, "Development on Open-RAN Simulator with 5G-LENA," in *2024 International Conference on Information Networking (ICOIN)*, IEEE, Jan. 2024, pp. 176–178. doi: 10.1109/ICOIN59985.2024.10572115.

[11] C. Stanescu, T. Paunescu, G. Predusca, L. D. Circiumarescu, N. Angelescu, and D. C. Puchianu, "Performance Evaluation of CDMA and GSM Systems through NetSim Simulations," in *2025 33rd Mediterranean Conference on Control and Automation (MED)*, IEEE, Jun. 2025, pp. 209–214. doi: 10.1109/MED64031.2025.11073525.

[12] V. Stoynov, D. Mihaylova, Z. Valkova-Jarvis, G. Iliev, and V. Poulkov, "An Investigation of Flexible Waveform Numerologies for 5G V2I Cellular Networks from a Physical Layer Perspective," in *2019 IEEE International Conference on Microwaves, Antennas, Communications and Electronic Systems (COMCAS)*, IEEE, Nov. 2019, pp. 1–6. doi: 10.1109/COMCAS44984.2019.8958075.

[13] S. H. Oleiwi, S. S. Gunasekaran, K. I. Abdulameer, M. Abed Mohammed, and M. A. Mahmoud, "Securing Real-Time Data Transfer in Healthcare IoT Environments with Blockchain Technology," *Mesopotamian Journal of CyberSecurity*, vol. 4, no. 3, pp. 291–317, Dec. 2024, doi: 10.58496/MJCS/2024/028.

[14] I.-S. Comșa, P. Bergamin, G.-M. Muntean, P. Shah, and R. Trestian, "FAIR-Q: Fairness and Adaptive Intelligent Resource Management with QoS Optimization in Dynamic 6G Radio Access Networks," in *2025 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, IEEE, Jun. 2025, pp. 1–7. doi: 10.1109/BMSB65076.2025.11165543.

[15] R. Tuninato, G. Maiolini Capez, N. Mazzali, and R. Garello, "5G New Radio for Non-Terrestrial Networks: Analysis and Comparison of HARQ and RLC ARQ Performance Over Satellite Links," *IEEE Access*, vol. 13, pp. 75400–75415, 2025, doi: 10.1109/ACCESS.2025.3563983.

[16] Z. Liu, Z. Yue, F. Li, Y. Yuan, and X. Guan, "Joint Optimization of Adaptive Time Slot Resource Segmentation and Route Scheduling with CQF Mechanism in Time-Sensitive Networks," *IEEE Trans Veh Technol*, pp. 1–11, 2025, doi: 10.1109/TVT.2025.3611967.