

CNN-Based Learning System in a Generalized Fading Environment

Samuel Borges Ferreira Gomes and Michel Daoud Yacoub.

Abstract—In this paper, we investigate the Block Error Rate (BLER) performance of a Convolutional Neural Network (CNN)-based autoencoder as a self-learning communication system under a generalized fading condition. To this end, the α - μ model is chosen, both for its flexibility and applicability to practical fading scenarios. For comparison and consistency purposes, the analytical BLER is also explored under the same environment for different classical modulation schemes. Our simulation results show that the studied architecture is able to converge quickly maintaining its generalization capability under various fading environments.

Keywords—Communication systems, α - μ fading, end-to-end learning, autoencoder, convolutional neural network.

I. INTRODUCTION

Deep Learning (DL) has been showing an enormous and increasing number of applications in diverse areas, including computer vision, image, natural language processing, and many others. Particularly, in communications, concrete mathematical theories and models ranging from information theory to channel modeling have been developed and well explored in the literature. However, the gap between theory and practice is motivational for the emergence of intelligent communications, driven by artificial intelligence, big data, power computation, and expert domain of wireless communications. Hence, this type of system is now considered to be one of the candidate technologies for further development in advanced wireless networks, i.e., beyond fifth generation (5G).

The power of DL applied to communication systems has been revealed through previous researches applied to the network layer (e.g., network optimization, resource allocation and management) as well as to the application layer (e.g., network prediction, facial recognition and data mining).

On the physical layer side, aiming to reduce system complexity and focusing on the specific function of each processing block, the conventional wireless communication system architecture adopts the divide-and-conquer strategy based on mathematical models. It split transmitter and receiver into subtasks, such as source/channel coding, modulation, and equalization. This architecture, whose advantage is to allow for the optimization of each component individually, has been very successful as demonstrated for the various communications systems, wireless communications included. On the other hand, in practice, complex systems, like those, undergo unknown effects, difficult to be mathematically modeled.

S. B. F. Gomes and M. D. Yacoub are with the Wireless Technology Laboratory (WissTek), Department of Communications, School of Electrical and Computer Engineering, State University of Campinas, 13083-852, Campinas, SP, Brazil (e-mails: [samuelbf,michel]@decom.fee.unicamp.br)

Therefore, a more flexible and adaptive framework may be required to deal with these challenges.

DL applied in communication systems has been studied in different areas. Among others, we cite signal detection [1], [2], signal classification [3], and channel encoding [4], [5], where Recurrent Neural Networks (RNNs) were used at the receiver to decode channel-coded information bits, such as Turbo codes and Polar codes. Moreover, channel estimation [6], [7], positioning [8], and security through jamming [9] exploitation and adversarial [10] attacks were also explored.

Although the chain of multiple independent processing blocks as a communication paradigm has led to successful and efficient systems we own today [11], it is not clear whether individually optimized processing blocks achieve the best-possible end-to-end performance.

Motivated by this, an innovative perspective considers wireless communications system as an end-to-end autoencoder [12]. As far as communications are concerned, the goal of an autoencoder is to find representations of the inputs (transmitted signals) at some intermediate layer that are robust w.r.t. the channel impairments mapping (e.g., noise, fading, and distortion), which allows reconstruction at the output (received signal) with small probability of error. The self-learning system can be seen as weights learned from an optimized neural network using end-to-end loss functions. The experiments in [13] show that the Neural Network (NN)-based autoencoder can achieve similar performance with the conventional method in the presence of additive white Gaussian noise (AWGN) channels without any prior specific domain information. This idea was further evolved with the radio transformer networks [14] to combine expert domain knowledge in a deep learning model. In addition, several extensions of the original idea have been explored, such as the implementation of the self-learning paradigm on a hardware system [15], and its applications to Multiple Input Multiple Output (MIMO) transmission [16], essential for 5G wireless communication systems.

The use of convolutional layers [17] as the main building blocks for the autoencoder-based communications systems was first introduced by [18]. However, the performance suffered from a irreducible error floor in high signal-to-noise ratio (SNR) regimes. Built upon that, [19] integrated communications engineering insights to propose an intelligent framework with a better generalization capability. Nevertheless, none of these works considered the propagation medium in terms of its non-linearity.

In wireless communications, short term fading occurs whenever signals reach a receiver via multiple paths. As widely known, in such a case, the signal behavior can be modeled us-

ing stochastic models. A large number of distributions describe the statistics of wireless communications channels including most notably Hoyt, Rayleigh, Weibull, Nakagami- m , and Rice. Each model tries to capture the real world effects in terms of different assumptions, such as line-of-sight wave condition and propagation medium linearity. The α - μ [20] is a general fading distribution used to represent the small scale variation of the fading signal in a non-linear environment in which clusters of multipaths are present. It includes important distributions such as Nakagami- m , Exponential, Weibull, Rayleigh.

In this paper, we investigate the Block Error Rate (BLER) performance under flat α - μ channels of the CNN-based autoencoder (CNN-AE) [19]. The analysis is given by means of its generalization capability under generalized fading scenario for block length, training SNR, and code rate. For comparison purposes, analytical BLER performance for traditional modulation schemes is considered. It is shown that the adapted CNN-based model can match the performance of existing optimal human engineered solutions under flat α - μ fading channel. Furthermore, constellation points of the learned representation is drawn. To the best of the authors' knowledge, the proposed system setup has not been investigated in the technical literature yet and this paper aims to fill this gap.

The remainder of this paper is organized as follows. In Section II, the autoencoder architecture as well as how it can be seen as a self-learning system are shown. Section III the α - μ model is revisited. Section IV presents simulated and analytical results, upon which some fundamental guidelines are highlighted. Finally, Section V concludes the paper and points out some potential future works.

II. AUTOENCODER AS AN END-TO-END COMMUNICATION SYSTEM

A point-to-point communication system consists of three main processing blocks, namely transmitter, channel, receiver. The transmitter sends a message m over a channel to the receiver. The message consists of a sequence of L symbols (block length), each symbol conveying k information bits, making n discrete uses of the channel. Hence, the rate of the system is $R = k/n$ (bits/channel use). To this end, the transmitter applies a transformation $\mathbf{x} = f(m) \in \mathbb{C}^n$, where \mathbf{x} is the generated transmitted signal. Generally, the hardware of the transmitter imposes certain constraints on \mathbf{x} . In this work, we shall consider this as a power constraint $\|\mathbf{x}\|^2 \leq n$. The channel acts as a stochastic system, whose output follows a conditional probability function distribution $\mathbf{y} \sim p(\mathbf{y}|\mathbf{x})$, where $\mathbf{y} \in \mathbb{C}^n$ denotes the received signal. Lastly, the receiver, which applies the transformation $\hat{m} = g(\mathbf{y})$, aiming to produce estimate of the original message m as closely as possible. This scheme is illustrated in Fig. 1.

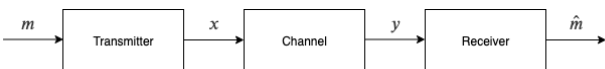


Fig. 1: Simplified point-to-point communication system.

In the DL terminology, inspired by Fig. 1, the transmitter and receiver are respectively called encoder and decoder, and

they are implemented using neural networks. Feedforward networks describes a mapping $f(\mathbf{x}_0; \mathbf{W}) : \mathbb{R}^{N_0} \mapsto \mathbb{R}^{N_L}$ of an input vector $\mathbf{x}_0 \in \mathbb{R}^{N_0}$ to an output $\mathbf{x}_L \in \mathbb{R}^{N_L}$, along L layers, describing different iterative processing steps. This kind of mapping depends not only on the output vector from the previous layer, but also on a set of parameters (or tensor of all weights) $\mathbf{W} = \{W_1, W_2, \dots, W_L\}$.

In this work, we consider convolutional layers, which, in general, consists of a set of F filter weights $Q^f \in \mathbb{R}^{a \times b}$, $f = 1, \dots, F$, generating each a so-called *feature map* $Y^f \in \mathbb{R}^{n' \times m'}$ from an input matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$, following the convolution:

$$Y_{i,j}^f = \sum_{k=0}^{a-1} \sum_{l=0}^{b-1} Q_{a-k,b-l}^f X_{1+s(i-1)-k, 1+s(j-1)-l}, \quad (1)$$

where $s \geq 1$ is the *stride* parameter, $n' = 1 + \lfloor \frac{n+a-2}{s} \rfloor$ and $m' = 1 + \lfloor \frac{m+b-2}{s} \rfloor$.

A convolutional neural network (CNN)-based autoencoder is assumed, whose structure is illustrated in Table I and Fig. 2. The convolutional layer, whose transformation is described in Eq. 1, allows the transmitter to process a sequence of symbols S , where a total number of $k \times L$ bits are handled simultaneously. Furthermore, each symbol of the sequence S is encoded as an *one-hot vector* $\mathcal{O}_s \in \mathbb{R}^{2^k}$, i.e., an 2^k -dimensional vector, the s th element of which is equal to one and zero otherwise.

In addition to facilitating linear/non-linear block encoding of the input sequence through the use of Exponential Linear Unit (ELU) activation functions, the convolutional layers at the transmitter transforms the one-hot input symbols sequence S to a new signal representation $\mathbf{X} = f(S)$, which occupies n channel use. Therefore, signal constellation points are mapped into $2n$ -dimensional space. Each convolutional layer is followed by a batch normalization layer before activation to increase the network stability, and has 256 filters, allowing mapping each one-hot vector to 256 -dimensional space and search for the most suitable representation of the input symbol.

Normalization layer is used to guarantee the transmitter power constraint, compressing the symbol representation to $2n$ -dimensional space, considering that each n channel slot has I/Q channels.

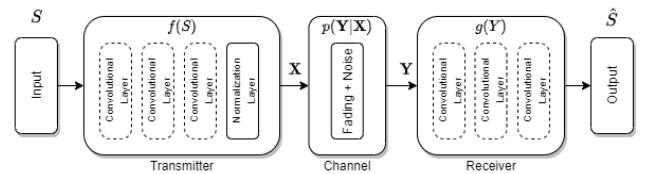


Fig. 2: Autoencoder representation.

The channel layer of Fig. 2 is described as the conditional probability density function $p(\mathbf{Y}|\mathbf{X})$ following an α - μ distribution, which will be revisited in the next section. Furthermore, an additive Gaussian white noise with a fixed variance $\sigma^2 = (2RE_b/N_0)^{-1}$ is added to the signals, where $R = k/n$ (bits/channel use) denotes the rate and E_b/N_0 SNR.

TABLE I: Autoencoder Layout

Block	Layer	Activation	Output Dim.
Encoder	Input		$L \times 2^k$
	Conv1D	ELU	$L \times 256$
	Conv1D	ELU	$L \times 256$
	Conv1D	LINEAR	$L \times 2n$
	Normalization		$L \times 2n$
Channel	Fading + Noise		$L \times 2n$
Decoder	Conv1D	ELU	$L \times 256$
	Conv1D	ELU	$L \times 256$
	Conv1D	SOFTMAX	$L \times 2^k$

The receiver, whose architecture is similar to the transmitter but without the normalization layer, decompresses the received signal \mathbf{Y} in order to extract adequate information, classifying each signal out of 2^k possibilities. A perfect Channel State Information (CSI) is assumed, which is fed into the receiver network together with \mathbf{Y} . The signals are mapped to an one-hot vector for soft decision using soft-max activation, whose output is a probability vector over all possible input sequences S , as illustrated in Table I. The entire transformation is denoted as $\hat{S} = g(\mathbf{Y})$, and it corresponds to the index of the probability vector element with the highest probability, as indicated in Fig. 2.

The goal of the self-learning system is to find the best set of parameters \mathbf{W}^* which minimizes the loss function $L(\mathbf{W})$, i.e.,

$$\mathbf{W}^* = \arg \min_{\mathbf{W}} L(\mathbf{W}). \quad (2)$$

This is done by means of the well suited stochastic gradient descent (SGD) algorithm, which starts with some random initial values of $\mathbf{W} = \mathbf{W}_0$ and then updates \mathbf{W} iteratively as

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \eta \nabla_{\mathbf{W}} \tilde{L}(\mathbf{W}_t) \quad (3)$$

$\eta > 0$ denotes the *learning rate*, while $\nabla_{\mathbf{W}}$ is the gradient of the approximated Binary Cross-Entropy (BCE) $\tilde{L}(\mathbf{W})$, which is computed for a random *minibatch* of training examples $N_t \subset \{1, 2, \dots, N\}$ of size N_t at each iteration, i.e.,

$$\tilde{L}(\mathbf{W}) = -\frac{1}{N_t} \sum_{i \in N_t} S_i \log(g(\mathbf{Y})_i) \quad (4)$$

which can be (under the assumptions) calculated using back-propagation [21] through the entire dataset of size N .

III. CHANNEL MODEL

This section revisits the α - μ physical model. Furthermore, theoretical block error rate (BLER) under α - μ fading channel are shown for classic modulation schemes, such as BPSK and QAM.

A. α - μ Physical Model

The α - μ is a generalized fading distribution for small-scale variation of the fading signal in a non line-of-sight fading condition. Besides considering the signals composed by clusters of multipath waves, the α - μ distribution accounts

for the non-linearity of a propagation medium as well. Many important small-scale fading channels are special cases of $\alpha - \mu$ distribution, such as Exponential, Rayleigh, Nakagami- m , Gamma, and Weibull. The power parameter ($\alpha > 0$) is related to the non-homogeneity of the environment, whereas the parameter ($\mu > 0$) is associated to the number of multipath clusters. As a result, the obtained envelope is a non-linear function of the modulus of the sum of the multipath components, i.e.,

$$R^\alpha = \sum_{i=0}^{\mu} X_i^2 + Y_i^2 \quad (5)$$

X_i and Y_i are mutually independent Gaussian processes with zero mean and normalized variance $\sigma_i = 1/2\mu$.

The probability density distribution $f_R(r)$ of a fading α - μ envelope R is given by [20] as

$$f_R(r) = \frac{\alpha \mu^\mu r^{\alpha\mu-1}}{\hat{r}^{\alpha\mu} \Gamma(\mu)} \exp\left(-\mu \frac{r^\alpha}{\hat{r}^\alpha}\right). \quad (6)$$

The parameter $\hat{r} = \sqrt[\alpha]{\mathbb{E}[R^\alpha]}$ is the α -root mean value of the channel envelope, $\Gamma(z) = \int_0^\infty t^{z-1} \exp(-t) dt$ is the Gamma function, and $\mathbb{E}[\cdot]$ the expectation operator.

The Weibull distribution can be obtained from the $\alpha - \mu$ distribution by setting $\mu = 1$. Rayleigh distribution arises from Weibull when $\alpha = 2$. The Nakagami- m distribution can be obtained from $\alpha - \mu$ by setting $\alpha = 2$ for different $m = \mu$ representing the multipath clusters.

B. Block Error Rate

In order to express the BLER of flat $\alpha - \mu$ fading channels with envelope R , we first define γ as the instantaneous SNR $\gamma = R^2$, and the average SNR $\bar{\gamma} = \mathbb{E}[\gamma]$. Hence, the probability density function of γ can be found by following the standard procedure of random variables transformation from (6)

$$f_\gamma(\gamma) = \frac{\alpha \mu^\mu \gamma^{\frac{\alpha\mu}{2}-1}}{2\Gamma(\mu) \bar{\gamma}^{\frac{\alpha\mu}{2}}} \exp\left(-\mu \left(\frac{\gamma}{\bar{\gamma}}\right)^{\frac{\alpha}{2}}\right). \quad (7)$$

The unconditional error probability P_e is defined as

$$P_e = \int_0^\infty p(\text{error}|\gamma) f_\gamma(\gamma) d\gamma, \quad (8)$$

where $p(\text{error}|\gamma)$ differs for each distinct modulation scheme. For BPSK and QPSK systems, the error probability can be approximated [22] as

$$p(\text{error}|\gamma) = Q(\sqrt{2\gamma}),$$

whereas for M-QAM systems:

$$p(\text{error}|\gamma) = \frac{4}{\log_2 M} \left(1 - 2^{-\frac{\log_2 M}{2}}\right) Q\left(\sqrt{\frac{3\gamma \log_2 M}{M-1}}\right).$$

where $Q(\cdot)$ is the Q -function $Q(x) = \frac{1}{2\pi} \int_x^\infty \exp(-\frac{u^2}{2}) du$.

Combining the required equations with (7) and (8), the BLER or probability of error in a message is followed by

$$P_{e^M} = 1 - (1 - P_e)^k, \quad (9)$$

where k is the number of bits per symbol.

IV. PERFORMANCE ANALYSIS

In this section, in order to investigate the performance of the CNN-based autoencoder over the said non-homogeneous fading environment, numerous simulation results were conducted. The system model of Fig. 2 was employed and all the neural network parameters were given in Table I, where the kernel size and the stride were set to 1, handling each symbol individually. The training and validation datasets are generated randomly using independent and identically distributed (i.i.d.) binary bit sequences, drawn from a uniform distribution. The self-learning system was trained using 16000 data messages, where each message contains a block length of L symbols, conveying k information bits each. The network was tested through 80000 data messages. The optimization used to train the end-to-end system uses Adam [23] to guarantee fast convergence, where the learning rate was set to $\eta = 0.001$ and decayed by a factor of 10 when saturated. 100 epochs were used for training. For all other implementation details, we refer to the source code [24].

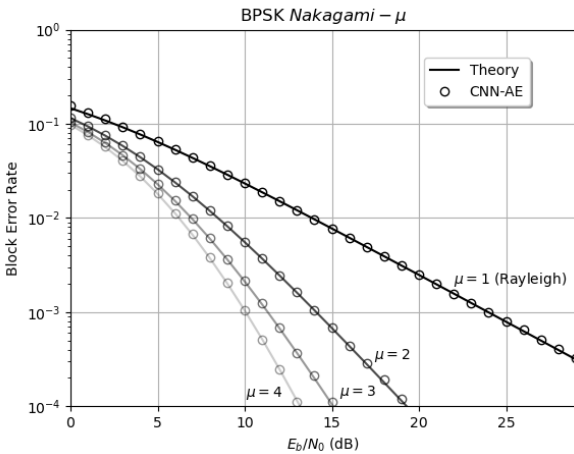


Fig. 3: BLER performance of the self-learning system under Nakagami- μ fading channel.

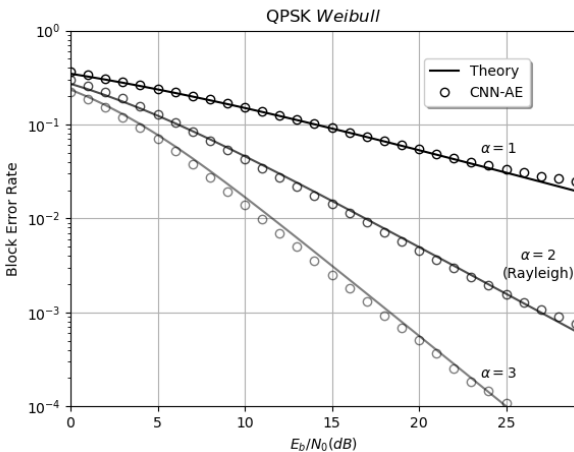


Fig. 4: BLER performance of the self-learning system under Weibull fading channel.

Setting the power parameter $\alpha = 2$, the Nakagami- m distribution can be obtained from the α - μ distribution, for different $m = \mu$. Under this scenario, Fig. 3 shows the BLER performance of the CNN-AE with fixed rate $R = 1$ (bit/channel use), trained at a fixed $E_b/N_0 = 14$ dB. Fixing μ , the Weibull distribution arises for different α . This scenario is shown in Fig. 4, where the self-learning system was trained at $E_b/N_0 = 16$ dB and $R = 2$. In both schemes, the receiver learns to equalize the severe fading effects before decoding, and the resultant BLERs match the performance of conventional BPSK modulation (for $R = 1$) and QPSK ($R = 2$), illustrated as solid lines. Lastly, Rayleigh fading comes up in both scenarios when we set $\alpha = 2$ and $\mu = 1$, and we consider only $n = 1$ channel use.

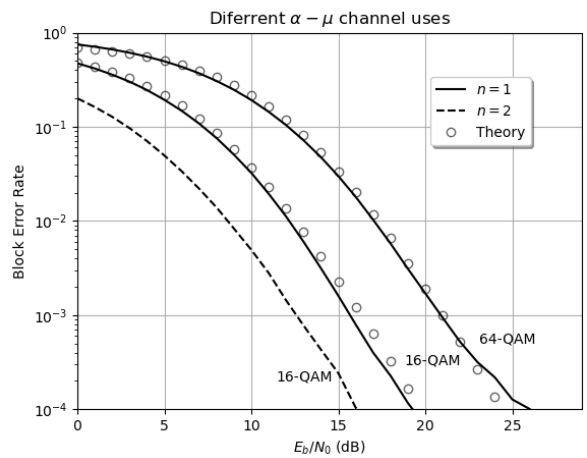


Fig. 5: BLER performance of the CNN-based system under $\alpha - \mu$ fading for different channel uses and rates. The circles indicate the theoretical correspondence.

Increasing channel uses, it is expected that a better BLER can be achieved. This scenario is illustrated in Fig. 5, showing the significant BLER gain performance when we set $n = 2$ to the intelligent system of Fig. 2, trained at a fixed rate $R = 4$ (bits/channel use), and training $E_b/N_0 = 27$ dB, transmitted over flat $\alpha - \mu$ channels. It also illustrates the performance of $R = 6$ systems, trained at $E_b/N_0 = 27$ dB. The CNN-based system was able to learn suitable symbol transformations, achieving the BLER performance that corresponds to the conventional 16-QAM and 64-QAM counterparts, as expected.

Fig. 6 shows an example of the learned representations for $R = 2$ as complex constellation points. It illustrates that the system converges quickly to classical QPSK constellation with some arbitrary rotation.

V. CONCLUSIONS

In this paper, we investigated the BLER performance of a CNN-based autoencoder as an end-to-end communication system, adapted to comprise a generalized fading scenario, namely the α - μ model. More explicitly, several constraints were imposed to the neural layers that compose the system, such as power constraints, time constraints, and the α - μ fading model. The system self-learned in a supervised manner with

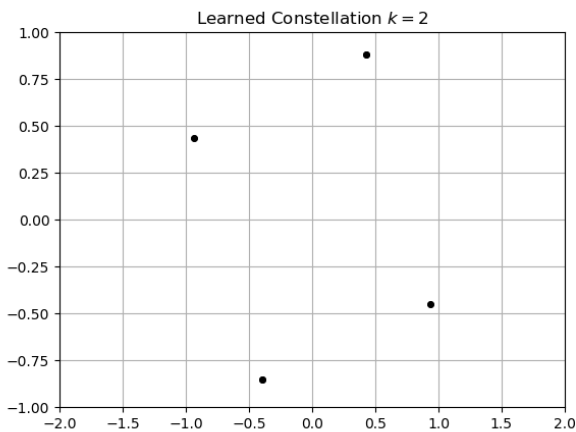


Fig. 6: Constellation produced by the intelligent system for $k = 2$ and $n = 1$.

sufficient data, converging to the exact network parameters that are suitable for pre-defined communication requirements. Moreover, the generalization capability, regardless of block length, training SNR, code rates, and channel uses, is validated under the generalized fading scenario, for which CSI is assumed. Finally, analytical results were obtained so as to validate the analysed system.

As future works, in order to eliminate the need for CSI at the receiver, a Differential CNN-based Autoencoder (DCNN-AE) system analysis under the same fading assumptions can be investigated. In addition, the impact of multiple destination/sources as well as more comprehensive fading models can be considered. Finally, the effect of multiple antennas on the overall system performance arises as an interesting subject for investigation.

REFERENCES

- [1] N. Samuel, T. Diskin and A. Wiesel, "Deep MIMO detection," *Proc. IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, July 2017.
- [2] N. Samuel, T. Diskin and A. Wiesel, "Learning to Detect," *IEEE Transactions on Signal Processing*, vol. 67, no. 10, pp. 2554-2564, May 2019.
- [3] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders and S. Pollin, "Deep Learning Models for Wireless Signal Classification With Distributed Low-Cost Spectrum Sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433-445, Sept. 2018.
- [4] N. Farsad, M. Rao and A. Goldsmith, "Deep learning for joint source-channel coding of text," *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2018.
- [5] T. Gruber, S. Cammerer, J. Hoydis and S. ten Brink, "On deep learning-based channel decoding," *Proc. Information Sciences and Systems (CISS)*, March 2017.
- [6] D. Neumann, T. Wiese, and W. Utschick, "Learning the MMSE channel estimator," *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 2905-2917, Jun. 2018.
- [7] M. Soltani, V. Pourahmadi, A. Mirzaei and H. Sheikhzadeh, "Deep Learning-Based Channel Estimation," *IEEE Communications Letters*, vol. 23, no. 4, pp. 652-655, April 2019.
- [8] X. Wang, L. Gao, S. Mao and S. Pandey, "CSI-based fingerprinting for indoor localization: a deep learning approach," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 763-776, January 2017.
- [9] T. Erpek, Y. E. Sagduyu and Y. Shi, "Deep learning for launching and mitigating wireless jamming attacks," *IEEE Transactions on Cognitive Communications and Networking*, 2018
- [10] M. Sadeghi and E. G. Larsson, "Adversarial attacks on deep-learning based radio signal classification," *IEEE Wireless Communications Letters*, vol. 8, no. 1, pp. 213-216, Feb. 2019.
- [11] E. Zehavi, "8-psk trellis codes for a rayleigh channel," *IEEE Transactions on Communications*, vol. 40, no. 5, pp. 873-884, May 1992.
- [12] T. J. O'Shea, K. Karra and T. C. Clancy, "Learning to communicate: Channel auto-encoders, domain specific regularizers, and attention," *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, Limassol, pp. 223-228, 2016.
- [13] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563-575, Dec 2017.
- [14] T. J. O'Shea, L. Pemula, D. Batra and T. C. Clancy, "Radio transformer networks: Attention models for learning to synchronize in wireless systems," *50th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, 2016, pp. 662-666, 2016.
- [15] S. Dörner, S. Cammerer, J. Hoydis and S. t. Brink, "Deep Learning Based Communication Over the Air," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 132-143, Feb. 2018.
- [16] G. Gui, H. Huang, Y. Song and H. Sari, "Deep Learning for an Effective Nonorthogonal Multiple Access Scheme," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8440-8450, Sept. 2018.
- [17] Y. LeCun, "Generalization and network design strategies," *Connectionism in Perspective*, vol. 19. Amsterdam The Netherlands: Elsevier, 19289.
- [18] B. Zhu, J. Wang, L. He and J. Song, "Joint Transceiver Optimization for Wireless Communication PHY Using Neural Network," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1364-1373, June 2019.
- [19] N. Wu, X. Wang, B. Lin and K. Zhang, "A CNN-Based End-to-End Learning Framework Toward Intelligent Communication Systems," in *IEEE Access*, vol. 7, pp. 110197-110204, 2019.
- [20] M. D. Yacoub, "The α - μ Distribution: A Physical Fading Model for the Stacy Distribution," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 1, pp. 27-34, Jan. 2007, doi: 10.1109/TVT.2006.883753
- [21] R. Hecht-Nielsen, "Theory of the backpropagation neural network," in *Neural network for perception*. Elsevier, 1992, pp. 65-93.
- [22] J. R. Barry, E. A. Lee, and D. G. Messerschmitt, "Digital Communication." Springer US, 2012.
- [23] D. P. Kingma, and J. L. Ba, "Adam : A method for stochastic optimization," 2014, arXiv:1412.6980v9.
- [24] S. B. F. Gomes, and M. D. Yacoub. (2020). *Source Code*. [Online]. Available: <https://github.com/SamuelBFG/cnn-based-autoencoder/>