Intelligent and Converged Networks ISSN 2708-6240
2022, 3(3): 271−281 DOI: 10.23919/ICN.2022.0012

Addressing the CQI feedback delay in 5G/6G networks via machine learning and evolutionary computing

Andson Balieiro*, Kelvin Dias, and Paulo Guarda

Abstract: 5G networks apply adaptive modulation and coding according to the channel condition reported by the user in order to keep the mobile communication quality. However, the delay incurred by the feedback may make the channel quality indicator (CQI) obsolete. This paper addresses this issue by proposing two approaches, one based on machine learning and another on evolutionary computing, which considers the user context and signal-to-interference-plus-noise ratio (SINR) besides the delay length to estimate the updated SINR to be mapped into a CQI value. Our proposals are designed to run at the user equipment (UE) side, neither requiring any change in the signalling between the base station (gNB) and UE nor overloading the gNB. They are evaluated in terms of mean squared error by adopting 5G network simulation data and the results show their high accuracy and feasibility to be employed in 5G/6G systems.

Key words: channel quality indicator (CQI) feedback delay; 5G/6G networks; machine learning; evolutionary computing

1 Introduction

The fifth and sixth generations of mobile networks (5G/6G) are designed to support applications that demand different levels of latency, connection density, reliability, and throughput[1]. However, the unpredictability and dynamism of the wireless environment (e.g., user mobility and signal reflection) make the process of holding the mobile communication quality challenging. In this respect, 5G networks apply adaptive modulation and coding (AMC) to dynamically select the downlink modulation order and coding rate, targeting high throughput and spectral efficiency but keeping the block error rate (BLER) under control[2]. The AMC is based on the channel quality indicator (CQI), a 4-bit value[3] reported by the user equipment (UE) that aims at expressing the current channel condition. By using this feedback, the 5G base station (gNB) allocates the resources and defines the modulation and coding schemes to be used by the UE.

Having a CQI value that faithfully reflects the channel quality is paramount important for 5G communications since that an inaccurate CQI may result in imbalanced resource distribution and unsuitable modulation and coding scheme (MCS), degrading the UE communication[4]. An overestimated (underestimated) CQI leads to selection of a higher (lower) order MCS by the gNB, which may cause higher BLERs and excessive retransmissions (reduction in data rate and spectral efficiency).

Besides the way how the channel quality is estimated (e.g., via measurement of the signal-to-interference-plus-noise ratio (SINR) of a reference signal sent by gNB[5]), the delay incurred by the CQI feedback may also make the CQI inaccurate since that variations in the channel quality may take place between the transmission and reception of the CQI, especially in scenarios with high mobility users.

This paper extends our previous one[6] by designing two approaches to address the CQI feedback delay, one based on machine learning and another on evolutionary computing. Different from solutions presented in the

• Andson Balieiro and Kelvin Dias are with the Centro de Informática (CIn), Universidade Federal de Pernambuco, Recife 50740-560, Brasil. E-mail: {amb4, kld}@cin.ufpe.br.
• Paulo Guarda is with the Motorola Mobility, Mogi Mirim 13800-206, Brazil. E-mail: pauloguarda@motorola.com.
∗ To whom correspondence should be addressed.
Manuscript received: 2022-02-18; revised: 2022-06-18; accepted: 2022-07-18

literature, which only adopt a single input (SINR, signal-to-noise ratio (SNR), or CQI) to predict its future value, our proposals consider the UE context (expressed in term of velocity, direction, and position) and delay length besides the SINR to estimate the updated SINR to be mapped into a CQI value. They act as regressive models and functions that map five input variables into an output (updated SINR), whereas the previous schemes are basically time-series forecasters that get data samples equally spaced in time. In addition, our solutions are designed to run at the UE side, neither requiring any change in the signalling between gNB and UE nor overcharging the base station. The proposed approaches are evaluated in terms of mean squared error (*MSE*), by using 5G network simulation data, and the results show their high accuracy and feasibility to be adopted in 5G/6G networks.

This paper is organized as follows. Section 2 presents works that deal with the CQI feedback process. Section 3 describes the proposed solutions for the CQI feedback delay problem, which are evaluated in Section 4. Section 5 concludes this paper and outlines future directions.

2 Related work

The CQI is an important indicator used by the 5G base station to define the MCS and the amount of radio resources to be considered in the downlink. In this aspect, it is imperative that its value expresses the current channel quality, but there are three important issues that emerge in the feedback process and make it challenging, which are: (1) the CQI accuracy, i.e., how the channel quality is gotten and translated into a channel quality indicator; (2) the obsolete CQI value, which may be a consequence of changes in the wireless environment during the reporting time (feedback delay); (3) the CQI feedback overhead, an excessive signalling in the uplink, caused by too frequent reports (e.g., in scenarios with high user density).

Different studies have proposed solutions for these issues, such as in Ref. [7], which addressed the obsolete CQI problem by using linear extrapolation to predict the signal-to-noise ratio (SNR) based on previous values. This is a low complexity scheme that fails when employed in scenario with moderate or high speed users. By addressing the same issue, Ref. [8] used a long short term memory (LSTM) neural network to predict the CQI and considered online retraining to keep the high accuracy of the schemes even in dynamic scenarios. Similar to Ref. [7], Ref. [8] was a single input type forecaster, but presented two differences: it predicted the CQI and ran at the base station.

The CQI accuracy issue was addressed in Ref. [4, 5, 9]. Reference [5] proposed a scheme that considers the SNR and the maximum multipath delay spread of instantaneous channel state to precisely compute the channel quality indicator under fading channels. Reference [4] adopted three machine learning (ML) techniques (stochastic gradient descent, multilayer perceptron (MLP), and support vector machine) to predict the SNR in environments with different user speeds and channel models besides considering the SNR and CQI as inputs. Reference [9] in turn designed a CQI mapping scheme that balances the energy efficiency (EE) and spectral efficiency (SE) while keeping the block error rate under control. By employing this proposal, the operators may set their priorities via weight adjustments for SE and EE. In addition to the tackled problem, Refs. [4, 9] differ from our work as they are developed to run on the base station and do not adopt the user equipment speed as an input to define the SNR or CQI.

The CQI feedback overhead was addressed via spatial correlation of wireless channels in Refs. [10, 11], where the authors used Gaussian process regression (GPR) to estimate the CQI for some users based on the selected ones. To improve the scheme performance, Ref. [10] adjusted the model by considering the user density and prediction accuracy. Different from our approaches, Refs. [10, 11] are limited to scenarios with static users and strongly dependent on the user density, not working well in low user density cases. In Ref. [2], the subband level and aperiodic feedback were used to reduce the CQI feedback overhead. The authors proposed a GPR-based scheme to estimate the CQI and thus compensate for the feedback reduction, evaluating

it in scenario with moving users, but without considering the user speed as an input to the scheme.

The authors in Ref. [12] proposed a strategy for multicast unmanned aerial vehicle (UAV) systems in which the base station determines the modulation and coding schemes for UAVs in the same group based on their CQI feedbacks. Although the proposal is focused on UAVs, it does not consider their high mobility, which may impact on their positions and, consequently, lead to an obsolete CQI to be used in the MCS selection, making the scheme unsuitable for UAV systems. In Ref. [13], the feedback overhead was addressed by estimating the channel quality indicator of some users via CQI values reported by others. To do so, the authors used two neural networks (NNs) running on the base station, one for selecting the users and subbands to be used in the CQI estimation and another for estimating the CQI.

Table 1 points out the main characteristics of the previous works and compares them to the proposed one, clarifying their differences.

3 Proposed approaches

The channel quality indicator is a key-element used by the user to report the channel condition to the base station in 5G networks. The medium access layer scheduler adopts the CQI to allocate resources, determines the modulation and coding schemes along with the transport block size to be employed in the downlink and, consequently, defines the amount of data that will be transmitted at each time slot[8]. In this regard, having a CQI that faithfully denotes the channel quality when the base station makes decisions is fundamental. Notwithstanding, the delay incurred by the CQI transmission may make its value obsolete.

To tackle this issue, we develop two approaches, one based on machine learning and another on evolutionary computing. Both consider the user context expressed in terms of position, velocity, and movement direction, the SINR measured at instant t and also the delay length (τ) to estimate the SINR at the moment $t+\tau$ (updated), which is then mapped into a CQI value. There are different alternatives for the SINR-CQI mapping (e.g., via mapping table received from the base station or using schemes such as Refs. [4, 13, 15]) and our approaches support these different ways, not demanding any change in the gNB or signalling protocol, besides being planned to run on the user equipment. Figure 1 illustrates the approaches, which are described in Sections 3.1 and 3.2.

The current SINR is measured regarding the reference signal sent by the base station. The UE context in turn may be obtained via global positioning system since it is commonly embedded in the current mobile devices and, thus, it could be a natural solution. However, there are other alternatives to get the UE context such as databases of geo-tagged Wifi hotspots, sensor-based technologies (e.g., cameras), Wifi signal based localization, indoor positioning services (IPSs), as well as their combinations[16]. These systems differ from each other 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, which need to be considered in the selection of the most suitable one based on constraints such as accuracy threshold, energy budget, and desired privacy level. For instance, considering the energy consumption and privacy, the GPS consumes more energy than Wifi or cell network based solutions[17], but as it runs the localization directly on the UE with no location-sensitive information received or sent from/to the base station or external server, such as in IPS, it may offer better level of privacy to the user.

3.1 Machine learning based approach

For SINR estimation (see Fig. 1), we analyze two types of feedforward artificial neural networks (ANNs): multilayer perceptron (MLP) and radial basis function (RBF), which are widely used for regression and classification problems and able to build non-linear mappings. We conducted several tests with different ANN configurations by varying the key-parameters of each network type and evaluated them in order to define the best one by considering the ANN complexity in addition to the Formulas (1) and (2), which intend to point out the ANN that is able to learn the data characteristics used in the training stage and

Table 1 Approaches for CQI feedback related problems.

Reference	Proposal	CQI problem	Technique	Goal	Side	Input	Output
[2]	A packet loss and CQI prediction approach	Feedback overhead	Gaussian process regression	To minimize the packet loss and limit the CQI signalling overhead	BS	CQI and packet loss	CQI and packet loss
[4]	An ML-based approach to predict SNR	Innaccuracy	Machine learning (SVR, MLP, SGD)	To provide a more accurate channel quality estimation	BS	CQI and SNR	SNR
[5]	A multipath delay spread-aware CQI scheme for LTE system	Innaccuracy	Empirical SNR/delay - CQI mapping based on simulation data	To achieve more precise CQIs under fading channels	UE	SNR and multipath delay spread	CQI
[7]	A linear extrapolation-based scheme to predict SNR and map it into a CQI value	Feedback delay	Linear extrapolation	To improve the network throughput	UE	SNR	SNR and derived CQI
[8]	An LSTM-based CQI prediction method and an online training module in ns-3	Feedback delay	LSTM artificial neural network	To improve the CQI prediction accuracy	BS	CQI	CQI
[9]	A CQI mapping algorithm that considers the spectral efficiency and energy efficiency tradeoff	Estimation	The VIKOR ranking method and weighted sum	To balance EE and SE in the MCS selection	BS	CQI and weights for EE, SE, and coding rate	CQI
[10]	An SNR prediction scheme based on the SNR reported by spatially correlated UEs.	Feedback overhead	Gaussian process regression	To reduce the CQI feedback overhead and improve the prediction quality	BS	SNR and spatial correlation of users	SNR and CQI
[11]	An SIR prediction scheme that uses SIR from spatial correlated users	Feedback overhead	Gaussian process regression	To reduce the CQI signaling overhead	BS	SIR and spatial correlation from a set of users	SIR
[12]	A CQI feedback scheme for UAV multicast system	Feedback overhead	Minimum function and fixed channel for CQI feeback per group	To reduce the signalling overhead and increase the spectral efficiency	BS	CQI	CQI
[13]	A CQI report scheme for URLLC	Estimation	The worst-case estimation	To accurately estimate and report the the worst-case SINR conditions	UE	SIRN	CQI
[14]	An ANN-based selector and a dense ANN-based for users/subbands selection and CQI estimation	Feedback overhead	Artificial neural networks	To reduce the CQI feedback signalling overhead	BS	CQI of selected subbands/ users	CQI
This work	ML and EC-based approaches with multiple inputs for CQI feedback delay	Feedback delay	MLP and RBF artificial neural networks and genetic algorithms	To accurately estimate the SINR considering the CQI feedback delay	UE	UE velocity, movement direction and position, delay length, and SINR	SINR

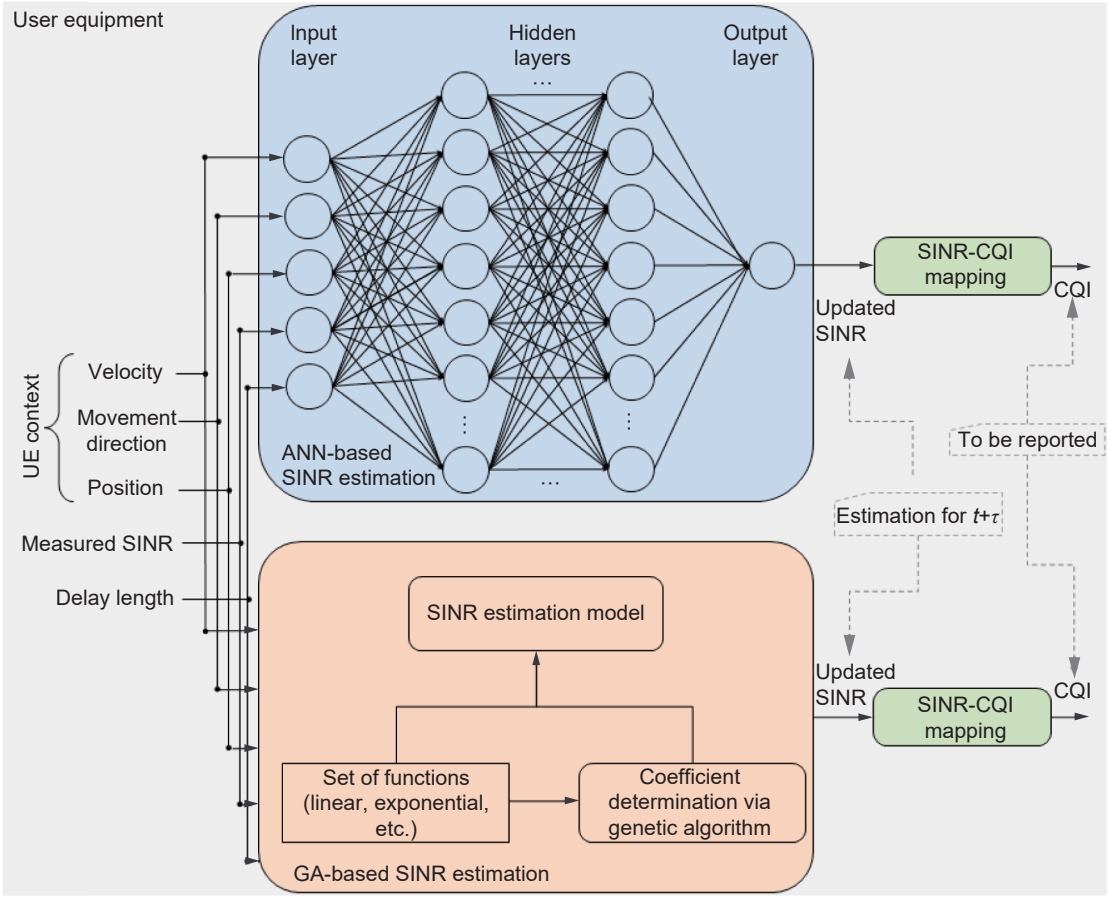

Fig. 1 Proposed schemes.

generalize them when fed with new data.

$$(MSE_{train} \text{ and } MSE_{valid}) \leqslant MSE_{ref} \quad (1)$$

$$\text{Minimize } |MSE_{train} - MSE_{valid}| \quad (2)$$

where the MSE_{ref} denotes the desired mean squared error (*MSE*), which was considered as 0.01 in this paper; MSE_{train} and MSE_{valid} mean the *MSE* got by the ANN in the training and validation phases, respectively.

3.1.1 MLP topology

The multi-layer perceptrons (MLPs) are neural networks with one or more hidden layers that adopt non-linear function neurons[18]. They have been widely used in diverse problems in 5G networks such as channel estimation in massive MIMO systems[19], throughput prediction[20] as well as in CQI-related issues[4, 6].

In order to define the most suitable MLP topology for dealing with the CQI feedback delay, we analyzed different MLP ANNs, by testing parameters such as the number of hidden layers (NHL), neurons in the hidden layer (NHLN) and their activation functions (AFHL) as well as the learning rate (LR). The sigmoid (Eq. (3)) and hyperbolic tangent (Eq. (4)) functions were the options for NHLN. In all cases, the input layer was set with five neurons, which are associated to the input variables, and the output layer was defined with one linear activation function (OLAF) based neuron, which denotes the SINR at the moment $t+\tau$. Moreover, the MLP ANNs were trained by using the backpropagation learning algorithm[18]. Table 2 presents the parameters and tested values.

Table 2 Tested MLP ANN hyper-parameters and values.

Hyper-parameter	Value
Number of hidden layers (*NHL*)	1; 2; 3
Number of hidden layer neurons (*NHLN*)	5; 10; 15; 20; 25; 30
Activation function of hidden layer neuron (AFHL)	Tansig; Logsig
Learning rate (*LR*)	0.01; 0.045; 0.1

$$Logsig(x) = \frac{1}{1+e^{-x}} \quad (3)$$

$$Tansig(x) = \frac{2}{1+e^{-2x}} - 1 \quad (4)$$

The number of neurons in the hidden layer impacts on the neural network performance. Adopting so few hidden neurons would result in a neural network without the capacity to learn the data structure. In contrast, too many hidden neurons would severely increase the neural network complexity and its learning time, without improving the performance significantly[18]. Similarly, the ANN's efficiency and complexity are also impacted by the number of hidden layers, where conditions of overfitting or underfitting may emerge as a consequence of adopting too many or few hidden layers in the ANN, not following the problem complexity, and, thus, causing generalization loss over new data or inefficient results[21]. On these points, we tested six and three values for NHLN and NHL, respectively, which are listed in Table 2.

Figure 2 presents the average results for all 144 MLP configurations, computed by considering 30 executions. Almost all configurations met Formula (1). The configuration #48, which comprises $NHL = 1$, $NHLN = 35$ neurons, sigmoid AFHL, and $LR = 0.2$, obtained the lowest MSE_{train} (0.003217), but its $MSE_{valid} = 0.005$ made its error difference (0.001782) bigger than the configuration #38, which got 0.0004008, 0.004127, and 0.004528 for error difference, MSE_{train}, and MSE_{valid}, respectively.

The configuration #91 got the best performance with

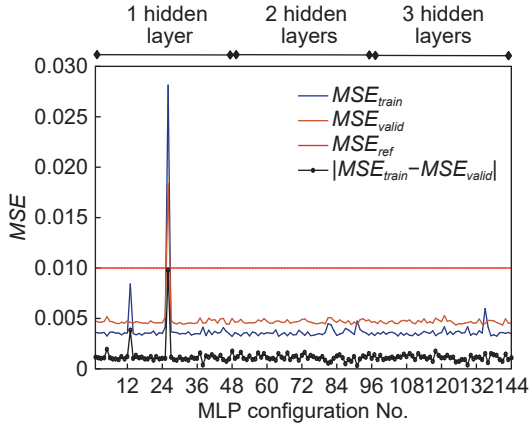

Fig. 2 *MSE* values got by the MLP under different configurations.

regard to Formula (2), error difference equals to 0.0003183, but being more complex (double hidden layer with 30 neurons) and achieving an MSE_{train} slight higher than the configuration #38. With this in mind, we adopted the configuration #38 in our MLP-based scheme, which is summarized in Table 3. The data used in this evaluation are described in Section 4.

3.1.2 RBF topology

The RBF ANNs are composed of input, hidden, and output layers. The first connects the neural networks to the environment. Each neuron in the hidden layer represents a center (cluster) for the input space and adopts a radial basis activation function, such as the Gaussian one, in which its output is given by the euclidean distance between the center and the input data. Each activation function requires two parameters: center and width. The outputs of the hidden layer are combined linearly by the second layer neurons[22].

Similarly to the MLPs, we also analyzed different RBF configurations, varying two important parameters, the spread factor (SF) and the maximum number of neurons in the hidden layer in order to define the proper configuration for the CQI feedback delay problem. The spread factor controls the width of the activation function, i.e., the response area in the input space associated to each hidden layer neuron. Table 4 summarizes the values tested.

Figure 3 presents the average *MSE* obtained by the RBF configurations in the training stage, considering 30 executions. All configurations achieved MSE_{train}

Table 3 Selected MLP configuration.

Parameter	Value
Number of input layer neurons (*NILN*)	5
Number of hidden layers (*NHL*)	1
Number of hidden layer neurons (*NHLN*)	25
Activation function of hidden layer neurons	Sigmoid (Eq. (3))
Number of output layer neurons (*NOLN*)	1
Activation function of output layer neuron	Linear
Learning rate (*LR*)	0.1

Table 4 RBF hyper-parameters and values.

Hyper-parameter	Value
Spread factor (*SF*)	0.1; 0.5; 0.7; 0.8; 0.9; 1; 1.5; 2; 5; 10
Maximum number of hidden neurons (*MN*)	200; 500; 700; 800; 1000

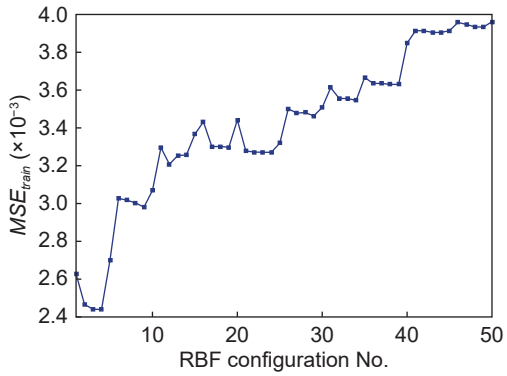

Fig. 3 MSE_{train} **for different RBF configurations.**

lower than the reference one (0.1), being the lowest value got by the configuration #4. However, only the configurations #1 and #5 were able to meet Formula (1), i.e., getting both MSE_{train} and MSE_{valid} lower than 0.1, being Formula (1) better satisfied by configuration #1, which achieved an absolute difference between the errors being equal to 0.006 25 instead of 0.018 51, got by the RBF configuration #4. Besides that, the fist configuration also presented less complexity, with $SF = 0.1$ and $MN = 200$.

3.2 Evolutionary computing based approach

Besides the machine learning based approach, we also propose a genetic algorithm (GA) based solution for the CQI feedback delay problem. It comprises a set of base functions (e.g., linear and exponential) to map the input values (e.g., UE context, measured SINR, and delay) into an output one, the updated SINR, and employs the GA to define the most suitable coefficient values for the SINR estimation model, as shown in Fig. 1.

This GA-based proposal admits different types of functions (e.g., linear, non-linear, and their combinations), without the cost/need of linearizing the models, which differs from works that are based on the ordinary least squares method. This flexibility allows that each input variable (or a subset of them) may be handled by the function that best describes its relationship with the output (updated SINR). In this paper, we instantiate our proposal by adopting a linear model to compute the updated SINR ($SINR^*$), which is given in Eq. (5),

$$(a_0 + a_1 v_1 + a_2 v_2 + a_3 v_3 + \cdots + a_n v_n) = SINR^* \quad (5)$$

where v_i and a_i represent the input variable i and its coefficient, respectively.

3.2.1 Chromosome structure and fitness function

GA is a search algorithm based on the natural selection principles that employs genetic operators (e.g., selection, crossover, and mutation) to evolve candidate solutions, represented by the so-called chromosomes, toward the optimal one. In this process, GA eventually finds good solutions by combining different chromosomes[23].

In our scheme, the individual or chromosome is represented by a sequence of coefficient values, where each position (gene) refers to a model coefficient to be defined. Figure 4 illustrates the chromosome structure for a model that presents k coefficients.

To evaluate the individuals (solutions), we defined the fitness function given in Eq. (6).

$$fitness(\cdot) = \frac{S}{1 + MSE} \quad (6)$$

where MSE is the mean squared error between the real SINR and that one computed by the model considering the coefficients given by the GA individual ($SINR^*$) and S is a scale factor. In this work, we adopted $S = 1000$.

3.2.2 Genetic operators and parameters

We adopted the roulette wheel as the selection operator for choosing individuals for the crossover process based on their fitness values, i.e., the higher fitness value, the higher probability of being selected. An arithmetic operator was employed for the crossover operation, which combines linearly the genetic material from parent chromosomes to create new ones (offsprings) and is expressed in Eq. (7).

$$F_i = \alpha P_i + (1 - \alpha) P_j \quad (7)$$

where P_i and P_j are two parent chromosomes and α is a number uniformly distributed between 0 and 1. In order to avoid the premature convergence to local optima, we used uniform mutation, in which each gene (coefficient a_i) may have its value modified according to the Eq. (8).

$$a_i = a_i^* + \beta \quad (8)$$

where a_i^* is the gene value before the mutation process

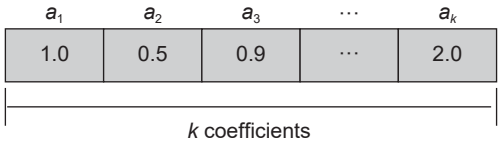

Fig. 4 Chromosome structure.

and β is a value uniformly distributed in a given interval.

Since that the frequency of crossover and mutation operations has great impact on the GA performance, multiple tests have been conducted to define their occurrence probabilities (pc and pm, respectively) in our GA-based scheme. In this way, 64 test cases obtained by the combination of values for the crossover and mutation probabilities (see Table 5) were analyzed and we selected the values that provided the highest average fitness for the last generation's population after 5 simulation instances to set our GA. Figure 5 presents the results, in which the test case 37 displayed the best performance for the GA, having crossover and mutation probabilities equal to 0.5 and 0.3, respectively. In addition to these parameters, the population size and number of generations were defined as being equal to 100 and 200, respectively.

3.2.3 GA flow execution

The execution flow of our GA-based scheme is shown in Fig. 6. Given an SINR model with coefficients to be determined, the initial population is randomly generated so as to provide candidate solutions (coefficient values). After that, the individuals are evaluated via fitness function (see Eq. (6)) that is based on the *MSE*. Thereupon, the individuals are submitted for selection, together with the crossover and mutation operators, and moreover, an elitist strategy is employed to ensure the best fitness individuals will not be lost during the selection process. Finally, a new generation of candidates will be created and the stop criterion, which is determined by the number of generations (G), is evaluated. If it is not satisfied, the process repeats from the fitness evaluation stage. Otherwise, the best individual is chosen as the final solution. This represents the coefficients to be adopted in the SINR model.

Table 5 Tested crossover and mutation probabilities.

Parameter	Value
pc	0.1/0.2/0.3/0.4/0.5/0.6/0.7/0.8
pm	0.01/0.03/0.05/0.1/0.3/0.5/0.6/0.7

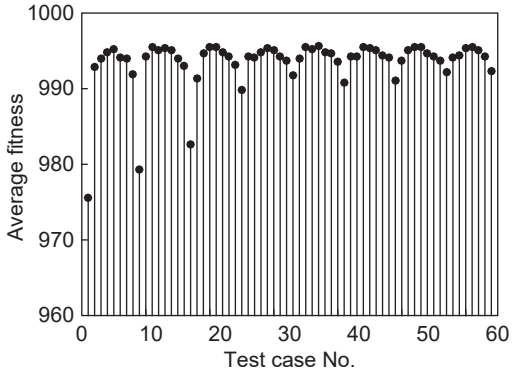

Fig. 5 Average fitness of the GA with linear model under different pc and pm values (test cases).

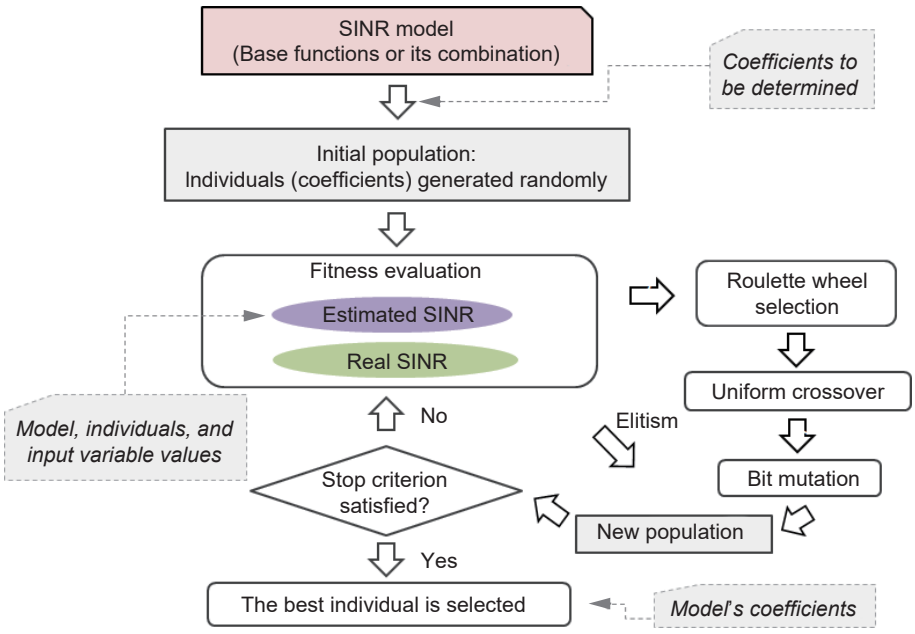

Fig. 6 GA execution flow.

4 Result

This section presents the results got by our approaches. First, they are analyzed in terms of *MSE* obtained in the training and validation stages. After that, a comparison between the SINR real (target) and that one defined by each proposed approach is conducted. The evaluation used data generated by the 5G/mmwave ns-3 simulation framework[24], considering a 5G network where the UE speed, movement direction, and position were varied during the simulations in order to produce different CQI and SINR values. 2633 samples were collected, being 70% and 30% used for training and validation, respectively.

Figure 7 compares the approaches in terms of *MSE* achieved in the validation and training stages. We noted that the GA-based scheme presents the best performance in both phases, achieving *MSE* values equal to 0.003 915 and 0.004 109, respectively. Besides that, GA also achieved a similar performance in both stages, which denotes it was able to properly define the linear model coefficients by using the training data, but not being addicted to them.

The MLP-based scheme also got a great performance, with MSE_{train} and MSE_{valid} assuming 0.004 127 and 0.004 528, respectively, denoting that the MLP-based scheme not only learned the data characteristics in the training phase, but also provided a great generalization capacity when faced the validation data. This close performance in both stages achieved by these two approaches was not followed by the RBF-based one in the same order (10^{-4}). Although it had gotten the lowest MSE_{train}, i.e., 0.002 627, its MSE_{valid} assumed a value equals to 0.008 878, denoting a difference between them in the scale of 10^{-3} and showing a slightly performance reduction in its generalization capacity.

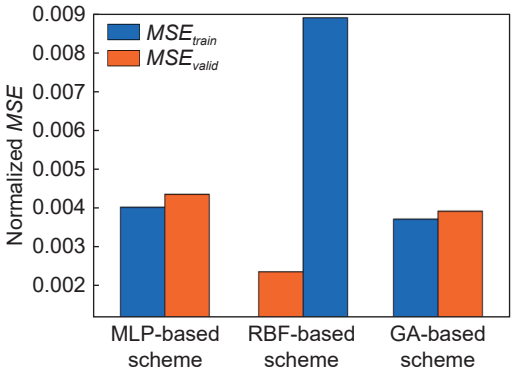

Fig. 7 *MSE* values achieved by the approaches.

Figures 8–10 compare the real SINR value (desired) to that one estimated by the schemes. In general, for all schemes, the curves presented similar behaviors with

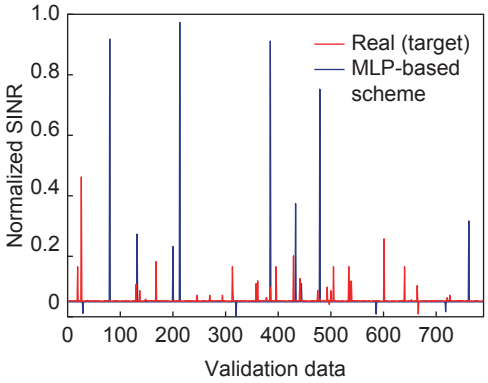

Fig. 8 SINR estimated by the MLP-based scheme and the real value.

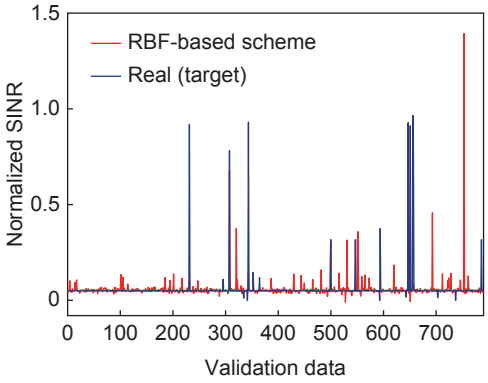

Fig. 9 SINR estimated by the RBF-based scheme and the real value.

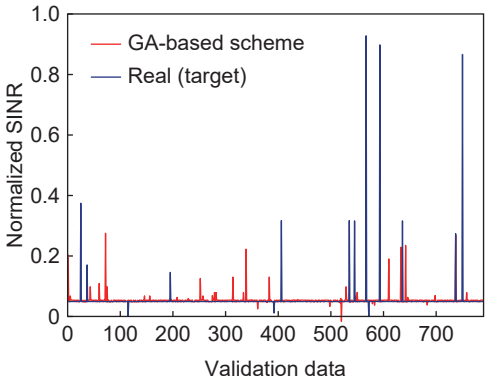

Fig. 10 SINR estimated by the GA-based scheme and the real value.

some points of mismatches between real and estimated values, but without causing high *MSEs* (as shown in Fig. 7). The results show that our proposed approaches are feasible to overcome the CQI feedback delay, either the machine learning based schemes or the evolutionary computing one.

Since our proposal is designed to be hosted at the UE, which has limited energy and processing capacities, it is worth mentioning that the main processing load (the neural network training or execution of genetic algorithms to determine the model coefficients) may be handled out of the UE, in a cloud or MEC (multi-access edge computing) server, for example. Once the model has already been defined (trained neural network or GA-based one), it may be embedded into device via sums and products. In this way, the processing load to compute the updated SINR is greatly reduced, being now predominant that one related to the UE context acquisition, in which different solutions may be adopted, as discussed in Section 3.

In addition, although the mapping of updated SINR into CQI value is not the focus of this paper, we point that different aspects may be considered in this process, such as spectral efficiency (SE) and BLER[15], SE and energy efficiency tradeoff[9], or application requirements[13]. Furthermore, when the SINR-CQI translation is based on intervals, the distance between the real and the value computed via proposed approaches may not cause a CQI error.

5 Conclusion

This paper proposed two approaches to deal with the CQI feedback delay, one based on machine learning and another on evolutionary computing. We took into account the user context, measured SINR, and delay length to faithfully estimate the SINR and conducted extensive tests to define the best configuration for each proposed scheme. All approaches presented high accuracy values, which indicate they are able to estimate the channel quality and thus assist the correct MCS selected by the base station. Although not discussed, our approaches support online re-training[8] in response to the wireless environment changes. Future works include embedding the proposed schemes in a simulator or testbed and combining them with different CQI mapping mechanisms in 5G networks with heterogeneous services. In addition, comparing our approaches to those ones presented in the literature regarding aspects such as complexity and performance (e.g., CQI accuracy, energy consumption, spectral efficiency, and BER) under different mobility scenarios is a work to be conducted.

Acknowledgment

This work was supported by the Motorola Mobility, the National Council for Scientific and Technological Development (No. 433142/2018-9), Research Productivity Fellowship (No. 312831/2020-0), and the Pernambuco Research Foundation (FACEPE)

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] A. Chiumento, M. Bennis, C. Desset, L. V. D. Perre, and S. Polin, Adaptive CSI and feedback estimation in LTE and beyond: A Gaussian process regression approach, *EURASIP Jounral on Wireless Communications and Networking*, doi: 10.1186/s13638-015-0388-0.

[3] 3GPP and ETSI, 5G; NR; Physical layer procedures for data, 3GPP TS 38.214 version 15.3. 0 Release 15, Technical specification ETSI TS 138 214 V15.3.0, 2018.

[4] K. Saija, S. Nethi, S. Chaudhuri, and R. M. Karthik, A machine learning approach for SNR prediction in 5G systems, in *Proc. 2019 IEEE International Conference on Advanced Networks and Telecommunications Systems* (*ANTS*), GOA, India, 2019, pp. 1–6.

[5] X. Chen, H. Yi, H. Luo, H. Yu, and H. Wang, A novel CQI calculation scheme in LTE\LTE-A systems, in *Proc. International Conference on Wireless Communications and Signal Processing* (*WCSP*), Nanjing, China, 2011, pp. 1–5.

[6] A. Balieiro, K. Dias, and P. Guarda, A machine learning approach for CQI feedback delay in 5G and beyond 5G networks, in *Proc. 2021 30th Wireless and Optical Communications Conference* (*WOCC*), 2021, Taiwan, Province of China, 2021, pp. 26–30.

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

[8] 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 preprint arXiv: 2008.01000,

2020.

[9] M. I. Salman, C. K. Ng, N. K. Noordin, B. M. Ali, and A. Sali, CQI-MCS mapping for green LTE downlink transmission, *Proceedings of the Asia-Pacific Advanced Network*, vol. 36, pp. 74–82, 2013.

[10] S. Homayouni, S. Schwarz, M. K. Mueller, and M. Rupp, CQI mapping optimization in spatial wireless channel prediction, in *Proc. 2018 IEEE 87th Vehicular Technology Conference* (*VTC Spring*), Porto, Portugal, 2018, pp. 1–5.

[11] S. Homayouni, S. Schwarz, and M. Rupp. Impact of SIR estimation on feedback reduction during heavy crowd events in 4G/5G networks, in *Proc. 2019 International Conference on Systems, Signals and Image Processing* (*IWSSIP*), Osijek, Croatia, 2019, pp. 129–132.

[12] K. Park, J. Rhee, M. Shin, and W. Hur, A novel CQI feedback channel for cellular UAV system, in *Proc. 2019 International Conference on Advanced Technologies for Communications* (*ATC*), Hanoi, Vietnam, 2019, pp. 84–88.

[13] G. Pocovi, A. A. Esswie, and K. I. Pedersen, Channel quality feedback enhancements for accurate URLLC link adaptation in 5G systems, in *Proc. 2020 IEEE 91st Vehicular Technology Conference* (*VTC2020-Spring*), Antwerp, Belgium, 2020, pp. 1–6.

[14] S. K. Vankayala and K. G. Shenoy, A neural network for estimating CQI in 5G communication systems, in *Proc. 2020 IEEE Wireless Communications and Networking Conference Workshops* (*WCNCW*), Seoul, Republic of Korea, 2020, pp. 1–5.

[15] G. Piro, L. A. Grieco, G. Boggia, and P. Camarda, A two-level scheduling algorithm for QoS support in the downlink of LTE cellular networks, in *Proc. 2010 European Wireless Conference* (*EW*), Lucca, Italy, 2010, pp. 246–253.

[16] 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.

[17] I. Constandache, S. Gaonkar, M. Sayler, R. R. Choudhury, and L. Cox, EnLoc: Energy-efficient localization for mobile phones, in *Proc. IEEE INFOCOM 2009*, Rio de Janeiro, Brazil, 2009, pp. 2716–2720.

[18] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, NJ, USA: Prentice Hall, 1994.

[19] M. Belgiovine, K. Sankhe, C. Bocanegra, D. Roy, and K. R. Chowdhury, Deep learning at the edge for channel estimation in beyond-5G massive MIMO, *IEEE Wireless Communications*, vol. 28, no. 2, pp. 19–25, 2021.

[20] D. Minovski, N. Ogren, C. Ahlund, and K. Mitra, Throughput prediction using machine learning in LTE and 5G networks, *IEEE Transactions on Mobile Computing*, doi: 10.1109/TMC.2021.3099397.

[21] M. Uzair and N. Jamil, Effects of hidden layers on the efficiency of neural networks, in *Proc. IEEE 23rd international multitopic conference* (*INMIC*), Bahawalpur, Pakistan, 2020, pp. 1–6.

[22] S. Haykin, *Neural Networks and Learning Machines, Third Edition*. Upper Saddle River, NJ, USA: Prentice Hall, 1999.

[23] A. Balieiro, P. Yoshioka, K. Dias, D. Cavalcanti, and C. Cordeiro, A multi-objective genetic optimization for spectrum sensing in cognitive radio, *Expert Systems with Applications*, vol. 41, no. 8, pp. 3640–3650, 2014.

[24] D. Raca, D. Leahy, C. J. Sreenan, and J. J. Quinlan, Beyond throughput, the next generation: A 5G dataset with channel and context metrics, in *Proc. 11th ACM Multimedia Systems Conference*, New York, NY, USA, 2020, pp. 303–308.

Kelvin Dias received the PhD degree in computer science from Federal University of Pernambuco (UFPE), Recife-Brazil, in 2004. Since 2010, he has been an associate professor at the Centro de Informática (CIn) of the UFPE. His current research interests are software-defined networking, network functions virtualization, cognitive radio, software-defined radio, fog computing, multi-access edge computing, and 5G networks.

Paulo Guarda specialized in network and telecommunication system engineering from Instituto Nacional de Telecomunicações (INATEL) in 2006. Since 2011 he has been a mobile phone product test project manager in Motorola Mobility.

Andson Balieiro received the PhD degree in computer science from Universidade Federal de Pernambuco, Brasil in 2015. Currently, he is a professor at the Centro de Informática (CIn) of the UFPE. He has worked in R&D projects funded by companies (e.g., Morotola Mobility and Ericsson) and governmental institution (e.g., Brazilian National Council for Scientific and Technological Development-CNPQ). He won the FET Best Paper Award in the 31st Wireless and Optical Communications Conference (WOCC). His research interests are ultra-reliable and low latency communications, 5G/6G networks and their key-enablers such as cognitive radio, network slicing, network function virtualization, multi-access edge computing, and software-defined networking besides the IA-based 5G/6G networks.