IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received January 13, 2020, accepted February 28, 2020, date of publication March 4, 2020, date of current version March 13, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2978253

Pilot-Assisted Channel Estimation and Signal Detection in Uplink Multi-User MIMO Systems With Deep Learning

XIAOMING WANG[1,2], **(Member, IEEE), HANG HUA**[1], **AND YOUYUN XU**[1], **(Senior Member, IEEE)**

[1]College of Telecommunication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
[2]National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China

Corresponding author: Youyun Xu (yyxu@njupt.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFE0200200, in part by the National Natural Science Foundation of China under Grant 61801240 and Grant 61971241, in part by the Natural Science Foundation of the Jiangsu Providence under Grant BK20180753, and in part by the Open Research Fund of the National Mobile Communications Research Laboratory, Southeast University under Grant 2019D16.

ABSTRACT In this paper, we propose two deep learning (DL) based receiver schemes in uplink multiple-input multiple-output (MIMO) systems. In the first scheme, we design a pilot-assisted MIMO receiver using a data-driven full connected neural network. This data-driven receiver can recover transmitted signal directly in an end-to-end manner without explicitly estimating channel. In the second scheme, we adopt a model-driven network which combines communication knowledge with DL. The model-driven scheme divides the MIMO receiver into channel estimation subnet and signal detection subnet, and each subnet is composed of a traditional solution as initialization and a DL network to further improve the accurate. The simulation results show that both of the two schemes achieve better bit error ratio (BER) performance than traditional methods. In particular, the data-driven scheme can achieve optimal BER performance in low-dimensional MIMO systems, while the model-driven scheme can be trained with fewer trainable parameters and outperforms the data-driven scheme in high-dimension MIMO systems.

INDEX TERMS Channel estimation and signal detection, MIMO, deep learning, model-driven, data-driven.

I. INTRODUCTION

High data-rate demands are becoming more and more challenging with the rapid development of mobile devices. To solve the problem of spectrum resources scarcity and increasing throughput requirements, multiple-input multiple-output (MIMO) has become one of the key technologies in the future network communication systems [1]–[4]. MIMO allows multiple antennas to send and receive messages simultaneously at transmitting and receiving terminals. It can effectively improve system capacity and spectrum efficiency without changing the system bandwidth and signal transmission power [5], [6].

A. BACKGROUND OF SIGNAL DETECTION AND CHANNEL ESTIMATION

In order to take advantage of the MIMO, efficient signal detection and channel estimation algorithms are always essential in design of MIMO receivers. With the increase of the number of transmit and receive antennas, the number of interference signals increases, which brings stronger co-channel interference. Therefore, signal detection algorithms are aimed to effectively suppress the channel interference and recover the transmitted signal in the MIMO systems. Maximum likelihood (ML) detection algorithm compares the received signal with all transmitted signals [7], and then estimates the comparison result according to the maximum likelihood principle to obtain the transmitted data. Although the ML algorithm has the optimal performance of signal detection, it is rarely used in practice because of its

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

high complexity. Linear detection algorithm is usually applied to reduce the computational complexity, such as matched filter method, zero forcing (ZF) method and minimum mean squared error (MMSE) method. However, the performance of these linear detectors is not good enough. Besides, it is worth mentioning that these signal detection methods are based on the assumption that channel matrix is known at the receivers. However, channel estimation is usually necessary before the signal detection. The channel estimation methods mainly include least square (LS) method, MMSE method and some compressive sensing-based methods (for massive MIMO systems) [8], [9]. To further improve the performance, combining them with deep learning (DL) is a feasible solution [10].

B. DEEP LEARNING

As a main technology in the field of artificial intelligence, DL has been widely concerned [11]. In recent years, the development of computers and the emergence of parallel computing have greatly reduced the cost and time of training the deep learning. It has enabled deep learning to develop rapidly in the areas of computer vision [12], [13], natural language processing [14], [15] and speech recognition [16], [17]. In addition, DL technology has gradually gained more attention in the field of wireless communications [18], such as channel estimation [19], [20], modulation identification [21], [22] and channel state information (CSI) feedback [23], [24]. In [25], a deep neural network (DNN) based scheme for beamforming in highly-mobile systems is proposed to reduce training overhead and achieve optimal performance. In [26], the DNN is integrated into hybrid precoding in millimeter wave (mmWave) MIMO systems to reduce the computational complexity. Moreover, deep learning for resource allocation has been considered in [27], where DNN is used to emulate the weighted MMSE algorithm. Literature [28] demonstrates that neural network can also learn the channel decoding algorithms.

Especially, literature [29] proposes a DNN detector which can be derived from the gradient descent method of unfolding projection for signal detection. Literature [30] employs a full-connection DL architecture to jointly replace signal detection and channel estimation modulation in orthogonal frequency division multiplexing (OFDM) systems. This method treats the function block of wireless communication as a black box and replaces it with a DL network. Besides, literature [31] uses a model-driven method named ComNet which integrates the communication knowledge into the deep learning to replace the OFDM receiver.

C. CONTRIBUTION

Inspired by the powerful feature extraction and mapping ability of deep learning, we design two MIMO receiver schemes based on the DL. In the first scheme, we use a data-driven DL network, named FullCon to replace the whole MIMO receiver. The proposed FullCon is a typical fully connected network which will be trained by a large amount of training data to obtain a mapping between the received signal and the transmitted signal. After the training is completed, the trained FullCon can recover the transmitted signal directly from the received signal. In the second scheme, we design a model-driven DL network named MdNet. Unlike the FullCon which regards the whole receiver as a black box, the MdNet divides the MIMO receiver into two subnetworks: channel estimation and signal detection by combining the expert communication knowledge and the DL. The experimental results show that both FullCon and MdNet have better bit error ratios (BERs) performance than the traditional schemes. To sum up, the main contributions of this paper are as follows:

- Inspired by the main methods of deep learning application in the field of communication, we designed two DL-based MIMO receivers: the data-driven DL architecture named FullCon and the model-driven DL architecture named MdNet.
- Different from the existing DL-based signal detector that is only adapted to the system with the fixed channel [32]–[34] or the known CSI [35]–[37], we propose two schemes considering both channel estimation and signal detection, which can be applied to time-varying MIMO channels.
- The proposed FullCon can directly recover the transmitted signal in an end-to-end manner. In addition, the proposed MdNet has fewer trainable parameters than the ComNet proposed in [31].
- Through simulation results, we sum up the application scenarios of the proposed MIMO receiver based on data-driven and model-driven DL respectively.

The rest of this paper is organized as follows. In Section II, we describe the system model and several conventional algorithms. Then we give the proposed data-driven FullCon and model-driven MdNet receivers for multi-user MIMO systems in Section III and Section IV respectively. In Section V, several simulation results are presented. Finally, we draw the conclusions in Section VI.

II. SYSTEM MODEL AND CONVENTIONAL ALGORITHMS

In this section, we first introduce the multi-user MIMO system model in Section II-A. Then, we present a briefly review the traditional schemes for channel estimation and signal detection in the MIMO systems.

A. SYSTEM MODEL

We consider the uplink transmission in a typical multi-user MIMO system, as shown in Fig. 1. The base station (BS) equipped with n antennas serves m users simultaneously where each user is equipped with a single antenna. The transmitted symbol vector is denoted as $\bar{\mathbf{x}} = [\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_m] \in \bar{\mathbb{S}}^m$, in which $\bar{x}_j$ ($j = 1, 2, \ldots m$) is a transmitted symbol from the jth transmitted antenna. $\bar{\mathbb{S}} = \{\bar{s}_1, \bar{s}_2, \ldots, \bar{s}_k\} \subset \mathbb{C}$ represents the symbol set of K-order modulation. $\bar{\mathbf{H}} \in \mathbb{C}^{n\times m}$ is the channel matrix whose (i, j)th entry $\bar{h}_{i,j}$ represents the path gain from the jth transmitted antenna to the ith received antenna. We consider a flat-fading channel

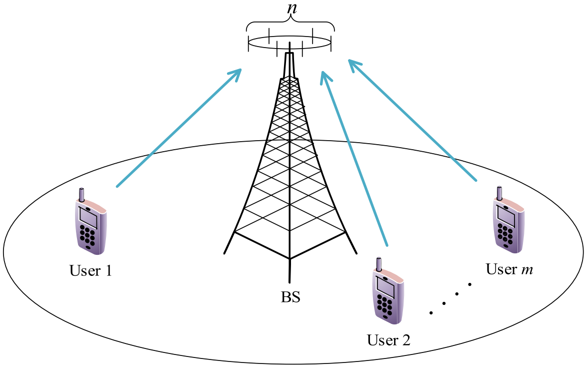


FIGURE 1. Uplink MIMO systems with one BS equipped *n* antennas and *m* single antenna users.

model with discrete-time and block fading where the channel information remains constant within a discrete time interval, i.e. channel matrix $\overline{\mathbf{H}}$ does not change in a time slot but changes with different blocks. The received symbol vector $\bar{\mathbf{y}} = [\bar{y}_1, \bar{y}_2, \ldots, \bar{y}_n] \in \mathbb{C}^n$ can be expressed as

$$\bar{\mathbf{y}} = \overline{\mathbf{H}}\bar{\mathbf{x}} + \bar{\mathbf{n}}, \tag{1}$$

where $\bar{\mathbf{n}}$ is the noise vector with independent, zero mean Gaussian variables of variance σ^2, and $\bar{y}_j$ is a received symbol from the jth receiving antenna.

B. SIGNAL DETECTION

The purpose of signal detection is to recover the transmitted data based on the signal received by the receiving antennas. The ML detection scheme is known for its optimal detection performance. Its principle is to transmit the signals through the known channel, and then detect them by checking the smallest distance metric. The specific rule of ML detection for detecting $\widetilde{\mathbf{x}}_{ML}$ is given by

$$\begin{aligned} \widetilde{\mathbf{x}}_{ML} &= \arg\max_{\mathbf{x}} P(\bar{\mathbf{y}}|\bar{\mathbf{x}}) \\ &= \arg\min_{\bar{\mathbf{x}}} \|\bar{\mathbf{y}} - \overline{\mathbf{H}}\bar{\mathbf{x}}\|^2. \end{aligned} \tag{2}$$

However, the ML detection is rarely used in practical applications due to its high computational complexity. On the other hand, the MMSE algorithm is a typical linear detection scheme with lower complexity, and its definition is to find the expected minimum mean square error when the transmitted signal and the received signal are linearly combined. The filter matrix of MMSE detection can be obtained by simplification as

$$\mathbf{G}_{MMSE} = \overline{\mathbf{H}}^{\dagger} = (\overline{\mathbf{H}}^H\overline{\mathbf{H}} + \sigma^2 I)^{-1}\overline{\mathbf{H}}^H, \tag{3}$$

where $(\cdot)^{\dagger}$ means the Moore-Penrose pseudo-inverse and $(\cdot)^H$ represents the conjugate transpose. Then, the transmitted signal recovered through this linear detector can be expressed as

$$\begin{aligned} \widetilde{\mathbf{x}}_{MMSE} &= \mathbf{G}_{MMSE}\bar{\mathbf{y}} \\ &= (\overline{\mathbf{H}}^H\overline{\mathbf{H}} + \sigma^2 I)^{-1}\overline{\mathbf{H}}^H\bar{\mathbf{y}}. \end{aligned} \tag{4}$$

It can be seen that the linear detection method mainly performs the pseudo-inverse with the channel matrix $\overline{\mathbf{H}}$, so the computational complexity is the order of cubic magnitude of the number of transmitting antennas. In addition, these conventional signal detection schemes are implemented under the assumption that the channel matrix is known.

C. CHANNEL ESTIMATION

Channel estimation is usually necessary when the channel information is unknown at receivers. We assume that pilot and data transmissions are done within a interval. N_p pilot vectors $\bar{\mathbf{x}}_p[n] \in \overline{\mathbb{S}}^{m\times 1}$ for $n = 1, \ldots, N_p$ are first transmitted from the users. Then, N_d data vectors $\bar{\mathbf{x}}_d[n] \in \overline{\mathbb{S}}^{m\times 1}$ for $n = 1, \ldots, N_d$ are transmitted following the pilot vectors within an interval. These vectors can also be expressed in matrix forms as $\bar{\mathbf{x}}_p = [\bar{\mathbf{x}}_p[1], \ldots, \bar{\mathbf{x}}_p[N_p]] \in \overline{\mathbb{S}}^{m\times N_p}$, $\bar{\mathbf{x}}_d = [\bar{\mathbf{x}}_d[1], \ldots, \bar{\mathbf{x}}_d[N_d]] \in \overline{\mathbb{S}}^{m\times N_d}$. The received pilot signal $\bar{\mathbf{y}}_p = [\bar{\mathbf{y}}_p[1], \ldots, \bar{\mathbf{y}}_p[N_p]] \in \mathbb{C}^{n\times N_p}$ and the received data signal $\bar{\mathbf{y}}_d = [\bar{\mathbf{y}}_d[1], \ldots, \bar{\mathbf{y}}_d[N_d]] \in \mathbb{C}^{n\times N_d}$ can be expressed as

$$\bar{\mathbf{y}}_p = \overline{\mathbf{H}}\bar{\mathbf{x}}_p + \bar{\mathbf{n}}_p, \tag{5}$$

$$\bar{\mathbf{y}}_d = \overline{\mathbf{H}}\bar{\mathbf{x}}_d + \bar{\mathbf{n}}_d, \tag{6}$$

where $\bar{\mathbf{n}}_p \in \mathbb{C}^{n\times N_p}$, $\bar{\mathbf{n}}_d \in \mathbb{C}^{n\times N_d}$ and $\overline{\mathbf{H}} \in \mathbb{C}^{n\times m}$. In order to obtain meaningful channel parameters, it is usually necessary to satisfy the condition that pilot length is not less than the number of transmitting antennas, i.e., $N_p \geq m$ [38]. LS estimator is a typical method for the channel estimation, and the channel $\widetilde{\mathbf{H}}_{LS}$ obtained by the LS estimator is given by

$$\widetilde{\mathbf{H}}_{LS} = \bar{\mathbf{y}}_{\mathbf{p}}\bar{\mathbf{x}}_{\mathbf{p}}{}^H(\bar{\mathbf{x}}_{\mathbf{p}}\bar{\mathbf{x}}_{\mathbf{p}}{}^H)^{-1}. \tag{7}$$

Its advantage is the simple structure and the low computational complexity. However, due to ignoring the influence of noise, the performance of this estimation algorithm is greatly reduced when the noise power is large.

III. PROPOSED DATE-DRIVEN RECEIVER FOR MIMO SYSTEMS

In this section, we present the architecture of the FullCon receiver for MIMO systems. In section III-A, we transform the system model of complex-value into an equivalent real-value model in order to combine MIMO with machine learning. The structure and details of the proposed FullCon are elaborated in Section III-B. After that, the training process of FullCon is presented in Section III-C.

A. REPARAMETERIZATION

A challenge in applying machine learning to MIMO systems is complex valued signals and channel parameters which are rare in the field of traditional machine learning. Taking into account this problem, we transform the complex system model into an equivalent real-valued channel model. As in [29], the real-valued model can be written as

$$\mathbf{y}_p = \mathbf{H}\mathbf{x}_p + \mathbf{n}_p, \tag{8}$$

$$\mathbf{y}_d = \mathbf{H}\mathbf{x}_d + \mathbf{n}_d, \tag{9}$$

where

$$\begin{aligned}
\mathbf{y}_p &= \begin{bmatrix} Re(\bar{\mathbf{y}}_p) \\ Im(\bar{\mathbf{y}}_p) \end{bmatrix} \in \mathbb{R}^{2n}, \quad \mathbf{H} = \begin{bmatrix} Re(\bar{\mathbf{H}}) & -Im(\bar{\mathbf{H}}) \\ Im(\bar{\mathbf{H}}) & Re(\bar{\mathbf{H}}) \end{bmatrix} \in \mathbb{R}^{2n \times 2m}, \\
\mathbf{x}_p &= \begin{bmatrix} Re(\bar{\mathbf{x}}_p) \\ Im(\bar{\mathbf{x}}_p) \end{bmatrix} \in \mathbb{S}^{2m}, \quad \mathbf{n} = \begin{bmatrix} Re(\bar{\mathbf{n}}) \\ Im(\bar{\mathbf{n}}) \end{bmatrix} \in \mathbb{R}^{2n}, \\
\mathbf{y}_d &= \begin{bmatrix} Re(\bar{\mathbf{y}}_d) \\ Im(\bar{\mathbf{y}}_d) \end{bmatrix} \in \mathbb{R}^{2n}, \quad \mathbf{x}_d = \begin{bmatrix} Re(\bar{\mathbf{x}}_d) \\ Im(\bar{\mathbf{x}}_d) \end{bmatrix} \in \mathbb{S}^{2m}. \qquad (10)
\end{aligned}$$

By separating the complex values into imaginary part and real part, we can observe that the elements of transmitted signal belong to a real constellation $\mathbb{S} = \{s_1, s_2, \ldots, s_{\sqrt{k}}\}$ with size $\sqrt{K}$. For example, the quadrature phase shift keying (QPSK) constellation is defined as

$$\begin{aligned}
\bar{s}_1 &= -1 - j \Leftrightarrow \mathbf{u}_1 = [0, 0], \\
\bar{s}_2 &= -1 + j \Leftrightarrow \mathbf{u}_2 = [0, 1], \\
\bar{s}_3 &= 1 - j \Leftrightarrow \mathbf{u}_3 = [1, 0], \\
\bar{s}_4 &= 1 + j \Leftrightarrow \mathbf{u}_4 = [1, 1], \qquad (11)
\end{aligned}$$

where $\mathbf{u}_i$ for $i = 1, \ldots, 4$ is the binary code element. $\bar{\mathbb{S}}$ in the complex-valued channel model (5) and (6) with QPSK is equivalent to binary phase shift keying (BPSK) modulation $\mathbb{S} = \{-1, 1\}$ in the real-valued channel model (8) and (9).

B. FULLCON ARCHITECTURE

The DNN is considered as the representative of artificial intelligence technology. Neural network is inspired by the concept that a neuron's computation involves a weighted sum of the input values [39], and the typical DNN model can be seen as a multi-layer perceptron (MLP). The DNN consists of multiple hidden layers which has powerful learning and mapping capabilities than single-layer neural network. The computation of jth neuron at each layer can expressed as

$$z_j = f(\sum_{i=1}^{P} W_{ij} \times v_i + b_j), \qquad (12)$$

where b_j is a bias. z_j, v_i, W_{ij} and P are the output data, input data, weights and the number of neurons respectively. $f(\cdot)$ is a nonlinear function called activation function, which is typically applied after full connection layer.

The role of the activation function in the neural network is to generate nonlinear decision boundaries, so that the DNN has nonlinear mapping learning ability. Generally, activation functions include 'Sigmoid' or Rectified Linear Units ('ReLu') [40] as well as TanHyperbolic ('tanh'). Mathematically, 'Sigmoid' function has a larger signal gain in the central region and a smaller signal gain in the bilateral regions. It benefits the mapping of signal characteristic space. The definition of 'Sigmoid' can be expressed as

$$f_{Sigmoid}(x) = \frac{1}{1 + e^{-x}}, \qquad (13)$$

which can map the output value of the neural network to interval [0, 1]. However, some disadvantages exist in 'Sigmoid', such as high computational complexity and vanishing gradient in backpropagation (BP). To solve them, 'ReLu' is becoming popular as its simplicity and ability to solve the vanishing gradient. 'ReLu' function can be denoted as

$$f_{ReLu}(x) = \max(0, x). \qquad (14)$$

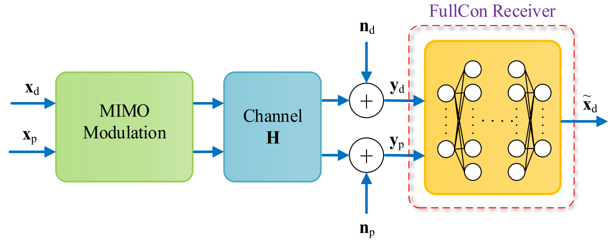


FIGURE 2. The diagram of the MIMO system with FullCon receiver.

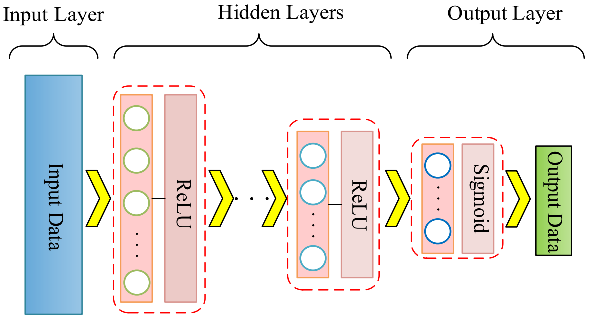


FIGURE 3. FullCon MIMO receiver architecture with hidden layers consisting of fully connected neural networks.

The diagram of MIMO systems with the FullCon receiver can be described as Fig. 2. In the receiver, the recovered data signal $\tilde{\mathbf{x}}_d$ can be estimated by using the received pilot signal $\mathbf{y}_p$ and the received data signal $\mathbf{y}_d$. Therefore, the input of FullCon is the combination of $\mathbf{y}_p$ and $\mathbf{y}_d$, and the output of FullCon is the estimated data signal $\tilde{\mathbf{x}}_d$. We denote this process of recovering signal as

$$\tilde{\mathbf{x}}_d = FullCon(\mathbf{y}_d, \mathbf{y}_p). \qquad (15)$$

The detail architecture of FullCon is shown in Fig. 3. The weight matrix and the bias vector in the lth layer of the FullCon are denoted by $\mathbf{W}_l$ and $\mathbf{b}_l$, where $l = 0$ represents the output layer. Setting the hidden layer number to L_1, the formula (15) can be rewritten as

$$\begin{aligned}
\tilde{\mathbf{x}}_d &= FullCon(\mathbf{y}_d, \mathbf{y}_p) \\
&= \psi_0(\mathbf{W}_0 \psi_{L_1}(\mathbf{W}_{L_1} \psi_{L_1 - 1}(\ldots \psi_1(\mathbf{W}_1(\mathbf{y}_d, \mathbf{y}_p) \\
&\quad + \mathbf{b}_1) \ldots) + \mathbf{b}_{L_1}) + \mathbf{b}_0), \qquad (16)
\end{aligned}$$

where $\psi_l(\cdot)$ with $l \neq 0$ represents the activation function at the nodes of the lth layer, and $\psi_0(\cdot)$ represents the activation function of output layer. Considering that the output of FullCon is transmitted binary 0-1 sequence streams, we set the activation function at the nodes of output layer $\psi_0(\cdot)$ as the 'Sigmoid' function. We then make the following judgment about the output value

$$x = \begin{cases} 0, & x < 0.5, \\ 1, & x \geq 0.5. \end{cases} \tag{17}$$

When the output value of the network is greater than 0.5, we judge it as symbol '1'. Otherwise, we judge it as symbol '0'. We use 'ReLu' as the activation function at the nodes of all hidden layers, i.e. $\psi_i(\cdot)$ is 'ReLu' function for $i = 1, \ldots, L_1$.

C. FULLCON TRAINING PROCESS

The FullCon receiver adopts a supervised learning approach, which is trained by a set of labeled training samples to obtain an optimal model. Our goal is to recover the transmitted signal through the FullCon receiver, thus the label of FullCon is the transmitted symbols $\mathbf{x}_d$ and the loss function in this scheme is set to the mean of squared errors (MSE) as follows

$$\mathbf{L} = \mathbf{MSE}(\widetilde{\mathbf{x}}_d, \mathbf{x}_d) = \frac{1}{|\mathcal{J}|} \sum_{\mathbf{x}_d \in \mathcal{J}} \|\widetilde{\mathbf{x}}_d - \mathbf{x}_d\|^2, \tag{18}$$

where $\mathcal{J}$ is the training dataset generated by simulation and $|\mathcal{J}|$ represents the size of training dataset. During the training stage, we use an optimization process called stochastic gradient descent (SGD) to update the weights and the biases of the neural network. The SGD algorithm updates the network by partial derivative of the gradient computed by back propagation. It will be iteratively repeated to update all of the weights and biases in the network to reduce the loss function.

Note that the pilot signal x_p should keep fixed in both the training and prediction stage. In the prediction stage, the complexity and consumption time of the training can be ignored. Once the FullCon is trained, $\widetilde{\mathbf{x}}_d$ can directly determined through Fullcon without explicitly estimating the channel.

IV. PROPOSED MODEL-DRIVEN RECEIVER FOR MIMO SYSTEMS

In this section, we report the proposed the model-driven MdNet receiver for MIMO systems. It divides the MIMO receiver into two subnet: channel estimation subnet and signal detection subnet. The structure of the proposed MdNet is described in Section IV-A and the detailed training process is presented in Section IV-B.

A. MDNET ARCHITECTURE

If we consider the proposed FullCon receiver scheme above as a black box, the proposed MdNet receiver architecture is a relatively bright box which is illustrated as Fig. 4.

In the MdNet receiver, both the channel estimation and the signal detection subnet are first initialized by low-complexity traditional methods, and then the preliminary results are sent to the deep learning module following these traditional methods to further improve the performance. Since this proposed method is based on the traditional methods, it has relatively superior robustness in the face of complex and various scenarios.

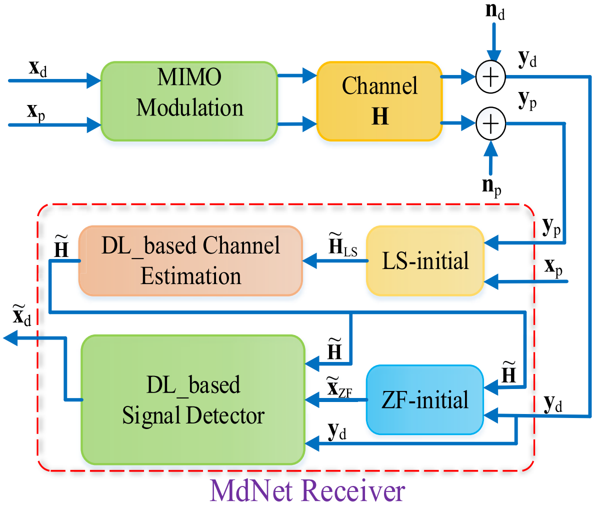


FIGURE 4. Architecture of MdNet MIMO receiver where the input is $\mathbf{x}_p$, $\mathbf{y}_p$ and $\mathbf{y}_d$, and the output is the estimated transmitted signal $\widetilde{\mathbf{x}}_d$.

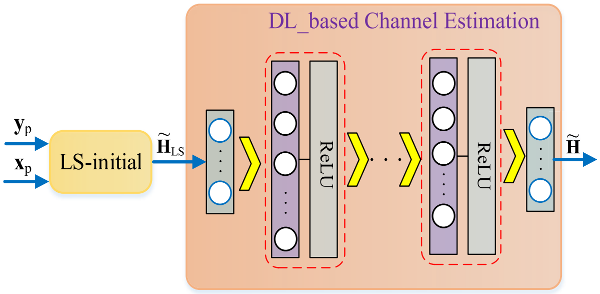


FIGURE 5. Architecture of channel estimation subnet. The subnet consists of a LS initializer and a DL channel refiner.

The detailed diagram of the channel estimation subnet is shown as Fig. 5. The first step is to get an initial channel matrix $\widetilde{\mathbf{H}}_{LS}$ by LS initialized channel estimator. The calculation process of $\widetilde{\mathbf{H}}_{LS}$ can be referred to the formula (7). Thus, the input of the LS initializer consists of the received pilot signal $\mathbf{y}_p$ and the pilot $\mathbf{x}_p$ which are known to the receiver. The initial $\widetilde{\mathbf{H}}_{LS}$ will be collected as the input of DL-based channel estimation (DL_CE) to obtain more accurate channel information. Here, the DL_CE is a fully connected deep learning network. The weight matrix and bias vector in the lth layer of the DL_CE are denoted by $\mathbf{Q}_l$ and $\mathbf{a}_l$, where $l = 0$ represent the output layer. Setting the hidden layers number to L_2, the final estimated channel matrix $\widetilde{\mathbf{H}}$ can be written as

$$\begin{aligned} \widetilde{\mathbf{H}} &= DL\_CE(\widetilde{\mathbf{H}}_{LS}) \\ &= \mathbf{Q}_0 \phi_{L_2}(\mathbf{Q}_{L_2} \phi_{L_2 - 1}(\ldots \phi_1(\mathbf{Q}_1 \widetilde{\mathbf{H}}_{LS} + \mathbf{a}_1) \ldots) + \mathbf{a}_{L_2}) + \mathbf{a}_0, \end{aligned} \tag{19}$$

where $\phi_l(\cdot)$ represents the activation function at the nodes of the lth layer. In the proposed DL_CE, we use 'ReLu' as activation function for all hidden layers, i.e., $\phi_i(\cdot)$ is 'ReLu' function for $i = 1, \ldots, L_2$.

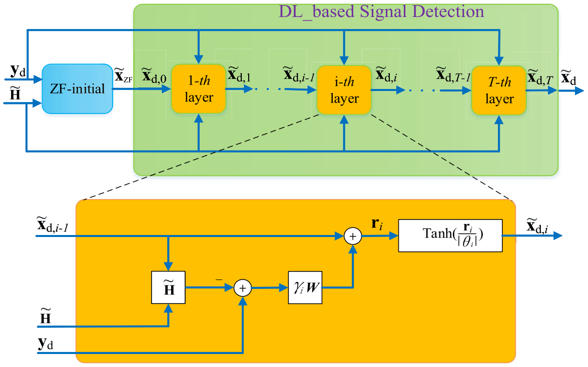

DL_based Signal Detection
$\mathbf{y}_d$
$\widetilde{\mathbf{H}}$
ZF-initial
$\widetilde{\mathbf{x}}_{ZF}$
$\widetilde{\mathbf{x}}_{d,0}$
1-th layer
$\widetilde{\mathbf{x}}_{d,1}$
$\widetilde{\mathbf{x}}_{d,i-1}$
i-th layer
$\widetilde{\mathbf{x}}_{d,i}$
$\widetilde{\mathbf{x}}_{d,T-1}$
T-th layer
$\widetilde{\mathbf{x}}_{d,T}$
$\widetilde{\mathbf{x}}_d$
$\widetilde{\mathbf{x}}_{d,i-1}$
$\mathbf{r}_i$
$\mathrm{Tanh}(\frac{\mathbf{r}_i}{|\theta_i|})$
$\widetilde{\mathbf{x}}_{d,i}$
$\widetilde{\mathbf{H}}$
$-$
$\gamma_i \boldsymbol{W}$
$\widetilde{\mathbf{H}}$
$\mathbf{y}_d$

FIGURE 6. Architecture of signal detection subnet consisting of a ZF initializer and a T-layers signal refiner. Each layer of DL_SD has the same structure, except for the learning parameters γ_i and θ_i.

In the signal detection subnet, the input is the received data signal $\mathbf{y}_d$ and the estimated channel matrix $\widetilde{\mathbf{H}}$ obtained by the channel estimation subnet. Fig. 6 shows the diagram of the signal detection subnet. Just like the channel estimation subnet, we first use a traditional ZF signal detector to obtain an insufficiently accurate signal detection result $\widetilde{\mathbf{x}}_{ZF}$ which will be used as the initialization data for the DL-based signal detection (DL_SD) iterative process. Different from the ComNet proposed in [31], which uses Bi-directional long short-term memory (BiLSTM) as signal detector refinenet, we used a detector refinenet named trainable projected gradient detector (TPG) proposed by [41] as the DL_SD to further reduce the number of training parameters and expand the advantages of model-driven scheme. The DL_SD is constructed by unfolding a variant of the projected gradient (PG) algorithm and introduces several trainable parameters to improve performance of the detection.

The PG algorithm has been applied in many fields [42], [43], including signal detection. This algorithm can be roughly described as finding the descent direction $\mathbf{d}^k$ of the objective function at the current iteration point $\mathbf{x}^k$, then starting from $\mathbf{x}^k$ and searching linearly along the direction $\mathbf{d}^k$, and finally getting the next iteration point by $\mathbf{x}^{k+1} = \mathbf{x}^k + a^k \mathbf{d}^k$ where parameter a^k is the step-size parameter. The PG-based MIMO detection is described by

$$\mathbf{r}_i = \widetilde{\mathbf{x}}_{d,i-1} + \gamma \mathbf{W}(\mathbf{y}_d - \widetilde{\mathbf{H}}\widetilde{\mathbf{x}}_{d,i-1}), \tag{20}$$

$$\widetilde{\mathbf{x}}_{d,i} = \tanh(\xi \mathbf{r}_i), \tag{21}$$

where $i = 1, \ldots, T$ represents the step of the iteration process and $\tanh(\cdot)$ is hyperbolic tangent function which can be expressed as

$$\tanh(x) = \frac{\sinh(x)}{\cosh(x)} = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \tag{22}$$

As can be seen from the formula (21), the soft projection function tanh can ensure the estimated $\widetilde{\mathbf{x}}_d$ to [-1, 1] interval. ξ is a parameter used for control the softness of soft projection. The matrix $\mathbf{W}$ in the formula (20) has many different forms, such as pseudo-inverse of $\widetilde{\mathbf{H}}$, linear MMSE matrix or just the transposed of $\widetilde{\mathbf{H}}$.

The TPG detector is composed of T cascade layers and each layer corresponds to one step iteration of traditional algorithm. The formula of the TPG is given by

$$\mathbf{r}_i = \widetilde{\mathbf{x}}_{d,i-1} + \gamma_i \mathbf{W}(\mathbf{y}_d - \widetilde{\mathbf{H}}\widetilde{\mathbf{x}}_{d,i-1}), \tag{23}$$

$$\widetilde{\mathbf{x}}_{d,i} = \tanh\left(\frac{\mathbf{r}_i}{|\theta_i|}\right), \tag{24}$$

where $i = 1, \ldots, T$. The matrix $\mathbf{W}$ in the formula (23) is defined by

$$\mathbf{W} = \widetilde{\mathbf{H}}^T (\widetilde{\mathbf{H}}\widetilde{\mathbf{H}}^T + \zeta \mathbf{I})^{-1}, \tag{25}$$

where $(\cdot)^T$ represents the transpose and ζ is a trainable parameter. Note that the MdNet aims to reduce training parameters and make full use of the knowledge of communication model. We adopt a same parameter ζ among the different layers in (25) to further reduce the complexity of the neural network. It can be clearly seen that the difference of TPG and PG is that the TPG adds training parameters $\{\gamma_i\}_{t=1}^T$ and $\{\theta_i\}_{t=1}^T$ to the step of gradient descent and soft projection respectively. That is to say, if γ_i and θ_i keep fixed in all layers, TPG will be equivalent to PG. In general, there are $2T + 1$ trainable scalar parameters in TPG, which are $\{\gamma_i\}_{t=1}^T$, $\{\theta_i\}_{t=1}^T$ and ζ. The number of trainable parameters is only related to the number of layers T. Compared with the traditional fully connected network, the number of training parameters is greatly reduced. In the proposed MdNet, we replace the initial detection signal $\widetilde{\mathbf{x}}_{d,0} = 0$ in TPG with $\widetilde{\mathbf{x}}_{ZF}$ obtained by the ZF signal detector, i.e., $\widetilde{\mathbf{x}}_{d,0} = \widetilde{\mathbf{x}}_{ZF}$, to speed up the convergence of iterations. Thus the received data signal $\mathbf{y}_d$, estimated channel matrix $\widetilde{\mathbf{H}}$ and signal $\widetilde{\mathbf{x}}_{ZF}$ are combined as the input of DL_SD and the output is the final transmitted signal $\widetilde{\mathbf{x}}_d$.

B. MDNET TRAINING PROCESS

The proposed DL_CE is to obtain more accurate channel information from the initial channel matrix $\mathbf{H}_{LS}$. Therefore, the label of the DL_CE is the real channel $\mathbf{H}$. The loss function is set to MSE as follows

$$\mathbf{L}(\Theta_Q) = \mathbf{MSE}(\widetilde{\mathbf{H}}, \mathbf{H}) = \frac{1}{|\mathcal{D}|} \sum_{\mathbf{H} \in \mathcal{D}} \|\widetilde{\mathbf{H}} - \mathbf{H}\|^2, \tag{26}$$

where $\mathcal{D}$ is the training dataset generated by simulation and $|\mathcal{D}|$ represents the size of training dataset. Θ_Q is a vector containing all the weights $\mathbf{Q}$ and biases $\mathbf{a}$ that need to be trained. we update Θ_Q as $\Theta_Q \leftarrow \Theta_Q - \eta \nabla_{\Theta_Q}(\mathbf{L}(\Theta_Q))$ by using SGD, where η is the learning rate and ∇ represents gradient. This process will iteratively repeat to update Θ_Q of the network to reduce the loss function.

In order to solve the problem of gradient vanishing, a training mode named incremental training is used in the DL_SD training process. The incremental training can be understood as follows: the first step is to train the first layer of network, and then increase the number of layers of training in each round. The last training result is used as the initial value of the next training. The loss function in the i training round can be expressed as

$$\mathbf{L}(\Theta_i) = \mathbf{MSE}(\mathbf{x}_d, \widetilde{\mathbf{x}}_{d,i}) = \frac{1}{|\mathcal{X}|}\sum_{\mathbf{x}_d \in \mathcal{X}} \|\mathbf{x}_d - \widetilde{\mathbf{x}}_{d,i}\|^2, \qquad (27)$$

where $\mathcal{X}$ is the training dataset generated by simulation and $|\mathcal{X}|$ represents the size of training dataset. Θ_i is a vector containing trainable parameters up to the ith layer, i.e., $\Theta_t = \{\gamma_1, \ldots, \gamma_i, \theta_1, \ldots, \theta_i, \zeta\}$. Within each round, the training process uses the SGD method to update the training parameters aiming to reduce the loss function, just like the training of the general neural networks. More details of the incremental training can be found in the literature [41].

V. SIMULATION RESULTS

In this section, we first introduce the setups of training in Section V-A and then describe the generation of dataset for training in Section V-B. The experimental results of the performance comparison between the two proposed schemes and the traditional methods are presented in Section V-C.

A. TRAINING METHOD

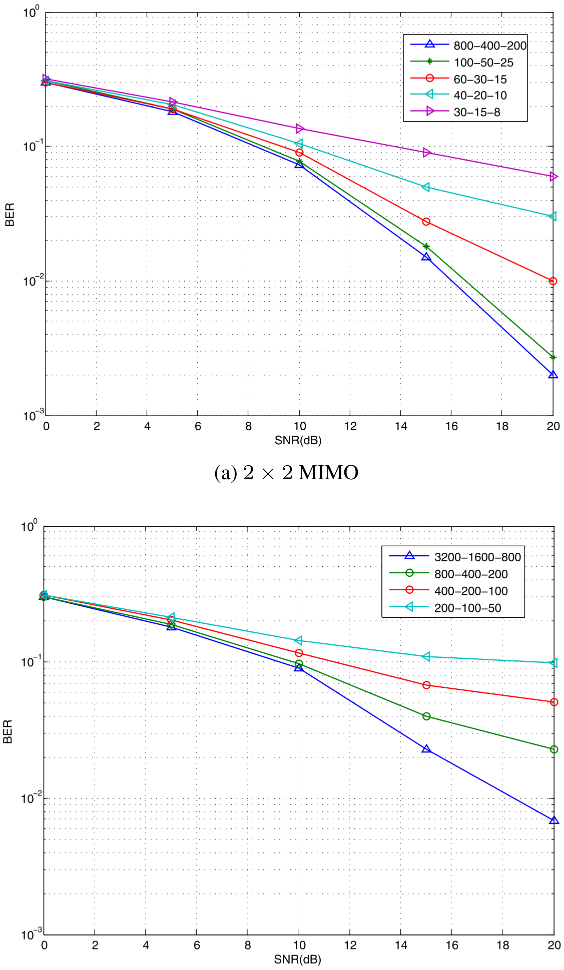
In all of the experiments, the simulation environment is based on Python with Tensorflow framework [44]. We use the graphic processing unit (GPU) method in the model training phase to speed up the training on a computer with i5-8300H CPU Core, one NVIDIA GeForce GTX 1050Ti GPU and 8GB RAM. Both in the training of FullCon and MdNet, we adopt mini-batch method and adaptive moment estimation (Adam) optimizer [45].

In the FullCon receiver, the number of hidden layers is set to 3. The number of training epoch is 2000. Each epoch consists of 20 batches and one batch contains 1000 training samples. In order to train the model better, we use the ladder learning rates. The initial learning rate is set $\lambda = 0.001$ and the learning rate λ drops five times for every 500 epochs in the following training process. Meanwhile, for every 100 epoches, we generate 10^5 testing data to evaluate the model performance.

In the MdNet receiver, the training of channel estimation subnet have 200 epochs. Each epoch consists of 10 batches and one batch contains 1000 training sample. We also use a ladder learning rate form in channel estimation subnet training. The initial learning rate is set $\lambda = 0.001$ and the learning rate λ drops five times for every 40 epochs in the following training process. In the training of signal detection subnet, we train 1000 epochs for each layer. So the total number of epochs is $1000T$. Each epoch consists of 1250 training sample.

B. GENERATION OF TRAINING DATASET

During the offline training, the training data is generated by simulating Rayleigh fading channels which is a reasonable channel model in urban environments. The channel $\mathbf{H}$ is time-varying and generated randomly following a typical independent and identically distributed (i.i.d.) Gaussian random variables. Since each training dataset is obtained using different channel information, the trained model can be applied to time-varying channels. The transmitted data $\mathbf{x}_d$ is randomly generated from the 0-1 distribution. The number of pilot symbols is set to the total number of users K, i.e., $N_p = m$ and $\mathbf{x}_d$ adopts Hadamard matrix. Finally, the QPSK modulation is employed in the simulation.



(a) 2 × 2 MIMO

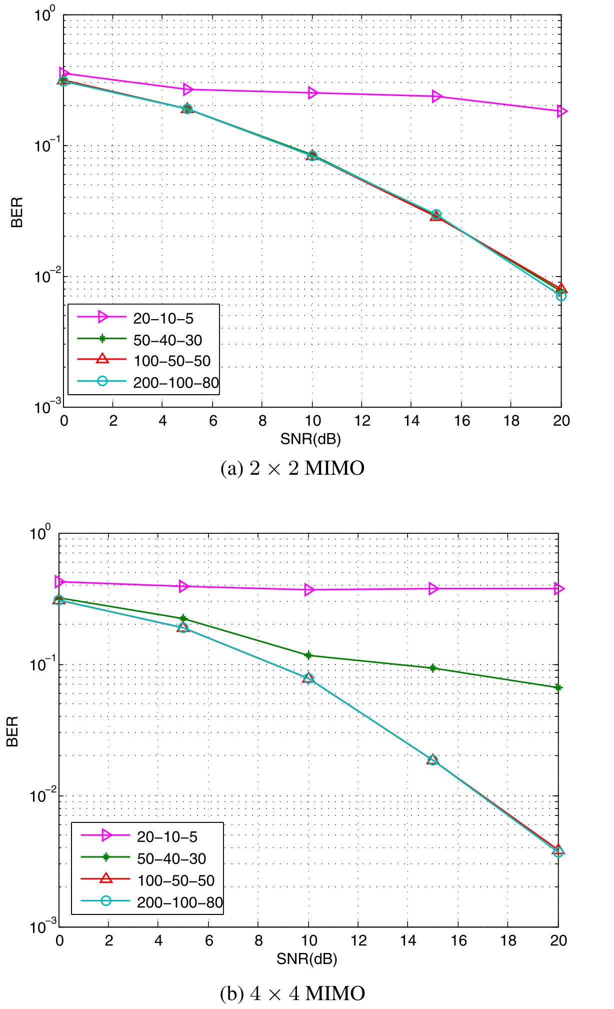
(b) 4 × 4 MIMO

FIGURE 7. BERs performance comparison of different hidden layer neurons in the FullCon: (a) BER value versus SNR in 2 × 2 MIMO; (b) BER value versus SNR in 4 × 4 MIMO.

C. MAIN RESULTS

In this section, we first investigate the effect of the number of neurons on the BER performance of FullCon and MdNet methods to find a balance between the complexity and accuracy of the proposed DL-based receivers. Then, we investigate the BER performance of the methods with different layers. Finally, the performance comparison between the proposed method and the traditional method is investigated. The abbreviations used in the following experiments are as follows:

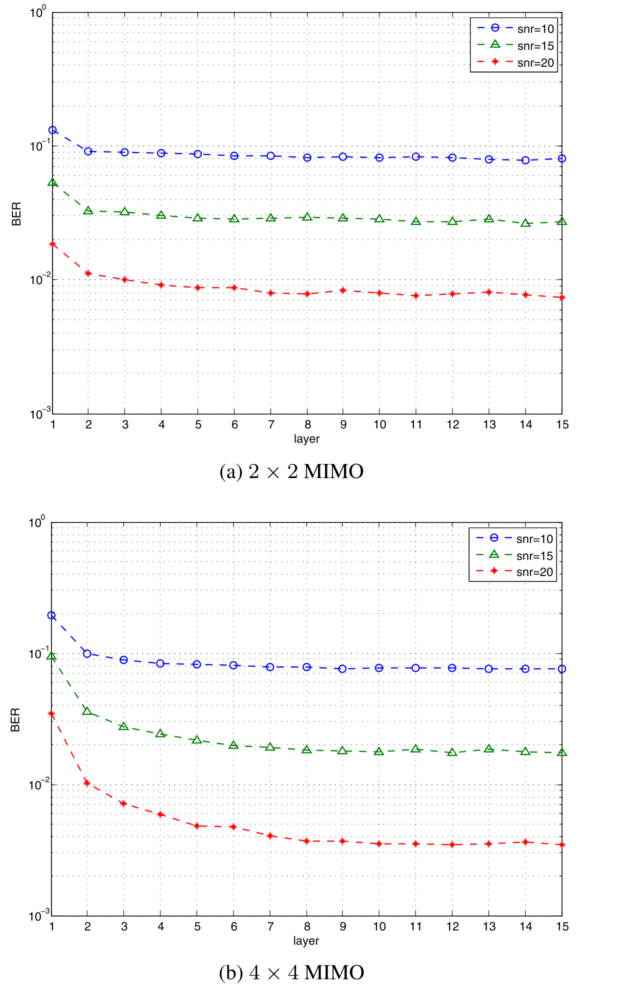
- LS_MMSE: Traditional LS channel estimation and MMSE signal detection;



(a) 2 × 2 MIMO

(b) 4 × 4 MIMO

FIGURE 8. **BERs performance comparison of different hidden layer neurons of the DL_CE subnet in MdNet: (a) BER value versus SNR in 2 × 2 MIMO, (b) BER value versus SNR in 4 × 4 MIMO.**



(a) 2 × 2 MIMO

(b) 4 × 4 MIMO

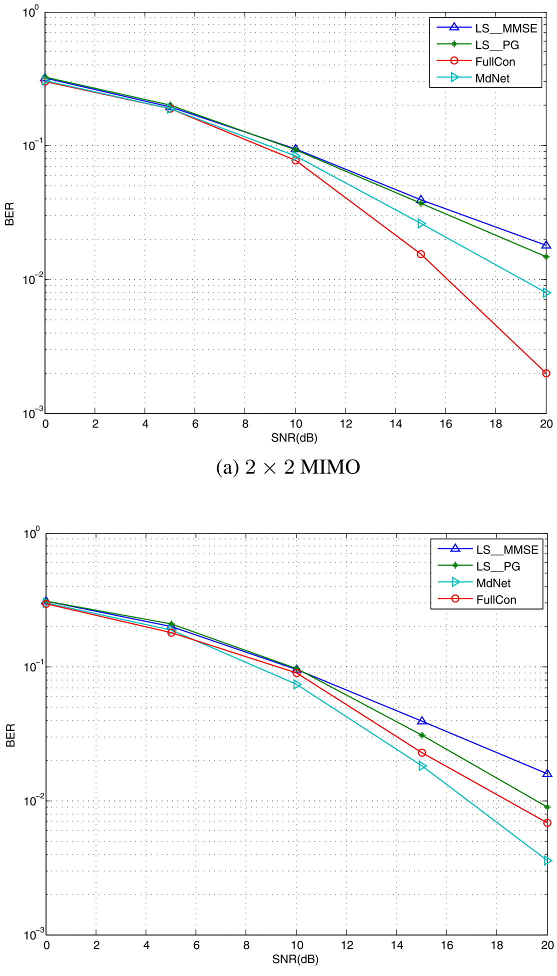
FIGURE 9. **BERs performance versus the number of layers in the MdNet under SNR = 10dB, 15dB and 20dB: (a) 2 × 2 MIMO; (b) 4 × 4 MIMO.**

- LS_PG: Traditional LS channel estimation and PG iterative signal detection;
- FullCon: Using the proposed FullCon instead of traditional MIMO receiver;
- MdNet: Using the proposed MdNet instead of traditional MIMO receiver.

1) IMPACT OF NEURONS NUMBERS IN FULLCON AND MDNET

Fig. 7 illustrates BERs performance versus signal-to-noise ratio (SNR) with different neuron numbers of each hidden layer in FullCon. In order to investigate the impact of the number of neurons in the hidden layer of FullCon on performance, and try to further reduce the complexity of the network while satisfying the appropriate BER performance, we test the performance of several groups of neurons. In this figure, we use the form of 'x_1-x_2-x_3' to represent the number of neurons in the three hidden layers respectively. In the scenario of 2 × 2 MIMO, the BER performance increases as the number of neurons increases. This is because with the increase of the number of neurons, the feature extraction and mapping capabilities of deep learning network will also be improved. However, Fig. 7(a) shows that the performance gap between 800-400-200 and 100-50-25 is very narrow, which indicates that when the number of neurons increases to a certain number, the performance improvement will become very slow. In the scenario of 4×4 MIMO system, the BER performance of the system also improves as the number of neurons increases shown as Fig. 7(b). However, at 3200-1600-800, the complexity of the model is already very high and the convergence time required for training is very long, so we do not continue to increase the number of neurons.

Fig. 8 presents BERs performance comparison of different hidden layer neurons of the DL_CE subnet in MdNet. It can be seen from Fig. 8 that the BERs performance increases as the number of neurons increases for both the scenario of 2 × 2 and 4 × 4 MIMO. This indicates that the performance of the channel estimation subnet has a great impact on the performance of subsequent signal detection. Although the



(a) 2 × 2 MIMO

(b) 4 × 4 MIMO

FIGURE 10. BERs performance comparison of the LS_MMSE, LS_PG, FullCon and MdNet over the Rayleigh fading MIMO channel: (a) BER value versus SNR in 2 × 2 MIMO; (b) BER value versus SNR in 4 × 4 MIMO.

trend of the curve in the figure is similar to that in Fig. 7, the number of trainable neurons of the model-driven method has been greatly reduced compared with the data-driven method.

2) IMPACT OF LAYER NUMBERS IN MDNET

Fig. 9 depicts BERs performance versus the number of layers in MdNet under SNR = 10dB, 15dB and 20dB. It can be seen from the Fig. 9 that with the increase of the number of layers, the BER performance is improved and converges after a certain number of layers. In particular, the performance improvement brought by the increase of the number of layers in 4 × 4 MIMO is more obvious than that of 2 × 2 MIMO. Besides, in the same MIMO system, the higher SNR, the greater the BER performance improvement caused by the increases in the number of layers. But a problem that comes with it is that the convergence speed is slow and more layers are needed. Based on the experiments, we find that all cases converge within 8 layers. So in the next performance comparison, we set the number of layers in MdNet to 8.

3) PERFORMANCE COMPARISON

In order to evaluate the performance of the proposed data-driven and model-driven DL schemes, we provide the BERs performance comparison of LS_MMSE, LS_PG, FullCon and MdNet over Rayleigh fading MIMO channel versus SNR in Fig. 10. Both the 2 × 2 and the 4 × 4 MIMO system are investigated. As is shown in Fig. 10, the BER performance of all scheme decreases as the SNR increases. Both the FullCon and the MdNet receivers achieve better BER performance than the traditional solutions in all setting. This reflects the superiority of deep learning to solve communication problems. It is worth mentioning that the FullCon scheme outperforms the MdNet scheme in 2 × 2 MIMO system and the advantage will be more obvious when the SNR increases. But when the dimension of the system increases, such as 4 × 4 MIMO system in the figure, the number of neurons in the FullCon must be multiplied, and the BER performance is still difficult to catch up with the MdNet scheme. This phenomenon reflects that the proposed FullCon which replace the whole MIMO receiving system with full connection network has an absolute advantage when the system dimension is low. However, when the antenna dimension increases, the MdNet method has the characteristics of lower training difficulty and higher BER performance compared with the FullCon method since the MdNet combines the advantages of traditional communication and machine learning.

VI. CONCLUSION

In this paper, we proposed two DL-based receiver schemes called FullCon and MdNet respectively in uplink multi-user MIMO scenario. The FullCon can be seen a data-driven black box, which uses a fully connected deep learning network to replace the whole MIMO receiver including channel estimator, signal detector and demodulator. The MdNet receiver is a model-driven deep learning method which combines traditional communication knowledge with DL. The simulation results show that both the FullCon and the MdNet outperform the traditional solutions. In addition, the FullCon can recover transmitted signal directly in an end-to-end manner without explicitly estimating channel and achieve the optimal BER performance when the MIMO dimension is low. As the MIMO dimension increases, the complexity and time required of training in FullCon are greatly large since the number of neurons must be multiplied to increase BER performance. However, the number of trainable parameters of MdNet is very small and independent of MIMO dimension, so the training of the MdNet is very fast. Moreover, the MdNet scheme can get best BER performance in high MIMO-dimension. In the future, we will extend the data-driven and model-driven DL architectures to hybrid MIMO systems.

REFERENCES

[1] A. Ghosh, R. Ratasuk, B. Mondal, N. Mangalvedhe, and T. Thomas, "LTE-advanced: Next-generation wireless broadband technology [invited paper]," *IEEE Wireless Commun.*, vol. 17, no. 3, pp. 10–22, Jun. 2010.

[2] C. Xu, S. Sugiura, S. X. Ng, P. Zhang, L. Wang, and L. Hanzo, "Two decades of MIMO design tradeoffs and reduced-complexity MIMO detection in near-capacity systems," *IEEE Access*, vol. 5, pp. 18564–18632, 2017.

[3] L. Hanzo, M. El-Hajjar, and O. Alamri, "Near-capacity wireless transceivers and cooperative communications in the MIMO era: Evolution of standards, waveform design, and future perspectives," *Proc. IEEE*, vol. 99, no. 8, pp. 1343–1385, Aug. 2011.

[4] Z. Gao, L. Dai, S. Han, C.-L. I, Z. Wang, and L. Hanzo, "Compressive sensing techniques for next-generation wireless communications," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 144–153, Jun. 2018.

[5] C.-K. Wen, S. Jin, and K.-K. Wong, "On the sum-rate of multiuser MIMO uplink channels with jointly-correlated rician fading," *IEEE Trans. Commun.*, vol. 59, no. 10, pp. 2883–2895, Oct. 2011.

[6] A. J. Paulraj, D. A. Gore, R. U. Nabar, and H. Bolcskei, "An overview of MIMO Communications—A key to gigabit wireless," *Proc. IEEE*, vol. 92, no. 2, pp. 198–218, Feb. 2004.

[7] S. Yang and L. Hanzo, "Fifty years of MIMO detection: The road to large-scale MIMOs," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1941–1988, 4th Quart., 2015.

[8] M. Ke, Z. Gao, Y. Wu, X. Gao, and R. Schober, "Compressive sensing based adaptive active user detection and channel estimation: Massive access meets massive MIMO," Jun. 2019, *arXiv:1906.09867*. [Online]. Available: http://arxiv.org/abs/1906.09867

[9] A. Liao, Z. Gao, H. Wang, S. Chen, M.-S. Alouini, and H. Yin, "Closed-loop sparse channel estimation for wideband millimeter-wave full-dimensional MIMO systems," *IEEE Trans. Commun.*, vol. 67, no. 12, pp. 8329–8345, Dec. 2019.

[10] H. Huang, S. Guo, G. Gui, Z. Yang, J. Zhang, H. Sari, and F. Adachi, "Deep learning for physical-layer 5G wireless techniques: Opportunities, challenges and solutions," *IEEE Wireless Commun.*, to be published, doi: 10.1109/MWC.2019.1900027.

[11] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[13] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1646–1654.

[14] R. Sarikaya, G. E. Hinton, and A. Deoras, "Application of deep belief networks for natural language understanding," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 4, pp. 778–784, Apr. 2014.

[15] A. R. Sharma and P. Kaushik, "Literature survey of statistical, deep and reinforcement learning in natural language processing," in *Proc. Int. Conf. Comput., Commun. Automat. (ICCCA)*, May 2017, pp. 350–354.

[16] G. Hinton, L. Deng, D. Yu, G. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

[17] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 6645–6649.

[18] 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.

[19] C.-J. Chun, J.-M. Kang, and I.-M. Kim, "Deep learning-based channel estimation for massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 8, no. 4, pp. 1228–1231, Aug. 2019.

[20] 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.

[21] J. Li, L. Qi, and Y. Lin, "Research on modulation identification of digital signals based on deep learning," in *Proc. IEEE Int. Conf. Electron. Inf. Commun. Technol. (ICEICT)*, Aug. 2016, pp. 402–405.

[22] S. Hong, Y. Zhang, Y. Wang, H. Gu, G. Gui, and H. Sari, "Deep learning-based signal modulation identification in OFDM systems," *IEEE Access*, vol. 7, pp. 114631–114638, 2019.

[23] C. Qing, B. Cai, Q. Yang, J. Wang, and C. Huang, "Deep learning for CSI feedback based on superimposed coding," *IEEE Access*, vol. 7, pp. 93723–93733, 2019.

[24] T. Wang, C.-K. Wen, S. Jin, and G. Y. Li, "Deep learning-based CSI feedback approach for time-varying massive MIMO channels," *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 416–419, Apr. 2019.

[25] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep learning coordinated beamforming for highly-mobile millimeter wave systems," *IEEE Access*, vol. 6, pp. 37328–37348, 2018.

[26] H. Huang, Y. Song, J. Yang, G. Gui, and F. Adachi, "Deep-learning-based millimeter-wave massive MIMO for hybrid precoding," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 3027–3032, Mar. 2019.

[27] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to optimize: Training deep neural networks for interference management," *IEEE Trans. Signal Process.*, vol. 66, no. 20, pp. 5438–5453, Oct. 2018.

[28] T. Gruber, S. Cammerer, J. Hoydis, and S. T. Brink, "On deep learning-based channel decoding," in *Proc. 51st Annu. Conf. Inf. Sci. Syst. (CISS)*, Mar. 2017, pp. 1–6.

[29] N. Samuel, T. Diskin, and A. Wiesel, "Learning to detect," *IEEE Trans. Signal Process.*, vol. 67, no. 10, pp. 2554–2564, May 2019.

[30] 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.

[31] X. Gao, S. Jin, C.-K. Wen, and G. Y. Li, "ComNet: Combination of deep learning and expert knowledge in OFDM receivers," *IEEE Commun. Lett.*, vol. 22, no. 12, pp. 2627–2630, Dec. 2018.

[32] M.-S. Baek, S. Kwak, J.-Y. Jung, H. M. Kim, and D.-J. Choi, "Implementation methodologies of deep learning-based signal detection for conventional MIMO transmitters," *IEEE Trans. Broadcast.*, vol. 65, no. 3, pp. 636–642, Sep. 2019.

[33] E. Nachmani, Y. Beery, and D. Burshtein, "Learning to decode linear codes using deep learning," 2016, *arXiv:1607.04793*. [Online]. Available: http://arxiv.org/abs/1607.04793

[34] Z. Chen, D. Li, and Y. Xu, "Deep MIMO detection scheme for high-speed railways with wireless big data," in *Proc. IEEE 89th Veh. Technol. Conf. (VTC-Spring)*, Apr. 2019, pp. 1–5.

[35] 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.

[36] M.-W. Un, M. Shao, W.-K. Ma, and P. C. Ching, "Deep MIMO detection using ADMM unfolding," in *Proc. IEEE Data Sci. Workshop (DSW)*, Jun. 2019, pp. 333–337.

[37] N. Samuel, T. Diskin, and A. Wiesel, "Deep MIMO detection," in *Proc. IEEE 18th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jul. 2017, pp. 1–5.

[38] B. Hassibi and B. M. Hochwald, "How much training is needed in multiple-antenna wireless links?" *IEEE Trans. Inf. Theory*, vol. 49, no. 4, pp. 951–963, Apr. 2003.

[39] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proc. IEEE*, vol. 105, no. 12, pp. 2295–2329, Dec. 2017.

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.

[41] S. Takabe, M. Imanishi, T. Wadayama, and K. Hayashi, "Deep learning-aided projected gradient detector for massive overloaded MIMO channels," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–6.

[42] C.-T. Chu, J.-N. Hwang, H.-I. Pai, and K.-M. Lan, "Robust video object tracking based on multiple kernels with projected gradients," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 1421–1424.

[43] F. Kizel, M. Shoshany, N. S. Netanyahu, G. Even-Tzur, and J. A. Benediktsson, "A stepwise analytical projected gradient descent search for hyperspectral unmixing and its code vectorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 4925–4943, Sep. 2017.

[44] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," Mar. 2016, *arXiv:1603.04467*. [Online]. Available: http://arxiv.org/abs/1603.04467

[45] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," Dec. 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

XIAOMING WANG (Member, IEEE) received the Ph.D. degree in information and communication engineering from the National Mobile Communications Research Laboratory, Southeast University, Nanjing, China, in 2016. He is currently a Lecturer with the Nanjing University of Posts and Telecommunications (NJUPT), Nanjing. His research interests include radio resource management, green communications, and machine learning in communications.

HANG HUA received the B.E. degree in rail transportation signal and control from Dalian Jiaotong University (DJTU), Dalian, China, in 2018. He is currently pursuing the master's degree with the Nanjing University of Posts and Telecommunications (NJUPT), Nanjing, China. His research interests include machine learning, MIMO, channel estimation, and signal detection.

YOUYUN XU (Senior Member, IEEE) received the Ph.D. degree in information and communication engineering from Shanghai Jiao Tong University (SJTU), in 1999. He is currently a Professor with the Nanjing University of Posts and Telecommunications. He is also a part-time Professor at the Institute of Wireless Communication Technologies, SJTU, China. He has over 20 years of professional experience in teaching and researching in communication theory and engineering with research and development achievement, such as the WCDMA Trial System under C3G Framework, China, in 1999, the B3G-TDD Trial System under FuTURE Framework, China, in 2006, and the Chinese Digital TV Broadcasting System. His current research interests include new generation wireless mobile communication systems (LTE, IM-T Advanced, and related), advanced channel coding and modulation techniques, multiuser information theory and radio resource management, wireless sensor networks, and cognitive radio networks. He is a Senior Member of the Chinese Institute of Electronics and a member of IEICE.