IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 15 August 2023, accepted 26 August 2023, date of publication 29 August 2023, date of current version 5 September 2023.
Digital Object Identifier 10.1109/ACCESS.2023.3309870

RESEARCH ARTICLE

Using Neural Network and Levenberg–Marquardt Algorithm for Link Adaptation Strategy in Vehicular Ad Hoc Network

ETIENNE ALAIN FEUKEU, (Member, IEEE), AND MBUYU SUMBWANYAMBE, (Member, IEEE)
Department of Electrical and Mining Engineering, University of South Africa (UNISA), Florida Campus, Johannesburg 1709, South Africa
Corresponding author: Etienne Alain Feukeu (feukeu@gmail.com)

ABSTRACT Vehicular Ad Hoc Network (VANET) was initiated about two decades ago in view of saving lives by mitigating and reducing the number of accidents and incidents on public roads. Moreover, this objective can only be achieved if VANETs mobiles regularly exchange Road State Information (RSI) with their neighborhood and take decisive actions based on the RSI received. Therefore, it becomes paramount to ensure that the transmitted message is well received. And this is only possible if the quality of the sharing medium or link is controlled, and transmission performed while taking into consideration the Channel State Information (CSI). The CSI provides information related to channel quality, Signal-to-Noise Ratio (SNR), and so forth. The process of adapting the payload as a function of the CSI is called Link Adaptation (LA). Several LA works have already been published in VANETs, but almost without serious consideration of the effect of the relative mobility amongst the nodes. Hence, while taking into consideration the Doppler Shift induced by the relative velocity, the current work presents a link adaptation strategy using a Neural Network (NN) and the Levenberg-Marquardt algorithm in VANETs. The simulation results definitively demonstrate that the NN approach outperforms its counterparts by a significant margin. It achieves a performance of 1075% in transmission duration, 180% in transmitted bit, and 115% in model efficiency when compared to the Cte, ARF, and AMC algorithms, respectively.

INDEX TERMS VANET, machine learning, link adaptation, WAVE, V2V, V2I, neural networks.

I. INTRODUCTION

Vehicles are regarded as wireless nodes in VANETs and regularly share Road State Information (RSI) with their neighbors. The RSI enables the participating mobile to constantly be aware of the position, velocity, and direction of all other mobiles nearby. The Wireless Access in Vehicular Environment (WAVE) standard, also known as Dedicated Short-Range Communication (DSRC), was created to enable and facilitate inter-vehicular communication [1], [2]. Operating at a frequency of 5.9 GHz, the WAVE standard is part of the Federal Highway Administration's Vehicle Infrastructure Integration (VII) for developing Intelligent Transportation Systems (ITS) [3]. Under the WAVE standard, a vehicle can connect with other vehicles using the Vehicle-to-Vehicle (V2V) protocol or with other wireless communication infrastructure using the Vehicle-to- Infrastructure (V2I) protocol. The WAVE standard is composed of two other standards named IEEE 802.11p, designed to handle all operations related to the Medium Access Control (MAC) and the physical layers (PHY), and the IEEE 1609 standard, which focuses more on handling all operations performed by the upper layers [3].

In the context of the safety application, the success of the VANETs relies essentially on two types of message dissemination between the involved nodes. The first type, known as Cooperative Awareness Messages (CAMs) which are transmitted regularly [4], possesses all information related to the whereabouts of all other mobiles in the vicinity of the involved nodes. CAMs are sort of the RSI awareness notification messages transmitted at the frequency of 1 to 10 Hz. The second type, named Decentralized Environmental

The associate editor coordinating the review of this manuscript and approving it for publication was Jie Gao.

Notification Messages (DENMs), are high-priority notification messages with extremely low latency which are only generated in case of emergency (accident, collision). DENMs notifications are time sensitive and will perish if not delivered on time. The DENMs are very important messages notification type that carry information about preventing accidents or other incidents that might assist arriving cars before they reach the accident or incident scene. These two message types are of prime importance in VANETs and are often exchanged between nodes in the context of safety applications [5]. Besides, some auxiliary advantages like electronic toll gate payment, a la carte service, infotainment, and so on, the successful transmission, effective handling, and management of the CAMs and the DENMs messages constitute the central part of the VANETs.

Taking advantage of this technology and in the seek to reduce the number of accident risks on public roads, the DSRC transceiver has since become mandatory for new vehicles manufactured in North America [6]. With more than 30 zettabytes of data generated by connected cars, it is estimated that the worldwide V2X industry will exceed US$100 billion in the coming years [7].

In view to improve on the current standard, the IEEE802.11 bd was developed to enhance and replace the actual IEEE802.11p standard. Nevertheless, despite many advancements, there is still a crucial problem of matching the transmission rate and mode to the extremely dynamic channel conditions in a vehicular environment that must be resolved [8]. Any mobile communication subjected to this network will have sporadic signal degradation proportional to the incurred relative speeds. The other big issue is that to design an effective LA model, one needs to simulate the communication module under all possible variations of channel realization. Hence, such exercise is very cumbersome, fastidious, and time-consuming. In the past, these ITS characteristics were and continue to be a mere challenge hindering the development of the LA algorithm for the ITS environment. Additionally, besides the higher mobility of the involved nodes, the IEEE 802.11p PHY layer also makes use of Orthogonal Frequency Division Multiplexing (OFDM) which is extremely sensitive to Doppler Shift (DS) [9]. Consequently, the design of an effective LA strategy that takes into consideration the higher mobility factor of the vehicular environment remains of prime importance [8].

Unlike the work proposed by some authors [10], [11], and [12] which focused on LA and Doppler Shift mitigation in VANETs, the current work proposes a Link Adaptation strategy using a Neural Network and the Levenberg-Marquardt Algorithm to address the challenging LA problem in VANETs. The proposed work starts by analyzing the problem to find out how such a complex problem can be modeled using ML. The dataset used in this work was obtained from the work generated by [12]. The dataset is arranged, and the data preparation is performed in line with the ML criteria. Then the LA model is proposed and the choice of the selected NN algorithm is motivated and justified. The model is trained, tested, and validated using the provided dataset to obtain the final model response. Thereafter, the work ended by evaluating and testing the final response against its peers LA strategies ARF, AMC, and Cte. The results from validation tests demonstrated the outperformance of the NN over its selected peers with 1075 %, 180%, and 115% performance in relation to the transmission duration, transmitted bit, and model efficiency, respectively.

As a main contribution, this work:

1-Presents a state-of-the-art method to translate a real-life problem into a mathematical model.

2-Demonstrates and presents how to use the Levenberg-Marquardt algorithm to solve LA problems.

3-Demonstrates a convenient way to model LA strategy using NN.

The paper is arranged as follows: a literature review is presented in Sect. II. The data preparation is presented in Sect. III. The proposed ML algorithm is presented in Sect. IV. Followed by the model training in Sect. V. The model testing and evaluation are described in Sect. VI. And finally, the conclusion is presented in Sect. VII.

II. LITERATURE REVIEW

Several works have already been published in the field of LA. Moreover, some recent works, such as [13], [14], and [15], do not employ ML in their strategies and are not tailored for the VANETs environment. Many well-known works, such as [16], [17], and [18], were developed for the legacy IEEE 802.11 that was created for conventional Wireless Local Area Networks (WLAN). Auto Rate Fallback (ARF) is one of the most used rate adaptation mechanisms in wireless environments [16]. In the ARF strategy, a rate upshift is performed after ten consecutive successful frame transmissions, whereas a rate downshift is performed after two consecutive frame transmission failures. ARF, however, is unable to respond quickly to a channel that is changing rapidly since it takes 10 successful frame transmissions to boost the transmission rate. Numerous ARF techniques based on various up/down counter algorithms have been incorporated in most firmware [19]. The adaptive auto rate fallback (AARF) is one of the enhancements to the ARF [17]. The AARF's goal is to improve ARF performance in a channel with gradual fading. Every time AARF tries to increase the transmission rate and the ensuing packet transfer fails, the threshold is doubled.

In general, very few research works have successfully addressed the LA in the context of VANETs while taking increased mobility into account. In view to develop the mathematical model for each MCS, Khaldoun examined the effects of Doppler Shift (DS) as a function of relative speed on the signal quality [20]. Furthermore, simulation experiments were conducted, and an Adaptive Modulation Coding (AMC) scheme was developed, simulated, and tested. Additionally, although restricted to a maximum DS range of 500 Hz, which is comparable to about 92 km/h, the AMC approach exhibits improved performance compared to its peers. Alternatively, the minstrel algorithm was proposed by [21] to solve the LA problem encountered in IEEE 802.11 standards. This algorithm selects its bit rate based on which rate can reach maximum throughput. It considers the expected number of retransmissions based on the statistical history of the wireless channel. To achieve the optimal data rate, the minstrel algorithm uses three main components made of the retry chain mechanism, the rate decision process, and the statistical calculations [21]. First, a multi-rate retry chain is used to select the best data rate whenever a short-term variation in channel quality occurs. Second, a rate selection which defines the rate of normal and sampling transmission used. Then, the third part focuses on statistical calculations.

The Packet Rate Adaptation based on the Bloom filter (PRAB) protocol to mitigate the hidden terminal collision by adapting the packet generation rate is presented in [22]. The paper derived the optimal formulation for the packet generation rate based on the number of hidden terminals experienced by each transmitter-receiver pair. To calculate the number of hidden terminals, the protocol used the efficient Bloom filter data structure for piggybacking 1-hop neighbor set. The simulation results demonstrated that the PRAB protocol increases the packet reception probability to 90% in contrast to IEEE 802.11p's 68%, NORAC's 75%, and FABRIC's 72.5% even in very high-density networks. In addition, it delivers BSM packets faster (lower PRD). The performance improvement in PRP and PRD comes at the negligible cost of marginally higher PRI.

In terms of VANETs LA strategies using Machine Learning (ML) techniques, very few works have been published in the literature. The authors in [23] investigated a machine learning method of link adaptation which analytically characterizes the adaptation problem to maximize the system throughput while maintaining transmission reliability. In order to maximize the whole system throughput, the setup uses an Nt x Nr MIMO system and employs an autoencoder model and a multi-class SVM to select MCS through SNRs for MIMO-OFDM systems. Furthermore, a scheme based on a channel matrix is investigated to select spatial mode and MCS for MIMO systems, where an autoencoder is used to extract features from CSI and the match probability of each MCS is then derived through a SoftMax model. Simulation results demonstrated the improved performance of the proposed algorithms. Despite the approach's performance, due to its extremely low applicable mobile velocity (3 km/h), the model is not fit for a VANET environment. A Study on Link Adaptation Techniques for IEEE 802.11bd Based eV2X Communications was presented in [24] to evaluate various link adaptation techniques. In the proposed approach, the communication between two vehicles is realized using single-link and multi-connectivity communications. The link-level performance of IEEE 802.11bd is abstracted using different effective SINR mapping techniques. The model also includes the DCM and multi-connectivity communications with a maximum ratio combining (MRC). Then the link adaptation performance

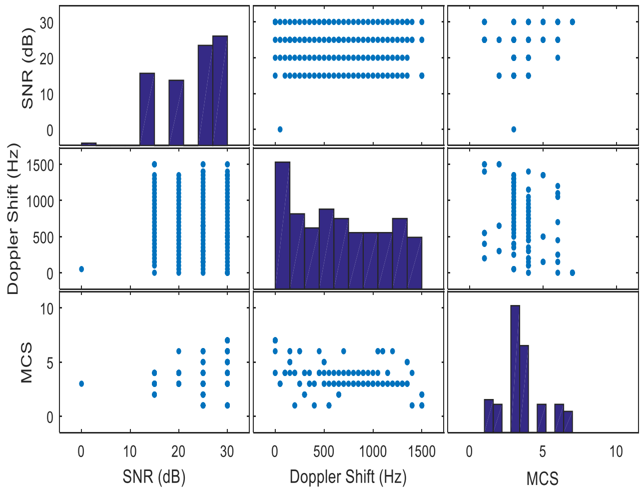

FIGURE 1. Data exploration.

of these mapping techniques is evaluated and compared to the optimal link adaptation in terms of achieved data rates and reliability. In the case of multi-connectivity, various link adaptation schemes are considered to adapt MCS and the number of links. Subsequently, an algorithm for joint adaptation of MCS and several multiple links is proposed to enable ultra-reliable communications. The evaluation results show that higher data rates can be achieved by utilizing the same or a lower number of simultaneous links.

A new link adaptation algorithm employing Deep Learning (DL) based channel prediction was proposed in [25]. The algorithm is derived by using two trained Deep Neural Networks (DNN) at the receiver to first estimate L and subsequent channel coefficients from past estimates. Then, considering the predicted channel condition, for every instant, the MCS L is chosen based on the SNR stylized profile and sent to the transmitter through a feedback link. In fact, to significantly reduces the number of feedback transmissions from the receiver to the transmitter. The proposed algorithm chooses the best MCS for a longer period of data transmission. The proposed DL-based algorithm achieves almost similar reliability and higher throughput compared to the rule-based link adaptation method using the Minimum Mean Square Error (MMSE) one-step channel predictor as the optimal link adaptation method in terms of reliability. Moreover, this algorithm heavily relies on a closed-loop system which is not appropriate in a VANETs environment to perform at its best.

This review demonstrates and pinpoints some limitations of the existing works. Many proposed LAs do not consider higher mobility and some few advanced works which consider the dynamics of the ITS environment make use of closed-loop feedback, historical data, or a combination of both. Conversely, any mobile in VANETs is very temporal, and communication between any two mobiles cannot always last for long. Consequently, building historical data becomes a challenge. Additionally, since each vehicle is independent and can move freely in any direction at any time, relying

TABLE 1. Refined ADSA MCS.

Doppler Shift value for SNR =15, 20, 25 and 30 dB																															
SNR	0	50	100	150	200	250	300	350	400	450	500	550	600	650	700	750	800	850	900	950	1000	1050	1100	1150	1200	1250	1300	1350	1400	1450	1500
15	4	4	4	4	4	3	4	4	4	4	3	3	3	3	4	3	3	4	3	3	3	3	3	4	4	3	3	3	3	3	6
20	4	6	6	6	6	3	6	6	4	4	4	4	4	3	3	4	3	3	3	4	4	4	3	3	3	3	3	3	3	3	3
25	6	4	4	5	1	6	2	3	3	4	3	6	5	2	4	5	3	6	6	3	3	6	6	4	3	4	4	3	1	4	1
30	7	3	4	6	4	6	4	3	1	6	5	1	3	4	6	4	4	4	4	6	6	6	6	4	6	3	3	5	4	5	6

on close-loop feedback becomes unrealistic in such an environment. Therefore, to overcome some of the drawbacks, the current work proposes an efficient LA for VANETs that, while taking into consideration the relative mobility of the involved nodes, does not actively rely on any feedback, historical data, or closed-loop method.

III. DATA PREPARATION

Making use of the refined Automatic Doppler Shift Adaptation (ADSA) MCS table generated by the authors in [12], the current work employs AI (ML) techniques to analyze the data in order to create a realistic model applicable in various VANET environments and instances. The data presented in Table 1 was generated after intense simulation under variable channel conditions in [12].

It should be noted that each entry in Table 1 corresponds to the interpolation of three distinct parameters: SNR, DS, and MCS. In this table, the MCS that ensures high throughput is selected based on the mobile velocity and the available SNR. This Table, which is our training set, cannot be directly used in ML due to its non-compliance with the ML data representation convention. To utilize it in ML, the columns and rows need to be rearranged according to the ML data presentation convention. The data should be divided into two groups: one representing the input features and the other representing the target (labels).

In our case, the input features group consists of SNR values and DS values, while the target group consists of MCS values.

In the training set table, it can be observed that this data has 30 columns of 4 rows. Each entry on the table corresponds to a specific SNR and DS which ensure successful data transmission under these channel conditions. For instance, considering the first entry (row 0), at the SNR of 15 dB, with a DS of 0 Hz, the maximum achievable MCS is a QPSK rate 3/4.

Similarly, in the last entry (row 4), at the SNR of 30 dB, and a DS of 1500 Hz, the maximum achievable MCS is a 16QAM rate 3/4. It should be noted that the DS range of 0 to 1500 Hz is to cater for a relative mobile velocity of 0 to 250 km/h.

The relationship between DS and the relative mobile velocity is defined in equation 1.

$$DS = \pm \frac{f_c V cos\beta}{C} \tag{1}$$

where DS is the change in frequency of the source seen at the receiver, fc the frequency of the source, V the speed difference between the source and transmitter, C the speed of light, and $\beta =$ angle of velocity vector.

To explore and get more insight into the data, Figure 1 was computed to clarify the correlation between the SNR, the DS, and the MCS. In that Figure, looking at the relationship between the SNR and the MCS, a general observation does not really provide any recognized pattern.

However, it can be observed that higher MCS is only achieved with SNR above 20 dB, where MCS 5, and 6 can be selected apart from MCS 7, which is only possible at 30 dB. Now looking at MCS and the DS, it can be seen that MCS 7 is only possible at a DS of 0 Hz. It can also be observed that MCS 3, 4, and 6 are denser in comparison to others. The same Figure also shows that as the DS increases beyond 1200 Hz, higher MCS is no longer possible. Finally, since the scatter plot of these Figures does not reveal any consistent recognized pattern, the only way to generate a realistic model is to turn to the power of the ML which can deal with complex problems.

The procedure will consist of first selecting the right ML algorithm to use, then training the selected algorithm with the available data so that the ML can learn from the data, understand it, and generate an acceptable, realistic model accordingly. The choice of the selected ML algorithm is discussed in the subsequent section.

IV. MACHINE LEARNING ALGORITHM

Before choosing the suitable algorithm, it is very important to first and foremost understand the problem that we are trying to solve. In the IEEE802.11p standard which defines the wireless parameters based on the VANET's characteristics, 8 MCS are available as presented in Table 2 [6].

In Table 2, MCS 1 represents BPSK rate 1/2 (lowest modulation with higher robustness), generally used in case of fluctuation or bad wireless channel quality while MCS 8 represents 64 QAM rate 3/4 (highest modulation with very low resilience) generally used in case of very good wireless channel quality. It should be noted that the higher the MCS, the higher the throughput, the faster the communication, and the higher the system efficiency. In general, the communication system will try to always achieve the best efficiency for the given channel condition. In our case, the suitable algorithm will always be able to select the right MCS which will provide higher throughput given the SNR and the DS values. Simply speaking, the problem is to select a particular MCS given a SNR and a DS value. Since the selected MCS

TABLE 2. Modulation code scheme IEEE802.11p.

Modulation	Coded Bit Rate (Mbps)	Coding Rate	Data Rate (Mbps)	Data Bit per OFDM Symbol	MCS
BPSK	6	1/2	3	24	1
BPSK	6	3/4	4.5	36	2
DPSK	12	1/2	6	48	3
QPSK	24	3/4	9	72	4
16-QAM	24	1/2	12	96	5
16-QAM	24	3/4	18	144	6
64-QAM	36	2/3	24	192	7
64-QAM	36	3/4	27	216	8

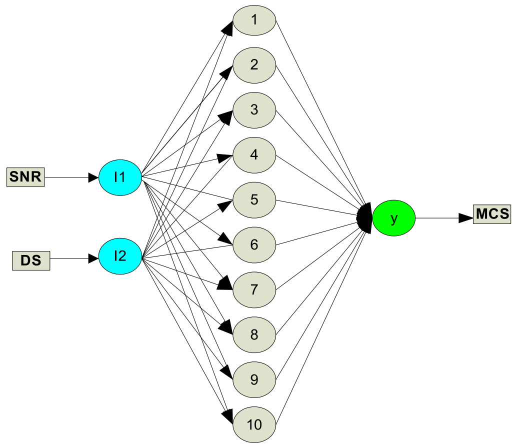

FIGURE 2. Feed forward neural network.

has a specific invariable value, it can be seen as a categorical nominal variable while the SNR and the DS can be seen as continuous variables. Therefore, a specific MCS will only be selected if the SNR and the DS values have reached a certain threshold value. Viewed from this perspective, it can be said that continuous variables are used to select a categorical variable. In the field of AI, any problem having these specific characteristics can be modeled using basic ML techniques (linear or logistic regression) or advanced ML such as Neural Networks (NN). Moreover, based on the complexity, the NN will be the suitable choice in this case. In general, NNs can learn and model nonlinear and complicated interactions between inputs and outputs; make generalizations and inferences; uncover hidden correlations, patterns, and predictions; and model highly volatile data and variances required to anticipate unusual occurrences [26]. Since we are dealing with a single target or label (MCS) selection based on input parameter values or features (SNR and DS), the model will have two input layers, one hidden layer made of 10 neurons, and one output layer.

A. NEURAL NETWORK MODEL

Inspired by biological nervous systems, a neural network combines several processing layers, using simple elements operating in parallel. The network consists of an input layer, one or more hidden layers, and an output layer. In each layer, there are several nodes or neurons, and the nodes in each layer use the outputs of all nodes in the previous layer as inputs, such that all neurons interconnect with each other through the different layers. Each neuron is typically assigned a weight that is adjusted during the learning process. These weights are automatically adjusted during training according to a specified learning rule until the artificial neural network performs the desired task correctly. A decrease or increase in the weight changes the strength of the neuron's signal [27]. The proposed and implemented NN is illustrated in Figure 2.

The type of NN implemented in the current work can also be called a FeedForward shallow Neural Network (FFNN) because it is made of a single hidden layer. The input layers I1 and I2 are connected to the SNR and DS vectors, respectively. The hidden layer consists of 10 neurons, numbered from 1 to 10, and the output layer denoted as "y", will output the selected MCS. It is noted that to facilitate error calculation and minimize the loss using the chain rules and backpropagation technique, the middle layer neurons use a sigmoid activation function while the upper layer uses the Rectified Linear Unit (ReLU) activation function. The sigmoid function is essential because the step function contains only flat segments, so there is no gradient to work with (Gradient Descent cannot move on a flat surface), while the sigmoid function has a well-defined nonzero derivative everywhere, allowing Gradient Descent to make some progress at every step. On the other hand, although the ReLU (z) function is continuous and not differentiable at $z = 0$, it works very well and is fast to compute. After building the NN, the next step is choosing of the optimization algorithm. Therefore, to ensure fast convergence, a smart optimization algorithm called the "Levenberg-Marquardt Algorithm (LMA) was chosen. The original description of the Levenberg-Marquardt algorithm is given in [28]. The application of Levenberg-Marquardt to neural network training is described in [29] and starts on pages 12-19 of [30]. This algorithm appears to be the fastest method for training moderate-sized feedforward neural networks (up to several hundred weights). The particularity of LMA is that it uses the combination of steepest descent and the Gauss-Newton method to always achieve the best result. The LMA is an iterative technique that locates the minimum of a function that is expressed as the sum of squares of nonlinear functions. Due to its performance in the field, it has become a standard technique for nonlinear least-squares problems. Further details on the proposed optimization algorithm (LMA) can be found in [28], [29], and [30].

In a NN, to minimize the network loss or error, one needs to iteratively find the weights that satisfy the optimal condition. This is done by computing the weights using equation 2.

$$W^* = \underset{W}{\operatorname{argmin}} \frac{1}{n} \sum\nolimits_{i=1}^{n} \mathcal{L}\left(f\left(x^{(i)}; W\right), y^{(i)}\right)$$
$$W^* = \underset{W}{\operatorname{argmin}} \, \mathrm{J}(W)$$

Remember:

$$W = \left\{ W^{(0)}, W^{(1)}, \ldots .. \right\} \tag{2}$$

Having described the NN, the next step will consist of applying the Levenberg-Marquardt (LMA) optimization algorithm to train the network in order to achieve the best performance with minimal error. For simplicity, the LMA employed in this work as described in [28], [29], and [30] is summarized in the subsequent sentences. Based on the problem at hand, the specific steps required to predict and select the right MCS as a function of the SNR and the DS are as follows.

1) Start by choosing initial parameters for training error ε, μ_0, θ and the weight w^0. Then let k=0 and $\mu = \mu_0$.

2) Then the error output and network index function $E\left(w^k\right)$ is calculated using equation 3 as follows.

$$E(w) = \frac{1}{2}\sum_{q=1}^{N}\left(y_q - \hat{y}_q\right)^2 = \frac{1}{2}\sum_{q=1}^{N} e^2 \tag{3}$$

3) And the Jacobian matrix J(w) of the partial derivative e(w) is calculated using equation 4.

$$J = \begin{bmatrix} \frac{\partial e_1}{\partial w_1} & \frac{\partial e_1}{\partial w_2} & & \frac{\partial e_1}{\partial w_n} \\ \frac{\partial e_2}{\partial w_1} & \frac{\partial e_2}{\partial w_2} & \cdots & \frac{\partial e_2}{\partial w_n} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial e_N}{\partial w_1} & \frac{\partial e_N}{\partial w_2} & & \frac{\partial e_N}{\partial w_n} \end{bmatrix}$$
$$= \begin{bmatrix} \frac{\partial e_1}{\partial w^h_{1,1}} & \frac{\partial e_1}{\partial w^h_{1,2}} & & \frac{\partial e_1}{\partial w^h_{s^h,R}} & \frac{\partial e_1}{\partial w^o_{1,1}} & & \frac{\partial e_1}{\partial w^o_{1,s^h}} \\ \frac{\partial e_2}{\partial w^h_{1,1}} & \frac{\partial e_2}{\partial w^h_{1,2}} & \cdots & \frac{\partial e_2}{\partial w^h_{s^h,R}} & \frac{\partial e_2}{\partial w^o_{1,1}} & \cdots & \frac{\partial e_2}{\partial w^o_{1,s^h}} \\ \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ \frac{\partial e_N}{\partial w^h_{1,1}} & \frac{\partial e_N}{\partial w^h_{1,2}} & & \frac{\partial e_N}{\partial w^h_{s^h,R}} & \frac{\partial e_N}{\partial w^o_{1,1}} & & \frac{\partial e_N}{\partial w^o_{1,s^h}} \end{bmatrix} \tag{4}$$

4) Calculate the weights increment of the network Δw using equation 5 as presented below.

$$\Delta w = -\left(J^T(w) J(w) + \mu I\right)^{-1} J^T(w) e(w) \tag{5}$$

5) Then the error output and network index function $E\left(w^k\right)$ is calculated using equation 3.

6) If $E\left(w^k\right) < \varepsilon$ go to step 8. Else calculate w^{k+1} and $E\left(w^{k+1}\right)$ using equation 6 and 3, respectively.

$$w^{k+1} = w^k + \Delta w \tag{6}$$

7) If $E\left(w^{k+1}\right) < E\left(w^k\right)$, let k=k+1, $\mu = \frac{\mu}{\theta}$, go back to step 2. Otherwise $w^{k+1} = w^k$, then $\mu = \theta\mu$ and go to step 4.

8) The stop condition is achieved if $E\left(w^k\right) < \varepsilon$ or when the maximum number of iterations has been reached.

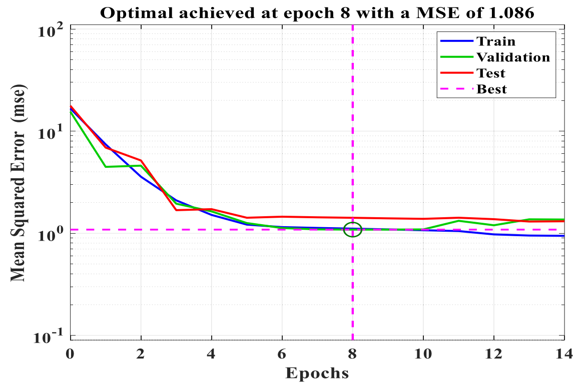

FIGURE 3. Training performance curve.

V. MODEL TRAINING

The proposed NN was built and implemented in MATLAB. The obtained training and validation results are depicted in Figures 3 and 4, while the model response after training is presented in Figure 5. In both Figures 3 and 4, the training data made up of 70% of the whole dataset is represented in blue color while the test data (15% of the whole dataset) is represented in red, and the validation data (15% of the whole dataset) is represented in green. These figures help us to analyze and appreciate the performance of the developed model in relation to our dataset.

The observation of the training performance curve presented in Figure 3 shows how the MSE starts at around 18 and diminishes progressively as the training continues until the optimal point of 1.086. At the optimal point which corresponds to 8 epochs and the MSE of 1.086, it was no longer possible to achieve better than this point. Conversely, a continuous training beyond the optimal point will instead degrade the model's performance. This is why the validation error starts rising from the 10^{th} epoch onward. Therefore, the best parameters that ensure the optimal performance of our model correspond to epoch 8 where the validation error is minimal.

The error histogram is presented in Figure 4. This Figure presents the error (training, test, and validation) distribution over the whole dataset. The observations in this Figure reveal that most data are centered between the bin -1.526 and 1.658 where the maximum number of instances occurred with minimal errors. Due to the nature of the data, this histogram also shows that there are a total of 5 training (bin) and 1 validation (bin) errors on both extremes of the distribution. This is surely due to the outlier's presence in the training dataset. Because the outliers are just some few points placed far away from the mean, and since the splitting is performed automatically over the whole dataset range, that is why some outliers didn't have sufficient testing and validation data. This lack of enough information about the outliers' variances then resulted in high or maximum number errors at these specific points. This can also be proven by observing the scatterplot Figure obtained before the feature mean normalization.

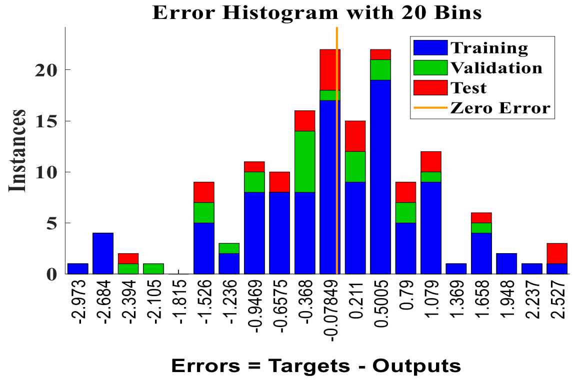

FIGURE 4. Error histogram.

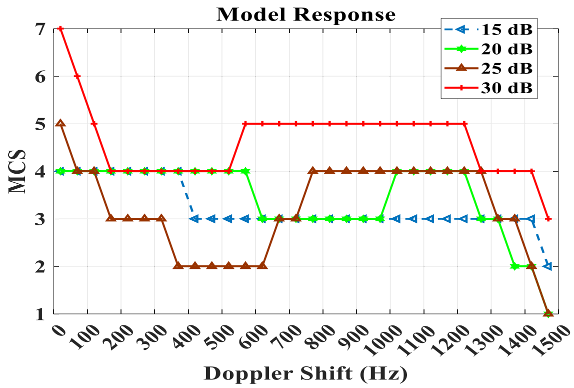

FIGURE 5. Trained model response.

After the training and validation phase using the supplied data, the final model response in terms of SNR, DS and MCS was computed and depicted in Figure 5. This Figure describes the dynamic behavior of the model response in relation to its inputs (SNR, DS) and the label (MCS). The overall observation of this response shows that MCS 6 and 7 are only possible for a DS less than 100 Hz with a SNR greater or equal to 30 dB.

The non-linearity behavior induced by the DS in a VANET can also be perceived and justified by the fact that each curve is affected differently at the same DS value in comparison to others. This reality is clarified by observing the DS values ranging from 400 to 600 Hz. This Figure further shows that beyond a DS value of 1300 Hz and a SNR less than 30 dB, the maximum achievable label is MCS3 which technically corresponds to QPSK rate 1/2. Having trained, evaluated, and validated our model, the next step will consist of evaluating and comparing the performance of the developed model against its peers in the field of VANETs. This model response is particularly important in the sense that it will be used during the testing phase to verify the performance of our developed NN approach.

VI. MODEL TESTING AND EVALUATION

To evaluate and appreciate the real performance of the developed NN, further simulations were performed to compare the strengths of the NN against its peers Auto Rate Fallback (ARF) [16], Constant modulation (Cte) which is a BPSK rate 1/2, and the Adaptive Modulation Code (AMC) proposed in [20]. The work makes use of a channel model presented in Figure 6 as developed and proposed by [12] where all channel components and subsystems were described and explained. To recall, the channel model as depicted in Figure 6 was made of four main block systems, namely Transmitter (TX), Channel, Receiver (RX), and Bit Error Rate (BER) calculator. The model used a combination of an Additive White Gaussian Noise (AWGN) and a Rayleigh channel. Further clarification can be found in [12].

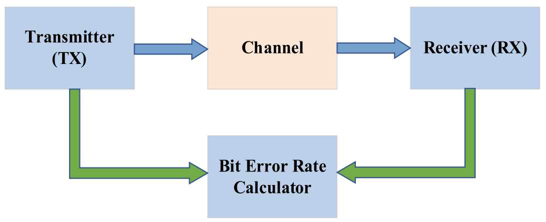

FIGURE 6. Channel model.

TABLE 3. Simulation parameters.

Parameter	Value	Parameter	Value
Pilot	2 Bytes	Number of Frame	100
Carrier Frequency	5.85 GHz	Frame Size	8000 symbols
Speed of light	300 000 Km/s	SNR Max	30 dB
Angle of signal arrival	0 deg	Delay Spread	3 us
Sample time	1 us	OFDM Sym duration	8 us
Number of subcarrier	48	Relative velocity	0 to 140 Km/h

Making use of the simulation parameters presented in Table 3, the model was evaluated in terms of transmission duration, transmitted bits, transmission efficiency and model performance. The simulation of each approach was carried out under variable Signal to Noise Ratio (SNR) and variable mobility expressed in Doppler Shift (DS).

Figure 7 represents the transmission duration while the number of transmitted bits for each approach is depicted in Figure 8. The analysis of Figure 7 shows how the DS affects each approach differently. The worst performer in this case is the AMC which spent a large amount of time (45 seconds or more) transmitting at DS values around 400, 1000 and 1250 Hz. This deficient performance is followed by the Cte strategy. Cte also spent about 35 seconds while transmitting at the DS value around 1250 Hz. However, both ARF and NN performed well despite the small outperformance of NN over ARF making NN the best performer in terms of transmission duration.

Now looking at Figure 8, although at different values, both Cte and ARF consistently transmitted the same number of bits under variable mobility over the whole simulation duration. However, for the case of Cte, this is not a surprise because it is always a fixed rate transmission. But in the case of ARF, this may be justified by its slow reaction to change. As for the other two strategies (AMC and NN), the variation of the

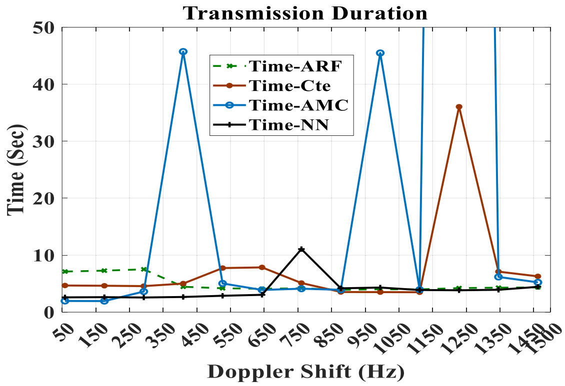

FIGURE 7. Transmission duration.

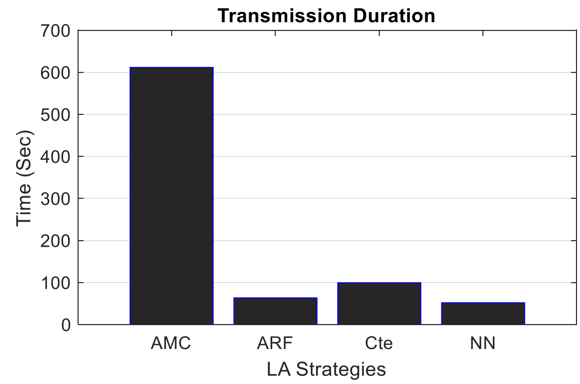

FIGURE 9. Transmission duration.

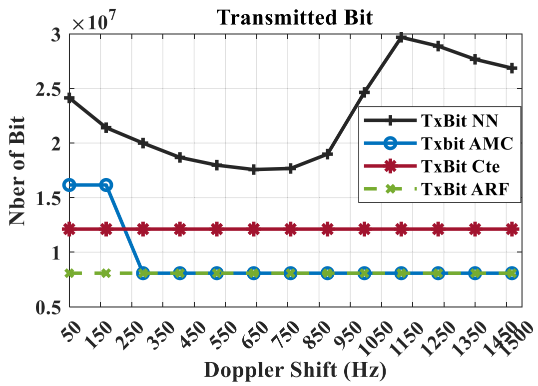

FIGURE 8. Transmitted bits.

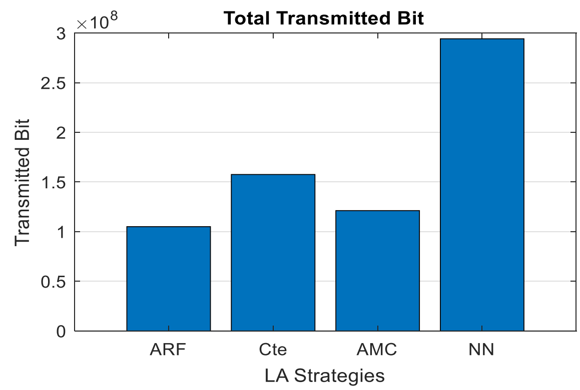

FIGURE 10. Transmitted bits.

transmitted number of bits can be clearly perceived. In this Figure, the best performer is still the NN with an extremely high performance compared to all other peers along the whole simulation range.

To further assess, confirm, and ascertain the robustness of the NN over its peers, Figures 9 and 10 were also computed. Figure 9 displays the total transmission duration while Figure 10 portrays the total number of bits transmitted by each LA strategy over the entire simulation runtime. The analysis of Figure 9 shows that the NN took a noticeably short time to transmit all its data in comparison to others. This performance is followed by the ARF and the Cte, respectively. The worst performer here is the AMC with more than 600 seconds in comparison to about 50 seconds spent by the NN during their respective data transmissions.

Now looking at the number of transmitted bits as depicted in Figure 10, it is remarkably interesting to see that although the NN spent the least amount of time on its data transmission, it also transmitted a higher number of bits in comparison to its peers. This behavior further confirms and approves the effectiveness of the AI over all other strategies. In this Figure, the best performer is still the NN followed by the Cte.

To clearly understand and perceive the whole simulation result, a recapitulative table summarizing all achieved results for each strategy is presented in Table 4.

Using the values presented in this table, the overall model performance was computed and depicted in Figure 11. In this Figure, ARF was taken as a reference point because, with the

TABLE 4. Simulation result.

	ARF	Cte	AMC	NN
Efficiency	0.4430	0.6295	0.9152	0.9551
Runtime	63.6840	99.6192	611.7129	52.0483
TxBit	105040000	157560000	121200000	294213000

exception of the transmission duration, its underperformance against its counterpart can be seen in all scenarios. Similarly, due to its large amount of time spent during transmission, the AMC was also taken as the reference point when computing the transmission duration performance. Then, comparing the results of each approach with those of ARF or AMC (transmission duration only), the overall performance of the whole simulation can be assessed using equations 7 or 8.

$$Ovef = \left(\frac{X_i - X_{\min}}{X_{\min}}\right) \times 100 \tag{7}$$

$$Ovef = \left(\frac{X_{Max} - X_i}{X_i}\right) \times 100 \tag{8}$$

In equation 7, $X_{\min}$ represents the ARF value because it has the minimum value compared to others in terms of efficiency and transmitted bit. In the same equation, X_i represent any other value (AMC, Cte or NN). In the case of transmission duration (runtime), equation 8 is used, and AMC is taken as a reference and represented as X_{Max} while all other values are represented by X_i.

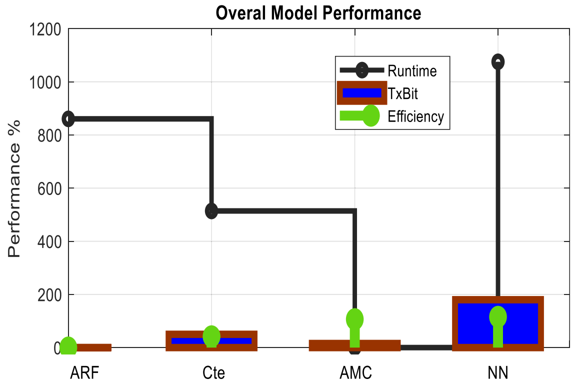

FIGURE 11. Overall model performance.

Figure 11 shows the global comparative performance (*Ovef*) of each approach in terms of transmission duration, transmitted bits, and strategy efficiency. The exploration of this Figure clearly confirms and ascertains the outperformance of the NN approach against its peers with 1075 %, 180%, and 115% performance in relation to the transmission duration, transmitted bit, and model efficiency, respectively. The second in line is the Cte with the best runtime and higher number of transmitted bits. The third best performer is AMC with higher efficiency and a higher number of transmitted bits in comparison to ARF.

Holistically, the exploration of all computed Figures as well as the derived findings proved that ML is a very powerful tool for complex problem resolution. The result of this work further emphasizes this reality by proving and confirming the outstanding performance of the NN against its peers in different simulation settings using variable performance metrics. This superior performance observed in various channel conditions and under variable mobile velocity was further confirmed from Figures 9 to 11. These results collectively emphasize the research's potential impact on improving VANET performance and addressing challenges in the domain of Intelligent Transportation Systems (ITS).

VII. CONCLUSION

The current work proposes a Link Adaptation (LA) strategy to mitigate one of the challenging problems that hinders the success of VANETs. An investigation was conducted to prove the significance of this problem in the WAVE environment. A solution was formulated, and a Neural Network (NN) algorithm was developed in accordance with the criteria for ML algorithm development. The proposed NN solution model was created, trained, and validated before being evaluated against its counterparts ARF, AMC and Cte in various channel conditions and under variable mobile velocity. A simulation result derived from this work demonstrated and proved the outperformance of the NN approach against its peers ARF, AMC and Cte with 1075 %, 180%, and 115% performance in relation to the transmission duration, transmitted bit, and model efficiency, respectively.

The significance of this study lies in its comprehensive approach to addressing a challenging problem in VANETs. It leverages mathematical modeling, advanced optimization algorithms, modern machine learning techniques, and practical implementation considerations. This study contributes to both the theoretical understanding of the problem and its practical application in real-world intelligent transportation systems.

As a main contribution, this work:

1-Presents a state-of-the-art method to translate a real-life problem into a mathematical model.

2-Demonstrates and presents how to use Levenberg-Marquardt algorithm to solve LA problems.

3-Demonstrates a convenient way to model LA strategy using NN.

REFERENCES

[1] *Vehicle Safety Communications -Applications (VSC-A) Final Report*, U.S. Dept. Transp., Nat. Highway Traffic Saf. Admin., Washington, DC, USA, 2011.

[2] D. Carona, A. Serrador, P. Mar, R. Abreu, N. Ferreira, T. Meireles, J. Matos, and J. Lopes, "A 802.11p prototype implementation," in *Proc. Intell. Vehicles Symp. (IV)*, 2010, pp. 1116–1121.

[3] K. Ram, C. Mujib, D. Matthew, A. Justin, M. Jim, S. Ian, B. Joe, A. Chris, W. Tim, and P. Frank. (Jan. 2, 2009). *Repository and Open Science Access Portal (ROSA P)*. [Online]. Available: https://rosap.ntl.bts.gov/view/dot/36804

[4] *(ITS), Intelligent Transport Systems; Communications, Vehicular; Applications, Basic Set of; Service, Part 2: Specification of Cooperative Awareness Basic*, Eur. Telecommun. Standards Inst., Sophia Antipolis, France, 2014.

[5] *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service*," Eur. Telecommun. Standards Inst., Sophia Antipolis, France, 2010.

[6] *IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks Specification Requirement—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY)*, IEEE Standard 802.11p, 2010.

[7] W. Xu, H. Zhou, N. Cheng, F. Lyu, W. Shi, J. Chen, and X. Chen, "Internet of Vehicles in big data era," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 1, pp. 19–35, Jan. 2018.

[8] W. Xu, H. Zhou, H. Wu, F. Lyu, N. Cheng, and X. Shen, "Intelligent link adaptation in 802.11 vehicular networks: Challenges and solutions," *IEEE Commun. Standards Mag.*, vol. 3, no. 1, pp. 12–18, Mar. 2019, doi: 10.1109/MCOMSTD.2019.1900005.

[9] E. Lawrey, "The suitability of OFDM as a modulation technique for wireless telecommunications, with a CDMA comparison," Ph.D. thesis, James Cook Univ., Townsville, QLD, Australia, 1997. [Online]. Available: http://skydsp.com/resources/Thesis_Eric_Lawrey_OFDM_vs_CDMA_old.pdf

[10] E. A. Feukeu, K. Djouani, and A. Kurien, "Doppler effect analysis and modulation code derivation," *Proc. Comput. Sci.*, vol. 10, pp. 1032–1038, Jan. 2012.

[11] E. A. Feukeu, K. Djouani. and A. Kurien, "Compensating the effect of Doppler shift in a vehicular network," in *Proc. AFRICON*, Pointe aux Piments, Mauritius, 2013, pp. 1–7.

[12] E. A. Feukeu, K. Djouani, and A. Kurien, "Performance evaluation of the ADSA in a vehicular network: MAC approach in IEEE 802.11p," *J. Ambient Intell. Hum. Comput.*, vol. 6, no. 3, pp. 351–360, Jun. 2015.

[13] M. M. I. Sheik and F. T. Pawan, "Performance analysis of link adaptation with MIMO and varying modulation and coderates for 5G systems," in *Proc. 3rd Int. Conf. Emerging Trends Elect., Electron. Commun. Eng. (ELECOM)*, Balaclava, Mauritius, 2020, pp. 222–228.

[14] M. Mitev, M. M. Butt, P. Sehier, A. Chorti, L. Rose, and A. Lehti, "Smart link adaptation and scheduling for IIoT," *IEEE Netw. Lett.*, vol. 4, no. 1, pp. 6–10, Mar. 2022.

[15] J. Khan and L. Jacob, "Link adaptation for multi-connectivity enabled 5G URLLC: Challenges and solutions," in *Proc. Int. Conf. Commun. Syst. Netw. (COMSNETS)*, Bengaluru, India, Jan. 2021, pp. 148–152.

[16] A. Kamerman and L. Monteban, "WaveLAN-II: A high-performance wireless LAN for the unlicensed band," *Bell Labs Tech. J.*, vol. 2, no. 3, pp. 118–133, 1997.
[17] M. Lacage, M. H. Manshaei, and T. Turletti, "IEEE 802.11 rate adaptation: A practical approach," in *Proc. 7th ACM Int. Symp. Modeling, Anal., Simulation Wireless Mobile Syst. (MSWiM)*, 2004, pp. 126–134.
[18] E. Feukeu, K. Djouani, and A. Kurien, "Using a greedy algorithm for link adaptation in MIMO wireless networks," in *Proc. IEEE Africon*, Victoria Falls, Zambia, Sep. 2011, pp. 1–7.
[19] K. Djouani, J. Chen, A. Akharraz, and K. Barkaoui, "Link adaptation for cooperative wireless LANs," in *Proc. 4th Int. Conf. Wireless Commun., Netw. Mobile Comput.*, Dalian, China, Oct. 2008, pp. 1–4.
[20] A. Khaldoun, M. Utayba, and A. Nizar, "Doppler shift impact on vehicular ad-hoc networks," *Can J. Multimedia Wireless Netw.*, vol. 2, no. 3, pp. 46–64, 2011.
[21] D. Xia, J. Hart, and Q. Fu, "Evaluation of the Minstrel rate adaptation algorithm in IEEE 802.11g WLANs," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Budapest, Hungary, Jun. 2013, pp. 2223–2228.
[22] S. Kumar and H. Kim, "Packet rate adaptation protocol based on Bloom filter for hidden node avoidance in vehicular ad-hoc networks," *IEEE Access*, vol. 7, pp. 137446–137460, 2019.
[23] Z. Dong, J. Shi, W. Wang, and X. Gao, "Machine learning based link adaptation method for MIMO system," in *Proc. IEEE 29th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Bologna, Italy, Sep. 2018, pp. 1226–1231.
[24] W. Anwar, N. Franchi, and G. Fettweis, "A study on link adaptation techniques for IEEE 802.11bd based eV2X communications," in *Proc. IEEE 94th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2021, pp. 1–7.
[25] M. Mehrabi, M. Mohammadkarimi, M. Ardakani, and Y. Jing, "Decision directed channel estimation based on deep neural network k-step predictor for MIMO communications in 5G," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 11, pp. 2443–2456, Nov. 2019.
[26] J. Kaur. (Nov. 17, 2022). *Deep Learning vs Machine Learning vs Neural Network—What's The Difference?* [Online]. Available: https://www.xenonstack.com/blog/deep-learning-vs-ml-vs-neural-network
[27] MathWorks. (May 15, 2023). *What is Neural Network? 3 Things You Need to Know*. [Online]. Available: https://www.mathworks.com/discovery/neural-network.html
[28] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *J. Soc. Ind. Appl. Math.*, vol. 11, no. 2, pp. 431–441, Jun. 1963.
[29] M. T. Hagan and M. Menhaj, "Training feedforward networks with the Marquardt algorithm," *IEEE Trans. Neural Netw.*, vol. 5, no. 6, pp. 989–993, Nov. 1994.
[30] M. Hagan, H. Demuth, and M. Beale, *Neural Network Design*. Boston, MA, USA: PWS Publishing, 1996.
[31] M. T. Hagan, H. B. Demuth, M. H. Beale and O. D. Jesús, *Neural Network Design*, 2nd ed., 2014. [Online]. Available: https://hagan.okstate.edu/NNDesign.pdf and https://hagan.okstate.edu/NNDesign.pdf

ETIENNE ALAIN FEUKEU (Member, IEEE) received the master's degree in telecommunication from the Tshwane University of technology, the Master of Science degree in electrical and electronic systems from ESIEE Paris, and the Ph.D. degree from the Tshwane University of Technology, in 2015.

He was a Research Fellow with Aalborg University. He is currently a Senior Lecturer with the University of South Africa (UNISA). His research interests include wireless networks, analysis, development, and design of link adaptation algorithms in WLAN, MIMO systems, vehicular ad hoc networks (VANETs), Doppler shift mitigation, congestion management, artificial intelligence (machine learning), the Internet of Things (IoT), and renewable energies. He is a Senior Member of SAIEE. He was a recipient of several research awards from the National Research Foundation of South Africa. He received the Post-Doctoral Fellowship from the University of Cape town in 2018.

MBUYU SUMBWANYAMBE (Member, IEEE) received the M.Eng. degree in electrical and electronic engineering and the Ph.D. degree in engineering management from the University of Johannesburg, South Africa. He is currently a Professor with the University of South Africa, where he is also the Head of Postgraduate and Research with the Department of Electrical and Mining Engineering. He has a wide range of industry experience, which he gained in South Africa and Zambia, as a Consultant Project Engineer and a Researcher with the University of Johannesburg. He is keen promoter of collaboration between industries, the university, and other universities in the SADC region. He has contributed to book chapters and published extensively in journal and conference proceedings. His research interests include risk management, application of game theory in telecommunication, digital communication, resource management, regulation and policy issues, telecommunication and ICT engineering, economic and social ICT delivery, energy efficiency and demand side management and data mining, and information systems evaluation. He is a member of SAIEE and International Association of Engineers (IAENG). He was a recipient of the UJ-Commonwealth Scholarship. He has held workshops with the government on the value of ICTs for social and economic development. He has reviewed several papers of IEEE and Taylor and Francis journals, as well as acted as the chair and invited panelist at numerous international conferences.