Distributed Hierarchical Slack-reduction in Slicing for railways FRMCS

David Kule Mukuhi*†, Leo Mendiboure‡, Rami Langar†§, Rodrigue Fargeon*, Sylvain Cherrier†, Marion Berbineau‡, Pierre-Yves Petton*
*DTIPG, SNCF, 93574 La Plaine-St-Denis, France
† LIGM-CNRS UMR-8049, University Gustave Eiffel, 77454 Marne-la-Vallée, France
‡COSYS/ERENA, University Gustave Eiffel, 33400 Talence, France
‡Software Eng. and IT Department, École de Technologie Supérieure (ÉTS), Montréal, QC, Canada

***Abstract*—The Future Railway Mobile Communication System (FRMCS), based on 5G technology, is being developed to support high-speed train mobility and increasing data-rate demands within the limited 5–10 MHz frequency bandwidth allocated to railway operators. The fluctuating channel conditions and limited radio resources present a significant challenge for the Radio Access Network (RAN). Railway communications are structured around application-specific requirements, categorized into three classes: critical (ultra-reliable train control and safety operations), performance (bandwidth-intensive services like video surveillance and infotainment), and business (on-board Wi-Fi). Existing RAN slicing solutions, are designed for public 5G Networks, assume wideband carriers or static quotas at slice level or user level, so they cannot guarantee application-level requirements or adapt to dynamic channel-quality indicator (CQI) swings in railways. That is why we propose HSRS (Distributed Hierarchical Slack-Reduction Slicing), which formulates this NP-hard allocation problem as an integer linear program and applies a two-phase heuristic: 1) minimize SLA slack for critical ; 2) maximize weighted throughput for performance and business slices. Simulations using SNR traces captured on real trains traveling up to 350 km/h demonstrate the improved application-level SLA satisfaction compared to baseline methods.**

***Index Terms*—FRMCS, railway communications, Network slicing, TOBA, RAN.**

I. INTRODUCTION

One of the most critical elements in railway communications is the radio-access link between trains and trackside infrastructure, today provided by the Global System for Mobile Communications – Railway (GSM-R). Although this 2G technology has served reliably for decades, it will reach end-of-life by 2035 due to discontinued vendor support and its inherently low throughputs, which cannot satisfy emerging high-throughput use cases [1]. In response, the International Union of Railways (UIC) has initiated the development of the Future Railway Mobile Communication System (FRMCS), a 5G-based technology designed to meet the rigorous requirements of next-generation smart railway systems [2].

With FRMCS, the UIC envisions a 5G-based end-to-end system in which a Telecom On-Board Architecture gateway equipment (TOBA) [3] receives both safety-critical and high-throughput application traffic from trackside gNBs, over spectrum as narrow as 5–10 MHz allocated to operators [2]. Railway services span from ultra-reliable signalling (e.g., ETCS (European Train Control System), ATO (Automatic Train Operation)) to high-bit-rate performance streams (e.g., CCTV) and best-effort business traffic (e.g., Wi-Fi onboard), each with very different throughput and latency demands [4]. Efficiently multiplexing these heterogeneous traffic on scarce spectrum, while trains travel at speeds of up to 350 km/h, poses unique challenges not well undertaken in conventional public-mobile networks.

To address this, FRMCS can leverage network slicing, a 5G feature that can partition a single physical RAN into multiple isolated logical networks, or slices, each tailored to a class of services. Based on the 3GPP FRMCS TR 22.889 [5] classification, three slices could be implemented in a railway context. A critical slice guarantees ultra-reliable, low-latency links for ETCS, ATO and emergency staff voice calls; a performance slice supports medium to high-bit-rate streams such as CCTV and passenger information system (PIS); and a business slice carries less vital traffic like onboard Wi-Fi. By assigning dedicated radio-resource blocks and control-plane functions per slice, the operator can satisfy SLAs even under rapid SNR fluctuations along a 350 km/h trajectory.

Despite the maturity of RAN slicing in public 5G networks, existing approaches fall short in the FRMCS context. Application-level schemes [6], [7] enforce per-flow rate targets but assume 100 MHz carriers and uniform SLAs within a slice, making them ill-suited for 5–10 MHz bands. Popular Channel-aware algorithms such as Best-CQI [8] and proportional-fairness [9] boost spectral efficiency yet ignore slice prioritization, risking starvation of critical traffic. Priority frameworks like NVS [10] enforce hierarchical quotas but rely on static partitions and fail to adapt to real-time CQI swings. Essentially, none of these methods have been evaluated on real high-speed railway subband CQI traces, leaving their performance under extreme mobility unverified.

To address these challenges and enable efficient slice management in radio access railway networks, we propose HSRS, a hierarchical slicing framework that combines rigorous modeling, efficient scheduling, and realistic validation. The main contributions of this paper are:

- An exact ILP (Integer-Linear Program): We formulate FRMCS downlink scheduling problem as an ILP;
- The HSRS (Hierarchical Slack-Reduction Slicing) heuristic design: a two-phase slice scheduler that (i) minimizes

the "slack" of critical slice to guarantee their application SLAs, and then (ii) solves a weighted rate-maximization problem to allocate remaining Resource Blocks among performance and business slices.
- A high-fidelity evaluation: We develop a simulation environment for channel-aware RB scheduling using our Subband CQI loader, based on the first real railway high-speed train SNR dataset [11] that captures a 350km/h train dynamic channel conditions. We compare and evaluate our approach with existing solutions.

The remainder of this paper is organized as follows. Section II surveys related work. Section III presents our ILP model. Section IV details the HSRS algorithm. Section V describes the simulation setup and discusses performance results. The conclusion is provided in section VI.

II. Related works

To implement high-performance RAN slicing for railways, we have identified five main criteria (cf. Table I): (a) application-level approach, (b) heterogeneous throughput requirements within a slice, (c) slice-level priority enforcement, (d) sub-band-level channel awareness, and (e) dynamic railway mobility. Due to the lack of work in the railway sector, we compared existing approaches in public 5G networks based on these criteria. These works can be classified into three categories: application-level schedulers, channel-aware algorithms, and priority-based frameworks.

TABLE I
Comparison of Existing RAN Slicing Solutions

Ref	App-level (a)	Hetero. App-level (b)	Priority (c)	Subband Aware (d)	Railway Mobility (e)
[6], [7]	✓	–	✓	–	–
[8], [9]	–	–	–	✓	–
[10], [12]	–	–	✓	✓	–
Our HSRS	✓	✓	✓	✓	✓

A. Slicing with Application-level assurance

Recent studies have increasingly focused on application-level service satisfaction. [7] introduced Zipper, a RAN slicing system that moves from slice-level to application-level service assurance to grant each mobile app its own throughput and latency guarantees. They employ Deep Neural Networks (DNNs) to predict resource availablity and rely on a Model Predictive Control (MPC) for RB allocation targeting video conferencing and file synchronisation applications. [6] addressed augmented reality application demands with a resource manager that adapts bandwidth and edge GPU frequency via a Multi-Armed Bandit algorithm. [7]. However, both assume uniform requirements across applications within a slice, which fails to capture the heterogeneity (b) observed in railway systems. Additionally, these solutions target public MNOs with wide bandwidths (e.g., 100MHz), unlike the narrowband (5–10MHz) available in FRMCS. In this narrowband context, subchannel-aware scheduling (d) becomes critical.

B. Channel-aware slicing

Regarding channel-aware schedulers, authors in [8] use Best CQI techniques that maximize throughput by allocating each RB to the UE with the best channel quality, but this can starve UEs in poor channel conditions and do not account for slice priorities (c). Proportional fairness approaches [9] offer more balanced UE allocation using historical throughput to avoid starvation of low-CQI UE but remain UE-centric, overlook application-level SLAs and slice priority (a,b,c).

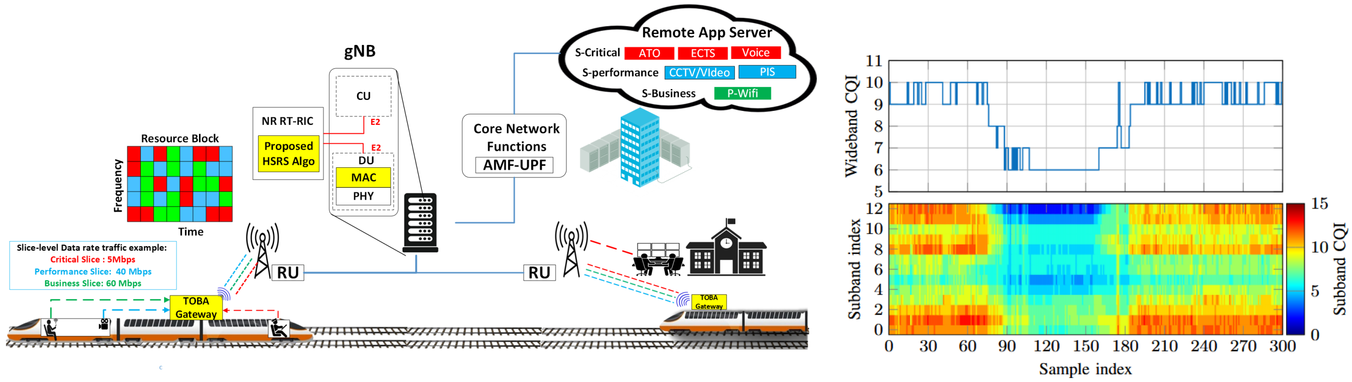
C. Prioritised slice-level assurance

Other slice-based solutions like the Popular *NVS* scheduler [10] rotate between slices using weighted round-robin scheduling based on target throughputs SLA. It assumes that initial slice quotas are defined by the operator as SLA. During resource allocation, any unused quota from a slice can be reallocated to the next slice in the priority hierarchy. However, allocating RB quotas at the slice level without accounting for the heterogeneity of application-level throughput requirements does not guarantee deterministic application-level satisfaction, which is crucial for railway use cases. Therefore, this solution is inadequate. To optimize slice-level RB quota isolation must also consider the varying CQI across subchannels, as it affects the achievable throughput of each RB. As shown in Figure 1b the same train, through its telecommunications equipment (TOBA), experiences varying subband CQI values corresponding to each RB. Therefore, RB allocation to various train applications should not simply consider the entire wideband CQI but account for Subband CQI with higher quality in order to maximize throughput per RB.

Authors in [12] consider a multi-tenant RAN. In their work, slice-level allocation is first performed by estimating slice quotas, followed by UE-level allocation within each slice using a subchannel-aware approach. The Quota estitmation is based a weight that regulate priority. However, this demand-based weight prioritization is correlated with the traffic demand of each slice, which fails to prioritize critical slices whose applications require low throughput but are of highest priority in the railway context. Critical Application cannot tolerate violations, as they are of vital importance for train operations. In addition to the limitations mentioned above , these existing proposals were not evaluated under realistic railway conditions. In contrast, our approach was tested using the first real SNR dataset [11] from a high-speed train traveling up to 350 km/h, capturing dynamic channel variations. Consequently, none of these approaches meets all the identified needs for railway appplication-level satisfaction. Our HSRS algoritm addresses these limitations by incorporating application-level requirements, subband channel awareness, and explicit slice priority in the resource allocation process for critical slice and performance slice applications.

III. Problem formulation

Let the three railway slices be: Critical $s = c$ with set of critical applications A_c, Performance ($s = p$, applications set A_p), and Business ($s = b$, applications set A_b), each



(a) Deployment scenario overview

(b) Subband CQI variations

Fig. 1. Train-to-infrastructure network slicing use case

with distinct throughput requirements to ensure optimal train operation. An UE corresponds to a TOBA gateway hosting multiple applications. To capture resource allocation decisions, we introduce the binary decision variable $x_{k,t,i,a,s}$. Here, $x_{k,t,i,a,s} = 1$ if Resource Block (RB) $k \in \{1,\ldots,K\}$ is allocated to application a of UE i during time slot t in slice $s \in \{c,p,b\}$; otherwise, $x_{k,t,i,a,s} = 0$. The total number of available RBs is K and the allocation episode spans T time slots. The achievable throughput for application a of UE i, on a RB k during time slot t, calculated compliant to Standard 3GPP TS 38.306, is given by:

$$\gamma_{i,a}^{k,t} = 10^{-3} \times J \times v_{\text{Llayers}} \times Q_m \times f \times R_{\max} \times \frac{12}{T_s} \times (1 - OH)$$

where J is the number of component carriers, v_{Llayers} represents the maximum number of MIMO layers, Q_m is the modulation order (determined by the subband CQI as in [13]), f is a scaling factor, $R_{\max}$ is the coding rate corresponding to the UE's subband CQI, T_s is the OFDM symbol duration, and OH is the overhead fraction. This expression represents the per-RB achievable throughput (in bits/ms) under given channel conditions. Our resource allocation problem aims to minimize the total "cost" of resource usage while satisfying the throughput SLAs for each application. The objective is to:

$$\min_{x,\{f_{i,a}^s\}} \sum_{k,t,i,a,s} w_s \, x_{k,t,i,a,s} + \sum_{i,a,s} \frac{1}{w_s} f_{i,a}^s$$

where the weight set $W = \{w_c, w_p, w_b\}$ is chosen such that $w_c < w_p < w_b$, thereby prioritizing the Critical slice over the Performance slice, and the Performance slice over the Business slice. The first term minimizes weighted RB usage cheaper for the Critical slice and the second penalizes unmet SLA via penality $f_{i,a}^s$ using the inverse of the slice weight so that higher-priority slices incur larger penalties. The following constraints must be satisfied:

1) Orthogonal allocation for a given RB across slices is enforced:

$$\sum_{s,a} x_{k,t,i,a,s} \leq 1, \quad \forall k,t,i$$

2) The throughput requirements (SLAs) $D_{i,a}^{\text{s}}$ for each application a in slice s must be met:

$$\sum_{k,t} x_{k,t,i,a,s} \, \gamma_{k,t,i,a} + f_{i,a}^s \geq D_{i,a}^s, \quad \forall i, a \in A_s, s$$

3) Violation Penalty :

$$f_{i,a}^s \geq 0, \quad x_{k,t,i,a,s} \in \{0,1\}.$$

In scenarios where the network is at capacity and these hard constraints become infeasible, we relax them by introducing fractional penalty functions. For example, for an application a in the slice s (e.g: s= critical slice):

$$f_{i,a}^{\text{s}} = \frac{\left|\min\Big(\sum_{k,t} x_{k,t,i,a,s} \gamma_{i,a}^{k,t} - D_{i,a}^{\text{s}}, 0\Big)\right|}{D_{i,a}^{\text{s}}},$$

In summary, the overall problem is to determine the RB allocation variables that minimize the weighted RB usage while ensuring that throughput SLAs are satisfied or the shortfall is minimized over the entire allocation episode.

IV. HSRS DESIGN

The proposed HSRS algorithm targets a distributeed deployment at the gNBs edge along railway tracks, as shown in Fig.1a. With 5G and beyond, railway networks are expected to adopt the O-RAN architecture, where a centralized unit (CU) connects to multiple distributed units (DUs), each managing several Radio Units (RU). O-RAN Near-Real-Time RIC enables real-time xApps, closed-loop RAN control vital for high-mobility. HSRS can be independently deployed as an xApp on each gNB, supporting distributed resource optimization as demonstrated in figure 1a. High speed train mobility are reflected by CQI variation which affect throughput capacity on allocated RBs. Our study accounts for train mobility by validating our model using the 5G SNR dataset [11], collected from real high-speed trains moving up to 350 km/h on a 174km 5G line, capturing realistic railway channel variations.

HSRS operation is based on slack reduction, where slack is the gap between the target throughput and the actual throughput provided by allocated RBs. At $t = 0$, slack for each application is initialized to its SLA target. Our slack-reduction approach allocates RBs to minimize this gap, ideally meeting the application's required throughput. Given the hierarchy of priority among slices in descending order (critical, performance and business), our hierarchical slack-reduction strategy proceeds in two phases, as outlined in Algorithm 1.

Algorithm 1 Hierarchical Slack-Reduction Slicer

Require: $\gamma_{i,a}^{k,t}$, $\mathrm{SLA}_{i,a,s}$, weights w_p, w_b, T, K
Ensure: `RB allocation`
1: initialize `allocs[]`, $\texttt{slack}_{i,a,s} \leftarrow \mathrm{SLA}_{i,a,s}$
2: **for** slots $t = 1 \rightarrow T$ **do**
3: **for** RBs $k = 1 \rightarrow K$ **do**
`#Phase1: minimal-slack`
4: **if** $\exists\,(i,a)\colon\ s=1,\ \texttt{slack}_{i,a,1} > 0$ **then**
5: choose $(i^*, a^*) \leftarrow \arg\max_{s=1} \min(\texttt{slack}_{i,a,1}, \gamma_{i,a}^{k,t})$
6: $\Delta \leftarrow \min(\texttt{slack}_{i^*,a^*,1}, \gamma_{i^*,a^*}^{k,t})$
7: $\texttt{slack}_{i^*,a^*,1} -= \Delta$
8: `allocs`.append$(t, k, i^*, a^*, 1, \gamma_{i^*,a^*}^{k,t})$
9: **continue**
10: **end if**
`#Phase2: weighted Max-Thrput`
11: best $\leftarrow 0$
12: **for all** $(i,a,s) \in \{2,3\}$ with $\texttt{slack}_{i,a,s} > 0$ **do**
13: $w \leftarrow (s = 2\ ?\ w_p : w_b)$
14: score $\leftarrow \gamma_{i,a}^{k,t} \times w$
15: **if** score $>$ best **then**
16: $(i^*, a^*, s^*) \leftarrow (i, a, s)$
17: best $\leftarrow$ score
18: **end if**
19: **end for**
20: **if** best > 0 **then**
21: $\Delta \leftarrow \min(\texttt{slack}_{i^*,a^*,s^*}, \gamma_{i^*,a^*}^{k,t})$
22: $\texttt{slack}_{i^*,a^*,s^*} -= \Delta$
23: `allocs`.append $(t, k, i^*, a^*, s^*, \gamma_{i^*,a^*}^{k,t})$
24: **end if**
25: **end for**
26: **end for**
27: **return** `allocs`

In the first phase (lines 4-10), RBs are allocated to critical applications by selecting the RBs that most effectively reduce the slack. The application that can achieve the highest slack reduction with the current RB is selected by evaluating the minimum between its slack and the achievable throughput (influenced by subband CQI) on that RB. This greedy choice ensures that the most suitable RBs are selected, therefore meeting critical needs while using a lesser number of RBs.

In the second phase (lines 11-24), remaining RBs are allocated between performance (s=2) and business slice (s=3) using a weighted throughput maximization. Each assignment is evaluated by the product of achievable throughput and a slice-specific weight, favoring performance slice. This favors performance applications, even with slightly lower CQI, preventing their starvation. Conversely, Business applications are selected when offering a higher weighted rate or when performance demands are met. Finally, allocated RBs are decremented to update the slack.

V. Performance evaluation

For benchmarking, we compare our HSRS algorithm with two widely used baselines: the Best CQI scheme and the NVS scheduler (cf. Section II). Note that due to the hierarchical priority of critical applications in railway networks, the Proportional Fairness approach[9] is not suitable as it may compromise the SLA of critical services in favor of fairness.

We also compare our HSRS with the optimal solutions. The RB allocation problem we address is NP-hard, meaning that in worst-case scenarios, finding the optimal solution may be computationally infeasible within a polytnomial time bound . Our choice of a hierarchical strategy was driven by the need to reduce computational complexity while still approaching optimal allocation performance. Table 2 compares the execution time (in milliseconds) of different algorithms. The ILP-based solution was computed using the Gurobi solver based on branch-and-bound algorithm. However, such an approach required several minutes of execution time unacceptable in real-time RAN operation. To facilitate comparison, the solver was capped at a 10-second runtime limit per allocation.

TABLE II
TIME COMPLEXITY AND EXECUTION TIME PER ALLOACTION SLOT

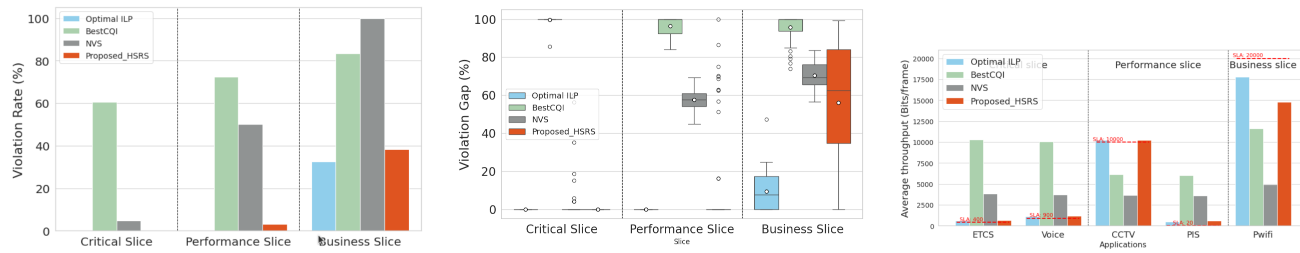
Algorithm	Complexity	Avg Exec. Time (ms)
Optimal ILP	$O(2^{(NKT)})$	694.4
Best-CQI	$O(TKN)$	3.4
Our HSRS	$O(TKN)$	3.0
NVS	$O(TK + NK)$	2.5

A. Simulation Environment

To realistically model spectrum limitations, we simulate a 5 MHz bandwidth, matching the allocation for railway operators in the 900 MHz band. We built a test environment on a Lab system with 36GB RAM memory and 64 core CPU using Python available in [14] for reproducibility and reuse by researchers in the community. In our simulation, the connected applications and gNB allocation decisions are evaluated on a per-frame basis with each frame lasting 10 ms, in accordance with 5G standards. Although railways application SLAs are typically expressed in Kbps for end-to-end communication, we convert these requirements to their per-frame equivalents for RAN-level measurement as shown in Table III.

TABLE III
SERVICE LEVEL AGREEMENTS

Slices	Applications	Throughput (Bits/frame)
Critical	ETCS	400
	Voice	900
Performance	CCTV	10 000
	PIS	20
Business	Passenger WiFi (PWiFi)	20 000



(a) Slice Violation Rate (b) Slice Violation Gap (c) Apps Avg vs SLA Throughput

Fig. 2. Throughput and Violation Metrics

We develop a subband CQI loader compliant with 5G standards based on real SNR dataset mentiond above. These SNR data are loaded by CQI loader, converted to subband CQI using SNR-CQI-MCS mapping from [13]. As specified in 3GPP TS 36.213, CQI can be reported to gnB periodically. To avoid control plane overhead, subband CQI is usually computed not for a single RB but for a Resource Block group (RBG) which can be a group of 4, 6 or 8 RB. Following this periodic CQI feedback in our environment, every configurable time index (default set to 10ms), each TOBA gateway in the simulation is assigned subband CQIs from the real dataset of a specific Train CSV file. This subband CQI loader enables the calculation of the achievable throughput per RB and facilitating channel-aware scheduling.

A block section in railway operations is the safety segment of the railroad where only one train is permitted on the track, which is between 1.5 and 2 km under dense traffic conditions to ensure safe distancing between trains. Knowing that a gNB typically covers 10–15 km in rural areas [15] (and less in urban settings), in a distributed approach a gNB must support about ten train block sections. Accordingly, as summarized in Table IV, we consider a dense scenario with 15 trains under a gNB, each generating traffic for multiple railway applications (26 applications in total). Under average channel conditions (CQI = 7), the cumulative throughput demand (SLA) of these 26 applications cannot be fully satisfied by the available 5Mhz (=25RB). Knowing that channel condition is updated every frame episode based on the CQI, this setup represents a dynamic overloaded scenario, allowing us to clearly observe resource contention and potential SLA violations.

TABLE IV
SCENARIO USED IN THE SIMULATIONS

Scenario	TOBA per Slices	Nbr of Apps across TOBAs	Total Apps
15 Trains (1 train = 1 TOBA gateway)	Critical: 6	6 ETCS + 6 Voice	26 apps
	Performance: 5	5 CCTV + 5 PIS	
	Business: 4	4 PWiFi	

B. Results and Discussion

We evaluate our algorithms using three key metrics shown in Figures 2:

(a) SLA Violations Rate: This measures the frequency at which an application's achieved throughput falls below its required SLA. The BestCQI approach prioritizes TOBAs with the highest channel quality (CQI), which leads to starvation of traffic from applications associated with TOBAs experiencing poor CQI regardless of their slice priority. Consequently, SLA violations are observed across slices depending on the TOBA's channel state. The NVS approach assumes that initial slice quotas are defined by the operator. To guarenty critical services, initial quotas of 50%, 30%, and 20% of available RB are allocated to critical, performance, and business slices, respectively. During resource allocation, any unused quota from a slice can be reallocated to the next slice in the priority hierarchy. This mechanism results in fewer SLA violations in critical and performance slices compared to BestCQI. On the other hand, HSRS ensures critical slice satisfaction while reducing SLA violations in performance and business slices through weighted maximum throughput in the second phase of our hierarchical approach.

(b) Violation Gap: This captures the ratio of the shortfall between the required and achieved throughputs in occured violations. The BestCQI scheduler exhibits binary behavior, allocating either sufficient resources or none, leading to wide violation gaps for applications in lower CQI. The NVS scheduler avoids strict violations gap across slices, due to its rigid priority quota system. In contrast, HSRS achieves a flexible allocation that minimizes the violation gap, even for applications experiencing poor channel conditions, due to their

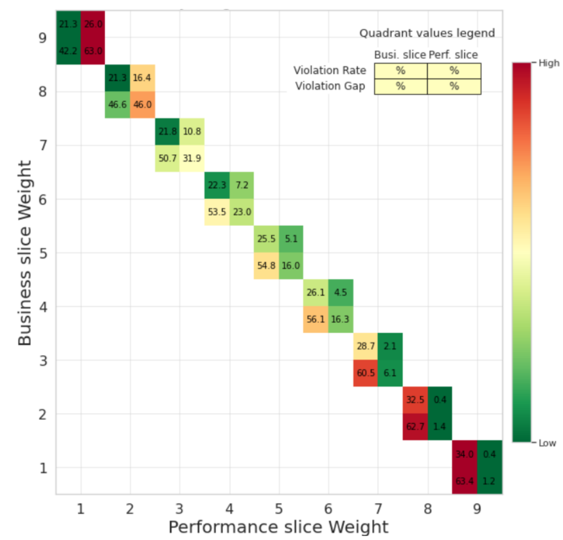


Fig. 3. Performance vs Business slice priority Weight trade-off on violations

respective slice priorities.

(c) Average throughput per Application: This shows the average throughput achieved by each application over multiple simulation episodes, illustrating the distribution of throughput across applications irrespective of any violations on each application instance. The Best CQI and NVS approaches demonstrate that high SLA apps, particularly in the performance slice (i.e: CCTV) are often violated. Our HSRS algorithm, however, distributes resources more proportionaly with application priorities, leading to a more conveniant average throughput distribution and better SLA satisfaction for critical and performance applications essential for train operations.

Our hierarchical approach ensures resource allocation for critical applications, while SLA violations in the second phase are managed using a weighted max-data-rate scheme across performance and business slices. Slice weights control the level of tolerated violations in the performance slice to accommodate business traffic. Figure 3 illustrates the SLA trade-off under different weight settings. As slice weight increases, both violation rate and gap decrease. Setting performance weight between 6 and 9 proved maintaining low violation levels while still serving business demands. In our simulations, weights of (3,7) were used for business and performance slices.

VI. Conclusion

We proposed a Hierarchical Slack-Reduction Slicing (HSRS) algorithm tailored for railway networks, addressing resource constraints and diverse application requirements. By formulating the scheduling problem as an ILP and introducing a low-complexity hierarchical approach, HSRS efficiently allocates RBs across critical, performance, and business slices while respecting priority and application-level heterogeneous requirements. Simulations under realistic overloaded conditions show that HSRS significantly outperforms traditional Best CQI and NVS schedulers. It consistently meets SLAs for critical and performance slices while accomodating business slice traffic, offering a balanced and effective resource allocation strategy. Overall, HSRS provides a robust, Subchannel-aware, and Application-level driven solution for railway RAN slicing. Future work may explore latency-aware scheduling, handling railways signalisation to prediction train arrival in nearby gnB, and MIMO support for enhanced efficiency.

References

[1] B. Ai, A. F. Molisch, M. Rupp, and Z.-D. Zhong, "5G Key Technologies for Smart Railways," *Proceedings of the IEEE*, vol. 108, no. 6, pp. 856–893, Jun. 2020.

[2] U. F. F. W. Group, *FRMCS User Requirements Specification*, 2023.

[3] 5. project, *5GRAIL Deliverable 2.1 - TOBA Architecture Report*, Jan. 2023.

[4] M. Narouwa, L. Mendiboure, H. Badis, *et al.*, "Enabling Network Technologies For Flexible Railway Connectivity," *IEEE Access*, vol. 12, pp. 151 532–151 553, 2024.

[5] "3GPP TR 22.889 Study on Future Railway Mobile Communication System," 3GPP, Technical Report, Mar. 2021.

[6] P. Nikolaidis, S. Mostafavi, J. Gross, and J. Baras, *A Proof of Concept Resource Management Scheme for Augmented Reality Applications in 5G Systems*, arXiv:2501.01398 [cs], Jan. 2025.

[7] A. Balasingam, M. Kotaru, and P. Bahl, "Application-Level Service Assurance with 5G RAN Slicing," *21st USENIX Symposium on Networked Systems Design and Implementation (NSDI 24)*, Apr. 2024.

[8] M. Asvial, G. Dewandaru, and A. N. Rachman, "Modification of Round Robin and Best CQI Scheduling Method for 3GPP LTE Downlink," *International Journal of Technology*, vol. 6, no. 2, p. 130, Apr. 2015.

[9] M. A. Ibraheem, N. ElShennawy, and A. M. Sarhan, "A Proposed Modified Proportional Fairness Scheduling (MPF-BCQI) Algorithm with Best CQI Consideration for LTE-A Networks," in *2018 13th International Conference on Computer Engineering and Systems (ICCES)*, Dec. 2018, pp. 360–368.

[10] R. Kokku, R. Mahindra, H. Zhang, and S. Rangarajan, "NVS: A virtualization substrate for WiMAX networks," in *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, Chicago Illinois USA: ACM, Sep. 2010, pp. 233–244.

[11] Y. Pan, R. Li, and C. Xu, "The First 5G-LTE Comparative Study in Extreme Mobility," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 6, no. 1, pp. 1–22, Feb. 2022.

[12] C. Yongzhou, Y. Ruihao, H. Haitham, and M. Radhika, "Channel-Aware 5G RAN Slicing with Customizable Schedulers," *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 24)*, pp. 1767–1782, Apr. 2023.

[13] A. K. Thyagarajan, P. Balasubramanian, V. D, and K. M, "SNR-CQI Mapping for 5G Downlink Network," in *2021 IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob)*, Apr. 2021, pp. 173–177.

[14] kulemandev, *HSRS github repository : Hierarchical Slack-Reduction Slicing for railways FRMCS*, original-date: 2025-05-13T12:04:30Z, May 2025. [Online]. Available: https://github.com/kulemandev/HSRS.

[15] GSMA, *Vision 2030: Low-Band Spectrum for 5G*, 2022. [Online]. Available: https://www.gsma.com/connectivity-for-good/spectrum/wp-content/uploads/2022/07/5G-Low-Band-Spectrum-1.pdf.