Int. J. Electron. Commun. (AEÜ) 155 (2022) 154350

Contents lists available at ScienceDirect

Int. J. Electron. Commun. (AEÜ)

journal homepage: www.elsevier.com/locate/aeue

Regular paper

DLNet: Deep learning-aided massive MIMO decoder

Satish Kumar *, Anurag Singh, Rajarshi Mahapatra

Department of Electronics and Communication Engineering, Dr. SPM International Institute of Information Technology-Naya Raipur, Nava Raipur, C.G., India

A R T I C L E I N F O

Keywords:
Correlated channel
Deep learning
Iterative network
Massive MIMO
MIMO decoding
Projected gradient descent

A B S T R A C T

Traditional MIMO decoding schemes are complex, impractical, and perform poorly for massive multiple-input multiple-output (M-MIMO) systems. Deep learning (DL) has recently emerged to perform many complex operations more efficiently within a shorter time. This paper proposes a learning-based network (DLNet) to design an M-MIMO decoder. The DLNet network architecture is designed by iteratively unfolding the gradient descent algorithm. The proposed DLNet decoder consists of 15 neural networks (NN) layers with some trainable parameters. This work considered uplink Rayleigh and correlated M-MIMO channels, which are perfectly known to the receiver. With the knowledge of the received signals and the M-MIMO channels, the proposed DLNet decoder decodes the messages of all the users. In the M-MIMO perspective, the proposed DLNet has been evaluated for symbol-error-rate (SER) performance, algorithm complexity, and run-time requirement. The simulations show that the proposed DLNet converges faster than other available decoders and performs better than other M-MIMO decoding schemes, by at least 2 dB in SER and at least 11 times faster than the baseline (OAMP-Net) and nine times less complex.

1. Introduction

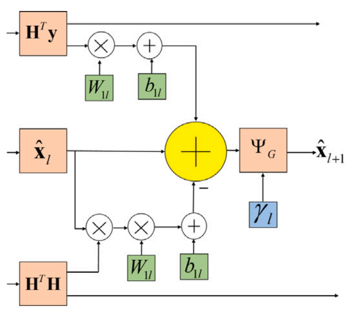
Massive multiple-input multiple-output (M-MIMO) is one of the primary transmission techniques for fifth-generation (5G) wireless communication systems. M-MIMO shows many advantages over classical MIMO schemes and can achieve all of its merits on a grander scale [1]. It mainly uses a large antenna array at a base station (BS) and a few antennas at user equipment (UE) to increase the data rate, link reliability, and coverage and reduce outage in the wireless systems. Additionally, using a sufficiently large number of antennas at BS and UE, the noise and intracell interference can be averaged out [2]. Conventional MIMO decoders are mainly developed using mathematical and information theory concepts that only capture the approximate behavior of the system. This makes it more challenging to perform end-to-end optimization of the communication system in practice and produces sub-optimal performance. These limitations on conventional MIMO decoders motivate researchers to investigate new M-MIMO decoding techniques that give near-optimal solutions, if not optimal, with less complex hardware implementation while faster processing speed. This paper proposes a Massive MIMO decoder using a deep neural network (DNN) that can satisfy the above criteria.

Recently, researchers have started using DL-based networks to address the problems related to communication systems [3,4]. While using DL-based techniques, mathematical and statistical models of the wireless channel and hardware systems are not required to implement the communication system. The data obtained from the real-world communication channels and hardware is used to optimize the system through rigorous training of the DNN. The DL-based systems capture the channel and hardware-based characteristics better than non-DL-based systems. In conventional wireless communication systems, the hardware and channels suffer from many impairments that make these systems highly nonlinear. As we know that neural networks can approximate any system function in a better way and are universal function approximators [5]. Such DNN detectors do not require a rigidly defined model for representation and transformation of information and can be optimized for a practical wireless system in a better manner [6].

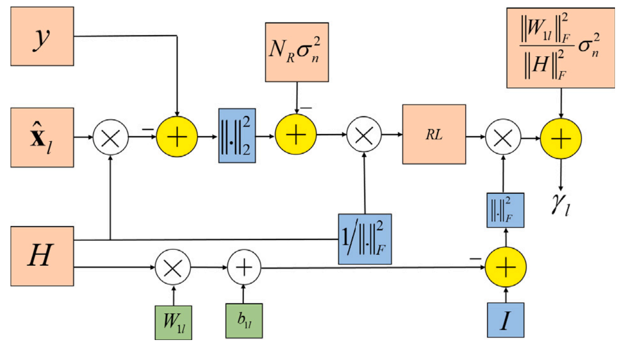
Many MIMO detectors have been investigated in the literature; the maximum likelihood (ML) decoder is one. ML decoder can give optimal error rate performance. However, it is an exhaustive search algorithm, and its complexity increases exponentially with the number of transmitted bits, making it nearly impossible to implement into practical systems. Some researchers have proposed the Sphere decoding [7,8] algorithm by limiting the search space. Its performance is close to ML decoder, but it is still very complex to implement practically. Zero-Forcing (ZF) and Minimum mean-squared error (MMSE) are two of many sub-optimal MIMO decoders. They are less complex in implementation but give a sub-optimal performance. Also, researchers have proposed expectation propagation (EP) [9] and approximate message passing (AMP) [10] as MIMO detection algorithms, and both of them

* Corresponding author.
E-mail addresses: satishkumarvatsa@gmail.com (S. Kumar), anurag@iiitnr.edu.in (A. Singh), rajarshim@ieee.org (R. Mahapatra).

https://doi.org/10.1016/j.aeue.2022.154350
Received 15 March 2022; Accepted 22 July 2022
Available online 27 July 2022



(a) Internal structure of l^{th} iteration



(b) Steps for calculating γ_l

Fig. 1. Internal structure of lth layer of the proposed DLNet decoder.

work iteratively. The AMP performs well on independent, ideally distributed (i.i.d.) Gaussian channels while EP decoder is more complex than AMP and performs well on unitary invariant MIMO channels.

Recently, researchers have used convolutional neural network (CNN) and DNN to detect MIMO signals in perfect and imperfect channel knowledge cases [11–13]. Some other iterative MIMO decoders with neural-network such as detection-network (DetNet) [14] and Orthogonal AMP (OAMP) [15] have been proposed. The DetNet is based upon a data-driven approach where knowledge about the channel model is not required. DetNet relies on a massive dataset for training and has many trainable variables. This large number of trainable variables makes the DetNet more complex, and thus it runs slower during training and decoding. In another work, quasi-ML decoding of MIMO signals has been discussed, for that two semi-definite relaxation (SDR) based models have been proposed [16]. These DetNet and SDR models provide near-optimal error rate performance but take more time while decoding and perform well on channels having i.i.d. Gaussian coefficients, whereas their error rate severely degrades on correlated MIMO channels. Based upon OAMP, in [17], authors proposed a model-driven deep learning network-based MIMO decoder OAMP-Net. OAMP-Net is trained offline and works well for both i.i.d Gaussian channel and correlated channels. OAMP-Net has only two trainable variables per layer, but it makes strong assumptions about the channel. The OAMP-Net is based on the MMSE model and needs to calculate the channel inverse in each iteration, which is too complex for large MIMO channels and thus takes more time during training and detection. A similar iterative and learning-based MIMO decoder has been proposed in [18]. In this, the authors have proposed two distinct NN architectures for i.i.d. Gaussian and spatially correlated channels. In i.i.d. Gaussian channels, the network (MMNet-iid) has very few trainable variables and needs to train offline on several channel realizations. However, the network (MMNet) needs to train online on every channel realization in spatially correlated channels. As we know, training neural networks is a time-consuming and highly resource greedy task. In MMNet, the proposed online training requires hardware with huge computational complexity and introduces latency, and thus cannot be used in practical systems.

1.1. Contribution of the paper

In this work, we propose a DL-based network (DLNet) architecture for signal decoding in the M-MIMO system. In particulars, the main contribution of this work is to design a DL decoder for M-MIMO, which is less complex, has a lower run time requirement, and provides better SER performance. This detector is built on DNN architecture which has exact knowledge of the channel. It takes the received signal as input to decode the transmitted bits. DLNet decoder is designed with several DNN layers, as shown in Fig. 1 (discussed in detail in Section 3). The selection of layers is essential for the optimum and accurate performance of the DLNet decoder. In the proposed DNN structure, the DLNet decoder uses 40 DNN layers, which depends on ML estimator's projected gradient descent-based solution. The simulation shows that the proposed DLNet achieves better performance, is less complex, and takes less time than many conventional MIMO decoders. This algorithm can be implemented in real-time and can decode an M-MIMO system with a large number of transmit–receive antennas. This presented M-MIMO decoder shows better SER performance, consumes at least 800 times less time while decoding and has 13 times fewer trainable parameters than DetNet.

In DLNet, we have used a different approach while choosing the loss function of the network for training than DetNet. DLNet's loss function is the sum of mean squared errors between transmitted and estimated message vectors at each layer of DLNet. The idea of this loss function is derived from the ML decoder that gives optimal performance. This loss function does not require matrix inverse calculation. Calculation of matrix inverse is computationally expensive. This further makes DLNet less complex and faster. (The loss function of DetNet requires matrix inverse calculation. See Ref [14]. Eq. (14)). Apart from this, the architecture of the proposed DLNet is based upon a learning-based solution of maximum likelihood (ML) decoder and uses a very less number of trainable variables than other M-MIMO decoders. These fewer trainable variables make the DLNet faster during training and decoding. It also requires less complex hardware for implementation. Thus the main contributions of this paper are as follows:

- A deep-learning-based decoder (DLNet) is proposed for M-MIMO signal decoding.
- We have chosen a loss function of the neural network that makes the MIMO decoder faster while decoding and less error-prone.
- Design the DLNet decoder architecture for M-MIMO with less trainable variables and less complex structure than the other available M-MIMO decoders.
- The number of the DNN layers, the number of the training iterations and the skip connection weight of the proposed DLNet decoder are estimated for optimal performance.

In this paper, we are denoting the mean and variance of a random variable as m and σ^2 respectively. Normal distribution with mean m and variance σ^2 as $\mathcal{N}(m, \sigma^2)$. The uniform distribution between a and b as $\mathcal{U}(a, b)$. Matrices are denoted by bold uppercase alphabets whereas bold lowercase alphabets are representing row or column vectors. x_i will be the ith element of a vector and $(.)^T$ is denoted as transpose of a matrix.

The organization of this paper is as follows. The conventional M-MIMO system model is discussed in Section 2. Section 3 presents the decoding structure of the proposed DLNet architecture. This section also presents the proposed M-MIMO decoder's mathematical model and network architecture. The simulation result and performance evaluation of the proposed DLNet have been discussed in Section 4. Section 5 concludes the work.

2. Massive-MIMO system model

Consider an M-MIMO system, where N_T user equipments (UEs) with single antenna transmit data to a base station (BS) of N_R receive antennas ($N_R > N_T \gg 1$). The channel between each transmit–receive antenna pair is assumed to be frequency flat fading and remain unchanged over one transmitted packet. In this work, we have considered the working frequency 5500 MHz, bandwidth is 80 MHz, channel model is Rayleigh and Correlated MIMO channels, and we have assumed that the channel state is perfectly known to the receiver.This M-MIMO system can be represented using the following mathematical expression:

$$\mathbf{y}_c = \mathbf{H}_c\mathbf{x}_c + \mathbf{n}_c \tag{1}$$

where $\mathbf{y}_c \in \mathbb{C}^{N_R}$ is received message vector, $\mathbf{x}_c \in \mathbb{S}^{N_T}$ is transmitted signal vector of equiprobable and independent symbols from a finite constellation $\mathbb{S}$, we assume that all the constellation points in constellation set $\mathbb{S}$ are normalized to unity power, $\mathbf{H}_c \in \mathbb{C}^{N_R \times N_T}$ is MIMO channel matrix and $\mathbf{n}_c \in \mathbb{C}^{N_R}$ is complex symmetric i.i.d. Gaussian noise of zero mean and variance σ_n^2. The real-valued equivalent of a complex MIMO system in (1) is given as

$$\begin{bmatrix}\mathrm{Re}\{\mathbf{y}_c\}\\ \mathrm{Im}\{\mathbf{y}_c\}\end{bmatrix} = \begin{bmatrix}\mathrm{Re}\{\mathbf{H}_c\} & -\mathrm{Im}\{\mathbf{H}_c\}\\ \mathrm{Im}\{\mathbf{H}_c\} & \mathrm{Re}\{\mathbf{H}_c\}\end{bmatrix} \times \begin{bmatrix}\mathrm{Re}\{\mathbf{x}_c\}\\ \mathrm{Im}\{\mathbf{x}_c\}\end{bmatrix} + \begin{bmatrix}\mathrm{Re}\{\mathbf{n}_c\}\\ \mathrm{Im}\{\mathbf{n}_c\}\end{bmatrix} \tag{2}$$

For conciseness, Eq. (2) can be represented as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \tag{3}$$

where $\mathbf{y} \in \mathbb{R}^{2N_R}$, $\mathbf{x} \in \mathbb{R}^{2N_T}$ and $\mathbf{n} \in \mathbb{R}^{2N_R}$. $\mathbf{H} \in \mathbb{R}^{2N_R \times 2N_T}$ is real-valued matrix. In this paper, the channel matrix $\mathbf{H}$ is stochastically generated, and it is assumed that the receiver perfectly knows it.

Upon receiving a message, the receiver needs to decode it properly. Among many traditional MIMO decoders, the ML decoder performs optimally. It minimizes the joint probability of error of all the symbols simultaneously. It uses the nearest Euclidean distance with the knowledge of the channel state and received signal to estimate the transmitted message. In order to maximize the probability of correctly calculating the $\mathbf{x}$, the error needs to be minimized. The joint probability of the error is represented below:

$$P(\mathbf{x} = \hat{\mathbf{x}}|\mathbf{y}, \mathbf{H}) = \frac{P(\mathbf{x} = \hat{\mathbf{x}}) f_{y|x,H}(\mathbf{y}|\mathbf{x} = \hat{\mathbf{x}}, \mathbf{H})}{f_{y|H}(\mathbf{y}|\mathbf{H})} \tag{4}$$

After maximizing expression (4), we will get the ML decoder which is given below:

$$\hat{\mathbf{x}}_{\mathbf{ML}} = \underset{\hat{\mathbf{x}} \in \mathbb{S}_{\mathbf{T}}^{\mathbf{N}}}{\arg\min} \quad \|\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}\|^2 \tag{5}$$

Since the ML estimator uses an exhaustive search approach, it searches for all possible combinations of transmitted symbols. Its computational complexity increases exponentially $O\left(M^{N_T}\right)$ with the increasing number of transmit antennas and modulation order ($M = \log_2 n(\mathbb{S})$). Where $\mathbb{S}$ is the constellation set. Some other linear decoders, such as ZF and MMSE have been proposed for decreasing complexities. Despite using higher complexity, the imperfect channel information or unavailability of channel state information leads to performance degradation. To overcome these limitations, people search for an efficient, fast, and less complex M-MIMO decoder. Dl-based DLNet is one such M-MIMO decoding technique.

3. DLNet M-MIMO decoder architecture

This section shows the network architecture of the proposed DLNet. The DNN approach has been incorporated into the classical ML estimator iteratively to make the DLNet architecture. The basic idea behind the DLNet decoder is the projected gradient descent-based solution of the ML estimator (5).

3.1. Network structure of DLNet

To estimate the transmitted symbol vector ($\hat{\mathbf{x}}$), we are mimicking a projected gradient descent-based solution for the ML optimization of MIMO systems. To estimate transmitted symbol vector $\mathbf{x}$, we formulate a projected gradient descent based expression given as:

$$\begin{aligned}\hat{\mathbf{x}}_{l+1} &= \Pi\left[\hat{\mathbf{x}}_l - \lambda \frac{\partial \|\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}_l\|^2}{\partial x}\right]\\ &= \Pi\left[\hat{\mathbf{x}}_l + 2\lambda\mathbf{H}^T\mathbf{y} - 2\lambda\mathbf{H}^T\mathbf{H}\hat{\mathbf{x}}_l\right]\end{aligned} \tag{6}$$

where, $\hat{\mathbf{x}}_{l+1}$ is the estimate of the transmitted message $\mathbf{x}$ after lth iteration, $\Pi[.]$ represents a nonlinear projection operator. In lth iteration, the step size and the estimate of $\mathbf{x}$ are λ and $\hat{\mathbf{x}}_l$ respectively. The gradient of Eq. (5) which is the objective function for any MIMO decoder with respect to transmitted symbols $\mathbf{x}$ is $2\mathbf{H}^T\mathbf{y} - 2\mathbf{H}^T\mathbf{H}\hat{\mathbf{x}}_l$. Now, the iterations using gradient descent will become $\mathbf{z}_{l+1} = \hat{\mathbf{x}}_l + 2\lambda\mathbf{H}^T\mathbf{y} - 2\lambda\mathbf{H}^T\mathbf{H}\hat{\mathbf{x}}_l$. Where, $\mathbf{z}_{l+1}$ is the noisy estimate of $\mathbf{x}$ after lth iteration. After this step a denoising process is required to remove the noise from $\mathbf{z}_{l+1}$. Here in DLNet, this denoising process is a series of non linear operations used to remove the residual error arisen due to deviation of $\mathbf{z}_{l+1}$ from the true value of $\mathbf{x}$ and the channel noise $\mathbf{n}$. With some loose bound, we assume that this mixture of noise have Gaussian pdf with $\gamma_l \mathbf{I}_{2N_t}$ as covariance matrix. Thus, the best denoiser function for Gaussian noise that minimizes $\mathbb{E}[\|\hat{\mathbf{x}}_l - \mathbf{x}\|_2 |\mathbf{z}_{l+1}]$ [19] is given in Eq. (9).

Fig. 1(a) presents the structure of a single iteration of our proposed DLNet. Here, first we perform some linear operation on $\hat{\mathbf{x}}_l$, $\mathbf{H}^T\mathbf{H}\hat{\mathbf{x}}_l$, and $\mathbf{H}^T\mathbf{y}$ with trainable variables $\mathbf{W}_{1l}$ (weights) and $\mathbf{b}_{1l}$ (biases). After this operation, the resultant estimate is then fed to a denoising function Ψ_G. Mathematically we can write these linear and denoising operation as:

$$\begin{aligned}&\mathbf{z}_{l+1} = \hat{\mathbf{x}}_l + \mathbf{W}_{1l}(\mathbf{H}^T\mathbf{y} + \mathbf{H}^T\mathbf{H}\hat{\mathbf{x}}_l) + \mathbf{b}_{1l}\\ &\hat{\mathbf{x}}_{l+1} = \Psi_G(\mathbf{z}_{l+1}; \gamma_l)\end{aligned} \tag{7}$$

Here the denoising function Ψ_G is a non-linear function $\eta_t : \mathbb{C}^{N_t} \rightarrow \mathbb{C}^{N_t}$ in general, however, the algorithm apply $\beta_t : \mathbb{C} \rightarrow \mathbb{C}$ to each element of z_{t+1}. A natural choice for the denoising function is the minimizer of $\mathbb{E}[\|\hat{\mathbf{x}} - \mathbf{x}\|_2 |\mathbf{z}_t]$, which is given by:

$$\eta_t(\mathbf{z}_t) = \mathbb{E}\left[\mathbf{x}|\mathbf{z}_t\right] \tag{8}$$

Optimal denoiser for Gaussian noise: Several existing MIMO detection schemes assume that the noise at the input of the denoiser $z_{t+1} - x$ has an i.i.d. Gaussian distribution with diagonal covariance matrix $\sigma_t^2 I_{N_t}$. Here, denoiser is a well known Gaussian denoiser [19] given as:

$$\Psi_G(z_{l+1}; \gamma_l) = \frac{1}{Z} \sum_{x_i \in \mathbb{S}} x_i \exp\left(-\frac{\|z_{l+1} - x_i\|^2}{\gamma_l}\right) \tag{9}$$

where $Z = \sum_{x_j \in \mathbb{S}} \exp\left(-\frac{\|z_{l+1} - x_j\|^2}{\gamma_l}\right)$ and γ_l is the estimated variance of the error due to channel noise $\mathbf{n}$ as well as residual error. During each iteration, estimation of γ_l is performed, which requires a series of operations. Fig. 1(b) represents the γ_l estimation steps. Mathematically, we can represent γ_l as:

$$\begin{aligned}\gamma_l = \frac{W_{2l}}{N_R}\Bigg(&\|\mathbf{I} - \mathbf{W}_{1l}\mathbf{H}^T\mathbf{H}\|_F^2 \times RL\left(\frac{\|\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}_t\|_2^2 - 2N_R\sigma_n^2}{\|\mathbf{H}\|_F^2}\right)\\ &+ \frac{\|\mathbf{W}_{1l}\mathbf{H}^T\mathbf{H}\|_F^2}{\|\mathbf{H}\|_F^2}\sigma_n^2\Bigg)\end{aligned} \tag{10}$$

Here, $\mathbf{W}_{1l}$ and $\mathbf{W}_{2l}$ are trainable variables and $RL(x) = max\{0, x\}$ is the rectified linear unit.

Such single iteration does not decode well. To improve the performance we cascade several such iterations to make proposed DLNet. The proposed DLNet consists of 15 such repetitive iterations. The final architecture of DLNet is shown in Fig. 2. This detector takes received signal $\mathbf{y}$, channel matrix $\mathbf{H}$, and $\hat{\mathbf{x}}_l$ as input where $\hat{\mathbf{x}}_l$ is the estimate of $\mathbf{x}$ in the $(l-1)$th layer of this detector. For first iteration we have taken $\mathbf{x}_0 = 0$.

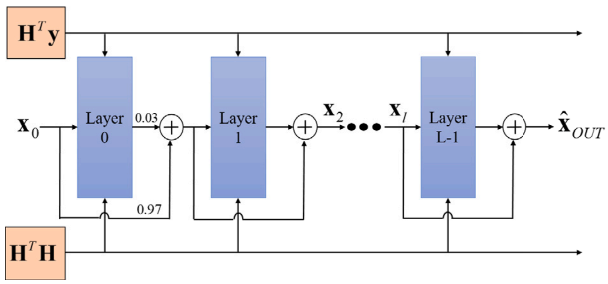


Fig. 2. Neural Network architecture of the DLNet decoder for M-MIMO system.

3.2. Selection of DLNet parameters

The expressions of $\mathbf{z}_{l+1}$ and $\hat{\mathbf{x}}_{l+1}$ are derived from gradient descent algorithm where weights $\mathbf{W}_{1l}$, W_{2l} and bias $\mathbf{b}_{1l}$ were introduced as trainable variables to lift the input to a higher dimension while neuron activation functions were used to introduce non-linearities in the system that are common in DL-networks. The dimension of $\mathbf{z}_{l+1}$ and $\mathbf{W}_{1l}$ are $2N_T \times 1$ and $2N_T \times 2N_T$ respectively whereas the dimension of W_{2l} is 1×1. The single layer of the proposed DLNet detector is represented in Eq. (7). Fifteen such layers are cascaded iteratively to make the proposed DLNet. The number of layers in the proposed DL architecture depends upon the desired accuracy and complexity. Fig. 3(a) shows the SER performance of the proposed DLNet having a different number of layers. This graph shows that the SER performance increases significantly up to 15 layers, and then a small increment is seen beyond 15 layers. With the increasing number of layers, the training variables, unit edges, and thus complexity increase. To make the DLNet decoder more accurate and less complex, we choose the network architecture, which is 15 layers deep. During training, the following parameters of the DLNet detector need to be optimized:

$\{\mathbf{W}_{1l}, \mathbf{b}_{1l}, W_{2l}\}_{l=1}^{l=L}$

where $L = 15$ is the layers in the DLNet detector.

Training is complex for DNN architecture. While training of a DNN converges, a degradation problem was observed. With an increasing number of layers at first, accuracy gets saturated and then degrades rapidly. The over-fitting of the NN is not the cause of the above degradation problem. While being trained, back-propagation of the gradient to previous layers occurs. Due to repeated multiplication, the gradient may become infinitely small. Also, increasing the number of layers results in higher training errors. During training, the objective function of the network may not be optimized due to wrong initialization and many more reasons. [20].

To overcome the above-discussed issues, two modifications have been made while making the network deeper. First, the proposed network is made residual, and second, a loss function that considers the outputs of all of the layers has been adopted. In [21] authors have shown that compared to swallow non-residual networks, residual networks can gain accuracy from the considerably increased depth and are easier to optimize. However, with increasing depth, the ability to backpropagate the gradients through all the layers in an effective manner is a concern [22]. Furthermore, to overcome repeated multiplication, it is required to amplify the gradient that propagated back along with additional regularization. Following this approach of GoogleNet [22], to address these challenges, the loss of all the layers has been taken into account to form the loss function of the DLNet.

For making the DLNet residual, the lth and $(l-1)$th layer output is weighted and averaged. To choose the weight factors of lth and $(l-1)$th layers, we have simulated the SER performance of the proposed DLNet for different weights at different SNR values and presented in Fig. 3(b). This simulation shows that the weight value of 0.97 gives a minimum BER for all SNR. Owing to this simulation, the weight of the current and previous layers are kept as 0.03 and 0.97, respectively. The loss function of the proposed DLNet is calculated as:

$$loss = \sum_{l=1}^{L} \|\mathbf{x} - \hat{\mathbf{x}}_l\|_2^2 \tag{11}$$

The $loss$ used during training is the sum of errors between the estimated transmitted messages at every DLNet layer. Stochastic gradient-descent algorithm [23] with Adam Optimizer [24] is used to train the proposed DLNet decoder for the M-MIMO system. The loss gradients are used to compute the weight updates at every training step. For training, batch processing has been used. In each batch, 1000 channel realizations have been used during each training iteration. In general, a large number of iterations may cause the proposed DLNet to over-fit over the training data. This means that in place of learning, the DLNet memorizes the data, and thus, performance degrades. To determine the iteration size, the BER of the proposed M-MIMO decoder with the increasing number of training iterations has been simulated and shown in Fig. 4. Upon investigating this result, it is found that 10 000 training iterations give optimal performance. For training, intel Xeon 4114 computer with 22 GB NVidia GTX 1080 Ti GPU has been used. The training took 3 min and 5 s for $N_T = 32$ and $N_R = 64$ M-MIMO system and 2 min and 34 s for $N_T = 16$ and $N_R = 64$ M-MIMO system.

4. Simulation results

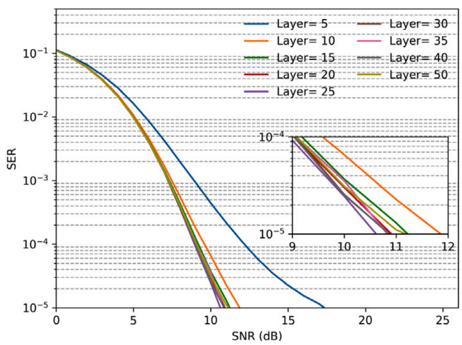
In the M-MIMO perspective, the proposed DLNet has been evaluated for complexity, SER performance, and run-time required over a wide SNRs range. The training phase of any DNN system is very critical. The training efficiency decides its performance. For efficient DNN training, the acceptable SNR range for training data generation is a crucial factor. To find this, normalized validation error (NVE) for various SNR ranges has been observed [25]. From NVE, it is found that neither too high nor too low SNR ranges are suitable for efficient training. owing this observation, the SNR values uniformly distributed from 5 dB to 18 dB, $\mathcal{U}(5, 18)$ dB have been chosen to train the DLNet. In our work, SNR is defined as:

$$SNR(dB) = 10 \log \frac{\|\mathbf{Hx}\|_2^2}{\|\mathbf{n}\|_2^2} \tag{12}$$

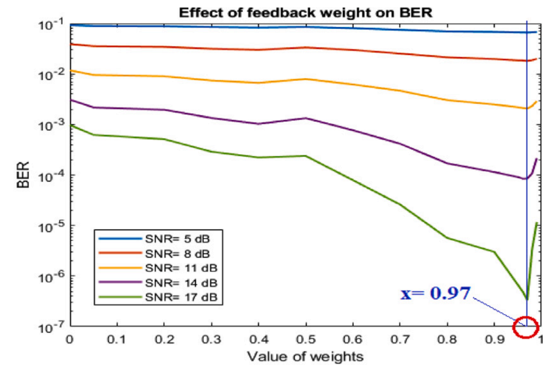
For training the presented DLNet, the transmit–receive (x–y) data pair has been randomly generated for all the transmitters. Each transmitter randomly generates transmit symbol x belonging to any QPSK, 16-QAM, 64-QAM, and 256-QAM modulation schemes. All the transmitters are assumed to transmit using the same modulation scheme. The channel matrices $\mathbf{H}$ are either time-varying where each element is sampled from Rayleigh channel having each column of $\mathbf{H} \sim \mathcal{CN}(0, (1/N_R)\mathbf{I}_{N_R})$ or they are generated via correlated MIMO channel described by the Kronecker model $\mathbf{H} = \theta_R^{0.5}\mathbf{A}\theta_T^{0.5}$. Where the spatial correlation matrix $\theta_T^{0.5}$ and $\theta_R^{0.5}$ at the transmitter and receiver are generated using exponential correlation model [26] having correlation coefficient $\rho = 0.5$ and matrices $\mathbf{A}$ as Rayleigh channel coefficients.

During training, the channel $\mathbf{H}_c$ is generated independently for each batch in each iteration. During the simulations, various performance measures are simulated and compared with the following existing M-MIMO decoding schemes:

1. **MMSE:** A linear decoder uses the pseudo inverse of SNR regularized channels to find the nearest point on the constellation set.
2. **MMNet_iid:** A DL-based iterative algorithm that has very few trainable variables for i.i.d. Gaussian channels. However, the network (MMNet) needs to train online on every channel realization in spatially correlated channels. In MMNet, the online training requires hardware with enormous computational complexity as well as introduces latency, thus difficult to use in practical systems [18].



(a) SER performance for different number of layers in DL-Net (Feedback weight=0.97, N_T =16, N_R = 64, Modulation= QPSK, Training Iteration=10,000).



(b) Feedback Weights and BER for DLNet (N_T =32, N_R =64, Modulation= 16-QAM, Training Iteration=10000, Number ofLayers=15)

Fig. 3. Selection of the DLNet parameters.

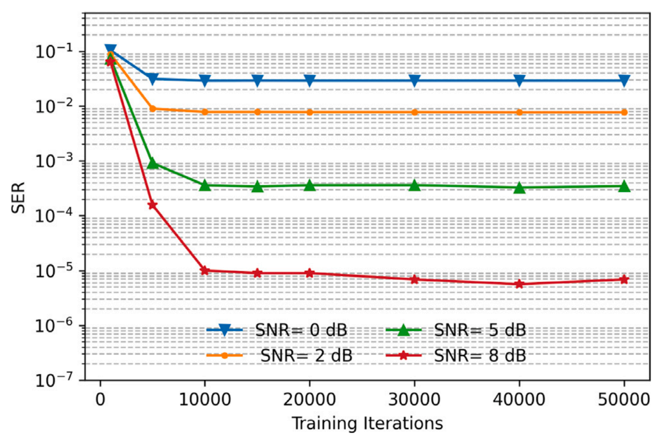


Fig. 4. SER performance for different number of training iterations (N_T = 16, N_R = 64, Modulation = QPSK, Number of Layers = 15).

3. **OAMP-Net** This is based on famous OAMP-based architecture [17]. It is also a DL-based iterative approach, but it makes strong assumptions about the channel. The OAMP-Net is based on the MMSE model and needs to calculate the channel inverse in each iteration, which is too complex for large MIMO channels and thus takes more time during training and detection.
4. **SDR:** Semi-definite relaxation is an optimization technique. Several researchers have used SDR to make M-MIMO decoders [16].
5. **AMP:** Approximate message passing algorithm based MIMO decoder [27].
6. **DetNet:** This has a data-driven DNN-based architecture. It also uses gradient-descent to make M-MIMO decoder [14].
7. **FS-Net:** A fast converging, sparsely connected deep learning-based network which is made by applying enhancements in DetNet [28].

Table 1 compares the various parameters of DetNet, OAMP-Net, and the proposed DLNet decoder.

4.1. SER performance

Figs. 5 and 6, 7 present the SER performance of the proposed DLNet decoder for two different channel models Rayleigh channel & correlated MIMO channels respectively. Fig. 5 presents the SER performance of the proposed DLNet for the Rayleigh channel model with two antenna configurations (Tx = 16, Rx = 64 and Tx = 32, Rx = 64) and different modulation schemes. The figure also compares the proposed DLNet with other decoders such as MMSE, AMP, SDR, OAMP-Net, MMNet_iid, DetNet, and FS-Net. From this figure, the following observations are made:

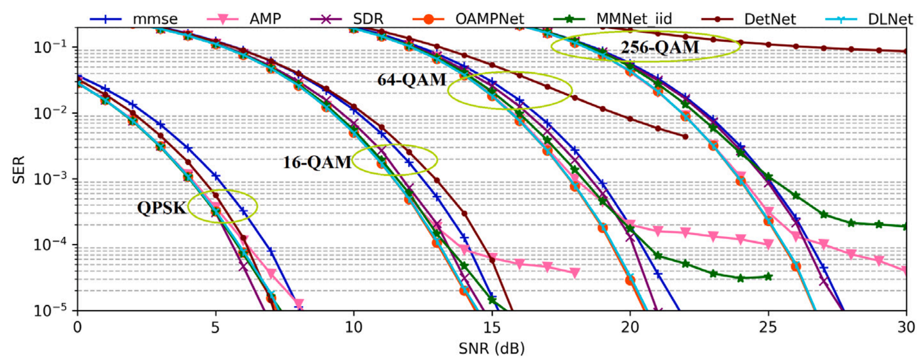
1. If we double the number of transmit antennas, there is more interference, and thus SNR required to achieve a particular SER will increase by 3–4 dB.
2. Performance of DetNet severely degrades on higher modulation order, especially for 64-QAM and higher.
3. At higher SNRs and higher modulation order (16-QAM and higher), the performance of MMNet_iid saturates.
4. AMP suffers from robustness issues, is unstable, and saturates at higher SNRs for all modulation schemes.
5. SDR performs best for QPSK, but its gap with MMSE decreases with increasing modulation order.
6. DLNet and OAMP-Net perform similarly throughout the SNR range for all modulation schemes and 16 transmit antennas. While doing timing and complexity analysis, we have found that OAMPNet takes more time for training and testing and requires more complex hardware to implement than DLNet.
7. DLNet performs superior to OAMP-Net at higher modulation (64-QAM and 256-QAM) for 32 transmit antennas.
8. DLNet performance is similar to SDR for QPSK, but for higher modulation schemes, DLNet outperforms SDR.

Figs. 6 & 7 show SER performance of the proposed DLNet for correlated channel model For correlated channels, we consider MMNet_iid, OAMP-Net, and MMSE only. SDR and DetNet training are highly unstable in the case of correlated channels. Also, the AMP decoder has not been designed for correlated channels and poorly performs on them. From this figure, the following observations are made:

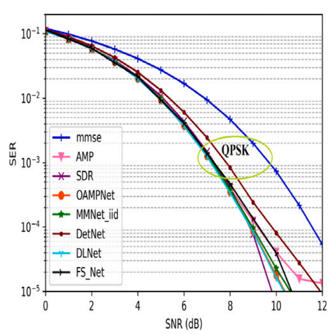
1. If we double the number of transmit antennas, nearly 4 dB increase in SNR is required to achieve the same SER. Correlated channels require 1.5 – 2 dB increase in SNR to match the performance with Rayleigh channels.
2. MMNet_iid suffers from robustness issues for correlated channels and performs poorer than MMSE for higher modulation orders (16-QAM and higher)throughout the SNR range.
3. The performance of DLNet and OAMP-Net is similar with 16 transmit antennas, whereas DLNet performs better than OAMP-Net with 32 transmit antennas for all modulation orders.
4. With the increase in the number of transmit antennas, DLNet outperforms OAMP-Net.
5. DLNet performs better than MMSE for all modulation orders, but its gap with MMSE decreases with increased modulation order.

Table 1
DetNet, OAMP-Net and proposed DLNet comparison.

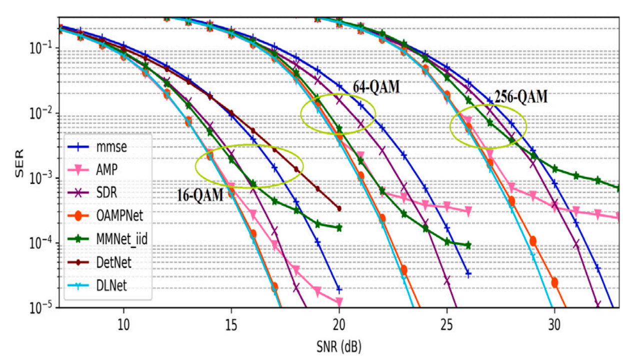
Parameters	DetNet	OAMP-Net	DLNet
Input shape	$5N_T \times 1$	$2N_T \times 1$	$2N_T \times 1$
Hidden layer neurons	$8N_T$	–	$2N_T$
Iterative layers	$3N_T$	10	15
Skip connections	Yes	No	Yes
Loss function	$l(\mathbf{x};\hat{\mathbf{x}}_\theta(\mathbf{H},\mathbf{y})) = \sum_{k=1}^{\mathbf{L}} \log(k)\frac{\|\mathbf{x}-\hat{\mathbf{x}}_k\|^2}{\|\mathbf{x}-\tilde{\mathbf{x}}\|^2}$ $\tilde{\mathbf{x}} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{y}$	$loss = \sum_{l=1}^{L} \|\mathbf{x} - \hat{x}_l\|^2$	$loss = \sum_{l=1}^{L} \|\mathbf{x} - \hat{x}_l\|^2$
Supported channel	Real and complex i.i.d. Gaussian channels Only	Real and complex i.i.d. Gaussian and Correlated Channels	Real and complex i.i.d. Gaussian and Correlated Channels
Matrix inverse	Not required	Required	Not required
Trainable Parameters	3 513 650 ($N_T = 32$, $N_R = 64$) 886 450 ($N_T = 16$ and $N_R = 64$)	20 ($N_T = 32$, $N_R = 64$) 20 ($N_T = 16$ and $N_R = 64$)	42 250 ($N_T = 32$, $N_R = 64$) 10 890 ($N_T = 16$ and $N_R = 64$)



(a) $N_T = 16$, $N_R = 64$



(b) $N_T = 32$, $N_R = 64$



(c) $N_T = 32$, $N_R = 64$

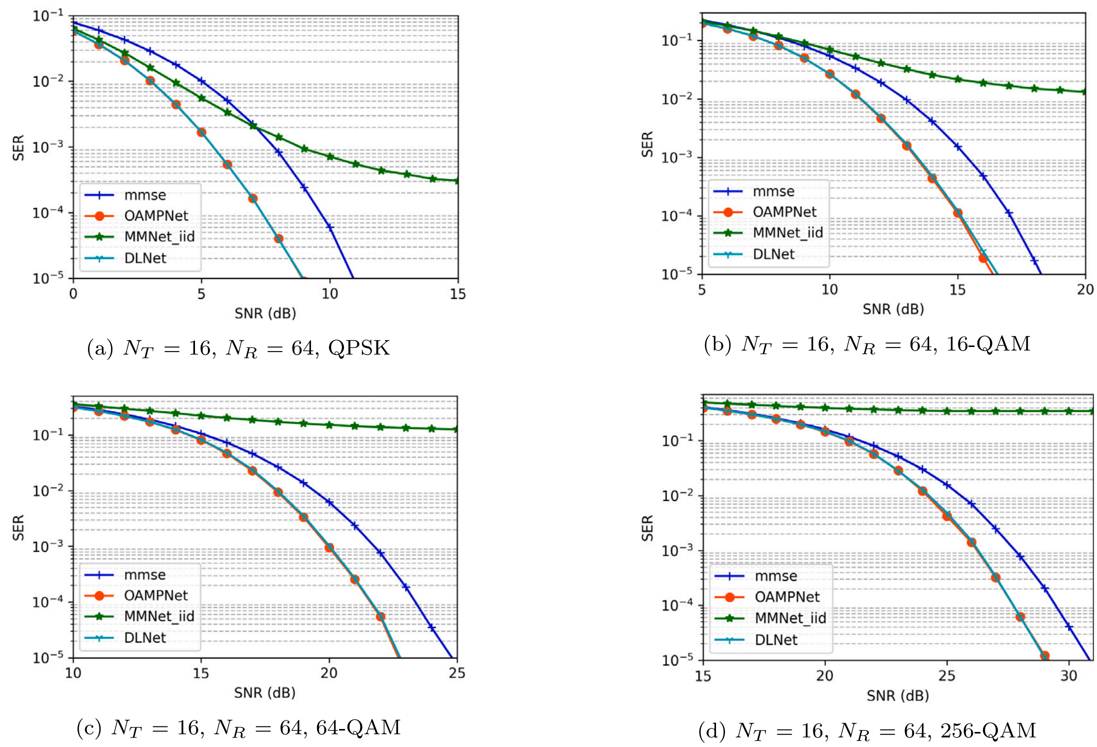
Fig. 5. Accuracy performance (SER vs. SNR) graph of the proposed DLNet decoder on Rayleigh channel model. (Training Iteration = 10 000, Number of Layers = 15).

NN needs to be trained as per the required channel model. A large dataset is required for efficient training, which takes a long time to train the network. Once trained, the network can be used for decoding. Retraining is required when there is a change in the number of antennas, operating environment (indoor, outdoor, static, mobile, etc.), operating frequency band, and other parameters that can impact the channel model. Once the channel model changes, NN needs to be retrained. The channel characteristics may be time-varying while working with the same channel model; DLNet need not be trained again.

4.2. Timing and complexity analysis

Timing and complexity are two main aspects of any signal decoder. In 5G, a message frame is needed to be decoded within a short time with minimum hardware complexity. SDR-based decoder, OAMPNet, MMNet_iid, DetNet, and proposed DLNet decoder have been simulated for timing analysis on a similar hardware platform. Fig. 8 represents the time required by various decoders to decode M-MIMO signals over a wide SNR range. For the shake of clarity, The y axis represents the normalized time of the decoders with respect to DLNet. This graph illustrates that the proposed DLNet decoder decodes at least 3 times faster than DetNet, on an average 90 times faster than SDR, and 14 times faster than OAMPNet for Rayleigh channels. At the same time, it is 11 times faster than OAMPNet for correlated channels. For both correlated and Rayleigh channels, MMNet_iid is 1.20 times faster than DLNet, but its SER performance is unsatisfactory than DLNet.

After timing analysis, the computational complexity of the M-MIMO detectors has been estimated. For that, their run time requirements were compared. Many factors such as hardware platforms and implementation details make the complexity comparison non-trivial. To ensure fairness among all, the same machine (See Section 3 for machine details) with python 3.7 environments, Tensorflow, and NumPy packages have been used to test them. While considering the run time of a DNN, the batch size plays an important role. DL-based decoders decode over an entire batch of data to increase the decoding speed, while conventional decoders such as SDR cannot work over the whole batch. AMP algorithm can also compute over an entire batch of signals at once. However, the batch introduced speed improvement is highly



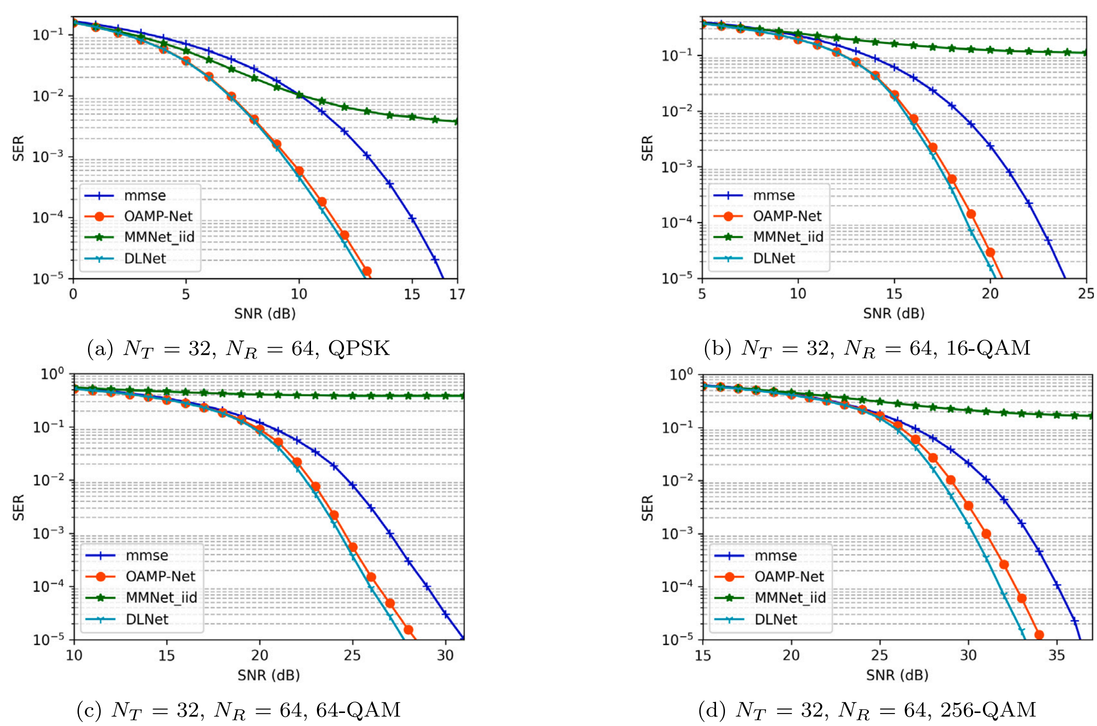
(a) $N_T = 16$, $N_R = 64$, QPSK

(b) $N_T = 16$, $N_R = 64$, 16-QAM

(c) $N_T = 16$, $N_R = 64$, 64-QAM

(d) $N_T = 16$, $N_R = 64$, 256-QAM

Fig. 6. Accuracy performance (BER vs. SNR)of DLNet decoder on Correlated channel model. (Training Iteration = 10 000, Number of Layers = 15).



(a) $N_T = 32$, $N_R = 64$, QPSK

(b) $N_T = 32$, $N_R = 64$, 16-QAM

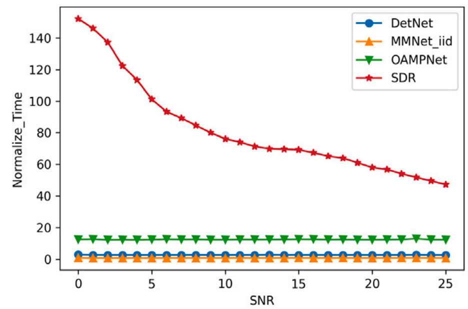
(c) $N_T = 32$, $N_R = 64$, 64-QAM

(d) $N_T = 32$, $N_R = 64$, 256-QAM

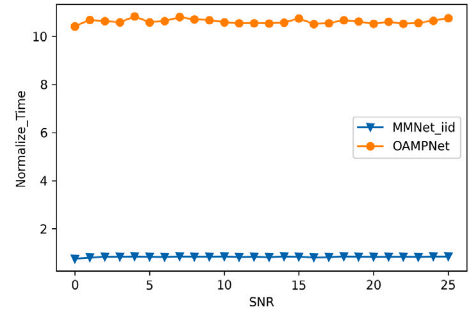
Fig. 7. Accuracy performance (SER vs. SNR) DLNet decoder on Correlated channel model.

dependent on the platform used (FPGA/GPU/CPU, etc.). Therefore, for completeness, 1000 as batch size has been chosen to present the results.

Table 2 presents the complexity comparison between DLNet and DetNet detectors. This table shows that the proposed DLNet has 82 times less trainable parameters (unit edges) than DetNet for the same M-MIMO configurations. Though the number of training variables is significantly less for OAMP-Net (20 variables), the complexity of OAMP-Net is dominated by matrix inverse operation in each iteration [17]. The fewer trainable variables (unit edges) and less complex processes make the proposed DLNet faster while training and computationally inexpensive i,e. it requires simple hardware to implement. The time required by DetNet and OAMP-Net for being trained is approx 28 and



(a) Rayleigh channels

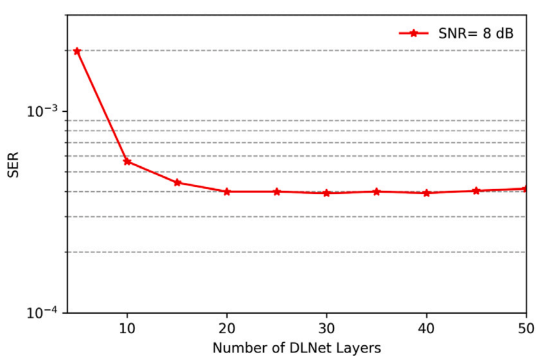


(b) Correlated channel

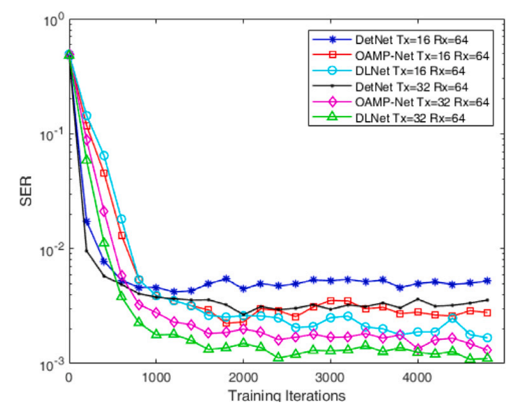
Fig. 8. Normalize Time (Time/Time taken by DLNet Decoder) vs. SNR (QPSK, $N_T = 32$ $N_R = 64$).

Table 2
Comparison with DetNet, DLNet and ML decoder.

N_T and N_R	Network	Unit edges	Training time	Deco ding time	Complexity
$N_T = 32$, $N_R = 64$	DLNet	42 250	3 min 5 s	15.29 μs	$4N_T^3 + 9N_T^2 - 5N_T$
	DetNet	3 513 650	1 h 23 min 45 s	45.27 μs	$784N_T^3 - 120N_T^2$
	OAMPNet	20	27 min 40 s	198 μs	ON_T^3 (dominated by Matrix Inverse)
	MMNet_iid	650	2 min 19 s	13.20 μs	Two magnitude lower than OAMPNet
	ML decoder	–	–	12.65 ms	Exponential with N_T
$N_T = 16$, $N_R = 64$	DLNet	10 890	2 min 34 s	12 μs	$4N_T^3 + 9N_T^2 - 5N_T$
	DetNet	886 450	1 h 12 min 11 s	25.75 μs	$784N_T^3 - 120N_T^2$
	OAMPNet	20	24 min 37 s	194 μs	ON_T^3 (dominated by Matrix Inverse)
	MMNet_iid	330	2 min 18 s	8.5 μs	Two magnitude lower than OAMPNet
	ML decoder	–	–	8.54 ms	Exponential with N_T



(a) SER Vs number of DLNet layers at SNR=8 dB, Modulation= QPSK.



(b) Network convergence speed.

Fig. 9. Accuracy-complexity tradeoff and Network convergence speed.

9 times higher than that of DLNet, respectively, whereas it is approx 3 times faster than DetNet and 14 times faster than OAMP-Net during decoding.

4.3. Accuracy and complexity tradeoff

One exciting feature of DLNet is that the complexity and accuracy trade-off can be achieved during run time by changing the number of layers. Fig. 9(a) represents the relationship between SER and the increasing number of layers of the proposed decoder at 8 dB SNR for QPSK modulation. The figure shows that the SER of the DLNet improves with an increasing number of layers (up to 15), then further minimal increment is seen. However, increasing DLNet layers results in more training variables, unit edges, and thus higher decoding complexity. To keep the DLNet computationally inexpensive, the output of any layer could be chosen as final on the cost of accuracy, e.g., 10th layer. This leads to fewer training variables in the decoder, and thus complexity also reduces but on the other hand, SER performance also degrades.

Fig. 9(b) shows the convergence speed of the DetNet, OAMP-Net, and the DLNet decoder during the training phase. These three networks were trained on uniformly distributed random SNRs varying from 5 dB to 18 dB as $\mathcal{U}(5, 18)$. Because of the random nature of SNR values, SER values will converge but never becomes minimum regardless of the number of iterations. However, the proposed DLNet M-MIMO decoder converges faster to a minimum SER value than DetNet for a given number of iterations.

5. Conclusion

This paper proposed a DL-based M-MIMO signal decoder DLNet. We have simulated its SER performance, implementation complexity, and decoding time requirement over frequency flat Rayleigh fading and correlated M-MIMO channels. The proposed DLNet is a 15 layers DNN based on the projected gradient descent-based solution of the ML decoder. The proposed DLNet provides near-optimal SER performance without knowing the SNR level and is computationally inexpensive. The training of DLNet was done on the entire channel distribution, which makes it robust and implementable in a system with changing channel values where the receiver perfectly knows the channel state. After doing single-time training, the proposed DLNet decoder can decode signals accurately over various channels. The data obtained from real-world channels and real hardware could be used to train the DLNet, and thus it can be optimized for specific practical M-MIMO environments. Its iterative layering structure enables a flexible complexity-accuracy trade-off required for many modern communication systems.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The authors acknowledge the financial assistance from the UGC, New Delhi, India, in the form of the NET-Senior Research Fellowship (UGC NET-SRF), which supports the work.

References

[1] Larsson EG, Edfors O, Tufvesson F, Marzetta TL. Massive MIMO for next generation wireless systems. IEEE Commun Mag 2014;52(2):186–95. http://dx.doi.org/10.1109/MCOM.2014.6736761.
[2] Ngo HQ, Larsson EG, Marzetta TL. Energy and spectral efficiency of very large multiuser MIMO systems. IEEE Trans Commun 2013;61(4):1436–49. http://dx.doi.org/10.1109/TCOMM.2013.020413.110848.
[3] Liu Y, Chai J, Zhang Y, Liu Z, Jin M, Qiu T. Low-complexity neural network based DOA estimation for wideband signals in massive MIMO systems. AEU - Int J Electron Commun 2021;138:153853. http://dx.doi.org/10.1016/j.aeue.2021.153853.
[4] Samuel N, Diskin T, Wiesel A. Learning to detect. IEEE Trans Signal Process 2019;67(10):2554–64. http://dx.doi.org/10.1109/TSP.2019.2899805.
[5] Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. Neural Netw 1989;2(5):359–66. http://dx.doi.org/10.1016/0893-6080(89)90020-8.
[6] O'Shea TJ, Hoydis J. An introduction to machine learning communications systems. 2017, ArXiv abs/1702.00832.
[7] Guo Z, Nilsson P. Algorithm and implementation of the K-best sphere decoding for MIMO detection. IEEE J Sel Areas Commun 2006;24(3):491–503. http://dx.doi.org/10.1109/JSAC.2005.862402.
[8] Karthikeyan M, Saraswady D. Reduced complexity sphere decoding using probabilistic threshold based Schnorr–Euchner enumeration. AEU - Int J Electron Commun 2016;70(4):449–55. http://dx.doi.org/10.1016/j.aeue.2016.01.007.
[9] Minka TP. A family of algorithms for approximate Bayesian inference (Ph.D. Dissertation), Cambridge, MA, USA: Dept. Elect. Eng. Comput. Sci., MIT; 2001.
[10] Donoho DL, Maleki A, Montanari A. Message passing algorithms for compressed sensing: I. motivation and construction. In: 2010 IEEE information theory workshop on information theory (ITW 2010, Cairo). 2010, p. 1–5. http://dx.doi.org/10.1109/ITWKSPS.2010.5503193.
[11] Carrera DF, Vargas-Rosales C, Azurdia-Meza CA, Morocho-Yaguana M. Extreme learning machine detector for millimeter-wave massive MIMO systems. AEU - Int J Electron Commun 2021;138:153875. http://dx.doi.org/10.1016/j.aeue.2021.153875.
[12] Amiri M, Akhavan A. An iterative detector based on sparse bayesian error recovery for uplink large-scale MIMO systems. AEU - Int J Electron Commun 2021;138:153848. http://dx.doi.org/10.1016/j.aeue.2021.153848.
[13] Chen Q, Zhang S, Xu S, Cao S. Efficient MIMO detection with imperfect channel knowledge - A deep learning approach. In: 2019 IEEE wireless communications and networking conference (WCNC). 2019, p. 1–6. http://dx.doi.org/10.1109/WCNC.2019.8885582.
[14] Samuel N, Diskin T, Wiesel A. Deep MIMO detection. In: 2017 IEEE 18th international workshop on signal processing advances in wireless communications (SPAWC). 2017, p. 1–5. http://dx.doi.org/10.1109/SPAWC.2017.8227772.
[15] Ma J, Ping L. Orthogonal AMP for compressed sensing with unitarily-invariant matrices. In: 2016 IEEE information theory workshop (ITW). 2016, p. 280–4. http://dx.doi.org/10.1109/ITW.2016.7606840.
[16] Mobasher A, Taherzadeh M, Sotirov R, Khandani A. A near maximum likelihood decoding algorithm for MIMO systems based on semi-definite programming. In: Proceedings. International symposium on information theory, 2005. ISIT 2005. 2005, p. 1686–90. http://dx.doi.org/10.1109/ISIT.2005.1523632.
[17] He H, Wen C-K, Jin S, Li GY. A model-driven deep learning network for MIMO detection. 2018, http://dx.doi.org/10.48550/arXiv.1809.09336, arXiv:1809.09336v1 [cs.IT].
[18] Khani M, Alizadeh M, Hoydis J, Fleming P. Adaptive neural signal detection for massive MIMO. IEEE Trans Wireless Commun 2020;19(8):5635–48. http://dx.doi.org/10.1109/TWC.2020.2996144.
[19] Jeon C, Ghods R, Maleki A, Studer C. Optimal data detection in large MIMO. 2018, http://dx.doi.org/10.48550/arXiv.1811.01917, arXiv:1811.01917v1 [cs.IT].
[20] Xavier Glorot YB. Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics, Vol. 9. 2010, p. 249–56.
[21] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: 2016 IEEE conference on computer vision and pattern recognition (CVPR). 2016, p. 770–8. http://dx.doi.org/10.1109/CVPR.2016.90.
[22] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. In: 2015 IEEE conference on computer vision and pattern recognition (CVPR). 2015, p. 1–9. http://dx.doi.org/10.1109/CVPR.2015.7298594.
[23] Bottou L. Large-scale machine learning with stochastic gradient descent. In: Proceedings of COMPSTAT'2010. 2010, http://dx.doi.org/10.1007/978-3-7908-2604-3_16.
[24] Diederik P. Kingma JB. Adam: A method for stochastic optimization. 2014, arXiv:1412.6980v9.
[25] Gruber T, Cammerer S, Hoydis J, Brink St. On deep learning-based channel decoding. In: 2017 51st annual conference on information sciences and systems (CISS). 2017, p. 1–6. http://dx.doi.org/10.1109/CISS.2017.7926071.
[26] Loyka S. Channel capacity of MIMO architecture using the exponential correlation matrix. IEEE Commun Lett 2001;5(9):369–71. http://dx.doi.org/10.1109/4234.951380.
[27] Jeon C, Ghods R, Maleki A, Studer C. Optimality of large MIMO detection via approximate message passing. In: 2015 IEEE international symposium on information theory (ISIT). 2015, p. 1227–31. http://dx.doi.org/10.1109/ISIT.2015.7282651.
[28] Nguyen NT, Lee K. Deep learning-aided tabu search detection for large MIMO systems. IEEE Trans Wireless Commun 2020;19(6):4262–75. http://dx.doi.org/10.1109/TWC.2020.2981919.