IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 17, Issue 2, Ser. I (Mar. – Apr. 2022), PP 34-42
www.iosrjournals.org

Performance Analysis of Adaptive Modulation and Codec Selection based Long-Short Term Memory Neural Network (LSTM-NN)

Mogahed A. M. Sharafuden[1*], Vinay Kumar [1], Mohammed E. Y. Abdalla[2]
[1](Electronics and Communication, Delhi Technological University, India)
[2](Electronics and Communication, Indian Institute of Technology (ISM) Dhanbad, India)
*(sharafmogahed@gmail.com)

Abstract:
This study presents the design and analysis of adaptive modulation and coding scheme intended for a codec used in short-range wireless communication systems; the study aims to investigate and analyze the existing channel prediction parameters for adapting the variation of wireless communication channels. Three modulation and coding schemes are used to develop the codec, namely 8QAM, 16QAM with a 1/3 BCH encoder, and 4QAM without error correction. MIMO antenna is used to increase the spectral efficiency of the codec within the fading channel.
The presented scheme operates at a BER threshold of 10^{-3}, with a channel correlation of 0.96 and a noise variance of 3 dB. The adaption parameter and the channel model provide the basis of the machine learning model. A perfect predictor with zero root means square error (RMSE) is used as a standard predictor. Long-short Terme memory Neural Network (LSTM-NN) and conventional Neural network (NN) predictors are used as prediction models for the adapter with RMSE of 3.9157 *and* 5.6121*, respectively.*
The result found that the adaptive modulation and codec parameters chosen under the LSTM-NN have a data rate of 15 Mbits/s and 10 Mbits/s for conventional NN predictor. The codec achieved the design requirements, and it can serve users operating within 2.4GHz. However, the AMC design can be improved by using more robust encoding and modulation techniques such as 64QAM and transmitting using more antennas.
Key Word*: Adaptive Modulation and coding scheme; LSTM-NN; Predictor; Codec adaptive.*

Date of Submission: 04-04-2022 Date of Acceptance: 20-04-2022

I. Introduction

Adaptive modulation and coding (AMC) technologies have been broadly used in wired and wireless communication systems to adjust the variations of the communication channel environments. Generally, in communication systems, wireless channels are more intensely time-variant than wired communication channels. The standards for wireless communications are recommended to apply AMC to enhance spectrum efficiency (SE) and consistency to achieve user demands. Recently, AMC combined some technologies such as orthogonal frequency division multiplexing (OFDM) and multiple-input multiple-output systems (MIMO) owing to improve consistency of wireless systems such as Wireless Fidelity (Wi-Fi) standards.
However, in the practical wireless systems, once AMC structures are combined with OFDM and MIMO, modulation and the convolutional coding process are highly challenging, and the design of the communication system divides the signal processing into many independent blocks. Consequently, optimizing these combinational systems leads to computationally complex systems [1]. In contrast, machine learning (ML) has the potential to enhance those independent blokes and make the computation for such complex systems possible. Hence, ML assists as a batter contender for optimizing AMC aided wireless systems with explicit hardware configurations when communicating over wireless channels.

Researchers have focused on developing approaches to switch between different modulation schemes to address the performance and adapt the variations of wireless channels. In [2], they use a signal to noise ratio (SNR) based adaption using M-array phase shift key (MPSK) and Reed Solomon (RS) code to make switching between schemes using a channel predictor given by $SNR_1 = \alpha^2(SNR)$ where $SNR_1$is the SNR value determined by the transmitter with the exists of the fading amplitude α. In [3], they present a design that considers the medium access control performance requested by an application. They used M-array quadrature amplitude modulation (M-QAM) with convolutional code bounds described in [10], and they defined the prediction model under the use of minimum mean square error (MMSE) as $\widehat{H} = H_{current} + \frac{L_{current}}{L_{Prev}} \Delta H$, where

$L_{current}$ and L_{prev} are the length of current and previous packet length and ΔH is the difference between the current and previous channel matrices. [4] proposed a model based on power and BER constraints using trellis codes. A more robust algorithm is presented in [5], which considers IEEE 802.11n guard interval sizes; they derived the optimal BER value for switching between the MSCs and concluded that their algorithm provides high throughput while reliable. Later, the authors of [6] used a fully connected neural network (NN) and convolutional neural network. They achieved a better result than traditional AMC designs for 64QAM using MMSE. The AMC model in [7] uses supervised learning to reduce complexities and a deep neural network for channel estimation using channel matrices as training data. Each channel matrix is mapped to its respective transmitter antenna; their results show that such a design outperforms conventional designs while being less complex. In [8], the authors used reinforcement learning-based AMC for underwater communication; their design reduces BER with less energy consumption. [9] explores a NN channel prediction model under different channel models, namely Jakes, Clarke/Gan's, and 3rd generation partnership project (3GPP) spatial channel model. Their model performs better when using the jakes model because of Jakes's substantial time domain correlation over different SNR values. In order to improve performance and system reliability, [10] suggests an online learning algorithm for AMC that keeps a database of post performances. This approach allows the training data to be generated while the system is in use.

In this work, adaptive modulation and coding scheme is the primary technique used to adapt the varying signal strengths. For the system to meet requirements and achieve a high data rate, the assumptions made are the linearity of the system channel and fading coefficients to be constant for the entire codeword. For designing a machine learning prediction model, the Long-Short term memory neural network (LSTM-NN) which is a variant of recurrent neural networks (RNN) used to model the time varied data for the reason that it allows the information data to persist in the neurons. In contrast to the simple neural networks (NN), which only depend on the current input. LSTM contain structures called memory cell, which modify the information in the network. The information in a memory cell is controlled by gates, namely the input gate, forget gate, and output gate. The forget gate is used to discard information no longer needed by the memory cell. The input gate dictates when new information is added to the memory cell. The output gate is responsible for determining the output based on input data. The operation used in an LSTM-NN is similar to a traditional NN with slight variations to accommodate additional computational units [11]. Detailed LSTM-NN processes are described in the coming section III, part A.

II. Technical Background

Wireless networks take advantage of a digital communication model, and this work tries to adapt the variation of the short-range wireless communication channel. This section introduced the fundamentals of wireless communication systems.

A. Digital Communication Model

Wireless networks take advantage of a digital communication model consisting of the main parts of digital communication to exchange information from the data source to the data sink. Nowadays, most digital systems use progressive encryption standards to present confusion and dispersion of information messages. Forward Error Correction (FEC) encoder is responsible for controlling error introduced in an encryption process in communication systems and intending k bits to encoder appends r to add redundancy. Then the encoder transforms the information into a unique $\boldsymbol{n}$ bitstream; FEC has a $\boldsymbol{k/n}$ code rate. Commonly used FEC codes include convolutions codes, LDPC, Turbo codes, RS, and BCH codes. The modulation performs bitstream maps to signal waveform; this process is completed by using a single or combination of parameters of the carrier signal.

B. Receiver Equalization

Receiver equalization is a method of restoring high-frequency signals components usually attenuated by channels. One of the most used Equalizers is a zero-forcing equalizer (ZF). This equalizer W is given by the equation below to reject inter-carrier and inter-symbol interference.

$$W = (H^H H)^{-1} H^H \quad (1)$$

Where the equalizer W is a complex matrix in the exact dimensions as H. the drawback of the equalizer is that it amplifies noise caused by linearly dependent columns[13]. ZF is also unfunctional if $H^H H$ is singular. The enhancement to equalization (ZF) detector is come inform of minimum mean square error (MMSE) detector, which introduces a regularization term $\lambda = \sigma^2$ to the central equalizer equation given above in (1), which enhanced the system to be less sensitive to the channel conditions [13].

$$W = (H^H H + \lambda I)^{-1} H^H \quad (2)$$

MMSE is applicable to work in both cases where Tx > Rx or Tx < Rx [13]. And the remaining part of the receiver operates in the inverse operation of their respective transmitter parts.

III. Procedures and considerations

The success of this work is mainly dependent on designing a simulation model that meets the flowing requirements.

1. The codec must achieve a rate of at least 2 bits/Hz.
2. The Rayleigh fading channel with a bandwidth of 4MHz.
3. The data rate must be at least 10Mbits/s.

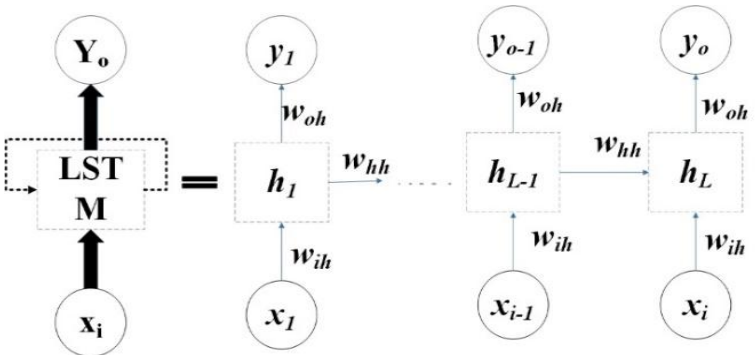
We consider the channel to be quasistatic, and the fading coefficients stay constant for the entire codeword; in addition, the antennas are independent of each other, which makes the channel have a unique coefficient for each antenna. The generated data to be transmitted has been compressed and encrypted under the existence of hardware components to support the proposed codec. So, the success of this work mainly stands on designing a simulation model that meets the requirements mentioned above. And MCS, LSTM-NN architecture as a machine learning model should be justified the design process and parameter choices.

A. Structure of LSTM-NN

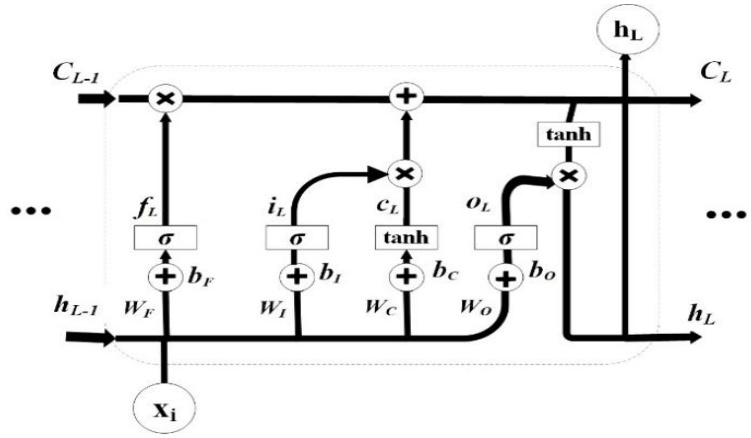
Since the conventional neural network is feed-forward (FNN), it does not have sequences or loops slightly, and its ability for sequential modeling is relatively poor. In contrast, RNN is a type of neural network capable of dealing with time-varied data because of its internal memory and the feedback chain structure that makes prediction output can be specified through the previous input sequences. For channel prediction purposes, RNN is enforceable to accomplish better performance compared with FNN. Nevertheless, in RNN, the back-propagated error speedily vanishes [14]. Therefore, RNN is not suitable to solve problems that require dependencies for long-term temporal learning. To overcome such an issue, the LSTM-NN is proposed [15]. Contrasted with RNN, LSTM-NN controls the passing data through three gates over the sequence, and its internal loops can store the long-term dependencies previously learned from input data. LSTM-NN basic architecture is shown in figure 1(a). during each time, the state of hidden layer $\mathbf{h_L}$ updates with input data $\mathbf{x_i}$ at present and the hidden state $\mathbf{h_L}$ of the last moment:

$$h_L = \sigma_h(w_{ih} + w_{hh}h_{L-1} + b_h) \qquad (3)$$

Where σ_h is the activation function, w_{ih} weight matrixes between the input to the hidden layer and w_{hh} are between two consecutive hidden states, and b_h representing the bias function.



(a)



(b)

Figure. 1. (a) Basic LSTM-NN architecture. (b) LSTM cell architecture [16].

Figure. 1. (a) shows the unfolded structure of LSTM-NN with Xi and Yo the input and output vector, respectively; where hL represents the hidden state; Wih and Woh are the weight matrix between the input-to-hidden layer hidden-to-output layer, respectively. LSTM cell structure is shown in the figure. 1. (b), LSTM has three gates: forget, input, and output gates. The forget gate f_L determines what state information at time L-1 (C_{L-1}) is to be forgotten. Then the input gate $\boldsymbol{i_L}$ adopts which information is updated, and the cell input state $\tilde{C}_L$ It will be produced by a tanh neural network layer. After that, the cell state at this time C_L It can be calculated. Lastly, the output gate $O_L$will filter the output h_L to determine the cell state C_L. During the reiteration of the all-time steps from 1 to L, the output sequence $y_0 = [h_1, h_2, \ldots., h_L]$ Is computed.

$$\boldsymbol{f}_L = \sigma_g(W_F[h_{L-1}, x_i] + b_F) \quad (4)$$

$$\boldsymbol{i}_L = \sigma_g(W_I[h_{L-1}, x_i] + b_I) \quad (5)$$

$$\boldsymbol{o}_L = \sigma_g(W_O[h_{L-1}, x_i] + b_O) \quad (6)$$

$$\tilde{\boldsymbol{c}}_L = tanh(W_c[h_{L-1}, X_i] + b_c) \quad (7)$$

$$\boldsymbol{c}_L = f_L * C_{L-1} + i_L * \tilde{C}_L \quad (8)$$

$$\boldsymbol{h}_L = O_L * tanh(c_L) \quad (9)$$

Where $\sigma_g$is the activation function of the gates, W_F, W_I, W_O, and W_c Represent the weights matrices between the hidden layer to these three gates. While b_F, b_I, b_O, and b_C are representing bias functions. LSTM-NN can learn from the previous input data because its cell structure can keep data for long-term dependencies. Therefore, it is suitable for the time-varied sequence prediction [16]. Because LSTM-NN cell structure has the ability to store the input information for log-term sequential dependencies learned from prior input data. In this work, the designed AMC predictor is adapted to flow the supervised machine at a specific time and can be precisely predicted from the signal randomly generated from the bits sequence at this moment and previous signal sequences using the LSTM-NN cell.

B. Codec Adaptive Modulation and Coding Model Design

The design runs at a frame level, with each frame consisting of a sequence of randomly generated bits. This work uses BCH codes with symbols from binary filed $GF(2^m)$ Where $m = 4$. The encoder used is a BCH with message length $k = 5$ and codeword length that is $n = 2^m - 1$, which is capable of correcting up to 3 errors, given by RS codes because of better performance over fading channels. The modulation scheme used is QAM; it utilizes both ASK and PSK to provide better spectrum efficiency than any of them separately. To increase channel capacity and spectral efficiency, the proposed AMC uses MIMO multiplexing. The AMC makes use of a 2×2 MIMO antenna. For the simulation purpose, two transmit antenna (Tx) and two receive antenna (Rx) were used in order to model a narrowband channel in the form of matrix H. these matrix elements represent amplitude and phase shift introduced by the channel and denoted by h_{ij} which represent MIMO channel connection between i^{th}receiver and j^{th} transmitter antennas. The antenna output signal (y) comes from groups of an input signal (x) via input antennas, where $y \; and \; x \in C^{N\times 1}$ comes from the assumption that $H \in C^{N\times N}$.

$$y = Hx + n \quad (10)$$

For simulation purposes, equation (10) is rewritten in (11) as the exact structure adopted in the code.

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} n1 \\ n2 \end{bmatrix} \quad (11)$$

Three different modulations and coding schemes (MCS) are combined to make up the AMC, and table 1 shows the structure for each MCS. MCS1 is the only one that does not make use of forwarding error correction (FEC). The MCS parameters are chosen for the codec to achieve a high data rate; equation (12) shows that the selected MCS meets the 2 bits/Hz rate. Using a pulse shaping coefficient of α_P=1, MIMO coefficient $\alpha_m = 2$, M is the modulation order, and *R* represents the FEC code rate. The spectral efficiencies for MCS1, MCS2, and MCS3 are shown in table no 1.

Table no 1. Modulation and coding schemes for AMC

MCS	Code rate	QAM order	MIMO	SE	Es
MCS1	1	4	2x2	4	2
MCS2	1/3	8	2x2	2	6
MCS3	1/3	16	2x2	2.67	10

$$SE = \frac{R\, log_2(M)\alpha_m}{\alpha_P} \quad (12)$$

In the simulation, the additive white gaussian noise (AWGN) is generated at the beginning of each frame, the noise variance σ_n^2 dependent on SNR (γ) and the MCS used. The noise variance for each MCS is determined using equation (13).

$$\sigma_n^2 = E_s \frac{n}{k\, log_2(M)} \cdot 10^{\left(-\frac{\gamma}{10}\right)} \quad (13)$$

Where Es is the average energy of the symbols after modulation. The energy is computed by taking the mean of the distances squared of the QAM constellations. The average energies for 4QAM, 8QAM, and 16QAM are 2,6 and 10 respectively.

$$C_{n,k} = \frac{n}{k} F \quad (14)$$

$$s_M = \frac{c_{n,\,k}}{log_2(M)} \quad (15)$$

$$s_{MIMO} = \frac{S_M}{Tx} \quad (16)$$

The frame structure of the simulated packets is described using equation (12 – 14). Where *F* is the frame size, $C_{n,k}$ is the number of coded by bits from an (*n, k*) encoder, S_M are the symbols from a M-array modulator and S_{MIMO} are symbols from each antenna. The simulation uses a frame size of 2000 bits for each MCS. Table no 2 shows the frame structure for each of the MCS.

Table no 2. Modulation and coding schemes frame structure

MCS	Coded bits	QAM symbols	Antenna symbols
MCS1	2000	1000	500
MCS2	6000	2000	1000
MCS3	6000	1500	750

We used an MMSE equalizer based on the reason discussed in section (II-part B) to remove errors introduced by the channel. The output signal X' from the MMSE equalizer is defined using equation (17). In order to recover the transmitted signal *X*, the respective inverse operations are performed on X'.

$$X' = wy \quad (17)$$

For the prosed model to be adaptive to channel conditions, a recursive model is provided in equation (18) that relates the current channel H_f to the previous frame channel $H_{f-1.}$ Equation (18) models a discrete channel update; it follows that the channel update can be written as an ARMA process which can be used to model the correlation in a discrete linear process. This channel update is deepened on the LSTM-NN gates discussed in section (III).

$$H_f = \alpha H_{f-1} + (1-\alpha) G_f \quad (18)$$

The extent to which the channel is correlated is controlled by $\alpha \in (0,1)$ which is the fading steady-state coefficient. A large α means a high channel correlation, and a lower one means a low channel correlation. The parameter G_f is used to introduce random channel errors using conditions states in LSTM-NN gets. The channel starts to fade from the second frame as such H_f is initialized to an identity matrix for the first frame. H_f and G_f are determined at the end of the frame.

We define the adaption parameter ξ, which is a measure of the effect of the channel and equalizer to the SNR. The adaption parameter is used to specify the range at which a given MCS is active. The adaption parameter is given a measure of the effect of the channel and equalizer Z = HW to SNR.

$$\xi = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\left|H_{ij}\right|\right)^2}{\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\left|W_{ij}\right|\right)^2} \quad (19)$$

A translation is done from the SNR vs BER curves to a set of ranges in ξ. The translation rule is the most spectral effective MCS should be active for the majority of the frames. Using the model in equation (19) shows that ξ has a variation centered at the initial value of $\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\left|H_{ij}\right|\right)^2$.

C. Predictors Design

In this work, two predictors have been introduced; one is based on the conventional NN, and the second is based on LSTM-NN. The NN has trained a window size of 4, which means each 4 ξ points are needed to predict the next value of ξ. The value of the predicted ξ will be used to select a specific MCS used in the next frame. The rest of the structure, such as the number of hidden layers, learning rate, and others are determined through experimentation. A similar approach was used for LSTM-NN. The final parameters are chosen for the output to form the loss function in minimum. Both predictors use root mean square error (RMSE) as a loss function given in equation (20) over n data points. MCS classification is done using ξ values determined by the predictor from the range of ξ values.

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y} - y)^2}{n}} \qquad (20)$$

The accuracy of the predictors is determined at the end of each forward propagation epoch. The cost or loss function is used to determine the difference between the predicted value and the expected value.

IV. Simulation Results and Analysis

In this work, the first task carried out is to determine the modulation orders for the different MCSs. Initially, the simulation is run to determine the effective ξ and SNR regions for each MCS. Using equation (18), we observed that if α is closed to a value of 1, then the channel is almost perfectly correlated since the noise term G_f will have a negligible effect on the channel conditions in the current frame. The α value is chosen such that it strikes a balance between the two extremes. Ultimately, we set on $\alpha = 0.96$ and $\sigma_p^2 = 3$. The channel variance σ_g^2 is kept constant, making comparison simpler. The specific value of σ_g^2 was experimentally determined through an iteration process. Figure. 2. shows ξ with the chosen channel model parameters settled on.

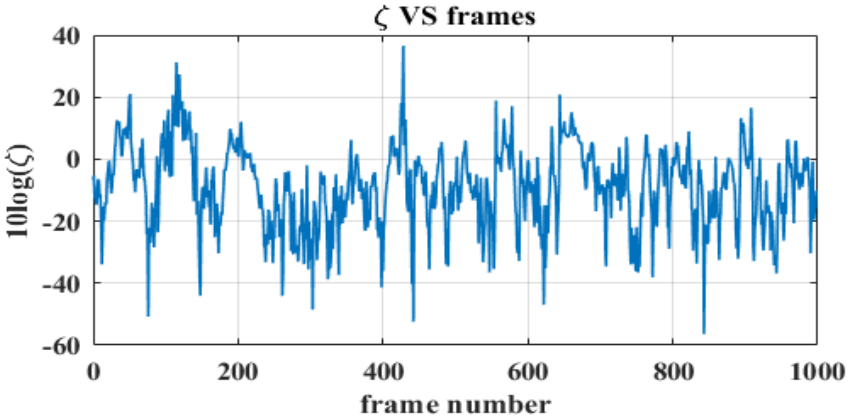


Figure. 2. Low correlation at $\alpha = 0.96$ and $\sigma_p^2 = 3$.

In figure 2. the MCS regions are defined to be MCS1 for ξ ϵ [0, ∞), MCS2 for ξ ϵ (0, -15) and MCS3 for ξ ϵ (-∞, −15]. Furthermore, we observe that the value of 10log (ξ) is centered around 10log ($\sum_{i=1}^{N}\sum_{j=1}^{N}(|H_{ij}|)^2$) = 0. Table 3 shows the operating ranges for each MCS subsystem. These ranges are used for the predictor to decide on which MCS to employ at a given frame.

Table no 3. MCS operational range

MCS	ξ **range**	**SNR range**
MCS1	[0, ∞)	(-∞, 32]
MCS2	(0, -15)	(32,37)
MCS3	(- ∞, -15,]	[37, ∞)

The SNR at which the received signal experiences significant BER curves are shifted to the right by a factor of 10log ($\Sigma_{i=1}^{N}\sum_{n=1}^{N}(|W_{ij}|)^2$) due to the fact that steady-state coefficient fading is less than 1. Therefore, the curves generated in this figure are used to determine the BER threshold for the AMC. We select a BER threshold of 10^{-3} which means 100 error bits for every 100,000 transmitted bits. The effective AMC SNR ranges can then determine using the BER threshold and MCS ξ regions as scaling conditions. The ξ = -11, which is determined through the simulation, corresponds translated to an SNR value of 30 dB, which is the point at which the leftmost curve meets the BER threshold. The SNR regions are then defined to span a proportional region to ξ. The regions are defined to be MCS1 for SNR ϵ (-∞, 32], MCS2 for SNR ϵ (32,37) and MCS3 for SNR ϵ [37, ∞).

The perfect predictor is developed to be the most optimal AMC. It predicts the channel conditions by calculating the ξ for the next frame and then selecting the appropriate MCS based on that value. As such, the perfect predictions can optimal MCS for a given frame. The perfect predictor is bounded by the 4QAM and 16QAM as expected with a shape that maximizes the effective data rate.

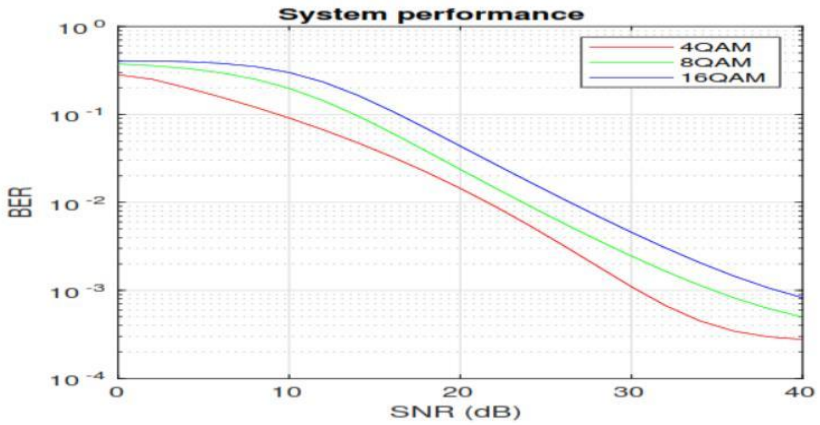


Figure. 3. SNR vs BER curves for the selected MCS at $\alpha = 0.96$ and $\sigma_p^2 = 3$.

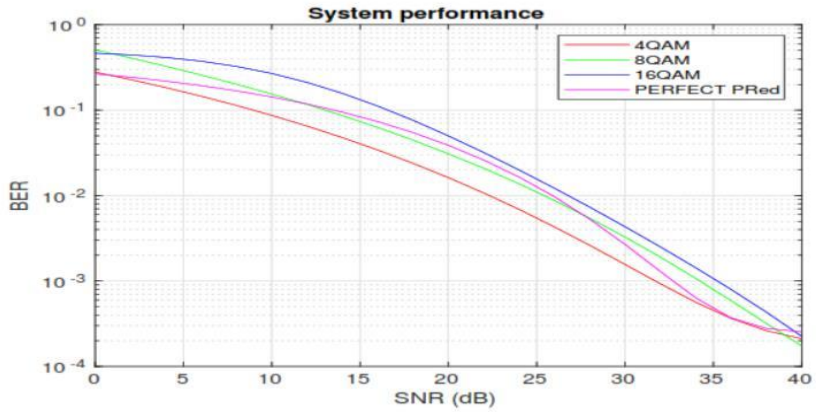


Figure. 4. Perfect AMC predictor.

The machine learning data set is generated by obtaining Figure 4.1. and Figure 4.2. the data set is slit such that 90% is used for training and the rest as testing data. Each of the predictors also gives the MCS classifications for each ξ. The specify parameters used for the conventional neural network and LSTM-NN were empirically determined. The predictor's performance is determined by computing the effective data rate of each system. The data rate is calculated as $R_b = T_x(BER_t)\Sigma(R)(FM)\,log_2\,M$.

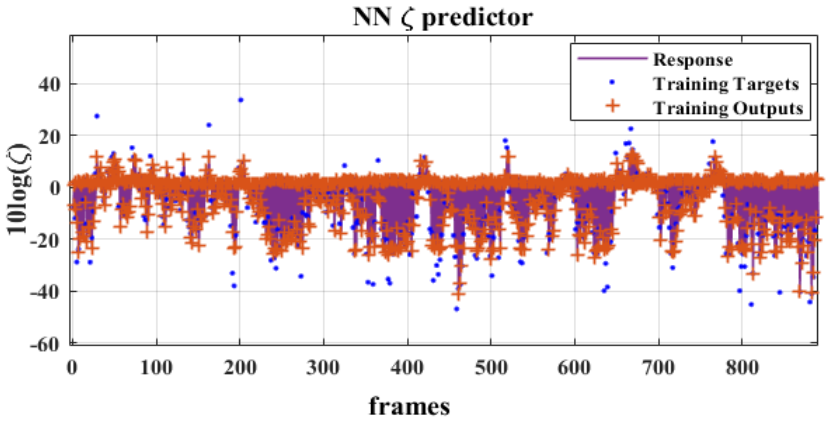


Figure. 5. conventional Neural network predictor.

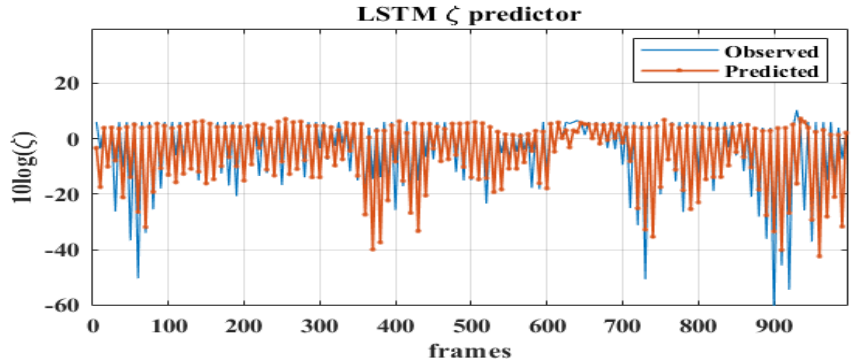


Figure. 6. Long-Term Memory predictor.

Where M is the modulation order of the MCS, R is the code rate, FM is the number of frames that activate MCS, and (BER_t) is the BER threshold for AMC. We concluded by considering the RMSE for each predictor since both predictors were trained using the same generated data. The RMSE for the perfect predictor is 0, whereas the LSTM-NN and NN predictors have an RMSE value of 3.9157 and 5.6121, respectively. We can conclude that the perfect predictor is more accurate, followed by the LSTM-NN predictor and, lastly, the conventional NN predictor.

V. Conclusion

This paper introduced the implementation of a machine learning-based adaptive codec using LSTM-NN and conventional NN predictors under the AMC technique to improve the channel prediction for the short-range wireless communication channel such as Wi-Fi. The system is simulated under a Rayleigh fading channel, and the simulation tool used is MATLAB. The designed codec used 8QAM and 16QAM with a 1/3 BCH encoder and 4QAM without FEC. Three different AMCs are implemented: a perfect predictor, LSTM-NN predictor, and NN predictor; these predictors have RMSE of 0, 3.9157, and 5.6121, and the data rate of *15 Mbits/s and 10 Mbits/s*, respectively. The RMSE values the codec can serve users operating in 2.4GHz. Furthermore, it improves data rate and reduces computation complexity. The AMC codec can improve by using more robust codecs such as turbo and LDPC codes.

References

[1]. L. Zhang and Z. Wu, "Machine Learning–Based Adaptive Modulation and Coding Design," *Mach. Learn. Futur. Wirel. Commun.*, pp. 157–180, 2020, doi: 10.1002/9781119562306.ch9.

[2]. K. M. S. Soyjaudah and B. Rajkumarsingh, "Adaptive coding and modulation using Reed Solomon codes for Rayleigh fading channels," *EUROCON 2001 - Int. Conf. Trends Commun. Proc.*, pp. 50–53, 2001, doi: 10.1109/EURCON.2001.937761.

[3]. F. Peng, J. Zhang, and W. E. Ryan, "Adaptive modulation and coding for IEEE 802.11n," *IEEE Wirel. Commun. Netw. Conf. WCNC*, no. 1, pp. 657–662, 2007, doi: 10.1109/WCNC.2007.126.

[4]. D. L. Goeckel, "Adaptive coded modulation for transmission over fading channels," *Adapt. Cross Layer Des. Wirel. Networks*, vol. 2, no. 5, pp. 71–94, 2018, doi: 10.1201/9781420046021.ch3.

[5]. P. Patil, M. Patil, S. Itraj, and U. Bombale, "IEEE 802.11n: Joint modulation-coding and guard interval adaptation scheme for throughput enhancement," *Int. J. Commun. Syst.*, vol. 33, no. 8, pp. 3–5, 2020, doi: 10.1002/dac.4347.

[6]. J. Chen, L. Zhang, Y. C. Liang, X. Kang, and R. Zhang, "Resource allocation for wireless-powered iot networks with short packet communication," *IEEE Trans. Wirel. Commun.*, vol. 18, no. 2, pp. 1447–1461, 2019, doi: 10.1109/TWC.2019.2893335.
[7]. P. Yang, Y. Xiao, M. Xiao, Y. L. Guan, S. Li, and W. Xiang, "Adaptive Spatial Modulation MIMO Based on Machine Learning," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 9, pp. 2117–2131, 2019, doi: 10.1109/JSAC.2019.2929404.
[8]. W. Su, J. Lin, K. Chen, L. Xiao, and C. En, "Reinforcement Learning-Based Adaptive Modulation and Coding for Efficient Underwater Communications," *IEEE Access*, vol. 7, pp. 67539–67550, 2019, doi: 10.1109/ACCESS.2019.2918506.
[9]. R. F. Liao, H. Wen, J. Wu, H. Song, F. Pan, and L. Dong, "The Rayleigh Fading Channel Prediction via Deep Learning," *Wirel. Commun. Mob. Comput.*, vol. 2018, 2018, doi: 10.1155/2018/6497340.
[10]. R. C. Daniels and J. Andrews, "Machine Learning for Link Adaptation in Wireless Networks," *LiLi - Zeitschrift für Lit. und Linguist.*, 2010.
[11]. J. Brownlee, "Long Short-Term Memory Networks With Python," *Mach. Learn. Mastery With Python*, vol. 1, no. 1, p. 228, 2017.
[12]. F. C. Vilar, "Implementation Of Zero Forcing And Mmse Equalization Techniques In OFDM," *Univ. Fortaleza*, 2014.
[13]. K. Rundstedt, "Measurements and Channel Modelling of Microwave Line-of-Sight MIMO Links," *MS thesis, Chalmers Univ. Technol.*, 2015.
[14]. F. A. Gers and F. Cummins, "A critique of neoclassical macroeconomics," *Choice Rev. Online*, vol. 27, no. 09, pp. 27-5238-27–5238, 1990, doi: 10.5860/choice.27-5238.
[15]. D. N. T. How, K. S. M. Sahari, H. Yuhuang, and L. C. Kiong, "Multiple sequence behavior recognition on humanoid robot using long short-term memory (LSTM)," *2014 IEEE Int. Symp. Robot. Manuf. Autom. IEEE-ROMA2014*, pp. 109–114, 2015, doi: 10.1109/ROMA.2014.7295871.
[16]. C. Wang, S. Fu, Z. Xiao, M. Tang, and D. Liu, "Long Short-Term Memory Neural Network (LSTM-NN) Enabled Accurate Optical Signal-To-Noise Ratio (OSNR) Monitoring," *J. Light. Technol.*, vol. 37, no. 16, pp. 4140–4146, 2019.
[17]. R. -F. Liao, H. Wen, J. Wu, H. Song, F. Pan and L. Dong, "The rayleigh fading channel prediction via deep learning," Wirelss Communications and Mobile Computtaing, vol. 2018, pp. 1-11, 07 2018.
[18]. F. Lone, A. Puri, and S. Kumar, "Performance comparison of reed Solomon code and BCH over Rayleigh fading channel", 072013.
[19]. Duckett, "Stop using 2.4 GHz and rely on GHz Wi-Fi: Acma nbn modem study", Jul 2019. [Online]. Available at: https://www.zdnet.com/article/stop-using-2.4ghz-and-rely-on-5-ghz-wi-fi-acma-nbn-modem-study/.

Mogahed A. M. Sharafuden, et. al. "Performance Analysis of Adaptive Modulation and Codec Selection based Long-Short Term Memory Neural Network (LSTM-NN)." *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)* 17(2), (2022): pp 34-42.