Joint Transceiver Optimization for Wireless Communication PHY with Convolutional Neural Network

Banghua Zhu, Jintao Wang, Longzhuang He, and Jian Song

August 10, 2018

Abstract

Deep Learning has a wide application in the area of natural language processing and image processing due to its strong ability of generalization. In this paper, we propose a novel neural network structure for jointly optimizing the transmitter and receiver in communication physical layer under fading channels. We build up a convolutional autoencoder to simultaneously conduct the role of modulation, equalization and demodulation. The proposed system is able to design different mapping scheme from input bit sequences of arbitrary length to constellation symbols according to different channel environments. The simulation results show that the performance of neural network based system is superior to traditional modulation and equalization methods in terms of time complexity and bit error rate (BER) under fading channels. The proposed system can also be combined with other coding techniques to further improve the performance. Furthermore, the proposed system network is more robust to channel variation than traditional communication methods.

Index terms— Deep learning, modulation, equalization, autoencoder, frequency selective fading

1 Introduction

One of the most critical problem in wireless communication techniques is how information gets transmitted from one end (transmitter) to another end (receiver) reliably. During the transmission, the signal suffers from distortion and noise due to complicated channel state and hardware imperfections. In the
study of physical layer of the OSI model (PHY), the whole system is optimized in a divide-and-conquer perspective [1]. The PHY transmitter usually includes source coding, channel coding, and modulation module, while the PHY receiver includes synchronization, channel estimation, equalization, demodulation, channel decoding, source decoding, and so on. Each part is optimized separately and requires a great amount of expert knowledge. Massive research has been focused on the optimization of each module for different channel environments and application demands. According to data processing theorem [2] in information theory, the optimization of sub-modules for communications cannot guarantee global optimality for the whole communication system. In fact, such an implementation is known to be sub-optimal [3].

In the past decades, deep learning has seen wide and successful application in computer vision and natural language processing due to its strong ability of generalization. In order to further improve the transmission and network performance, the fifth generation communication system (5G) will apply many new techniques, such as massive MIMO [4], mmWave [5], and ultra-dense wireless network [6]. As a result, it also raises a great number of challenges and opportunities. In communication area, deep learning is applied to communication network [7] and cognitive radio [8]. However, since physical layer is quite complex and requires real time and high accuracy transmission, the development of related technology has been insufficient until recent years [9]. With the development of neural network compression technique and specialized hardware such as GPU and FPGA, the cost and time complexity of deep learning related techniques are significantly decreased. It is possible to run neural network on mobile devices and antennas. There has been massive research on deep learning based techniques on individual modules such as channel decoding [10–14], signal detection [15–18], and etc.

In deep learning based communication systems, it is possible to optimize the transmitter and receiver jointly with a structure of autoencoder [19] instead of artificially introduced block schemes [20–22]. In previous efforts, the autoencoder is able to automatically design the mapping scheme from an one-hot encoded signal to constellation symbols and the way of demapping under AWGN channel. The network is proven to outperform the traditional coding techniques such as uncoded BPSK and Hamming coded BPSK scheme in the aspect of block error rate [21]. However, the application of this work is still limited since the input can only be short one-hot vector, which carries much less information than a random bit sequence of the same length. Also, uncoded BPSK and Hamming coded BPSK scheme is not specially designed for one-hot vector. The training of neural network has a natural advantage over the benchmark. Furthermore, the size of the dense layer based autoencoder would grow with the increased length of the input sequence, which would lead to quadratic growth in time complexity. Such issues as above have not been well addressed by the current research.

In another work on autoencoder communication system [23], the authors addressed the issue of receiver synchronization by introducing a frame synchronization scheme based on another neural network. The authors further proposed
a transfer learning based two-phase procedure to overcome the problem of missing channel gradient during training. The transmitter and receiver are trained under a stochastic channel model, and the receiver is then fine-tuned with respect to real channels. These work enable practical over-the-air transmission through pure neural networks.

In this paper, we focus on further exploring the application of autoencoder in physical layer. Our main contribution is listed as follows.

- We propose a novel structure based on convolutional autoencoder which is able to jointly optimize the transmitter and receiver design from overall system performance point of view which has the following virtues: 1. the property of convolutional autoencoder enables the trained network to process input bit sequence of any length, 2. the soft output of this structure can be the input of any soft decoder, which can be easily combined with any other soft-input-soft-output channel decoding schemes, and 3. the proposed structure can be flexibly applied to either time or frequency domains.

- We have confirmed that the proposed structure is able to handle the challenge of mapping scheme of different levels under various channels, including AWGN channel, fading channels and non-Gaussian noise channels.

- From various BER performance evaluation and time complexity analysis of the proposed system, the robustness of the system is demonstrated. The proposed system structure is less sensitive to channel variation compared with traditional design, showing the great potential for the design principle and methodology of future telecommunication systems.

**Notation.** Bold capital letters refer to matrices and bold lowercase letters represent vectors. The subscript on a lowercase letter $y_i$ represent the $i$-th element of vector $y$. For two real number $a < b$, $[a, b]$ refers to the set of all real number $x$ satisfying $a < x < b$ while $[a, b]^n$ refers to the set of all $n$ dimensional vector with each element in $[a, b]$. $\mathbb{R}^n$ represents the space of all $n$ dimensional real vectors. $\mathbb{C}^n$ represents the space of all $n$ dimensional complex vectors. Functions $\text{real}(\cdot) : \mathbb{C}^n \to \mathbb{R}^n$ and $\text{imag}(\cdot) : \mathbb{C}^n \to \mathbb{R}^n$ turn a complex vector to a real vector by taking the real or imaginary part of each element.

## 2 Deep Learning Basics

A feedforward neural network [24], or multilayer perceptron, describes a mapping $f(x_i; w_i) : \mathbb{R}^n_i \to \mathbb{R}^{n_{i+1}}$ in the $i$-th layer. Here the mapping is generally a linear transformation determined by the parameter $w_i$ plus activation function to introduce nonlinearity. The output of the $i$-th layer is fed into next layer as input. The feedforward neural network with total layers of $L$ can be represented in (1).

$$x_{i+1} = f(x_i; w_i), \quad i = 1, 2, \ldots, L - 1$$ (1)
Given enough amount of training data, i.e. multiple pairs of input vector $x_0$ and output vector $x_L$, the mapping from input vector $x_0$ to output $x_L$ is approximated by the cascaded function in equation (1). Denote $f_i(\cdot) = f(\cdot; w_i)$, we have $x_L = f_{L-1}f_{L-2} \ldots f_0(x_0)$.

The training of a neural network is based on mini-batch stochastic gradient descent and back propagation methods. The gradient is calculated and propagated backward from the last layer to the first layer. The parameters $w_i$ are updated accordingly.

There are multiple kinds of the layer $f(\cdot; w_i)$. Common choices are fully connected layer, convolutional layer \[25\] and recurrent layer \[26\] etc. A fully connected layer is also called a dense layer, where each element of the input is connected to all the elements of its output vector. The dense layer is common and effective in most of the application areas, but is relatively high in computational complexity. The convolutional layer consists of a set of learnable filters called kernel. It performs a simple convolution along the input data. In our work, a 1D convolutional layer is frequently used to convolve the sequence with learnable filters. An illustration of how a convolutional layer works is shown in Fig. 1.

We also use a layer wrapper called time-distributed wrapper. It applies a layer to every temporal slice of its input instead of all the elements of the input. The function of a time-distributed dense on a 2D array is similar to a 1D convolutional layer with kernel size = 1.

Figure 1: The illustration of a 1D convolutional neural network
3 System Model

We consider a typical communication system with block diagram in Fig. 2. The input data modeled as a sequence of i.i.d. bits $s$ is coded and modulated by the transmitter. The transmitted data $x$ is transmitted through a linear time-invariant channel with impulse response coefficients $h_n$. The distortion and noise introduced by channel can be modeled as equation (2).

$$y_i = \sum_{n=0}^{L_h-1} h_n x_{i-n} + n_i$$  \hspace{1cm} (2)

The received data $y$ is equalized, demodulated and decoded by the receiver to recover the original sequence $\hat{s}$.

There is a restriction on the power of transmitted symbol in the real transmitter. Usually the average power for the signal is constrained to 1, i.e. $\frac{1}{n||x||_2^2} = 1$.

The procedure of the communication system can be viewed as the cascade of three function $\hat{s} = g_3(g_2(g_1(s)))$. Here $g_1 : [0, 1]^n \rightarrow \mathbb{C}^m$ is the function of the operation in transmitter. $g_2 : \mathbb{C}^m \rightarrow \mathbb{C}^m$ is the channel function defined in equation (2). $g_3 : \mathbb{C}^m \rightarrow [0, 1]^n$ is the receiver function to recover the original information.

The similarity between the representation of communication system and neural network leads us to optimize the communication system using neural networks. In the rest part of the paper, we study the problem of jointly optimizing function $g_1$ and $g_3$ with fixed $g_2$. 

![Block diagram for a typical communication system](image-url)
4 Autoencoder for Time Domain Transmission

4.1 Network Structure

The property of neural network enables us to train the model using only input bit sequences under different channel state.

We jointly optimize the transmitter and receiver with an autoencoder structure. Convolutional neural network is used considering the sequential property of the input sequence. The network structure is shown in Fig. 3.

Assume that the system is transferring \( k \times M \) bits information \( s \in \{0, 1\}^{k \times M} \). Here \( k \) represents the number of bits each symbol carries. And \( M \) is the length of symbols. The transmitter maps the input bit stream to a sequential complex vector \( x \in \mathbb{C}^M \) and transmits \( x \) to the channel in the time domain. The received signal \( y \in \mathbb{C}^N \) with distortion and noise is then equalized and demapped in the receiver to recover the original bit stream \( s \).

We compress the length of the input from \( k \times M \) to \( M \) in the first convolutional layer with stride size = \( k \). And we combine the usage of time-distributed dense layer with convolutional layer to introduce further correlation and nonlinearity in the transmitter. The transmitted symbol \( X = [\text{real}(x); \text{imag}(x)] \)
is of size $M \times 2$ as a complex vector. In the channel layer, the input symbols are first normalized to satisfy the power constraint. For AWGN channel, only additive white gaussian noise is added to the normalized symbols. For fading channels, the normalized symbols first convolve with the impulse response in the time domain. The convolution in complex number is implemented as in equation (3). In neural networks, it can be represented by a 1D convolutional layer that convolves $X$ with a 3D tensor.

$$\begin{align*}
\text{real}(y_i) &= \sum_{n=0}^{L-1} (\text{real}(h_n) \text{real}(x_{i-n}) - \text{imag}(h_n) \text{imag}(x_{i-n})) + \text{real}(n_i) \\
\text{imag}(y_i) &= \sum_{n=0}^{L-1} (\text{real}(h_n) \text{imag}(x_{i-n}) - \text{imag}(h_n) \text{real}(x_{i-n})) + \text{imag}(n_i)
\end{align*}$$

Generally a neural network suffers from the restriction of input shape, i.e. the length of input for testing shall be the same as that in training procedure. However, due to the locally-connected property of convolutional layer and time-distributed layer, the proposed network structure is able to accept input sequences of any length without the need of retraining the whole model. Thus the system can process long sequences while trained on short ones.

We conduct massive experiments to analyze the performance of our model. We train our model separately on AWGN channel and fading channel. In the rest part of this chapter, we give thorough analysis on the result of the learned system.

## 4.2 Setting

In our experiment, we set $k = 6$ and compare the learned system with 64QAM for AWGN channel, and 64QAM plus minimum mean square error (MMSE) estimation [27] for fading channel. We also test the proposed model on $k = 8$ case with 256QAM+MMSE in fading channel to prove the extensibility of our model. The modulation scheme is selected considering the throughput fairness. For the training and testing our model, we randomly generate i.i.d. bit sequences. The generated dataset is separated arbitrarily into training set, validation set and test set. The property of convolutional neural network enables the network to process input sequence of any length without changing network parameters. The change in the length of the input sequence would not affect the performance of our system. Thus we fix $M$ to be 400.

We train the autoencoder with 30,000 training samples and test that with 10,000 test data. The learning rate is set to be 0.001 and the batch size is 32. We train the system at a specific signal to noise ratio (SNR) but test at a wide range of SNR, as well as robustness and adaptivity to deviations from the AWGN and fading setting. We would like to show that although the state
space of input bit sequence is as large as $2^{k \times M}$, the model can generalize well with mere 30,000 training samples.

In the time domain transmission system, we test three different channels, one AWGN channel and two fading channels. The amplitude and delay for two fading channels are plotted in Fig. 4. The integer in x-axis is the delay of each path in the unit of symbols. Note that the phase information is omitted in the figure.

![Figure 4: The amplitude of two fading channels tested. The left is referred to as channel A and the right is channel B.](image)

### 4.3 AWGN Channel

The BER performance for AWGN channel is shown in Fig. 5. The CNN based autoencoder can bring up to 1 dB gain when bit error rate (BER) ranges from $10^{-1}$ to $10^{-3}$.

The constellation diagram is shown in Fig. 6. We plot 40,000 modulated symbols in a complex plane. The system learns an amplitude and phase-shift keying (APSK) like constellation method. However, the symbols seem not to concentrate into the finite constellation points, but are distributed as clusters instead. The clustered structure of constellation symbols may have higher requirements for antennas, but may bring extra gain for low SNR regime. The learned constellation is close to a Gaussian distribution, which is optimal for a maximum likelihood receiver according to information theory.

### 4.4 Fading Channel

We train the system under two Rayleigh fading channels in Fig. 4. The benchmark system adopts 64QAM modulation and MMSE detection method. MMSE is assumed to have perfect channel state information.

The comparison between CNN based method and 64QAM for fading channel A is shown in Fig. 7.
Figure 5: The comparison between 64QAM and the trained system under AWGN channel.

Figure 6: The learned constellation diagram for AWGN channel.
The trained system is able to bring up to 4 dB gain. The system is outperformed by the benchmark in high SNR regime. This also happens in the case of AWGN channel. And this phenomenon is not related to the SNR used in training procedure. We would like to point out that the performance decay in high SNR would not affect the advantage of our method in practical system since our system can be easily combined with any coding module as long as the output sequence of the coding module is i.i.d., which is easy to satisfy with current interleaving technique. The output of the neural network is a real number in $[0, 1]$. This can be viewed as a probability and fed into any soft-decoding system as soft input. We add a standard LDPC coding system with code rate $\frac{1}{2}$ to both 64QAM+MMSE system and our scheme. We can see in Fig. 8 that after combining with a coding module, our system maintains a performance advantage over the entire SNR range.

Furthermore, our system is able to bring higher gain for a channel with higher delay. We train the neural network under fading channel B. The result is shown in Fig. 9.

Traditional 64QAM system is highly tailored for AWGN channel. For a known fading channel, the proposed neural network is able to introduce pre-coding in modulation procedure. The transmitter and receiver can be jointly adjusted to fit the channels, thus making the system more robust to fading.

The proposed structure can also be extended to different $k$. We train the neural network under $k = 8$ and compare the result with 256QAM under channel A. The comparison is shown in Fig. [10]. Although the neural network is not as good as 256QAM in high SNR regime, the performance is guaranteed to
Figure 8: BER comparison after combining with LDPC under fading channel A.

Figure 9: BER comparison between 64QAM+MMSE and the system under fading channel B.
maintain an advantage over the whole SNR regime after combining with coding system, as is the case shown above.

Figure 10: BER comparison between 256QAM+MMSE and the system under fading channel A.

4.5 Robustness

In this section, we show the neural network system is much more robust against the variations in the channel. This makes, among other things, the proposed system much more attractive alternative to traditional modulation methods in practice, where the channel model is not available. We consider two kinds of channel variations. One of them is the introduction of non-Gaussian noise. We test bursty AWGN channel where a small fraction of noise has much higher variance than other on the neural network system trained under AWGN channel, and compare that with 64QAM. Usually bursty AWGN channel is used to model inter-cell interference in OFDM cellular systems or co-channel radar interference [29]. The comparison result is shown in Fig. 11. The result is expected since convolutional neural network would introduce correlation in a longer range of symbols than 64QAM, making system more robust to burst noise.

Another kind of channel imperfections is related to the time-variant property of real channel and inaccurate channel estimation. Channel in real world may change within a single frame. The signal may go through a slightly different channel from the one previous pilot passes. Furthermore, channel estimation cannot ensure perfect recovery of channel state information. In our experiment, we train the model on a fixed channel A. And test it with a different channel that adds a white Gaussian white noise with standard deviation $= 0.05$ to channel
A. For MMSE detection, it takes the fixed channel A as channel state information. We test both 64QAM+MMSE and neural network system 100 times and take the average BER performance. The comparison of robustness between 64QAM+MMSE and neural network is shown in Fig. 12. The channel variation affects 64QAM+MMSE more significantly than neural network structure.

From the experiments above, we can see that the proposed method is especially promising under extreme channel condition, like severe fading, or frequent bursty noise. Furthermore, it is more robust to channel variation than traditional methods.

4.6 Time Complexity Comparison

We provide both simulation and numerical analysis for analyzing time complexity. We first test the time complexity by running demodulation plus detection algorithms and the receiver part in neural network on a Intel (R) Corel (TM) i7-7700HQ CPU @ 2.80GHz CPU and an NVIDIA GeForce GTX 1060 GPU. The platform in this experiment is python+keras [30]. For AWGN channel, only demodulation is needed to recover the bit sequence. For fading channel, MMSE is also included in the receiver, which takes long for the FFT step. We test both methods for 100 sets of data and take the average. The comparison is shown in Table 1.

For transmitted bit sequence with length $n$, the time complexity for both CNN and QAM demodulation is $O(n)$. However, for MMSE detection, FFT requires a $O(n\log n)$. CNN is able to substitute the demodulation plus detection
Figure 12: The comparison of robustness of 64QAM+MMSE (a) and neural network (b). Both methods are tested under original channel and changed channel.
Table 1: Time Complexity Comparison between traditional demodulation method and neural network.

| Channel | Method       | 64QAM(+MMSE) | CNN   |
|---------|--------------|--------------|-------|
| AWGN    |              | 1.501s       | 3.695s|
| Fading  |              | 8.709s       | 3.751s|

algorithm with a lower time complexity in theory and higher accuracy. Thus CNN based framework is quite suitable for designing communication system in fading channels.

With the development of neural network, the dimension of the CNN can be potentially reduced with techniques such as network pruning and distillation. Parallelization is also possible in the multiplicative units in the neural network, as well as pipelining. Neural network can be further accelerated by specially designed hardware framework like GPU, FPGA and TPU etc. These designs along with a careful analysis of the fixed point arithmetic requirements of the different weights are under active research. The efficiency of neural network can be further improved in the future.

5 Autoencoder for Frequency Domain Transmission

5.1 Network Structure

In the previous chapter, we studied the effect of the multipath fading channel. In Orthogonal Frequency Division Multiplexing (OFDM) [31] system, the inter-symbol interference can be eliminated by introducing guard interval between each subcarrier. Cyclic prefix is a typical guard interval, which makes each subcarrier orthogonal to each other. In the following part of this chapter, we assume perfect cyclic prefix as guard interval. Thus there is no inter-symbol inference and inter-channel inference. However, there is still fading on each subcarrier. One of the most common equalization method for this issue is zero forcing (ZF) [32]. Since the fading on individual subcarrier might be quite significant, it is hard to recover the signal from those subcarriers due to poor SNR. The information carried by some subcarriers might experience deep fading and thus get lost during transmission. In fact, a zero order equalization system would cause inevitable loss of information [33]. By introducing correlation between subcarriers, the information can be carried by nearby subcarriers. Thus the burst error on subcarriers with deep fading can be decreased.

Based on previous network structure, we design a frequency domain equalization system that is able to retrieve more information than ZF. Since the fading on each subcarrier is different, a convolutional layer that shares weight along the whole input sequence may not be suitable. Here locally connected layer is
used to substitute some of the convolutional layers in previous structure. The locally connected layer works similarly as the convolutional layer, except that weights of kernels are unshared. That is, a different set of filters is applied at each different patch of the input. The whole network structure is shown in Fig. 13.

Figure 13: DNN framework for frequency domain transmission in physical layer.

5.2 Simulation

We test the new structure under the case of an OFDM system transmission. The bit sequence is modulated and transmitted in frequency domain. All other settings are the same as the previous example. We consider the frequency transformation of channel B in Fig. 4. The frequency selective fading channel in frequency domain is shown in Fig. 14. We train and test our model on the same channel. And compare that with 64QAM+ZF method. Here ZF is assumed to have accurate channel state information. But the zero points in the fading plot would prohibit a significant number of subcarriers from transmitting information to the receiver.
The BER performance for both 64QAM+ZF and CNN under OFDM frequency selective fading channel is shown in Fig. 15. The BER for CNN based system can reach as low as $10^{-4}$ while the BER for traditional 64QAM+ZF method goes down slow. Because of the correlation introduced in convolutional layer, the neural network is able to assign different amount of information to different subcarriers according to the statistical property of the noise. The function of the neural network is similar to adaptive bit loading techniques [34] that assign bits efficiently based on subcarrier quality. This may explain the huge gain the neural network brings compared to traditional 64QAM+ZF system.

6 Conclusion

In this paper, we propose a convolutional autoencoder structure that is able to automatically design communication physical layer scheme according to different channel status. The system has no restriction on the length of input bit sequence. We conduct massive experiment to give empirical evidence for the superiority of the proposed system. The neural network has lower time complexity and higher accuracy especially for fading channel, and is also quite robust to channel variation. The framework can also be extended to OFDM system which transmits in frequency domain.

The trained autoencoder may not compete with the state-of-the-art system that is optimized over the past decades. But the autoencoder is able to learn
the way of mapping and demapping for any known channel without prior mathematical model and analysis. We may further explore the feasibility and utility of neural network based communication methods in following aspects.

- One of the most important goals of designing a communication system is to maximize the capacity, i.e. the mutual information between input and output. However, since the constellation diagram in neural network based system is continuously distributed in the complex plane. It is hard for us to estimate the mutual information accurately, let alone optimizing the mutual information within neural network. A framework for analyzing mutual information in neural network based communication system may significantly enlarge our knowledge about both neural network and communication system.

- An iterative, soft-input soft-output receiver can significantly improve the BER performance. Designing a receiver with both log likelihood ratio and received symbol as input using neural network may also enable iteration inside receiver, thus improving the current performance.

- The performance of neural network is still not as good in high SNR regime. It is important to figure out how autoencoder can learn a communication physical layer that outperforms existing communication techniques even in the high-SNR regions.

Figure 15: BER comparison between 64QAM+ZF and CNN under OFDM frequency selective fading channel.
References

[1] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge university press, 2005.

[2] J. Ziv and M. Zakai, “On functionals satisfying a data-processing theorem,” *IEEE Transactions on Information Theory*, vol. 19, no. 3, pp. 275–283, 1973.

[3] E. Zehavi, “8-PSK trellis codes for a rayleigh channel,” *IEEE Transactions on Communications*, vol. 40, no. 5, pp. 873–884, 1992.

[4] P. F. Driessen and G. J. Foschini, “On the capacity formula for multiple input-multiple output wireless channels: A geometric interpretation,” in *Communications, 1999. ICC’99. 1999 IEEE International Conference on*, vol. 3. IEEE, 1999, pp. 1603–1607.

[5] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, “Millimeter wave mobile communications for 5G cellular: It will work!” *IEEE access*, vol. 1, pp. 335–349, 2013.

[6] X. Ge, S. Tu, G. Mao, C.-X. Wang, and T. Han, “5G ultra-dense cellular networks,” *IEEE Wireless Communications*, vol. 23, no. 1, pp. 72–79, 2016.

[7] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, “Deep learning for wireless physical layer: Opportunities and challenges,” *China Communications*, vol. 14, no. 11, pp. 92–111, 2017.

[8] M. Bkassiny, Y. Li, and S. K. Jayaweera, “A survey on machine-learning techniques in cognitive radios,” *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1136–1159, 2013.

[9] M. Ibnkahla, “Applications of neural networks to digital communications—a survey,” *Signal processing*, vol. 80, no. 7, pp. 1185–1215, 2000.

[10] E. Nachmani, Y. Be’ery, and D. Burshtein, “Learning to decode linear codes using deep learning,” in *Communication, Control, and Computing (Allerton), 2016 54th Annual Allerton Conference on*. IEEE, 2016, pp. 341–346.

[11] S. Cammerer, T. Gruber, J. Hoydis, and S. t. Brink, “Scaling deep learning-based decoding of polar codes via partitioning,” *arXiv preprint arXiv:1702.06901*, 2017.

[12] T. Gruber, S. Cammerer, J. Hoydis, and S. ten Brink, “On deep learning-based channel decoding,” in *Information Sciences and Systems (CISS), 2017 51st Annual Conference on*. IEEE, 2017, pp. 1–6.
[13] F. Liang, C. Shen, and F. Wu, “An iterative BP-CNN architecture for channel decoding,” IEEE Journal of Selected Topics in Signal Processing, vol. 12, no. 1, pp. 144–159, 2018.

[14] H. Kim, Y. Jiang, R. Rana, S. Kannan, S. Oh, and P. Viswanath, “Communication algorithms via deep learning,” in The International Zurich Seminar on Information and Communication (IZS 2018) Proceedings. ETH Zurich, 2018, pp. 48–50.

[15] N. Samuel, T. Diskin, and A. Wiesel, “Deep MIMO detection,” arXiv preprint arXiv:1706.01151, 2017.

[16] Y.-S. Jeon, S.-N. Hong, and N. Lee, “Blind detection for MIMO systems with low-resolution ADCs using supervised learning,” in Communications (ICC), 2017 IEEE International Conference on. IEEE, 2017, pp. 1–6.

[17] S.-N. Hong, Y.-S. Jeon, and N. Lee, “MIMO systems with low-resolution adcs: Linear coding approach,” in Communications (ICC), 2017 IEEE International Conference on. IEEE, 2017, pp. 1–6.

[18] N. Farsad and A. Goldsmith, “Detection algorithms for communication systems using deep learning,” arXiv preprint arXiv:1705.08044, 2017.

[19] R. Salakhutdinov, “Nonlinear dimensionality reduction using neural networks,” RBM, vol. 2, p. 1000, 2000.

[20] T. J. O’Shea, K. Karra, and T. C. Clancy, “Learning to communicate: Channel auto-encoders, domain specific regularizers, and attention,” in Signal Processing and Information Technology (ISSPIT), 2016 IEEE International Symposium on. IEEE, 2016, pp. 223–228.

[21] T. J. 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, 2017.

[22] T. J. O’Shea, T. Erpek, and T. C. Clancy, “Deep learning based MIMO communications,” arXiv preprint arXiv:1707.07980, 2017.

[23] S. Dörner, S. Cammerer, J. Hoydis, and S. ten Brink, “Deep learning based communication over the air,” IEEE Journal of Selected Topics in Signal Processing, vol. 12, no. 1, pp. 132–143, 2018.

[24] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, Deep learning. MIT press Cambridge, 2016, vol. 1.

[25] Y. LeCun et al., “Generalization and network design strategies,” Connectionism in perspective, pp. 143–155, 1989.

[26] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997.
[27] D. Johnson, “Minimum mean squared error estimators,” Connexions, 2004.

[28] M. Livshitz, “Low density parity check (LDPC) code,” Jan. 17 2012, uS Patent 8,099,646.

[29] D. Fertonani and G. Colavolpe, “On reliable communications over channels impaired by bursty impulse noise,” 2008 IEEE International Symposium on Power Line Communications and Its Applications, pp. 357–362, 2008.

[30] F. Chollet et al., “Keras,” https://keras.io, 2015.

[31] B. Le Floch, M. Alard, and C. Berrou, “Coded orthogonal frequency division multiplex,” PROCEEDINGS-IEEE, vol. 83, pp. 982–982, 1995.

[32] C. B. Peel, B. M. Hochwald, and A. L. Swindlehurst, “A vector-perturbation technique for near-capacity multiantenna multiuser communication-part i: channel inversion and regularization,” IEEE Transactions on Communications, vol. 53, no. 1, pp. 195–202, 2005.

[33] I. Daubechies, A. Grossmann, and Y. Meyer, “Painless nonorthogonal expansions,” Journal of Mathematical Physics, vol. 27, no. 5, pp. 1271–1283, 1986.

[34] A. N. Barreto and S. Furrer, “Adaptive bit loading for wireless ofdm systems,” in Personal, Indoor and Mobile Radio Communications, 2001 12th IEEE International Symposium on, vol. 2. IEEE, 2001, pp. G–G.