A Decision-Tree Solution for the Outdated CQI Feedback Problem in 5G Networks

Peterson Yoshioka *, Maria Damasceno*§, Andson Balieiro*, Siba Narayan Swain†, Elton Alves‡
*Centro de Informática (CIn), Universidade Federal de Pernambuco (UFPE), Recife, Brazil
†Indian Institute of Technology Dharwad, Karnataka, India.
‡Faculdade de Engenharia da Computação, Universidade do Sul e Sudeste do Pará, Brazil.
§Sidia Institute of Science and Technology, Manaus, Brazil.
{pmsy, mgld, amb4}@cin.ufpe.br,
sibaswain@iitdh.ac.in, eltonalves@unifesspa.edu.br, maria.lima@sidia.com

***Abstract*—Accurately reporting a Channel Quality Indicator (CQI) value that reflects the current channel condition is crucial for 5G networks. However, the delay between measuring the channel condition and effectively utilizing it by the base station can render the CQI outdated, negatively impacting UE communication. This paper proposes a Decision-Tree solution that considers Signal-to-Interference plus Noise Ratio (SINR) and user context to estimate the updated SINR for translating into a CQI value. While various machine learning (ML) models are proposed in the literature, this study focuses on decision trees, which are explainable artificial intelligence (XAI) models capable of elucidating decision-making processes. The results demonstrate that our solution achieves high accuracy and performance similar to the ideal one, with an absolute difference of only 0.001 in both throughput and spectral efficiency metrics. This underscores the feasibility of our approach in addressing the outdated CQI feedback problem.**

***Index Terms*—Outdated CQI Feedback Problem, 5G Networks, Decision-Tree**

I. Introduction

The Fifth Generation (5G) of Mobile Networks is being employed to support a wide array of services, encompassing autonomous vehicles, ultra-high definition video streaming, and the Internet of Things (IoT). 5G networks aim to deliver high throughput (e.g., dozens of Gbps), ultra-low latency (e.g., on the order of a few milliseconds), high reliability (e.g., on the order of 99.99999%), low energy consumption, and the ability to handle high connection densities [1]. However, maintaining consistent high-quality service in wireless communications encounters challenges such as signal reflection, diffraction, user mobility, and interference from external sources.

To address this challenge, 5G base stations (gNodeB) adopt the Adaptive and Modulation Coding (AMC) technique. This approach dynamically adjusts the modulation and coding schemes (MCS) used for transmission based on the channel quality indicator (CQI) reported by the user equipment (UE). The CQI is intended to reflect the downlink channel condition, aiding the gNodeB in determining the proper MCS and radio resources for communication. As a result, the reported CQI indirectly influences key performance metrics such as throughput, block error rate (BLER), and spectral efficiency. Thus, accurately reporting a CQI value that reflects the current channel condition is crucial for effective 5G network link adaptation. However, the time delay between measuring the channel condition and utilizing this information at the gNodeB, which involves tasks such as CQI computation and transmission, may render the CQI obsolete or stale, failing to accurately represent the current channel quality [2] [3]. Consequently, this outdated information could degrade the UE communication performance, leading to issues such as reduced throughput and increased BLER.

Solutions addressing the outdated CQI feedback problem have been proposed in the literature [2] [3] [4] [5]. These solutions primarily focus on predicting CQI or Signal-to-Noise Ratio (SNR) values using techniques such as linear extrapolation [5], Long Short Term Memory (LSTM) Neural Networks [3] [4], or linear estimation with stochastic approximation [2]. However, these approaches often neglect influence of user context, such as position and distance to the BS, on the perceived channel condition. They typically rely solely on CQI or SNR for estimating future channel conditions. Additionally, some solutions are implemented at the gNodeB side [3] [4], which may lead to increased computational burden, especially in scenarios with high user density.

In contrast to existing approaches, our paper proposes a Decision Tree (DT)-based solution that not only considers the SINR but also incorporates user context, such as position and distance to the base station, to estimate the updated SINR to be translated into a CQI value. We predict the updated SINR instead because it provides a more granular and continuous measure of channel quality, allowing for more precise adjustments when translating to CQI values. Additionally, SINR is widely used in 5G for link adaptation and modulation processes, making it a more robust variable for predictions. Although various ML models have been developed, here we focus on decision trees, which are eXplainable Artificial Intelligence (XAI) and interpretable models capable of explaining, by design, how decisions are made. Indeed, DTs present transparency, i.e., the capability of understanding the structure of the model itself, which is a key requirement towards trustworthy AI [6] and a major pillar in the design of 5G/6G networks. Thus, due to their simplicity and interpretability, decision trees have also been used to address other problems in 5G/6G Networks. For instance, [7] proposes a real-time decision tree-based approach for jamming detection in 5G new radio. In [8], the authors combine clustering and decision tree-based learning to predict traffic anomaly and thus proactively prevent network failure events; and in [9], a Hoefding Decision Tree is employed to predict quality of experience (QoE) of video streaming applications in B5G/6G Networks.

We evaluated our solution by using data from 5G simulator [10] in terms of prediction accuracy, spectral efficiency, and throughput, comparing it to an ideal SINR predictor, i.e., one with perfect prediction, zero error. The results show that our solution presents a high level of accuracy and performance similar to an ideal one, with a minimal absolute difference of 0.001 in both throughput and spectral efficiency metrics, highlighting its feasibility to deal with the outdated CQI feedback problem. The remainder of this paper is organized as follows. Section II discusses related works. The outdated CQI feedback problem and the proposed Decision Tree-based solution proposed are presented in Section III. Results and Analysis are conducted in Section IV. Section V concludes this work.

II. Related Work

Solutions based on prediction have been proposed for the oudated CQI feedback problem in the literature. For instance, in [2], the authors analyze different techniques such as Linear, Linear with Stochastic Approximation (LSA), Kalman Filters, and Discrete Cosine Transform (DCT) Sequences to predict the CQI, with LSA outperforming the others in terms of the complexity-performance tradeoff. In [5], the authors use previous signal-to-noise ratio (SNR) values and linear extrapolation to predict future SNR. However, the proposal does not work properly in scenario with moderate or high speed users.

On the other hand, in [4], a Long Short Term Memory (LSTM) Neural Network is proposed to predict the CQI and online retraining is employed to achieve high accuracy even in dynamic scenarios. For predicting CQI, [3] proposes a deep recurrent neural networks (DRNNs) approach based on the time-series of previous CQI values. The authors emphasize that the solution is designed for Unmanned Aerial Vehicle (UAV) control information based on Ultra-reliable and Low Latency communication (URLLC) but do not consider the device context aspects such as device position to estimate the future CQI. These three approaches ([3], [4], and [5]) are single-type input forecasters, but differ in terms of the prediction technique. [3] employs Long Short Term With Memory (LSTM) and Gated Recurrent Unit (GRU) layer. Furthermore, [3] and [4] predict the CQI and operate at the base station, while [5] focuses on the SNR and is embedded into the device. [11], in turn, proposes a lightweight version of LSTM to predict the CQI and thus reduce the number of feedback transmissions (CQI feedback overhead) in IoT networks with aperiodic reports. Besides the focus, [11] differs from our current paper as it does not consider user context to predicts the CQI and it is designed to run at the BS.

In our previous paper [12], we tackled the outdated CQI feedback problem by proposing a Multi-Layer Perceptron (MLP) Neural solution. This MLP considers the UE mobility context (e.g. position, velocity, and movement direction), delay length, and the SINR to predict the updated SINR. In the current paper, we adopt the Decision-Tree Approach for predicting the updated SINR, mapping it into a CQI value. The selection of inputs for Decision-Tree is guided by Pearson and Spearman correlation coefficients to avoid unnecessary overhead or complexity. Unlike the MLP, which is considered a black-box model, the Decision Tree is an interpretable and transparent approach that allows for explaining, by design, how decisions are made and the model structure. Additionally, beyond evaluating the accuracy of the Decision Tree solution, we consider its impact on spectral efficiency and throughput, comparing it to an ideal predictor, which has zero prediction error.

III. Proposal

A. The Channel Quality Indicator (CQI)

The channel quality in 5G networks varies across cells due to factors such as position (proximity to the antenna), interference from other sources, signal reflection and diffraction, and user mobility. To respond to these changes and provide the best possible communication service to users, 5G base stations employ link adaptation (LA), where modulation and coding schemes (MCS) and the amount of resources are adjusted based on the channel quality [4]. For instance, in channels with good quality, higher-order MCS may be applied to achieve higher throughput and spectral efficiency. On the other hand, lower-order MCS are most suitable for handling poor channels and avoiding frequent retransmissions.

The channel condition is reported by the UE to the gNodeB in a Channel State Information (CSI) report, which comprises three main components, a Channel Quality Indicator (CQI), Precoding Matrix Index (PMI), and Rank Indicator (RI). Among these, the CQI holds particular significance for link adaptation as it indirectly defines UE communication performance, influencing factors such as data rate and error block rate. The UE determines the CQI based on the reference signal received from the gNodeB, with values ranging from 0 to 15. A higher score indicates better channel quality [13]. Based on the CQI, the gNode selects the best modulation and coding schemes along with determining the amount of resources for transmission in order to maximize the spectral efficiency, while targeting a certain block error rate (BLER) [14], for example.

In this respect, it is crucial that the CQI accurately reflects the current channel quality for a proper LA. However, the delay incurred by tasks performed between the reference signal reception at the UE and the MCS selection by the gNodeB may render the CQI obsolete or outdated. For instance, upon receiving the reference signal, the UE dedicates time to process measurements (e.g. computing the SINR) and translate them into a CQI value. Subsequently, the UE sends the CQI to the gNodeB, introducing delays associated with UL transmission scheduling, transmission itself, and signal propagation. Once the CQI reaches the gNodeB, the MCS selection adds an additional delay to this sequence of events [3]. These cumulative overheads contribute to the issue of the outdated CQI feedback problem.

B. The Decision Tree

The Decision Tree is a whitebox supervised learning algorithm composed of nodes and connections between them (branches), forming a hierarchy in a tree-like structure, with possible paths from the initial node (root) to the terminal nodes (leaves). Intermediate nodes (decision nodes) have a condition based on some attribute (feature), where it is possible to decide between two paths: an affirmative case or a negative one. Thus, its operation involves recursion in which the tree's depth increases as the splits occur, until reaching stopping points (the leaves), which contain the final value of an attribute's result [15]. Figure 1 shows an example of a decision tree with leaves represented by squares (t_3, t_4, and t_5), one intermediate node (t_2), the root node (t_1), and two attributes for evaluation (x_1 and x_2) .

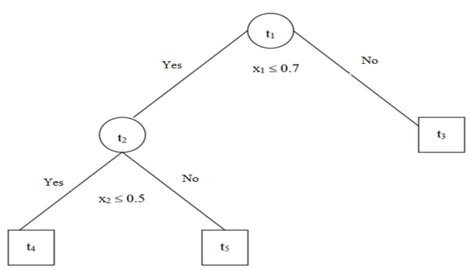

Fig. 1. An example of a Decision Tree

In the splits, selection criteria for features are used to determine which attributes will be taken into account in the intermediate nodes as well as the stop criterion to be considered to determine a terminal node. For Regression Decision Trees, the attributes are numeric and their intervals characterize the tree's splitting, where Mean Squared Error (MSE) is considered as the default cost for the split, along with the purity level for division [16]. For stop criteria, options include defining a minimum number of samples for a leaf node, the maximum tree height, and the minimum impurity decrease value, for instance [15]. To enhance the performance of a tree, pruning is employed, removing less important branches to prevent overfitting of the data and reduce the tree's complexity. One method of pruning is through the Minimum Cost Complexity Pruning, starting from a complexity parameter (alpha) that measures the complexity cost of nodes and branches. This allows for the removal of nodes and branches based on the cost of sub-trees [16].

C. Data and Input Variables

To train the Decision Tree model and estimate the updated SINR, mapping it into a CQI value, we adopted data generated via mmWave simulator [10]. Simulations were conducted with eight different UE initial positions, spanning values for x, y, and z coordinates from 20 to 100. Each simulation lasted 30 seconds, resulting in a total of 23,975 collected samples and balancing computational complexity with a realistic time frame for channel variations in dynamic environments. Additionally, we set the PDSHC (Physical Data Shared Channel) overhead to zero to focus on evaluating the pure data transmission capacity without interference from control channels. This configuration allows facilitates reproducibility by isolating the effects of link adaptation and allows for a clearer analysis of the modulation and coding scheme (MCS) impact on throughput and spectral efficiency. Throughout the simulation, we collected data on the UE's velocity, position (x, y, and z coordinates), angle, distance to the closest base station, movement direction, and SINR (in dB). These variables were considered as possible inputs for estimating the SINR at instant $t + \tau$, where τ denotes the delay length (feedback delay) and t the current time. This work defined τ as the time elapsed between two consecutive collected samples, but it also allows for consideration of other values. To select the proper input variables for the Decision Tree model, we computed the Pearson and Spearman correlation coefficients. These coefficients evaluate the influence of each input variable at instant t on the SINR at instant $t + \tau$ (target output) and are presented in Table I. The criterion for selection was set as having both correlation coefficients higher than 0.5 (absolute value), resulting in choosing the UE position (x,y,z), distance between UE and the BS, and SINR to compose the set of input variables.

TABLE I
PEARSON AND SPEARMAN CORRELATION COEFFICIENTS

Input Variable	Pearson	Spearman	Selected Variable
SINR	0.97	0.96	✓
Velocity	0.01	0.01	-
Angle (μ)	- 0.16	- 0.16	-
Direction(v)	0.04	0.004	-
Position (x,y,z)	- 0.61	- 0,585	✓
Distance to BS	- 0.90	- 0.92	✓

The first two variables can be obtained through the Global Positioning System (GPS), commonly embedded in current mobile devices, or by using alternative methods such as databases of geo-tagged WiFi hotspots, sensor-based technologies (e.g., cameras), Wifi signal-based localization, Indoor Positioning Services (IPS), as well as their combination [17]. These options vary in terms of position accuracy, adopted environment (outdoor and indoor), orientation mode (UE-based or network/server–based), measurement time, energy consumption, and privacy level. These factors must be considered in the selection of the most suitable one [18]. The current SINR may be measured by the UE based on the reference signal received from the gNodeB.

D. Analysis Metrics

a) Spectral Efficiency (SE): spectral efficiency (in bit/s/Hz) is defined as the ratio between the data rate and channel bandwidth [19]. It can be obtained via Eq. 1, which considers the linear SINR and the Block Error Rate (BLER).

$$SE = log_2(1 + \frac{SINR}{-ln(5BLER)/1.5}) \quad (1)$$

b) Throughput: measured in bits per second (bps), throughput denotes the amount of data transmitted in given period of time. It can be computed by considering the slot duration, based on the 5G numerology μ, and the number of bits per slot. The latter is determined by taking into account the downlink channel overhead (OH_{dw}) and the transport block size (TBS), as denoted in Eq. 2. The TBS is defined according to the MCS selected by the BS for use in downlink communication.

$$Th = \frac{bits_{slot}}{slot_{duration}} = \frac{(1 - OH_{dw}) * TBS}{(1/2^{\mu})10^3} \quad (2)$$

To compute the TBS [20], the first step involves determining the number of resource elements (REs) allocated for the physical downlink shared channel (PDSCH) within the slot (N_{RE}) using Eq. 3. It considers the number of REs allocated for PDSCH within a physical resource block (PRB), denoted as N'_{RE}, and the number of PRBs allocated to the UE (n_{PRB}). The expression for N'_{RE} is given by Eq. 4, where N_{SC} represents the number of sub-carriers per PRB (12 in 5G networks), N_{symb} is the number of symbols of the PDSHC allocation within the slot (which may be 12 for extended cyclic prefix or 14 for normal cyclic prefix), N_{DRMS} denoted the amount of REs per PRB for demodulation reference signals (DMRS), and the OH_{PDSCH} is the PDSHC overhead, which can assume 0, 6, 12, or 18. In this work, we set it as 0.

$$N_{RE} = min(156, N'_{RE}) * n_{PRB} \quad (3)$$

$$N'_{RE} = N_{SC} * N_{symb} - N_{DRMS} - OH_{PDSCH} \quad (4)$$

Subsequently, the unquantized intermediate variable (N_{info}) is obtained through Eq. 5, where R represents the code rate, Q_m denotes the modulation order, and v signifies the number of layers. Following this, N_{info} undergoes analysis to define how the quantized intermediate number of information bits (N'_{info}) should be computed and used to derive the TBS. If the value is less than or equal to 3840 then N'_{info} is determined by Eq. 6 and Table 5.1.3.2-1 (from 3GPP TS 38.214 version 16.2.0 Release 16 [20]) is referenced to find the closest TBS that is not less than N'_{info}. Otherwise, N'_{info} is obtained via Eq. 7 and the TBS is calculated using Eq. 8, where C is given by Eq.9.

$$N_{info} = N_{RE} * R * Q_m * v \quad (5)$$

$$N'_{info} = max\left(24, 2^n \left\lfloor \frac{N_{info}}{2^{max(3,\lfloor log_2(N_{info})\rfloor - 6)}} \right\rfloor\right) \quad (6)$$

$$N'_{info} = max\left(3840, 2^n * round\left(\frac{N_{info} - 24}{2^{(\lfloor log_2(N_{info} - 24)\rfloor - 5)}}\right)\right) \quad (7)$$

$$TBS = 8 * C \left\lceil \frac{N'_{info} + 24}{8C} \right\rceil - 24 \quad (8)$$

$$C = \begin{cases} \left\lceil \frac{N'_{info}+24}{3816} \right\rceil, & R \leq 1/4 \\ \left\lceil \frac{N'_{info}+24}{8424} \right\rceil, & R > 1/4 \ and \ N'_{info} > 8424 \\ 1, & \text{otherwise.} \end{cases} \quad (9)$$

c) Model Accuracy: we have adopted the Mean Squared Error (MSE) and R-Squared (R^2) metrics to assess the accuracy of the Decision Tree model and select the best configuration for our approach. The MSE is defined in Eq. 10 and measures the average squared difference between the predicted values ($\hat{y}_i$) and the target ones (y_i). The (R^2), given in Eq. 11, denotes the model's ability (in percentage) to explain or predict the relationship between the dependent and independent variables. A higher R^2 value indicates a better fit of the model to the data, demonstrating its ability to explain the dataset.

$$MSE = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n} \quad (10)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})} \quad (11)$$

IV. Results

To train and evaluate the Decision Tree aproach, the data was organized into two sets, training and test, with each one comprising 70% and 30% of the collected samples (23975), respectively. Since the input variables present different scales, we normalized them between 0 and 1 by using minmax operation [11]. In the next section (IV-A), we present the accuracy results obtained by different Decision Tree configurations in the training and test stages as well as the criteria adopted to select the best setting to compose our Decision Tree approach. Section IV-B compares the defined Decision Tree model to an optimal SINR predictor in terms of throughput and spectral efficiency, where it is considered that the optimal solution predicts all values perfectly, i.e, its MSE is zero. The aim is to show the performance proximity between our approach and the ideal one.

A. Model Configuration and Accuracy

To define the best Decision Tree configuration to be adopted in our scheme, we conducted several tests, varying different parameters, including the criteria for selecting the best split strategy of each node and maximum number of features to consider during the split, maximum depth of the tree, minimum number of samples for a leaf, maximum number of leaves, and complexity parameter of Minimum Cost Complexity Pruning. The tree's depth controls the overall complexity of a decision tree, allowing for getting a trade-off between an under-fitted and over-fitted decision tree. Two values for this parameters were analyzed. In terms of strategy used to choose the split at each node, the "best" and "random" were used. The former selects the best split whereas the latter chooses the best random split. The maximum number of features to consider when looking for the best split were computed considering two alternatives, square root ($sqrt$) and log_2. Table II summarizes the parameters and values analyzed to define the best DT configuration.

With a total of 216 settings for Decision Trees, the selection proceeded through four general steps, specifying further details later. The steps are: (1) Restrict configurations with Mean Squared Error (MSE) below the reference MSE (1%); (2) Select the one with the lowest absolute error difference between training and test MSEs; (3) Evaluate error metrics and consider the configuration with the lowest average MSE and highest average R^2 Score; and (4) Review the results of candidate configurations. To select the configuration to compose our Decision Tree scheme, we adopted the criteria defined in Eqs. 12 and 13, considering a MSE_{ref} equals 0.001, and took into account the configuration with the lowest average MSE and highest R^2.

TABLE II
TESTED DECISION TREE PARAMETERS

Parameter	Value
Split strategy of each node	best, random
Maximum number of features to consider during the split	sqrt, log2
Maximum depth of the tree	5, 10
Minimum number of samples for a leaf	20, 30, 40
Maximum number of leaves	30, 40, 50
Parameter of complexity for Minimum Cost Complexity Pruning	0.0, 0.01, 0.02

$$(MSE_{train}, MSE_{valid}) \leq MSE_{ref} \tag{12}$$

$$Minimize \quad |MSE_{train} - MSE_{valid}| \tag{13}$$

In Fig. 2, DT configurations with an MSE lower than 0.01 corresponds to ids 33, 42, 87, 96, and 105. Among these, configuration 42 exhibits the lowest absolute error difference between MSEs. Upon analyzing the error metrics of these selected configurations, it is observed that the values are quite close to each other, with configuration 42 demonstrating the lowest average MSE and the highest R^2 Score (see Fig. 3). This decision tree configuration has a height of nine with variability in leaf distribution. When comparing the predicted normalized SINR(dB) with the target values, it is observed that most samples are close, with some records showing partial deviation, as shown in Fig. 4. For configuration 42, the errors of iterations have values less than 0.1, except for the MSE in the first condition, while the R^2 Score mostly exceeds 0.92. Therefore, configuration 42 was selected, and its parameters are outlined in Table III.

Fig. 4 shows the SINR estimated by the Decision Tree scheme (selected configuration) in comparison to the target value for one execution. In general, the Decision Tree follows the target behavior, denoting that the proposed scheme learned the structure of the dataset and is able to deal with the CQI delay feedback problem. Since the Decision Tree presented a low MSE (as seen in Fig. 2) and the estimated SINR is quantized into a CQI value via a process based on SINR intervals, the small difference between the target and Decision Tree output may not lead a CQI error.

TABLE III
SELECTED DECISION TREE CONFIGURATION

Parameter	Value
Split strategy of each node	best
Maximum number of features to consider during the split	sqrt
Maximum depth of the tree	10
Minimum number of samples for a leaf	30
Maximum number of leaves	50
Parameter of complexity for Minimum Cost Complexity Pruning	0

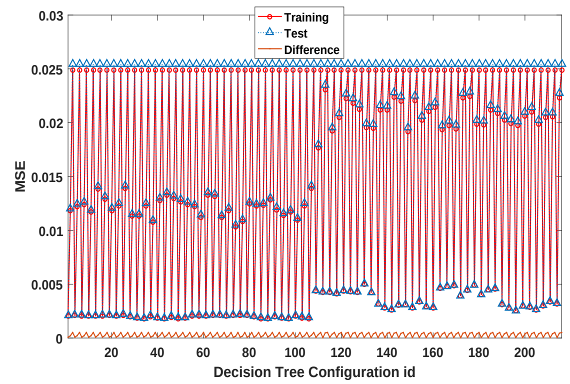

Fig. 2. MSE and Absolute Difference for Different Decision Tree Configurations

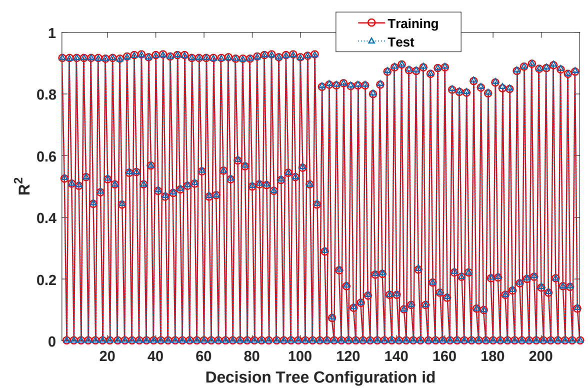

Fig. 3. R^2 score for Different Decision Tree Configurations

B. Throughput and Spectral Efficiency

Fig. 5 presents the results in terms of average spectral efficiency and throughput considering 30 executions and the parameters summarized in Table IV. Comparing to the ideal predictor, it is observed that our proposed solution offers similar spectral efficiency and throughput values, presenting an absolute difference of only 0.01 in both metrics. This finding reinforces the feasibility of our scheme to address the outdated CQI feedback problem in 5G networks.

V. CONCLUSION

This work proposed a Decision Tree-based solution for addressing the outdated CQI feedback problem in 5G net-

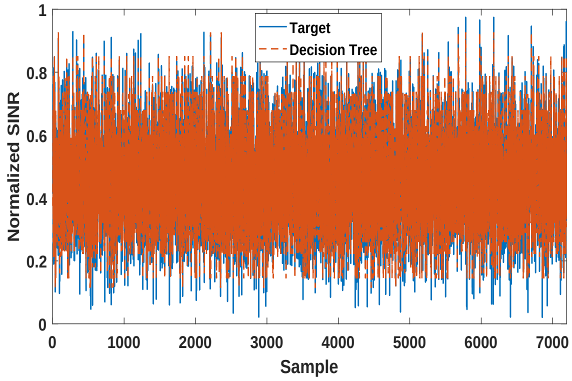

Fig. 4. SINR estimated by the Decision Tree in comparison to the target value

TABLE IV
PARAMETERS FOR TBS AND THROUGHPUT COMPUTATION

Parameter	Value
Downlink Overhead (OH_{dw})	0.18
Number of Allocated PRBs (n_{PRB})	1
Numerology (μ)	3
Number of Layers (v)	1
Number of Subcarriers (N_{SC})	12
Number of Symbols per slot ($N_s ymb$)	14
PDCH overhead (OH_{PDSCH})	0
N_{DRMS}	0
Target BLER	0.00005

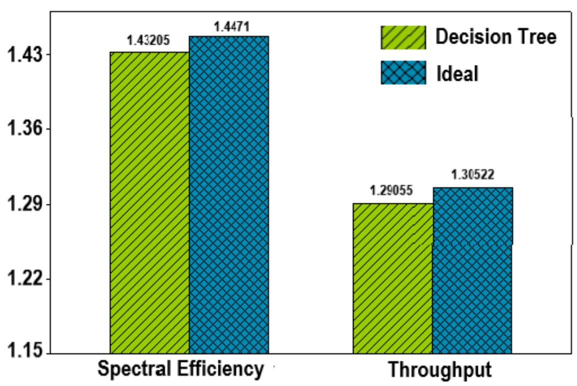

Fig. 5. Average Spectral Efficiency (in bit/s/Hz) and Throughput (in Mbps)

works. We utilized the Pearson and Spearman coefficients to carefully select the appropriate inputs for the Decision Tree, thus avoiding unnecessary overhead. We considered various configurations of Decision Trees to select the best one for our solution.Additionally, besides achieving a high level of accuracy, our solution demonstrated performance similar to the ideal one, with an absolute difference of only 0.001 in both throughput and spectral efficiency metrics. This demonstrates the feasibility of our approach in addressing the outdated CQI feedback problem. For future work, we suggest exploring hybrid approaches that combine different machine learning techniques as reference functions, as well as investigating other values for window size prediction.

ACKNOWLEDGMENT

This study was funded by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brazil (CAPES) - Code 001, and the Sidia Institute of Science and Technology.

REFERENCES

[1] I. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat, and H. Dai, "A survey on low latency towards 5g: Ran, core network and caching solutions," *IEEE Communications Surveys Tutorials*, vol. 20, no. 4, pp. 3098–3130, 2018.

[2] R. A. Akl, S. Valentin, G. Wunder, and S. Stánczak, "Compensating for cqi aging by channel prediction: The lte downlink," in *2012 IEEE Global Communications Conference (GLOBECOM)*, 2012, pp. 4821–4827.

[3] G. Bartoli and D. Marabissi, "Cqi prediction through recurrent neural network for uav control information exchange under urllc regime," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 5, pp. 5101–5110, 2022.

[4] H. Yin, X. Guo, P. Liu, X. Hei, and Y. Gao, "Predicting channel quality indicators for 5g downlink scheduling in a deep learning approach," *ArXiv*, vol. abs/2008.01000, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:220936414

[5] M. Ni, X. Xu, and R. Mathar, "A channel feedback model with robust sinr prediction for lte systems," in *2013 7th European Conference on Antennas and Propagation (EuCAP)*, 2013, pp. 1866–1870.

[6] E. Commission, "Ethics guidelines for trustworthy ai," European Commission - High Level Expert Group on AI, Tech. Rep., 2018.

[7] Y. Arjoune and S. Faruque, "Real-time machine learning based on hoeffding decision trees for jamming detection in 5g new radio," in *2020 IEEE International Conference on Big Data (Big Data)*, 2020, pp. 4988–4997.

[8] N. Koursioumpas, L. Magoula, S. Barmpounakis, and I. Stavrakakis, "Network traffic anomaly prediction for beyond 5g networks," in *2022 IEEE 33rd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2022, pp. 589–594.

[9] *Hoeffding Regression Trees for Forecasting Quality of Experience in B5G/6G Networks*. CEUR Workshop Proceedings, Aug. 2022. [Online]. Available: https://doi.org/10.5281/zenodo.7024541

[10] D. Raca, D. Leahy, C. J. Sreenan, and J. J. Quinlan, "Beyond throughput: The next generation a 5g dataset with channel and context metrics," 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:216653912

[11] N. Han, I.-M. Kim, and J. So, "Lightweight lstm-based adaptive cqi feedback scheme for iot devices," *Sensors*, vol. 23, no. 10, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/10/4929

[12] A. Balieiro, K. Dias, and P. Guarda, "A machine learning approach for cqi feedback delay in 5g and beyond 5g networks," in *2021 30th Wireless and Optical Communications Conference (WOCC)*, 2021, pp. 26–30.

[13] I. Tomic, E. Bleakley, and P. Ivanis, "Predictive capacity planning for mobile networksmdash;ml supported prediction of network performance and user experience evolution," *Electronics*, vol. 11, no. 4, 2022.

[14] M. Ramezani-Mayiami, J. Mohammadi, S. Mandelli, and A. Weber, "Cqi prediction via hidden markov model for link adaptation in ultra reliable low latency communications," in *WSA 2021; 25th International ITG Workshop on Smart Antennas*, 2021, pp. 1–5.

[15] P. GUPTA, "Towards data science," 2017. [Online]. Available: https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052

[16] L. E. d. Abreu, "People analytics: uso de árvores de decisão na retenção de talentos," Universidade Estadual Paulista (Unesp), 2022.

[17] A. Konstantinidis, G. Chatzimilioudis, D. Zeinalipour-Yazti, P. Mpeis, N. Pelekis, and Y. Theodoridis, "Privacy-preserving indoor localization on smartphones," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 11, pp. 3042–3055, 2015.

[18] I. Constandache, S. Gaonkar, M. Sayler, R. R. Choudhury, and L. Cox, "Enloc: Energy-efficient localization for mobile phones," in *IEEE INFOCOM 2009*, 2009, pp. 2716–2720.

[19] S. D. de Lima, "Análise e simulação dos aspectos de transmissão de sinais 5g em macro células em ambientes urbanos," Patos de Minas, MG, p. 97, 2023.

[20] ETSI and 3GPP, "'5g;nr;physical layer procedures for data (3gpp ts 38.214 version 16.2.0 release 16," ETSI and 3GPP, Tech. Rep., 2020.