

Deep-Waveform: A Learned OFDM Receiver Based on Deep Complex-Valued Convolutional Networks

Zhongyuan Zhao[✉], *Member, IEEE*, Mehmet Can Vuran[✉], *Member, IEEE*,
Fujian Guo, *Student Member, IEEE*, and Stephen D. Scott

Abstract—The (inverse) discrete Fourier transform (DFT/IDFT) is often perceived as essential to orthogonal frequency-division multiplexing (OFDM) systems. In this paper, a deep complex-valued convolutional network (DCCN) is developed to recover bits from time-domain OFDM signals without relying on any explicit DFT/IDFT. The DCCN can exploit the cyclic prefix (CP) of OFDM waveform for increased SNR by replacing DFT with a learned linear transform, and has the advantage of combining CP-exploitation, channel estimation, and intersymbol interference (ISI) mitigation, with a complexity of $\mathcal{O}(N^2)$. Numerical tests show that the DCCN receiver can outperform the legacy channel estimators based on ideal and approximate linear minimum mean square error (LMMSE) estimation and a conventional CP-enhanced technique in Rayleigh fading channels with various delay spreads and mobility. The proposed approach benefits from the expressive nature of complex-valued neural networks, which, however, currently lack support from popular deep learning platforms. In response, guidelines of exact and approximate implementations of a complex-valued convolutional layer are provided for the design and analysis of convolutional networks for wireless PHY. Furthermore, a suite of novel training techniques are developed to improve the convergence and generalizability of the trained model in fading channels. This work demonstrates the capability of deep neural networks in processing OFDM waveforms and the results suggest that the FFT processor in OFDM receivers can be replaced by a hardware AI accelerator.

Index Terms—Channel estimation, OFDM, deep learning, physical layer, wireless communications.

I. INTRODUCTION

DEEP learning for the physical layer (PHY) of wireless communications has been explored recently for various tasks [1]–[3], including signal classification [4], [5], parameter estimation [5]–[9], channel estimation [8]–[20], channel coding [16], [21], [22], detection [11], [16], [23]–[26], design of modulation constellation [7] and pilot design [27], [28].

Manuscript received December 26, 2020; revised March 20, 2021; accepted April 22, 2021. Date of publication June 7, 2021; date of current version July 16, 2021. This work was supported by the NSF under Grant CNS-1731833. (Corresponding author: Zhongyuan Zhao.)

Zhongyuan Zhao is with the Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005 USA (e-mail: zhongyuan.zhao@rice.edu).

Mehmet Can Vuran and Stephen D. Scott are with the Department of Computer Science and Engineering, University of Nebraska-Lincoln (UNL), Lincoln, NE 68588 USA (e-mail: mcv@unl.edu; sscott@cse.unl.edu).

Fujian Guo was with the Department of Computer Science and Engineering, University of Nebraska-Lincoln (UNL), Lincoln, NE 68588 USA. He is now with Qualcomm, San Diego, CA 92121 USA (e-mail: fguo@cse.unl.edu).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/JSAC.2021.3087241>.

Digital Object Identifier 10.1109/JSAC.2021.3087241

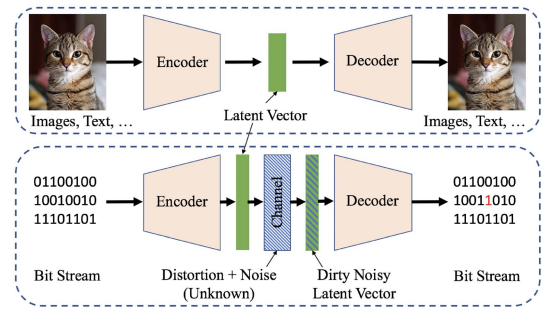


Fig. 1. General AutoEncoder (Top) v.s. Communication AutoEncoder.

Deep neural networks (DNNs) can not only enhance certain functionalities and components of wireless PHY, but also could be developed into an end-to-end novel communication architecture viewed as an autoencoder (AE) [6]–[8], [17], [21], [22]. Instead of seeking a compact embedding of structured data such as in an image and text (top of Fig. 1), a communication AE (bottom of Fig. 1) generates redundant representation of unstructured bits from which the original information can be recovered after being polluted by a noisy channel. A communication AE learns the behavior of a channel by the structure of data through self-supervised learning of a set of random bits as the training data. Above the PHY, deep learning is also used in resource allocation and network management, such as traffic prediction [29], interference alignment [30], power control [31], spectrum sharing [32], and scheduling [33].

The data-driven approach of deep learning offers the wireless PHY several advantages: i) achieving synergistic effects of combining cascading modules in the chain of signal processing [6], [7], ii) reducing the mismatch between the underlying model and reality, iii) building low complexity solutions by leveraging the non-linearity of DNN [11], [16], [25], [26], and iv) discovering irregular and/or adaptive designs, e.g., for pilot design [27], [28] and modulation constellation [6], [7].

A major distinction of wireless communication compared to image or text interpretation is the reliance on representation of wireless signals in complex field $\mathbb{C}$. However, currently, deep learning for wireless PHY lacks support of complex-valued neural networks (CVNNs) [34], [35], which is an emerging area in the machine learning discipline, from popular platforms such TensorFlow [36] and Keras [37]. Instead, existing studies treat the real and imaginary parts of a complex-valued tensor separately, $\mathbb{C} \Rightarrow \mathbb{R}^2$, such as *approx-D*: two parallel real-valued tensors [7], [11], [16], or *approx-C*: two channels of a real-valued tensor [28]. However, the complex field $\mathbb{C}$

differs from $\mathbb{R}^2$ space in multiplicative operations, which, if unaccounted for, could prevent the DNN from fully utilizing the relationship between the real and imaginary parts of signal samples in terms of phase and amplitude, resulting in increased complexity, reduced performance, and limited interpretability.

For example, orthogonal frequency-division multiplexing (OFDM) systems require (inverse) discrete Fourier transform (DFT/IDFT) and/or linear finite impulse-response (FIR) filters that are defined in the complex field $\mathbb{C}$. Without complex-valued representation, existing deep learning-based OFDM receivers [8], [10], [11], [13]–[16], [23] and AE [8] are limited to relying on DFT/IDFT in processing the OFDM waveform. Regarding technical solutions, convolutional neural networks (CNNs) [4], [5], [16], [23], [24], [28] are less often used than multilayer perceptron (MLP) [6]–[12], [21], [22], [25] in wireless PHY despite their higher efficiency, as CNN depends on the assumptions of the underlying process and is harder to design for the operations in the complex field.

In this paper, we propose a deep complex-valued convolutional network (DCCN) design to recover bits from synchronized time-domain OFDM signals. Instead of relying on DFT/IDFT, the developed end-to-end OFDM receiver learns a new way to receive OFDM waveforms with improved signal-to-noise ratio (SNR), which demonstrates the potential of deep learning for OFDM waveforms. The DCCN receiver outperforms the legacy receivers in Rayleigh fading channels with lower complexity by utilizing recent developments within the context of CVNNs. Moreover, many structural and dimensional hyperparameters of the DCCN are selected based on domain knowledge and OFDM frame structure, offering a transferable design template for other waveform structures.

The major contributions of this paper include the following: 1) We develop a learned linear transform that can replace DFT/IDFT with increased SNR in processing OFDM waveforms by exploiting its cyclic prefix (CP). The results suggest that a new hardware accelerator can potentially replace the FFT processor in OFDM receivers, and we demonstrate the potential of CVNN in learning communication waveforms. 2) We also design a data-driven interpretable DCCN channel equalizer that achieves superior performance than the legacy receivers and good generalizability at a low complexity of $\mathcal{O}(N^2)$ by combining CP exploitation [38]–[42], intersymbol interference (ISI) mitigation, and channel estimation. 3) We present a suite of training methods to improve the convergence and generalizability of the DNN-based receiver in fading channels, including a transfer learning scheme that trains the basic OFDM demodulation and channel equalization in two stages. The models are trained and evaluated in different settings of SNR values for white noise and fading channels, and using mixed Rayleigh fading models to smoothen the loss landscape of training. 4) To the best of our knowledge, DCCN is the first deep neural network-based wireless receiver that employs explicit complex-valued representation for the entire In-Phase and Quadrature (IQ) domain. A library of complex-valued dense and convolutional layers is provided in open-source software [43], and guidelines for processing complex-valued tensor with real-valued neural networks are provided to the community.

The rest of this paper is organized as follows: The related work is discussed in Section II. An overview of an OFDM system and channel estimation approaches are provided in Section III. In Section IV, the design and training approach of the DCCN receiver are presented. The numerical results are presented in Section V. Finally, the conclusions and future directions are discussed in Section VI.

II. RELATED WORK

A. Deep Neural Networks for Wireless Communications

An MLP makes no assumptions about the underlying processes, and is used in channel estimation [9]–[12], [19], detection [25], [26], and communication AE [6]–[8], [21], [22]. MLP in these studies typically has 3 to 5 layers, and follows two configurations in the signal path. The first configuration is *pass-through*: at the receiver (transmitter), MLP takes received data and pilot (input bits) and outputs estimation of the transmitted symbol or soft bits [7], [8], [11], [25], [26] (transmit IQ samples [6]–[8], [22]). Additional components may be included for synchronization in AE [6]–[8], [22]. The second configuration is the *estimator*: MLP only estimates channel [9], [10], [12], [19] or other parameters [7], [9] by utilizing the received pilot or signal, and the signal is recovered separately similar to the legacy systems. In this paper, the developed DCCN employs a 4-layer MLP with estimator configuration between convolutional layers for channel estimation.

CNN is used in signal classification [4] and recovery [5], channel estimation [16]–[18], [20], [28], [44], [45] and detection [16], [24]. Although CNNs may not outperform MLP [21], it is more efficient and scalable. For example, an OFDM receiver in [16] matches the linear minimum mean square error (LMMSE) estimator with linear complexity based on a depth-wise separable convolution cascaded in a residual architecture. CNNs are also configured as pass-through [16], [24] and estimator [44] in the signal path. Other types of DNNs, such as generalized regression neural network (GRNN) [12] and long short-term memory (LSTM) [14], [17] are also used in channel estimation, and model-based DNN that unfolds an iterative algorithm [23] is used for detection.

CVNNs are discussed in [34], [35] but they have not been supported in popular platforms [36], [37]. [35, p. 1] indicates that CVNNs “have been marginalized due to the absence of the building blocks”. In wireless PHY, most CNNs are real-valued [4], [16], [18], [20], [24], [28], [45], except that a single layer of complex-valued convolution was used in [5] to achieve amplitude-phase representation, which would require additional phase (un)wrapping in many applications. Based on the analysis in Section IV-A, the expressive nature of a CVNN can be preserved by an MLP as well as a CNN [16], [28] with proper input format and dimensions, e.g., reception field and number of filters. However, the lack of discussions on such settings makes it difficult to scale up the successful design of CNNs [16], [28] with respect to the system parameters including antenna number and DFT size, or transfer these experiences for future work. In this work, guidelines

for implementing complex-valued convolutional layers are provided and CVNN is used in the entire IQ domain.

B. OFDM System and Enhanced Channel Estimation

OFDM is the most popular system in modern wireless networks. In [46], [47], the OFDM physical layer and various channel estimation approaches are introduced. At the transmitter side, several enhanced waveforms are introduced to the OFDM family, such as Filter Bank MultiCarrier (FBMC), UFMC, GFDM for 5G and next generation communication systems [48]. These modified OFDM waveforms generally have better characteristics with regards to various interferences. A constellation enhancement approach [49] and deep learning (DL)-based coding system [13] are proposed to reduce the Peak to Average Power Ratio (PAPR) of the OFDM waveform. An MLP-based channel estimator for FBMC shows good performance in high mobility channels [50].

Most of the improvements take place at the receiver side. As a partial copy of the OFDM waveform, cyclic prefix (CP) is introduced for synchronization and ISI mitigation at the receiver, as shown in Fig. 2(b). CP has been exploited to enhance blind [39]–[42] and pilot-aided [38] channel estimation by improving SNR, frequency selectivity, and interference mitigation [41]. In [38], the decoded data of a previous OFDM symbol is used to recover the CP of the next symbol. However, the approach in [38] can only enhance LS channel estimates. Our model accomplishes this task by processing multiple OFDM symbols simultaneously with better overall performance and slightly improved efficiency without an explicit algorithm, which serves as a complementary to analytical approaches, such as maximum likelihood [39], [41] and factor graph [40], [42] in exploiting CP.

DNN is also used to enhance the OFDM receivers. In [11], a 5-layer MLP is used for channel estimation and symbol detection. Our work confirms the conclusions in [11] that MLP can match MMSE in high SNR and handle adversities related to CP, clipping noise, and channel mismatch better than MMSE. Different from [11], our channel estimator is linearly activated instead of using ReLU, thus can be interpreted as a low rank approximation of LMMSE. The MLP in [11] handles OFDM waveform without CP, while our model uses CP to simultaneously improve the SNR and mitigate ISI. Moreover, an independent MLP needs to be trained for every 16 bits in [11], while we only use one model to process an entire coherence slot, leading to better scalability. A number of studies use CNN for channel estimation in OFDM systems [18], [20], [44], [45], without being able to outperform the ideal LMMSE. However, for massive MIMO, a low-complexity channel estimator that can outperform approximate LMMSE [18] is still attractive. The OFDM receiver in [16] achieves linear complexity $\mathcal{O}(N)$, but it only matches the ideal LMMSE on trained channel models and underperforms on unseen channel models. Compared to these studies, our DCCN outperforms the ideal LMMSE with a complexity of only $\mathcal{O}(N^2)$ even with unseen channel models. In [19], a MLP-based channel estimator for non-OFDM system outperforms approximate LMMSE in doubly selective channels and exhibits better robustness to mobility, of which these

findings are confirmed in our work, despite the results are not directly comparable. In [14], long short-term memory (LSTM) is used to estimate channels by predicting future data and is reported to outperform LMMSE in high-speed scenarios. In comparison, our approach mitigates ISI by simultaneously processing multiple OFDM symbols. Deep learning has been shown being able to improve MIMO detection [23]–[25] and channel coding [17], while only single antenna is considered in this paper. Moreover, the aforementioned work [8], [10], [11], [13]–[16], [23] all rely on explicit DFT/IDFT, which is replaced by a learned linear transform in our approach for increased SNR. Our work first demonstrates the capability of DNNs in processing OFDM waveform, and our approaches of setting hyperparameters of NN layers based on OFDM frame structure and leveraging domain knowledge offer a transferable design template for other waveform structures.

III. OFDM COMMUNICATION SYSTEM

We first introduce the relevant concepts and notations in the lower physical layer of a legacy OFDM system, followed by channel estimation approaches in legacy receivers.

A. Physical Layer

The block diagram of the PHY of an OFDM system is illustrated in Fig. 2(a). At the transmitter, channel encoding is firstly applied to input bits $\mathbf{b} \in \{\pm 1\}$ for error detection and/or correction. The encoded bits are then converted to complex-valued in-phase and quadrature (IQ) data by mapping to a constellation on the IQ plane. A frequency-domain OFDM symbol, $\mathbf{X}$, is created by inserting training signals (pilots) and guard bands into the IQ data, and then $\mathbf{X}$ is transformed to a time-domain OFDM symbol, $\mathbf{x}$, via an N -point IDFT and a subsequent parallel to serial (P/S) conversion. Next, CP, a section of $\mathbf{x}$ at its end, is prepended to $\mathbf{x}$ to create a time-domain full OFDM symbol, $\mathbf{x}_{cp}$, as illustrated in Fig. 2(b). The baseband signal, $\mathbf{x}_{cp}$, is then up-converted to the radio frequency (RF) and transmitted over-the-air by the RF frontend. The radio signal propagates over the wireless channel and is picked and down-converted to baseband IQ samples by the receiver frontend. At the receiver, the received time-domain OFDM symbols, $\mathbf{y}_{cp}$, are recovered by a carrier synchronizer. Then, the CP is removed from $\mathbf{y}_{cp}$ and the rest of the IQ samples, $\mathbf{y}$, are transformed to the frequency-domain OFDM symbol, $\mathbf{Y}$, via DFT. Based on $\mathbf{Y}$, a channel equalizer outputs the estimated transmit frequency-domain IQ data $\hat{\mathbf{X}}$, which is then demodulated to soft bits (log-likelihood) $\hat{\mathbf{b}}$ and converted to binary output bits $\hat{\mathbf{b}}$ by a channel decoder. Finally, $\hat{\mathbf{b}}$ is passed to the next layer. Note that the focus of this paper is lower PHY, and channel coding is out of the scope. We refer the frequency and time domains in Fig. 2(a) as the *IQ domain* in which a signal is represented by complex-valued samples.

OFDM systems typically have a frame structure where a coherence slot (or “slot” for short) is composed of multiple OFDM symbols, as shown in Fig. 2(b). The notations related to the OFDM coherence slot are as follows: an OFDM symbol contains N subcarriers, where N is the size of DFT/IDFT.

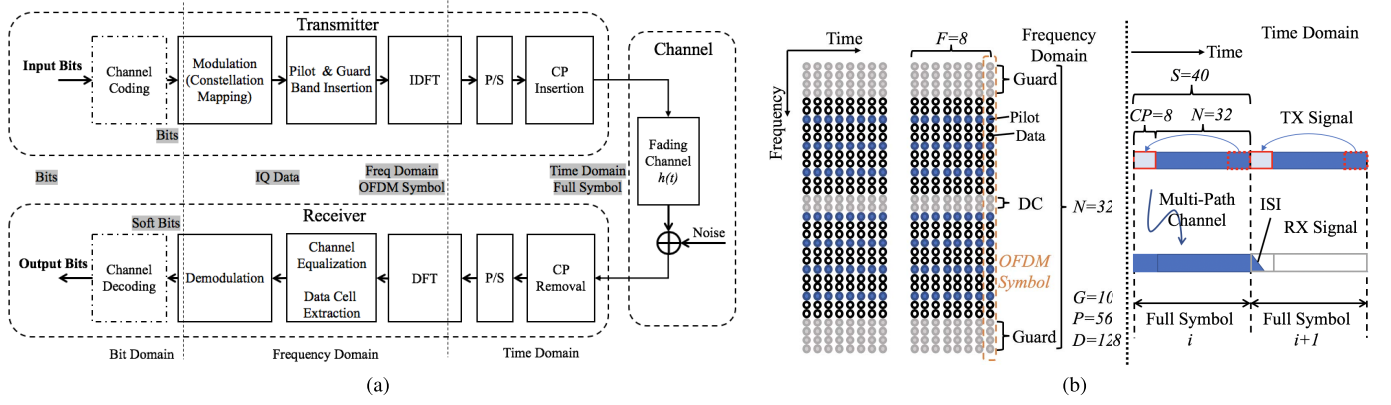


Fig. 2. Physical layer of OFDM system: (a) Diagram of legacy OFDM PHY [46], (b) Exemplary OFDM coherence slot and time-domain waveform [41].

Among the N subcarriers, a total of G nullified guard subcarriers are placed at the center (DC guard band) and the edge (edge guard band). A subcarrier in an OFDM symbol is referred to as a resource element (RE). A coherence slot contains F consecutive OFDM symbols, in which P and D REs are allocated to pilot and data, respectively. The length of a time-domain full OFDM symbol is $S = N + N_{cp}$ where N_{cp} is the length of CP. Under m -ary modulation, an IQ sample carries m bits, and the size of the constellation is 2^m .

B. Wireless Channel

A well accepted simplification of the wireless channel model describes the fading and noise processes as [46]:

$$\mathbf{y} = \mathbf{x} * \mathbf{h} + \mathbf{n}_o, \quad \mathbf{Y} = \mathbf{X} \odot \mathbf{H} + \mathbf{N}_o, \quad (1)$$

where $\mathbf{x}, \mathbf{y}, \mathbf{n}_o \in \mathbb{C}^{S \times 1}$ are the time-domain transmitted and received signals and white noise, respectively, $\mathbf{h} \in \mathbb{C}^{L \times 1}$ is the channel impulse response, $\mathbf{X}, \mathbf{Y}, \mathbf{H}, \mathbf{N}_o \in \mathbb{C}^{N \times F}$ are the frequency domain transformations of $\mathbf{x}, \mathbf{y}, \mathbf{h}$, and $\mathbf{n}_o$, e.g., $\mathbf{X} = DFT_N(\mathbf{x})$, and $*$ and $\odot$ are operators of convolution and element-wise product, respectively. Without loss of generality, (1) can also refer to an OFDM symbol, i.e., $\mathbf{x} \in \mathbb{C}^{N \times 1}$, $\mathbf{x}_{cp} \in \mathbb{C}^{S \times 1}$ and $\mathbf{X} \in \mathbb{C}^{N \times 1}$.

The multipath fading can be modeled as a linear finite impulse-response (FIR) filter [51]:

$$y_t = \sum_{l=0}^{L-1} x_{t-l} h_l, \quad \text{and} \quad h_l = \sum_{k=1}^K \sqrt{\Omega_k} z_k \text{sinc}\left(\frac{\tau_k}{T_s} - l\right), \quad (2)$$

where z_k is a complex-valued random variable, vectors Ω and τ represent the power-delay profile (PDP) of the fading process, and T_s is the sampling period of the discrete signal. The filter length L is chosen so that $|h_l|$ is small when $l < 0$ or $l \geq L$. For Rayleigh fading, the real and imaginary parts of z_k are i.i.d. Gaussian random variables, thus $|z_k|^2$ follows a Rayleigh distribution. K is the number of paths in a multipath fading channel. In a flat fading channel $K = 1$, the channel coefficients on all the subcarriers of an OFDM symbol are identical. In a multipath fading channel, $K > 1$, the channel coefficient varies by subcarrier so that the channel exhibits frequency selectivity. The effects of different fading processes on frequency-domain IQ samples are illustrated in Fig. 3.

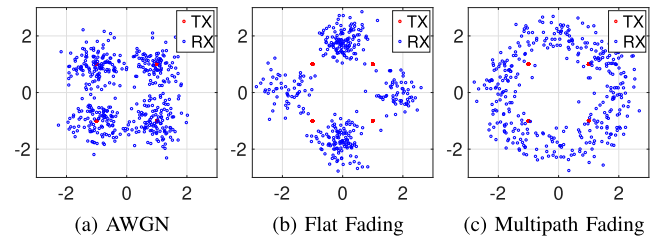


Fig. 3. The effects of different wireless channels on QPSK modulation, SNR = 6 dB, on IQ plane (x: In-Phase, y: Quadrature).

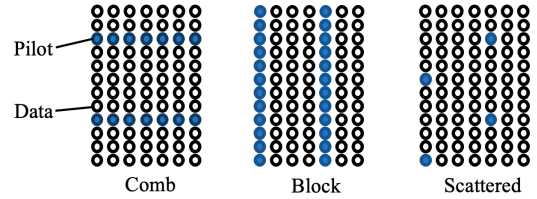


Fig. 4. Typical OFDM pilot patterns: comb, block, and scattered.

Note that only noise and fading are considered, while channel impairments for channel coding are left for future work.

The channel impulse response, $\mathbf{h}$, is time-variant, and its coherence time, T_c , is inversely proportional to the maximum Doppler frequency F_d , i.e., $T_c \approx 1/F_d$. An OFDM system is usually configured according to its applications, measured coherence time, T_c , coherence bandwidth, B_c , and total bandwidth, so that slow fading holds for a coherence slot.

C. Channel Estimation and Equalization

Consider a communication system with a pilot-aided channel estimation at the receiver. Different pilot patterns, such as block, comb, and scattered pilots, as illustrated in Fig. 4 [46], are designed to sample the channel distortion. Pilot signals are of either constant signal or low auto-correlation sequence (e.g., Zadoff-Chu sequence) known at the receiver. The basic pilot-aided channel equalizer in OFDM system is based on the least square (LS) estimator [46], [47]:

$$\hat{\mathbf{X}} = \frac{\mathbf{Y}}{\hat{\mathbf{H}}_{LS}}, \quad \text{where} \quad \hat{\mathbf{H}}_{LS} = \mathcal{F}\left(\frac{\mathbf{Y}_P}{\mathbf{X}_P}\right), \quad (3)$$

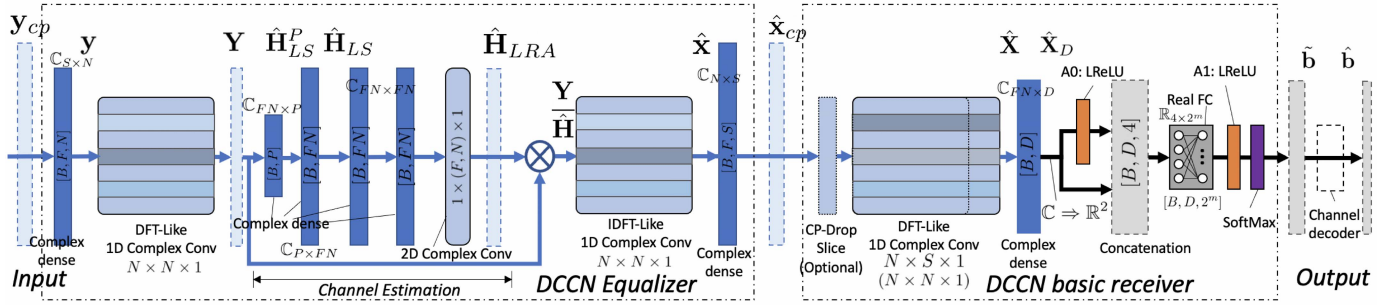


Fig. 5. DCCN OFDM receiver flow graph: a DCCN equalizer converts time domain received signal, $\mathbf{y}_{cp}$, to estimated transmitted signal, $\mathbf{x}_{cp}$, which is subsequently converted to estimated soft bits, $\tilde{\mathbf{b}}$, and hard bits, $\hat{\mathbf{b}}$, by a DCCN basic receiver. The blue color represents complex-valued domain, and gray for real-valued domain. The dimensions and data types of a dense layer are shown as e.g., $\mathbb{C}_{S \times N}$, and the shape of its output tensor as, e.g., $[B, F, N]$, the dimensions of a Conv layer is labeled beside the block, e.g., $N \times N \times 1$ represents N 1-D filters each of length N . A dashed block represents a data tensor.

where $\hat{\mathbf{X}}$ is the estimated signal, $\hat{\mathbf{H}}_{LS}$ contains the LS channel estimates, $\mathbf{X}_P$ and $\mathbf{Y}_P$ are transmitted and received pilots, respectively, and $\mathcal{F}(\cdot)$ is an interpolation operator, e.g., linear, spline, low-pass-filter, and DFT [46], [47]. The LS estimator is agnostic to channel statistics, while other estimators, such as LMMSE, maximal likelihood, and parametric channel modeling-based (PCMB) estimator, are based on LS estimation and/or prior channel knowledge [46], [47].

The ideal LMMSE estimator is expressed as [47], [52]:

$$\hat{\mathbf{H}}_{LMMSE} = \mathbf{R}_{HH} \left(\mathbf{R}_{HH} + \frac{\beta}{\alpha} \mathbf{I} \right)^{-1} \hat{\mathbf{H}}_{LS}, \quad (4)$$

where $\mathbf{R}_{HH} = \mathbb{E}\{\mathbf{H}_{*,i} \mathbf{H}_{*,i}^H\}$ is the frequency-domain covariance matrix of channel realization $\mathbf{H}$, α is the linear-domain SNR, and β is a constant defined based on a specific modulation scheme. The ideal LMMSE estimator in (4) requires prior channel knowledge $\mathbf{R}_{HH}$ and α , which are unavailable in practice. Moreover, the matrix inversion in (4) leads to a computational complexity of $\mathcal{O}(N^3)$. Low-rank approximation (LRA) of LMMSE [53] approximates the ideal LMMSE matrix $\mathbf{R}_{HH}(\mathbf{R}_{HH} + \frac{\beta}{\alpha} \mathbf{I})^{-1}$ via prescribed α and singular value decomposition (SVD) of $\mathbf{R}_{HH}$, which can tolerate PDP mismatch and achieve a lower complexity of $\mathcal{O}(N^2)$ at the cost of an irreducible error floor. Improvements on approximate LMMSE (ALMMSE) include using different PDPs [54], SNR estimation [52], [55], rank estimation and exploiting the circulant property of LMMSE matrix [52]. Fast ALMMSE can further reduce the complexity to $\mathcal{O}(N)$ with a pre-computed LMMSE matrix at the cost of degraded performance in a certain SNR range [56].

IV. DCCN-BASED OFDM RECEIVER

The DCCN-based OFDM receiver is denoted by a function $\hat{\mathbf{b}} = \mathcal{D}_S(\mathbf{y}_{cp}; \Theta)$, which is defined on a collection of hyperparameters, $\mathcal{S}$, for the configurations of an OFDM frame and DCCN flow-graph (Fig. 5), where $\hat{\mathbf{b}}$ is the log-likelihood of the transmitted bits, $\mathbf{y}_{cp}$ is the synchronized time-domain received OFDM symbols of a coherence slot including CP, and Θ is the collection of trainable parameters of DCCN. The DCCN-based OFDM receiver comprises a channel equalizer followed by a basic receiver. The hidden layers of DCCN are named after the signal processing modules in a legacy OFDM

receiver, while they might function differently. The forward network generalizes the signal processing in a legacy receiver and ensures that the search space of Θ contains at least an ALMMSE receiver by adding computational redundancy, which can be minimized during training by introducing a regularization loss. Structural redundancy is included in the flow-graph for research, and a simplified flow-graph is proposed in Section V-E for deployment.

A. Guidelines for Complex-Valued Layers

Omitting the bias and activation, a complex-valued neuron can be expressed as [34]:

$$\begin{bmatrix} \text{Re}(x^{OUT}) \\ \text{Im}(x^{OUT}) \end{bmatrix} = \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \begin{bmatrix} \text{Re}(x^{IN}) \\ \text{Im}(x^{IN}) \end{bmatrix}, \quad (5)$$

where $x^{IN}, x^{OUT} \in \mathbb{C}$ and weights $a, b \in \mathbb{R}$ are the real and imaginary parts of a complex weight, respectively. (5) can be approximated by a dense layer of $\mathbb{R}_{2 \times 2}$ with approx-D input and trained weights $\mathbf{w} \approx \begin{pmatrix} a & -b \\ b & a \end{pmatrix}$. Generalizing (5) to higher dimensions, we have the following principles:

Remark 1: A dense layer of $\mathbb{C}_{Z_i \times Z_o}$ can always be approximated by a dense layer of $\mathbb{R}_{2Z_i \times 2Z_o}$ with full expressive power.

Remark 2: A complex-valued convolutional (C-Conv) layer of size $f_n \times f_s \times f_c$ (stands for f_n filters of shape, f_s , and depth of f_c) can be approximated by a real-valued Conv layer of size $2f_n \times (f_s, 2) \times f_c$ for approx-D input or $2f_n \times f_s \times 2f_c$ for approx-C input without following (5), and can be exactly implemented by a real-valued Conv layer of size $2f_n \times f_s \times f_c$ with approx-D input by following (5).

By following (5), the exact implementation of CVNN layers reduces unnecessary degree of freedom and only needs half of the trainable real-valued weights that is required by the approximations. More specifically, three exemplary implementations of a 1D C-Conv layer of size $1 \times 1 \times 1$ are given in Fig. 6(a) to illustrate Remark 2. All three implementations meet the expressive requirement with the same computational complexity (four multiplications and two additions per complex sample). Each of the first two approximations requires four weights while the third exact implementation only needs two weights. The exact implementation has better spatial

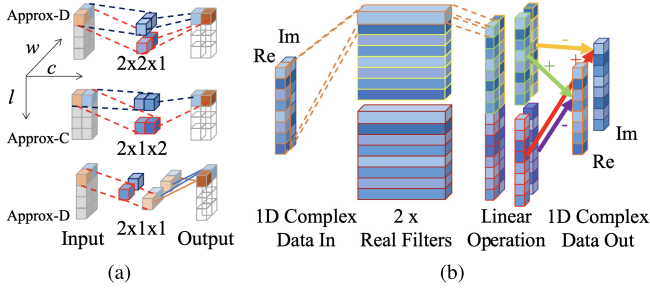


Fig. 6. (a) Three implementations of 1D complex-valued Conv (C-Conv) layer ($1 \times 1 \times 1$) that preserve its expressive power, (b) Implementation of a 1D C-Conv layer ($8 \times 8 \times 1$) based on a 2D real-valued Conv layer ($16 \times (1,8) \times 1$).

complexity and training efficiency with a reduced search space for the optimizer. For practitioners, an approximated C-Conv layer will be less expressive than the exact one if it fails to meet the required input format or the minimal size, e.g., the shape and number of filters. For complex-valued activation principles, we refer readers to [34], [35].

In Fig. 6(b), the implementation of our DFT-Like C-Conv layer is illustrated by an example of 1D C-Conv layer of size $8 \times 8 \times 1$ implemented by a 2D real-valued Conv layer of size $16 \times (1,8) \times 1$. In this work, real and imaginary parts of a complex tensor are in the last dimension.

B. Basic DCCN Receiver

The basic DCCN receiver is an OFDM receiver without a channel equalizer, as illustrated on the right half of Fig. 5. The forward network of the basic receiver begins with an optional slicing for dropping the CP, followed by a C-Conv layer of size $N \times S \times 1$ (or $N \times N \times 1$ if CP is dropped), which is designed to transform time-domain OFDM symbols, $\hat{\mathbf{x}}_{cp}$, to frequency domain $\hat{\mathbf{X}}$ and exploit CP for SNR gains. Next, a complex dense layer $\mathbb{C}_{FN \times D}$ is designed to extract all the data REs $\hat{\mathbf{X}}_D$ from a coherence slot. The rest of the forward network is essentially a classifier that converts IQ samples to soft bits, where an input IQ sample is treated as a vector of two real numbers, $\mathbb{C} \Rightarrow \mathbb{R}^2$. The extracted IQ vector and its non-linear (Leaky ReLU) activation, $A0$, are concatenated to a tensor of shape $[B, D, 4]$ and fed to a small dense layer of $\mathbb{R}_{4 \times 2^m}$ followed by another Leaky ReLU activation, $A1$, of which the output tensor is reshaped to $[B, D, m, 2]$ and then activated by a softmax function along its last dimension to produce a soft bit—a vector of likelihoods of ± 1 . B is the number of slots in a batch of input signal. Since the channel coding is out of our scope, the output bits are obtained by hard decisions on soft bits. In Section V-E, variations of the forward network are tested.

C. DCCN Channel Equalizer

The DCCN equalizer, with the input of $\mathbf{y}_{cp}$ and the output of $\hat{\mathbf{x}}_{cp}$, is prepended to the basic DCCN receiver, as illustrated on the left half of Fig. 5. The forward network of the DCCN equalizer contains four submodules: The first submodule comprises a dense layer of $\mathbb{C}_{S \times N}$ (or $\mathbb{C}_{N \times N}$ if CP is dropped),

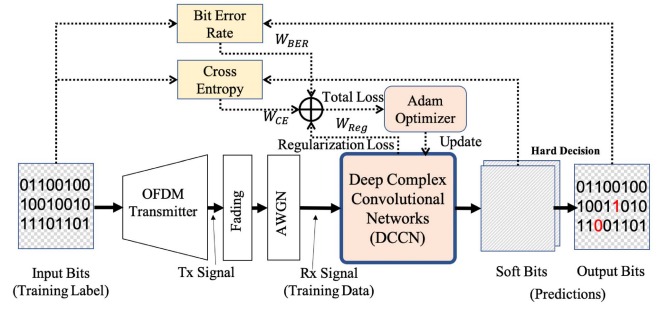


Fig. 7. Block diagram of the DCCN training System.

followed by a C-Conv layer of $N \times N \times 1$, and it converts $\mathbf{y}_{cp}$ to the frequency domain $\mathbf{Y}$. The second submodule estimates the channel frequency response $\hat{\mathbf{H}}$ with four dense layers followed by a 2D complex filter. The third submodule performs equalization with an element-wise complex division $\hat{\mathbf{X}} = \mathbf{Y}/\hat{\mathbf{H}}$. Finally, $\hat{\mathbf{X}}$ is converted to $\hat{\mathbf{x}}$ by an IDFT-Like 1D C-Conv layer of $N \times N \times 1$, and $\hat{\mathbf{x}}_{cp}$ is recovered by adding back CP with a dense layer of $\mathbb{C}_{N \times S}$.

In the channel estimation submodule, the first dense layer of $\mathbb{C}_{FN \times P}$ is designed to locate pilots and estimate channel coefficients on pilots $\hat{\mathbf{H}}_{LS}^P$. Then, $\hat{\mathbf{H}}$ is obtained by an interpolation of $\hat{\mathbf{H}}_{LS}^P$ to the entire coherence slot and channel estimation in the next three dense layers and a 2D filter of size (F, N) , which resembles the LRA-LMMSE [53]:

$$\hat{\mathbf{H}}_{LRA} = \mathbf{U} \mathbf{D}_p \mathbf{U}^H \hat{\mathbf{H}}_{LS}, \quad (6)$$

where $\mathbf{D}_p$ is a diagonal matrix with entries $\delta_k = \frac{\lambda_k}{\lambda_k + \beta/\alpha}$ for $k \in [1, p]$, and $\delta_k = 0$ for $k \in [p+1, N]$, $\mathbf{U}$ is a unitary matrix containing the singular vectors of $\mathbf{R}_{HH}$. Instead of setting λ , p , α , and $\mathbf{U}$ explicitly [52], [54], [55], the LMMSE matrices in (6) are trainable parameters of multiple dense layers to be learned from data. Note that the dimension of these dense layers are of a coherence slot for ISI mitigation. In Section V-C, our DCCN receiver also exhibits an error floor like ALMMSE with a prescribed SNR [52], [53], [55]. Other numbers of the dense layers of $\mathbb{C}_{FN \times FN}$ are tested in Section V-E.

Considering the number of OFDM symbols per coherence slot F as a constant in a protocol, the asymptotic computational complexity of the DCCN receiver is $\mathcal{O}(N^2)$ (or $\mathcal{O}(NS)$ for DCCN with CP), since DCCN is composed of cascading layers without any loops. The 1D complex convolutional layer and the fully connected layer have a maximum complexity of $\mathcal{O}(N^2)$ (or $\mathcal{O}(NS)$ for with CP). If F is considered as a variable, the complexity becomes $\mathcal{O}(N^2 F^2)$.

D. 2-Stage Training

The training setup of the DCCN receiver is illustrated in Fig. 7. An online random generator creates a random binary stream $\mathbf{b}$, which is translated by an OFDM transmitter into time domain OFDM symbols as the transmit signal, $\mathbf{x}_{cp}$. The received signal $\mathbf{y}_{cp}$ is created by a channel model that adds fading and noise to $\mathbf{x}$. $\mathbf{y}_{cp}$ and $\mathbf{b}$ are the training data and labels, respectively. The outputs of DCCN model are the soft

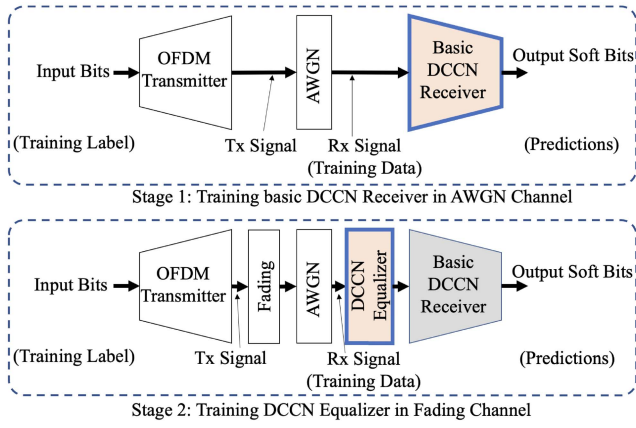


Fig. 8. Training the basic receiver and the equalizer in two stages.

bits, $\tilde{\mathbf{b}}$, and the output bits, $\hat{\mathbf{b}} \in \{\pm 1\}$, which are generated by hard decision on $\tilde{\mathbf{b}}$. The loss function $\mathcal{L}(\mathbf{b}, \tilde{\mathbf{b}}, \hat{\mathbf{b}}, \Theta)$ is the weighted sum of the cross entropy (CE) loss and regularization loss $\mathcal{L}_{reg}(\Theta)$:

$$\mathcal{L}(\mathbf{b}, \tilde{\mathbf{b}}, \hat{\mathbf{b}}, \Theta) = \mathcal{L}_{CE}(\mathbf{b}, \tilde{\mathbf{b}}) + \varepsilon \mathcal{L}_{reg}(\Theta), \quad (7)$$

where $\varepsilon \ll 1$ is a small constant. The cross entropy loss, $\mathcal{L}_{CE}(\mathbf{b}, \tilde{\mathbf{b}})$, is the average cross entropy of the training labels $\mathbf{b}$, and the soft bits $\tilde{\mathbf{b}}$. During training, Θ of DCCN receiver is randomly initialized and updated by an Adam optimizer that is running back-propagation based on the loss function in (7).

It is difficult and may be impractical to train the DCCN receiver directly in multipath fading channels due to the severe distortion shown in Fig. 3(c). Therefore, a transfer learning scheme, as illustrated in Fig. 8, is developed to train the basic receiver and the equalizer in two stages. In stage 1, the basic receiver is trained in an AWGN channel only. In stage 2, the flow-graph of the DCCN equalizer is first prepended to the pre-trained basic receiver in a TensorFlow session dedicated for graph-editing. Then, the equalized DCCN receiver is loaded and trained in another session, in which the trainable parameters of the basic receiver is frozen and the channel fading is included to generate the training data. The loss function is the same throughout stages 1 and 2. The graph-editing technique enables back-propagation when the second half of the forward network is frozen. Note that the two-stage training approach can increase the data efficiency by reusing the same pre-trained basic receiver in stage 2 for different fading settings. Similarly, the trained model from stage 2 can be fine-tuned for different realistic channels.

To improve training efficiency, several techniques are employed. The training data is fed to the model in mini batches, which leverages the high throughput of the parallel processing in the graphics processing units (GPUs) and minimizes the latency in memory copying. In the programming of NumPy-based OFDM transmitter and fading modules, data processing is vectorized and large loops are avoided. The learning rate is decayed exponentially for fine-tuning as the training proceeds. The random training labels and corresponding training data are generated online rather than feeding a pre-generated training dataset repeatedly. Therefore, we use an *iteration* instead of an epoch (a full training pass over

the entire dataset) to describe the outermost loop in training. An early stop mechanism is employed on top of a maximum number of training iterations to end the training if the key performance metric (i.e., BER) has not improved after a fixed number of iterations.

E. Channel Settings in Training

A clear guidance does not exist for setting the training SNR for a DL-based PHY. We use different SNR configurations in the two stages. In stage 1, the SNR does not influence the optimal basic receiver but the effectiveness of training. The noise in the channel creates bit errors that drive the gradient descent and acts as a regularizer to prevent overfitting. A higher SNR requires a larger size of the mini-batch to generate the same amount of bit errors, which could lower the data efficiency and prolong the convergence of training. On the other hand, low SNRs may hide small demodulation bias in relatively large BERs, i.e., consistently more 1s than -1s in the output bits. To this end, we recommend a training SNR value of $E_b/N_o = 5$ dB [21] for the training stage 1, specifically, the SNR $\eta_i = 5m$ dB is selected for m -ary modulation.

In the training stage 2, the optimal DCCN equalizer depends on the channel statistics such as SNR and PDP. For better generalizability of the trained model, a mixture of SNRs and fading models is employed. The SNR of each OFDM coherence slot is randomly selected from the working SNR regime based on a probability function that prefers high SNR values without excluding middle and low SNR values. For example, $\eta_i \in \{0, 3, \dots, 30\}$ with $P(\eta_i \geq 17) = 90\%$. The rationale is that the DCCN equalizer resembles a fast ALMMSE algorithm [52], which has preferable performance when designed for high SNR (i.e., 20 dB) other than low SNR (i.e., 5 dB). Meanwhile, since LMMSE is robust to PDP mismatch, the optimal points of the DCCN equalizer for different PDPs would be located closely. A mixture of Rayleigh fading models improves not only the generalizability but also the convergence time. The fading models with shorter delay spread can smooth the overall loss landscape to help the optimizer overcome the local minima associated with those with richer multipath. In Sections V-D and V-E, models trained with different channel settings in both stages are compared.

V. EVALUATION RESULTS

A. Methodology

The DCCN receiver is compared to legacy receivers with different channel estimators [43] in numerical evaluations. We first present the results of the DCCN basic receiver for m -ary Quadrature Amplitude Modulation (QAM) modulation ($m \leq 4$) in AWGN channels, then the equalized DCCN receiver in Rayleigh fading channels with different settings of PDP, ISI leakage, and mobility. We use the notations DCCN and DCCN-CP to refer to DCCN receivers without and with CP exploitation, respectively.

The evaluated OFDM system emulates a simplified LTE downlink frame structure [57] as detailed in Table I, in which the DFT size is 64, the total number of guard subcarriers (SCs) at the edge and direct current (DC) is 16, the sampling rate

TABLE I
CONFIGURATIONS OF THE EVALUATED OFDM SYSTEM

FFT Size	$N = 64$
Sample rate (Mps)	0.96
Guard SCs per symbol	14 (Outer) + 2 (DC)
Pilot REs per slot	$P = 16$
Data REs per slot	$D = 320$
Subcarrier bandwidth	15 KHz
Symbols per slot	$F = 7$
CP length	long: $0.25N$, short: $0.07N$
PAPR limit	9 dB
Pilot	LTE downlink frame pilot pattern [57] with constant value $\sqrt{1/2}(1 + i)$
Modulation	QAM with orders of 2, 4, 8, 16, Gray code
Rayleigh fading model	Flat, EPA, EVA, ETU [58]
Channel coherence	$T_c > 1ms$, $B_c \geq 200KHz$ [58]
Max Doppler spread	$F_d = 0$ Hz by default, or if specified EVA 70 Hz, ETU 70 Hz, 300 Hz [58]

TABLE II
TRAINING CONFIGURATIONS FOR m -ARY MODULATION

Setting	Basic Receiver	DCCN Equalizer
Maximum iterations	$1200m$	$4000m$
Early stop window	200 iterations	200 iterations
Initial learning rate	0.001	0.001
Learning rate decay	Exponential, rate 2%, step 500 (mini-batches)	
SNR (dB)	5m	Customized random variable
Iteration	200 mini-batches	
Mini-batch size	$72 \times 320 \times m$ bits	
Testing bits per SNR	$20000 \times 320 \times m$ bits	
Testing SNR	-10 to 30 dB	
Optimizer	SGD with Adam	

is 0.96 MHz, and the bandwidth of a subcarrier is 15 KHz. A coherence slot contains 7 OFDM symbols, in which the numbers of REs assigned to pilot and data are $P = 16$ and $D = 320$, respectively. A scattered pilot pattern, which is consistent with LTE protocol as illustrated in the rightmost of Fig. 4, is used, and the pilot signal has a constant value of $\sqrt{1/2}(1 + i)$. QAM with orders of 2, 4, 8, 16 and Gray code are used for constellation mapping, where the maximum amplitude of the constellation is 1. The peak to average power ratio (PAPR) of the OFDM waveform after the transmitter is limited to 9 dB.

The training configurations for the basic receiver and equalizer with m -ary modulation are listed in Table II. A stochastic gradient descent-based Adam optimizer with a mini-batch size of 72 coherence slots are used. In each iteration, a new random bit stream (training labels) is generated and converted to the training data by a legacy OFDM transmitter and a channel model. Since training labels are random, the loss function for early stop mechanism is based on the training data rather than a separate test data set. Training terminates either by reaching the maximum number of iterations, or triggering an early stop which mostly happens in practice. The learning rate is set to 0.001 initially and decayed by 2% every 500 mini-batches (steps) or 2.5 outer iterations. The SNR is set according to the description in Section IV-D. Each outer training iteration contains 200 mini-batches, and the testing data is of 20,000 coherence slots for each SNR point from -10 to 30 dB. The Rayleigh fading for training stage 2 is set as alternating

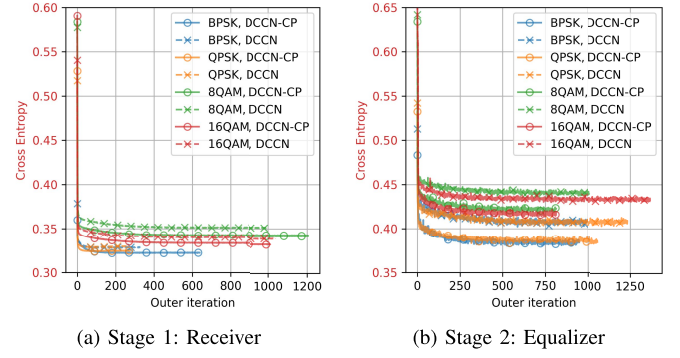


Fig. 9. Evolution of cross entropy loss in the training of DCCN models.

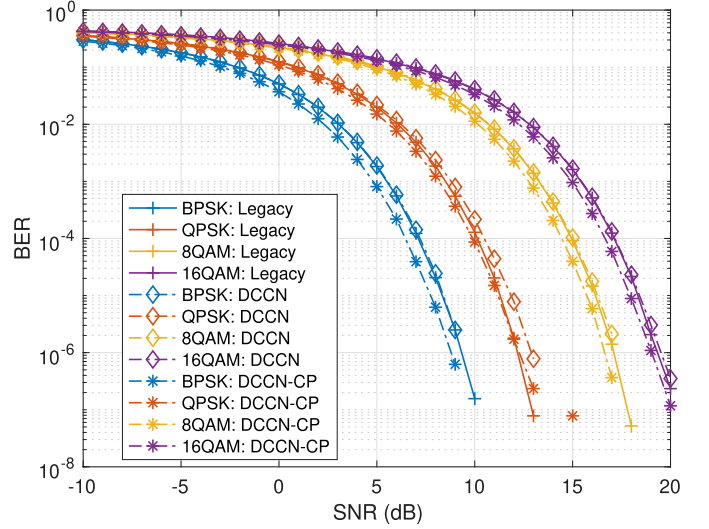


Fig. 10. BER of legacy and DCCN OFDM receivers vs SNR, in AWGN channel, with long CP.

models of flat fading, EPA, EVA, and ETU in consecutive slots.

Training a DCCN model lasts for 250–1300 iterations, as shown in Figs. 9. The training process starts with a quick fitting followed by a long fine-tuning phase: the cross entropy first decreases drastically in the first 10–50 iterations, then decreases slowly but steadily until hitting a floor. The source code is available at [43].

B. Additive White Gaussian Noise Channel

The BERs of our basic receivers and the legacy OFDM receiver [43] with modulations of BPSK, QPSK, 8QAM and 16QAM in AWGN channels with SNR from -10 to 20 dB are presented in Fig. 10, where long CP is considered for DCCN-CP. Without the help of CP, we do not expect DCCN to outperform the legacy demodulation since the latter is optimal in an AWGN channel. For BPSK, 8QAM and 16QAM, DCCN performs closely to the legacy receiver with a negligible gap (≤ 0.16 dB) when BER is below 10^{-5} . However, for QPSK, DCCN begins to underperform the legacy receiver at the BER level of 10^{-2} and the gap is increased to 0.7 dB at BER of 10^{-6} . On the other hand, the DCCN-CP consistently outperforms DCCN by a gap that decreases from 0.7 dB for BPSK to 0.5 dB for 16QAM. For QPSK, DCCN-CP starts

to underperform legacy receiver at BER of 10^{-6} . Note that when BER is very small, e.g., $\leq 10^{-5}$, the relative error due to limited data size in simulation increases. The CP carries information on all subcarriers and experiences independent random noise, thus can theoretically improve the power of data signal by $N_{cp}D/(FN^2)$, which corresponds to an increase of 0.75 dB on SNR in AWGN channel for a CP length of $0.25N$. DCCN-CP brings an improvement of SNR to DCCN very close to 0.75 dB for BPSK. It shows that using C-Conv layer to transform time-domain OFDM symbol to frequency-domain has the advantage of exploiting redundancy carried by CP, which cannot be accomplished by explicit DFT.

C. Rayleigh Fading Channels

Next, the equalized DCCN receivers are evaluated in Rayleigh fading channels. The benchmarks are legacy OFDM receivers with different channel estimators: LS estimator (LS-Spline), LS estimator enhanced with CP based on analytical approach in [38] (LS-CP), the ideal LMMSE, and an ALMMSE. The channel covariance matrix is updated per OFDM symbol for the ideal LMMSE as $\mathbf{R}_{\mathbf{H}\mathbf{H}} = \mathbf{H}_{*,i} \mathbf{H}_{*,i}^H$ based on true channel realization $\mathbf{H}_{*,i}$, and per slot for ALMMSE as $\hat{\mathbf{R}}_{\mathbf{H}\mathbf{H}} = \frac{1}{F} \hat{\mathbf{H}}_{LS} \hat{\mathbf{H}}_{LS}^H$. The $\hat{\mathbf{H}}_{LS}$ in the ideal LMMSE and ALMMSE is from LS-Spline. The tested ALMMSE represents the upper bound of ALMMSE of complexity $\mathcal{O}(N^2)$ [52] with full rank approximation and perfect SNR estimates. The ISI is set to be only between consecutive OFDM symbols in the same coherence slot. For simplicity, BPSK modulation and perfect synchronization are considered at the receiver. An independent channel realization is generated per coherence slot and is invariant within the slot by default ($F_d = 0$ Hz) [11]. If $F_d > 0$ Hz, the time-varying channel is generated by the Jake's model and the fading technique of sum of 48 sinusoids [59]. The flat fading and 3GPP multipath fading models [58] are evaluated. With the DFT size of 64, the channel filter lengths, L , for flat fading, extended pedestrian A model (EPA), extended vehicular A model (EVA), and extended typical urban model (ETU) are 1, 9, 11 and 13, respectively.

In Rayleigh fading channels of small delay spread, such as flat fading and EPA model [58], the BER performances of the equalized DCCN receivers and benchmarks with long CP are presented in Figs. 11(a) and 11(b), respectively. Since the length of channel filter L is smaller than length of CP 16, the ISI can be completely removed by dropping CP. Compared to the ideal LMMSE, the DCCN receiver that drops CP is similar or slightly better in low to mid-range SNR regime (≤ 15 dB), and underperforms in high SNR regime with a tiny performance gap in flat fading and a larger gap of 1 to 3 dB in EPA channel. The DCCN-CP receiver outperforms the ideal LMMSE by around 1 dB and almost overlap with the ideal receiver with perfect channel estimates in the low to mid-range SNR regime, but its performance also deteriorates in high SNR regime. It shows that DCCN-CP receiver is able to exploit the redundancy information carried in CP for performance gain. Despite that DCCN-CP receiver coincidentally overlaps with perfect channel estimates here, its performance gain in

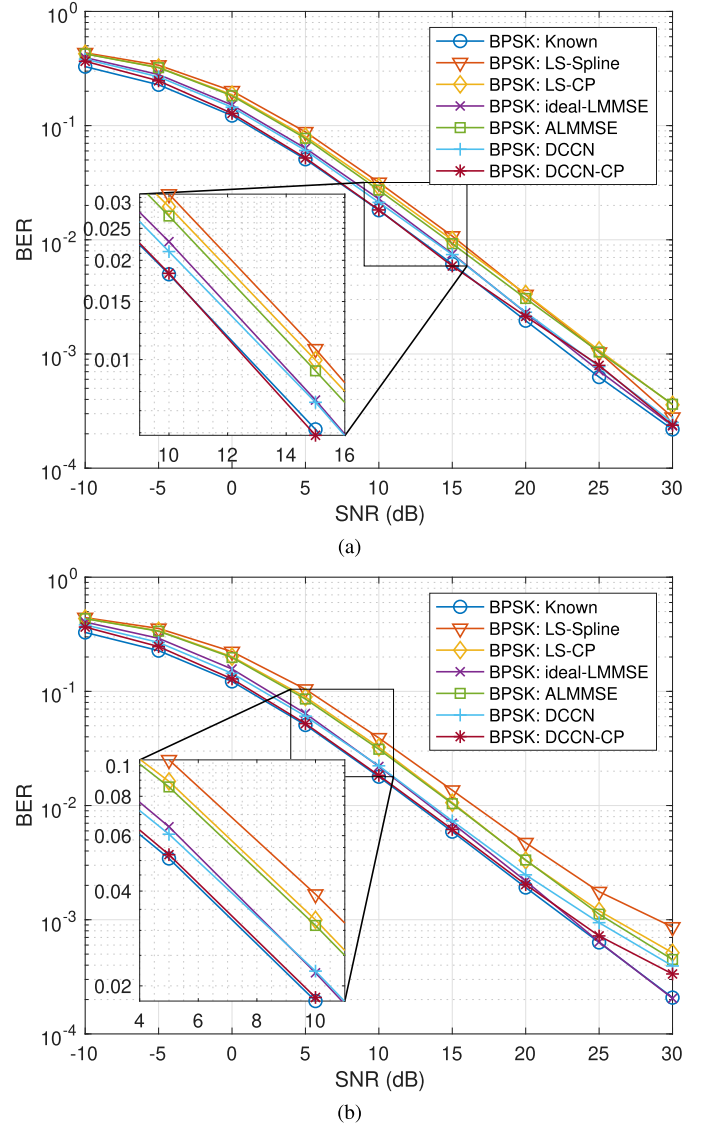


Fig. 11. BER of equalized DCCN receivers and benchmarks with long CP $N_{cp} = 16$ in low-mobility Rayleigh fading channels: (a) flat fading $L = 1$, and (b) multipath fading with low RMS delay spread (EPA [58]), $L = 9$.

fact depends on the amount of redundancy in CP, which is discussed later in Fig. 13. Compared to flat fading, the delay spread in EPA channel enlarges the performance gap between the ideal LMMSE and ALMMSE from 0.8 to 1.4 dB, and the gap between ALMMSE and LS-Spline from 0.7 to 1 dB, while LS-CP stays closely to ALMMSE. Although LS-CP underperforms ALMMSE by about 0.2 dB, LS-CP requires no prior information while ALMMSE is based on perfect SNR. Notice that SNR of 30 dB may not be common in practice, and the uncertainty of tested BER at the level of 10^{-4} is also higher. The fact that DCCN performs similar to the ideal LMMSE with prior knowledge of channel covariance matrix when ISI is completely removed, shows that DCCN can learn the channel statistics from data.

In a multipath channel of large delay spread, such as EVA and ETU models [58], the BER performance by SNR of the equalized DCCN receivers and benchmarks with short

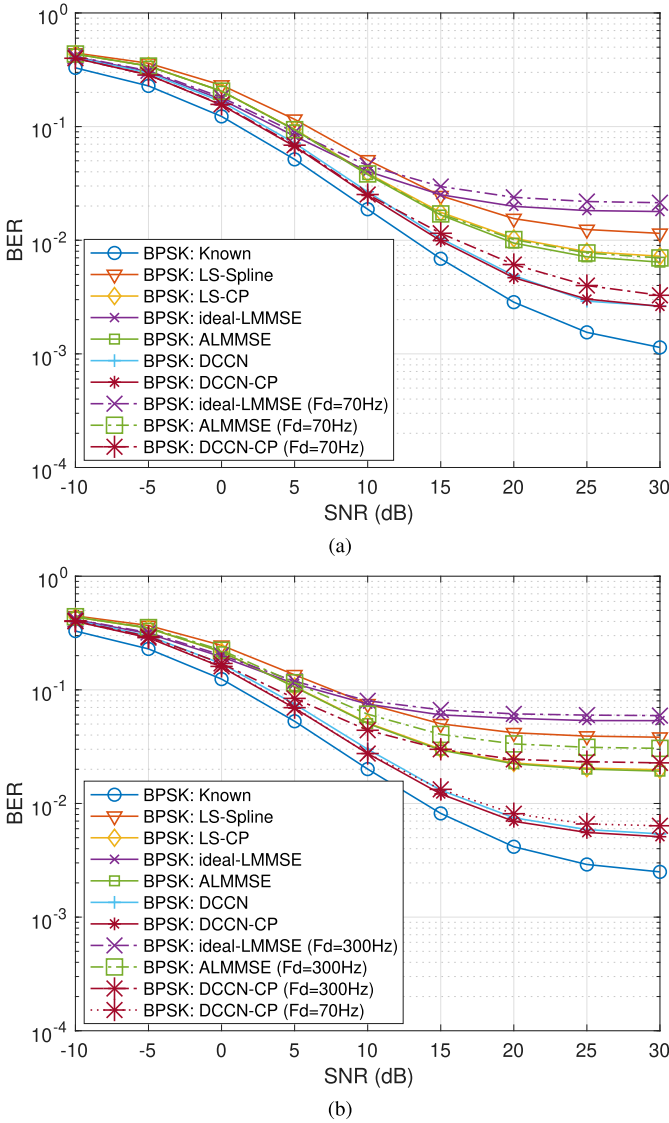


Fig. 12. BER of equalized DCCN receivers and benchmarks with short CP $N_{cp} = 4$ in multipath Rayleigh fading channels with large RMS delay spread and leakage of ISI: (a) EVA [58] $L = 11$, and (b) ETU [58] $L = 13$.

CPs are presented in Figs. 12(a) and 12(b), respectively. Notably, in the case of the length of the channel exceeds the length of CP, the ISI cannot be removed by dropping CP, resulting performance degradation in high SNR regime even with perfect channel knowledge. With the presence of ISI, the ideal-LMMSE underperforms LS and all the other channel estimators in mid-to-high SNR regime, similar results was found in [60]. Meanwhile, by alleviating the notch in frequency in frequency-selective fading [38], the performance gain of CP-enhanced LS estimator over the baseline LS estimator increases by the delay spread of the channels. In a frequency-selective channel with large delay spread, DCCN receiver outperforms the ALMMSE, LS, and CP-enhanced LS, with a large margin in high SNR regime, e.g., leading ALMMSE by 5 dB at BER of 0.01. The DCCN equalizer can take advantage of processing a whole coherence slot instead of a single OFDM symbol, thus further reduce the ISI. Similar mechanism is employed in conventional CP-enhanced

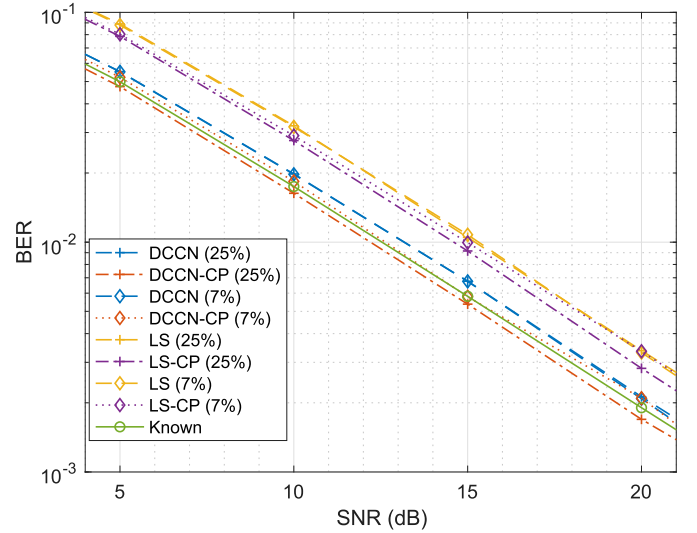


Fig. 13. BER of DCCN receivers with different CP lengths vs SNR in flat fading channel. The DCCN equalizer is trained only in flat fading channel. The gains of DCCN receiver with long CP (25% N) and short CP (7% N) are 0.88 dB and 0.33 dB, respectively. The DCCN receiver that drops CP is on average 0.5 dB worse than the receiver with known channel information. Legacy receiver with CP-enhanced LS equalizer [38] gains 0.7 dB and 0.3 dB over baseline LS equalizer in long and short CPs, respectively.

equalizer [38], in which decoded data of the previous OFDM symbol is used to mitigate the ISI in the next one. DCCN can learn to mitigate ISI and frequency nulling together in frequency-selective fading channels, yielding superior performance from synergy. In comparison, CP-enhancement approach in [38] can only enhance baseline LS estimation but fails to improve ideal-LMMSE and ALMMSE in our experiment.

The BERs of the tested receivers in doubly selective channels, simulated by including Doppler spread in the EVA and ETU channels, are shown with the dashed lines in Figs. 12(a) and 12(b). In the mobile channels, the Doppler effect increases the temporal channel variation and causes frequency dispersion that leads to inter-carrier interference at the OFDM receiver. With a maximum Doppler frequency of 70 Hz in both EVA and ETU channels, the BERs of ideal-LMMSE, ALMMSE, and DCCN-CP rise by 0.003, 0.0008, 0.001, respectively, in middle-to-high SNRs. The DCCN-CP still leads by a gap similar to the static cases. However, with a maximum Doppler spread of 300Hz in the ETU channel, the BERs of ideal-LMMSE, ALMMSE and DCCN-CP rise by 0.007, 0.01, and 0.018, respectively, where DCCN-CP suffers the most. These results show that DCCN-CP is robust to channel mobility, but with a relatively low pilot density of 4.7%, its advantage over legacy receivers can be compromised in high mobility scenarios.

The effectiveness of CP enhancement of both equalized DCCN receivers and CP-enhanced LS receivers [38] are evaluated in OFDM systems with long and short CPs, as illustrated in Fig. 13. For a fair comparison, flat fading channel is selected since no ISI will be leaked into the main OFDM symbol with short CP. The equalized DCCN receivers are trained in flat fading channels only. Both the baseline DCCN and LS receivers that drop CP has identical performance regardless

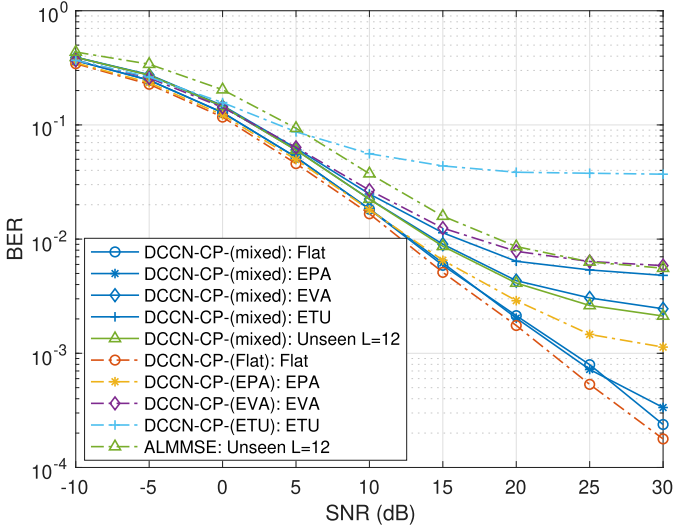


Fig. 14. Comparison of DCCN equalizers trained with mixed Rayleigh fading models vs trained with single channel model and tested on the same channel model, the four Rayleigh fading models are: flat fading, EPA, EVA, ETU [58].

of the CP length. In the settings of long and short CP, the DCCN-CP receiver gains 0.88 dB and 0.33 dB, respectively, and the CP-enhanced legacy LS receiver [38] gains 0.7 dB and 0.3 dB, respectively. The DCCN-CP receiver utilizes CP slightly more effective than the conventional approach [38].

D. Generalizability

The evaluation of DCCN and benchmarks in different channels in Section V-C already shows that a trained equalized DCCN receiver generalizes well to different fading channels. To further illustrate the impact of different training settings on the generalizability of DCCN, we present the BER of a DCCN-CP receiver trained in mixed Rayleigh fading channels in all four different fading channels and an unseen fading channel with $L = 12$, in comparison with four other DCCN-CP receivers trained in single Rayleigh fading channel and then tested in the same fading model keeping everything else the same, as illustrated in Fig. 14. When $\text{SNR} \geq 5$ dB, the receiver trained on mixed fading channels outperforms those trained in single multipath fading channel, except flat fading in which the DCCN-CP receiver trained in mixed fading models underperforms the one trained only in flat fading by 0.5 dB on average. Mixed fading models could help DCCN overcome local minima during training, as opposed to those trained in single fading model with large delay spread, as shown in Fig. 14, at the cost of precision in flat fading. The advantage of DCCN-CP receiver trained on mixed fading can also be generalized to unseen PDP (green curves in Fig. 14).

E. Ablation Studies and Alternative Training Methods

We test 6 alternative structures of the DCCN basic receiver, ($a-f$), with some of the 6 components being changed. Their detailed structures and test results for 4 modulations in AWGN channel as listed in Table III (–: included, X: removed, ✓: test pass, F: test fail). The two consecutive Conv layers are

TABLE III
ALTERNATIVE FLOW-GRAPHS OF DCCN BASIC RECEIVER WITH TEST RESULTS

Layers	original	a	b	c	d	e	f
DFT-Like C-Conv	–	dense	–	–	–	–	–
Conv $2^m \times 1 \times 2$	X	–	–	X	X	–	–
Conv $2^m \times 1 \times 2^m$	X	–	–	X	X	–	–
A0: Leaky ReLU	–	–	–	X	–	–	–
Concat(IQ, A0)	–	–	–	IQ	A0	–	A0
A1: Leaky ReLU	–	–	–	–	–	X	–
AWGN: BPSK	✓	✓	✓	✓	✓	✓	✓
AWGN: QPSK	✓	✓	✓	✓	✓	✓	✓
AWGN: 8QAM	✓	✓	✓	F	F	✓	✓
AWGN: 16QAM	✓	✓	✓	F	F	✓	✓

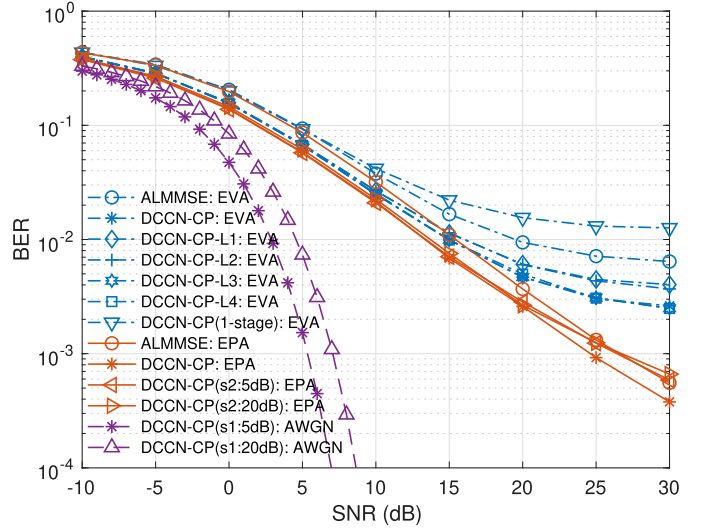


Fig. 15. BER performance of DCCN-CP receivers (short CP) with different hyperparameters or training SNRs in EPA and EVA channels.

located right before Leaky ReLU A0. Except c and d , the other 4 alternatives perform identically to the original one. Flow-graph a shows that the DFT-Like C-Conv layer can be replaced by a dense layer $\mathbb{C}_{S \times N}$. Large error floor appears in flow-graph c and d , which shows that only IQ vector (c) or its Leaky ReLU-activation (d) cannot replace their combination, which, however, can be replaced by linear convolution that expands IQ data from $\mathbb{R}^2$ to $\mathbb{R}^{2^m}$ followed by a non-linear activation in f . Alternative e shows that the last Leaky ReLU activation is unessential despite it improves training. Non-linear activation A0 helps to reduce the amount of trainable parameters as in the original flow-graph.

Next, we modify the number of dense layers of $\mathbb{C}_{FN \times FN}$ in DCCN-CP equalizers from 0 (DCCN-CP-L1) to 3 (DCCN-CP-L4), and remove the 2D C-Conv layer. DCCN-CP-L1 and DCCN-CP-L2 underperforms those with 2 and 3 layers (L3, L4, and the original) in EVA channels, as the blue lines in Fig. 15, while have similar performance in EPA channels. It shows that two dense layers of $\mathbb{C}_{FN \times FN}$ as in the original one is best for performance and complexity. It is also verified that by replacing C-Conv layers with an over-simplified implementation of processing real and imaginary parts with two separated Conv layers, DCCN could not be successfully trained, i.e., with a BER of 0.47.

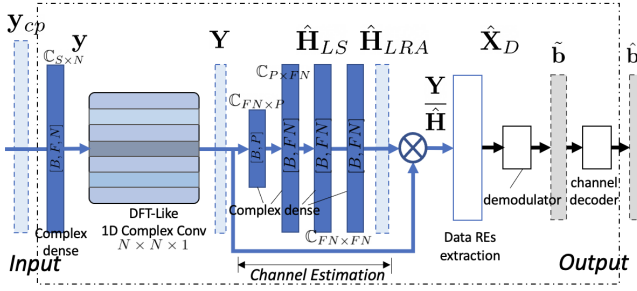


Fig. 16. Simplified DCCN-CP receiver for deployment, using legacy data REs extraction and demodulation, trained in single stage (stage 2).

We also test different training methods for both stages. In stage 1, DCCN-CP trained with an SNR of 20 dB underperforms the baseline by 2 dB in AWGN channels as shown by the purple dashed lines in Fig. 16. In stage 2, the same initial model trained in mixed fading channels with SNRs of 5 and 20 dB, respectively, exhibit larger error floors (0.0003) than the baseline trained in mixed SNRs, as shown by the orange lines in Fig. 16. Finally, training the DCCN-CP in a single stage, by mixing the channels and SNRs of the two stages, can significantly degrade the performance, i.e., BER increases by 0.01, as illustrated by the line of *DCCN-CP (1-stage)* in Fig. 16 in EVA channel. These results justify our 2-stage training approach and settings.

For deployment, the DCCN-CP receiver structure in Fig. 5 can be further simplified by replacing DCCN basic receiver with legacy data extraction and demodulation, as illustrated in Fig. 16. Compared to [11], the simplified DCCN-CP receiver requires no explicit DFT, can exploit CP for performance.

VI. CONCLUSION AND FUTURE WORK

In this paper, an end-to-end OFDM receiver, Deep Complex-valued Convolutional Networks (DCCN), is developed to recover uncoded bits from synchronized time-domain OFDM signal. By following the rule of multiplication in complex field instead of treating the real and imaginary parts of IQ samples as separated streams, DCCN is able to replace DFT/IDFT in OFDM system and exploit the redundancy of cyclic prefix in OFDM waveform for increased SNR. With the expressive power and synergistic advantage of complex-valued neural networks, DCCN is able to combine the tasks of CP-exploitation, low-rank approximation of LMMSE, and inter-symbol interference mitigation, thus outperform the legacy receivers with LMMSE and conventional CP-enhanced channel estimation in doubly-selective Rayleigh fading channels, with a lower computational complexity of $\mathcal{O}(N^2)$. This work also offers transferable experience for similar work. Practical guidelines is provided for approximated implementation of complex-valued convolutional networks, especially on setting the dimensions of a convolutional layer with respect to the parameters of OFDM system. A suite of novel training methods are developed for deep learning-based wireless transceiver, including a transfer learning scheme, an end-to-end loss function that can prevent vanishing gradient problem in training, and use of mixed SNR and fading models to smooth the loss land-

scape. It demonstrates the capability of deep neural network in processing sophisticated communication waveform, and suggests that the FFT processor in OFDM receiver can be replaced by a hardware AI accelerator.

Possible future directions with regard to complex-valued neural networks include: 1) explore non-linear signal processing for better performance and/or complexity, 2) improve the scalability by using convolutional layers for channel estimation, 3) extend to spatial domain such as massive MIMO, and 4) explore waveform design and channel coding through complex-valued neural networks-based communication autoencoder.

ACKNOWLEDGMENT

The majority of this work was completed during the Ph.D. studies of the author Zhongyuan Zhao at UNL.

REFERENCES

- [1] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," *China Commun.*, vol. 14, no. 11, pp. 92–111, Nov. 2017.
- [2] Q. Mao, F. Hu, and Q. Hao, "Deep learning for intelligent wireless networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2595–2621, 4th Quart., 2018.
- [3] D. Gunduz, P. de Kerret, N. Sidiropoulos, D. Gesbert, C. Murthy, and M. van der Schaar, "Machine learning in the air," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2184–2199, Oct. 2019.
- [4] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Engineering Applications of Neural Networks*, C. Jayne and L. Iliadis, Eds. Cham, Switzerland: Springer, 2016, pp. 213–226.
- [5] T. J. O'Shea, L. Pemula, D. Batra, and T. C. Clancy, "Radio transformer networks: Attention models for learning to synchronize in wireless systems," in *Proc. 50th Asilomar Conf. Signals, Syst. Comput.*, Nov. 2016, pp. 662–666.
- [6] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. Cognit. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Dec. 2017.
- [7] S. Dorner, S. Cammerer, J. Hoydis, and S. ten Brink, "On deep learning-based communication over the air," in *Proc. 51st Asilomar Conf. Signals, Syst., Comput.*, Oct. 2017, pp. 1791–1795.
- [8] A. Felix, S. Cammerer, S. Dorner, J. Hoydis, and S. Ten Brink, "OFDM-autoencoder for end-to-end learning of communications systems," in *Proc. IEEE 19th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jun. 2018, pp. 1–5.
- [9] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and DOA estimation based massive MIMO system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8549–8560, Sep. 2018.
- [10] C.-H. Cheng, Y.-H. Huang, and H.-C. Chen, "Channel estimation in OFDM systems using neural network technology combined with a genetic algorithm," *Soft Comput.*, vol. 20, no. 10, pp. 4139–4148, Oct. 2016.
- [11] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.
- [12] K. Yang, M. C. Vuran, S. Scott, F. Guo, and C. R. Ahn, "Neural network-based channel estimation for 2×2 and 4×4 MIMO communication in noisy channels," in *Proc. Int. Balkan Conf. Commun. Netw.*, May 2018, pp. 1–5.
- [13] M. Kim, W. Lee, and D.-H. Cho, "A novel PAPR reduction scheme for OFDM system based on deep learning," *IEEE Commun. Lett.*, vol. 22, no. 3, pp. 510–513, Mar. 2018.
- [14] Y. Liao, Y. Hua, X. Dai, H. Yao, and X. Yang, "ChanEstNet: A deep learning based channel estimation for high-speed scenarios," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–6.
- [15] H. He, S. Jin, C.-K. Wen, F. Gao, G. Y. Li, and Z. Xu, "Model-driven deep learning for physical layer communications," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 77–83, Oct. 2019.
- [16] M. Honkala, D. Korpi, and J. M. J. Huttunen, "DeepRx: Fully convolutional deep learning receiver," *IEEE Trans. Wireless Commun.*, early access, Feb. 2, 2021, doi: 10.1109/TWC.2021.3054520.

- [17] B. Lin, X. Wang, W. Yuan, and N. Wu, "A novel OFDM autoencoder featuring CNN-based channel estimation for Internet of vessels," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7601–7611, Aug. 2020.
- [18] P. Dong, H. Zhang, G. Y. Li, I. S. Gaspar, and N. NaderiAlizadeh, "Deep CNN-based channel estimation for mmWave massive MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 5, pp. 989–1000, Sep. 2019.
- [19] Y. Yang, F. Gao, X. Ma, and S. Zhang, "Deep learning-based channel estimation for doubly selective fading channels," *IEEE Access*, vol. 7, pp. 36579–36589, 2019.
- [20] M. Soltani, V. Pourahmadi, A. Mirzaei, and H. Sheikhzadeh, "Deep learning-based channel estimation," *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 652–655, Apr. 2019.
- [21] T. J. O'Shea, K. Karra, and T. C. Clancy, "Learning to communicate: Channel auto-encoders, domain specific regularizers, and attention," in *Proc. IEEE Int. Symp. Signal Process. Inf. Technol. (ISSPIT)*, Dec. 2016, pp. 223–228.
- [22] T. J. O'Shea, T. Erpek, and T. Charles Clancy, "Deep learning based MIMO communications," 2017, *arXiv:1707.07980*. [Online]. Available: <http://arxiv.org/abs/1707.07980>
- [23] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "A model-driven deep learning network for MIMO detection," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Nov. 2018, pp. 584–588.
- [24] C. Fan, X. Yuan, and Y.-J. Zhang, "CNN-based signal detection for banded linear systems," *IEEE Trans. Wireless Commun.*, vol. 18, no. 9, pp. 4394–4407, Sep. 2019.
- [25] Y. Zhang, M. Alrabeiah, and A. Alkhateeb, "Deep learning for massive MIMO with 1-bit ADCs: When more antennas need fewer pilots," *IEEE Wireless Commun. Lett.*, vol. 9, no. 8, pp. 1273–1277, Aug. 2020.
- [26] E. Balevi and J. G. Andrews, "One-bit OFDM receivers via deep learning," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4326–4336, Jun. 2019.
- [27] J. Xu, P. Zhu, J. Li, and X. You, "Deep learning-based pilot design for multi-user distributed massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 8, no. 4, pp. 1016–1019, Aug. 2019.
- [28] M. B. Mashhadi and D. Gündüz, "Pruning the pilots: Deep learning-based pilot design and channel estimation for MIMO-OFDM systems," *IEEE Trans. Wireless Commun.*, to be published, doi: [10.1109/TWC.2021.3073309](https://doi.org/10.1109/TWC.2021.3073309).
- [29] C. Zhang, H. Zhang, D. Yuan, and M. Zhang, "Citywide cellular traffic prediction based on densely connected convolutional neural networks," *IEEE Commun. Lett.*, vol. 22, no. 8, pp. 1656–1659, Aug. 2018.
- [30] Y. He *et al.*, "Deep-reinforcement-learning-based optimization for cache-enabled opportunistic interference alignment wireless networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10433–10445, Nov. 2017.
- [31] A. Chowdhury, G. Verma, C. Rao, A. Swami, and S. Segarra, "Unfolding WMMSE using graph neural networks for efficient power allocation," 2020, *arXiv:2009.10812*. [Online]. Available: <http://arxiv.org/abs/2009.10812>
- [32] X. Li, J. Fang, W. Cheng, H. Duan, Z. Chen, and H. Li, "Intelligent power control for spectrum sharing in cognitive radios: A deep reinforcement learning approach," *IEEE Access*, vol. 6, pp. 25463–25473, 2018.
- [33] Z. Zhao, G. Verma, C. Rao, A. Swami, and S. Segarra, "Distributed scheduling using graph neural networks," 2020, *arXiv:2011.09430*. [Online]. Available: <http://arxiv.org/abs/2011.09430>
- [34] A. Hirose, *Complex-Valued Neural Networks*, vol. 400, 2nd ed. Berlin, Germany: Springer, 2012, doi: [10.1007/978-3-642-27632-3](https://doi.org/10.1007/978-3-642-27632-3).
- [35] C. Trabelsi *et al.*, "Deep complex networks," in *Proc. Int. Conf. Learn. Represent.*, 2018. [Online]. Available: <https://openreview.net/forum?id=H1T2hmZAb>
- [36] M. Abadi *et al.* (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. [Online]. Available: <https://www.tensorflow.org/>
- [37] F. Chollet *et al.* (2015). *Keras*. [Online]. Available: <https://keras.io>
- [38] A. A. Quadeer, "Enhanced equalization in OFDM systems using cyclic prefix," in *Proc. IEEE Int. Conf. Wireless Commun., Netw. Inf. Secur.*, Jun. 2010, pp. 40–44.
- [39] T. Y. Al-Naffouri and A. A. Quadeer, "Cyclic prefix based enhanced data recovery in OFDM," *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3406–3410, Jun. 2010.
- [40] J. Yang, Q. Guo, D. D. Huang, and S. Nordholm, "A factor graph approach to exploiting cyclic prefix for equalization in OFDM systems," *IEEE Trans. Commun.*, vol. 61, no. 12, pp. 4972–4983, Dec. 2013.
- [41] S. Rathinakumar, B. Radunovic, and M. K. Marina, "CPRecycle: Recycling cyclic prefix for versatile interference mitigation in OFDM based wireless systems," in *Proc. 12th Int. Conf. Emerg. Netw. Exp. Technol.*, Dec. 2016, pp. 67–81.
- [42] L. Xu, J. Yang, D. Huang, and A. Cantoni, "Exploiting cyclic prefix for turbo-OFDM receiver design," *IEEE Access*, vol. 5, pp. 15762–15775, 2017.
- [43] Z. Zhao. *Source Code: An OFDM Receiver Based on Complex Deep Neuron Networks*. Accessed: Mar. 21, 2021. [Online]. Available: https://github.com/zhongyuanzhao/dl_ofdm
- [44] M. Koller, C. Hellings, M. Knodlseder, T. Wiese, D. Neumann, and W. Utschick, "Machine learning for channel estimation from compressed measurements," in *Proc. 15th Int. Symp. Wireless Commun. Syst. (ISWCS)*, Aug. 2018, pp. 1–5.
- [45] L. Li, H. Chen, H.-H. Chang, and L. Liu, "Deep residual learning meets OFDM channel estimation," *IEEE Wireless Commun. Lett.*, vol. 9, no. 5, pp. 615–618, May 2020.
- [46] Y. Shen and E. Martinez, "Channel estimation in OFDM systems," *Freescalc Semicond.*, Austin, TX, USA, Appl. Note AN3059, 2006, pp. 1–15. [Online]. Available: <https://www.nxp.com/docs/en/application-note/AN3059.pdf>
- [47] J.-J. van de Beek, "Synchronization and channel estimation in OFDM systems," Ph.D. dissertation, Dept. Comput. Sci., Elect. Space Eng., Signals Syst., Luleå Univ. Technol., Luleå, Sweden, 1998.
- [48] Z. E. Ankaralı, B. Peköz, and H. Arslan, "Enhanced OFDM for 5G RAN," *ZTE Commun.*, vol. 15, no. S1, pp. 1–10, 2017.
- [49] W. Wang, M. Hu, J. Yi, H. Zhang, and Z. Li, "Improved cross-entropy-based tone injection scheme with structured constellation extension design for PAPR reduction of OFDM signals," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 3284–3294, Apr. 2018.
- [50] X. Cheng, D. Liu, C. Wang, S. Yan, and Z. Zhu, "Deep learning-based channel estimation and equalization scheme for FBMC/OQAM systems," *IEEE Wireless Commun. Lett.*, vol. 8, no. 3, pp. 881–884, Jun. 2019.
- [51] M. C. Jeruchim, P. Balaban, and K. S. Shanmugan, *Simulation of Communication Systems: Modeling, Methodology and Techniques*. New York, NY, USA: Springer, 2006, doi: [10.1007/b117713](https://doi.org/10.1007/b117713).
- [52] W. Zhou and W. H. Lam, "A fast LMMSE channel estimation method for OFDM systems," *EURASIP J. Wireless Commun. Netw.*, vol. 2009, no. 1, Dec. 2009, Art. no. 752895.
- [53] O. Edfors, M. Sandell, J.-J. van de Beek, S. K. Wilson, and P. O. Borjesson, "OFDM channel estimation by singular value decomposition," *IEEE Trans. Commun.*, vol. 46, no. 7, pp. 931–939, Jul. 1998.
- [54] K.-C. Hung and D. W. Lin, "Pilot-based LMMSE channel estimation for OFDM systems with power-delay profile approximation," *IEEE Trans. Veh. Technol.*, vol. 59, no. 1, pp. 150–159, Jan. 2010.
- [55] V. Savvaux, F. Bader, and Y. Louet, "A joint MMSE channel and noise variance estimation for OFDM/OQAM modulation," *IEEE Trans. Commun.*, vol. 63, no. 11, pp. 4254–4266, Nov. 2015.
- [56] S. Ohno, S. Munesada, and E. Manasseh, "Low-complexity approximate LMMSE channel estimation for OFDM systems," in *Proc. Asia Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, Dec. 2012, pp. 1–4.
- [57] *Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation*, Standard (TS) 36.211, Version 15.12.0, 3GPP, 3rd Generation Partnership Project (3GPP), Tech. Specification, Feb. 2021.
- [58] *Evolved Universal Terrestrial Radio Access (E-UTRA); Base Station (BS) Radio Transmission and Reception*, Standard (TS) 36.104, Version 15.10.0, 3GPP, 3rd Generation Partnership Project (3GPP), Tech. Specification, Jan. 2021.
- [59] MathWorks. (2021). *Communication Toolbox: Fading Channels (R2021a)*. Accessed: Mar. 14, 2021. [Online]. Available: <https://www.mathworks.com/help/comm/ug/fading-channels.html>
- [60] A. Khlifi and R. Bouallegue, "Performance analysis of LS and LMMSE channel estimation techniques for LTE downlink systems," *Int. J. Wireless Mobile Netw.*, vol. 3, no. 5, pp. 141–149, Oct. 2011.



Zhongyuan Zhao (Member, IEEE) received the B.Sc. and M.S. degrees in electronic engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2006 and 2009, respectively, and the Ph.D. degree in computer engineering from the University of Nebraska-Lincoln, Lincoln, NE, USA, in 2019, under the guidance of Prof. Mehmet Can Vuran. From 2009 to 2013, he worked as an Engineer with ArrayComm and Ericsson, respectively, developing 4G base-station. He is currently a Postdoctoral Research Associate with the Department of Electrical and Computer Engineering, Rice University. His current research interest includes machine learning for wireless networks.



Mehmet Can Vuran (Member, IEEE) was born in Istanbul, Turkey. He received the B.Sc. degree in electrical and electronics engineering from Bilkent University, Ankara, Turkey, in 2002, and the M.S. and Ph.D. degrees in electrical and computer engineering from the Broadband and Wireless Networking Laboratory, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA, in 2004 and 2007, respectively, under the guidance of Prof. Ian F. Akyildiz.

He is currently the Susan J. Rosowski Professor of computer science and engineering with the University of Nebraska-Lincoln. He is a Daugherty Water for Food Institute Fellow. He was awarded an NSF CAREER Award for the project "Bringing Wireless Sensor Networks Underground."



Stephen D. Scott received the D.Sc. degree in computer science from Washington University in St. Louis in 1998. He is currently an Associate Professor with the Department of Computer Science and Engineering, University of Nebraska-Lincoln. His current research interests include deep learning and application of machine learning in image and text analysis.



Fujuan Guo (Student Member, IEEE) received the B.Sc. degree in electrical engineering from the Harbin University of Science and Technology in 2010, the M.S. degree in electrical engineering from the Harbin Institute of Technology, Harbin, China, in 2012, and the Ph.D. degree in computer engineering from the University of Nebraska-Lincoln, Lincoln, NE, USA, in 2019, under the guidance of Prof. Mehmet Can Vuran. She is currently a Senior Engineer with Qualcomm. Her research interests include wireless communications,

channel estimation, time synchronization for the Internet of Things (IoT) systems, the IoT-based localization, and machine learning in wireless communication. She was a recipient of the Best in Session Presentation Award of IEEE INFOCOM in 2017.