

A Fast and Effective MIMO Algorithm Using CLR-RNN for Hybrid MDM and WDM Optical Communication System

Danni Zhang¹, Zhongwei Tan¹, Xinyuan Ma¹, Shun Lu¹, Wenhua Ren¹, and Fengping Yan¹

Abstract—There is an increasing demand for data with the development of the world, and various fiber optic multiplexing techniques have become an important research direction to improve transmission capacity. However, the transmitted signals are subject to great interference due to mode coupling and mode dispersion, which require multiple-input multiple-output (MIMO) digital signal processing techniques to restore the quality of the transmitted signals. In this paper, a novel MIMO detector is designed using an adaptive learning recurrent neural network and successfully implemented in a mixed wavelength-division-mode-division-multiplexing (WDM-MDM) optical transmission system, and its performance is compared with that of the forced-zero detector and the minimum-mean-square-error detector. The results show that the introduction of an adaptive machine learning model in MIMO detection for WDM-MDM optical transmission systems can significantly improve the quality of the transmitted signals and achieve better performance than other MIMO detection algorithms while maintaining a faster computational speed and a lower number of parameters.

Index Terms—MIMO, WDM, SDM, communication system, machine learning.

I. INTRODUCTION

THE limited bandwidth in optical communication systems is being rapidly consumed as the demand for data flow in people's lives continues to grow. Moreover, since single-mode fiber-based optical communication systems have never been able to break through their inherent nonlinear Shannon limits and reach bottlenecks, the expected capacity growth rate of single-mode fiber transmission systems is still lower than the growth rate of Internet traffic even though the single-mode fiber technology has been significantly improved [1]. In recent

decades, the system capacity has increased by orders of magnitude whenever a new parameter is added as a reuse dimension. However, the utilization of the frequency/wavelength [2], time [3], phase [4], and polarization dimensions [5] are currently approaching their limits. As the only underutilized dimension, the spatial dimension has become considered the most promising multiplexing method. [6] When space-division multiplexing (SDM) and wave-division multiplexing (WDM) are combined, the transmission capacity of existing optical communication systems can be greatly expanded.

SDM has three types of transmission fibers: multi-core fiber (MCF) [7], [8], few-mode fiber (FMF) or multi-mode fiber (MMF) [9], [10], [11], and few-mode multi-core fiber (FM-MCF) [12]. For SDM systems, FMF supports several modes between SMF and MMF and can transmit over longer distances than MMF. Meanwhile, FMF is easier to fabricate fiber and related optical devices compared to MCF and FM-MCF. Therefore, FMF-based mode division multiplexing (MDM) technology has greater advantages. Under ideal conditions, the FMF channels of each mode are independent of each other without crosstalk, which can realize a multiplier increase in system transmission capacity. However, problems such as bending and twisting can also occur during practical application, resulting in the orthogonality of modes in the fiber being destroyed, causing crosstalk between modes such as mode coupling, mode group delay, and other crosstalk. In addition, loss, noise, and other damage can also occur due to defects in the manufacturing process of optical fibers or optical devices [13]. The superposition of various types of impairing factors makes the channel of the FMF MDM system complex and requires the use of multiple-input multiple-output (MIMO) signal processing techniques at the receiver side to eliminate crosstalk and recover damage to achieve more optimized transmission. Besides that, because of the variations of the environments, like the temperature or the stress of fiber link, the mode dispersion and mode coupling may change from time to time. Therefore, the MIMO algorithms may need to be update consequently, which requires to reduce the complexity and improve the speed of algorithms.

Machine learning (ML) algorithms have evolved dramatically in recent years as hardware computing power has increased and specialized datasets have expanded [14]. ML is capable of systematically mining valuable information from flow data and automatically discovering correlations that would be too complex

Manuscript received 25 February 2024; revised 22 March 2024; accepted 24 March 2024. Date of publication 27 March 2024; date of current version 18 April 2024. This work was supported in part by the National Key Research and Development Program of China under Contract 2021YFB28009 and in part by the National Natural Science Foundation of China under Contract 61875008. (Corresponding author: Zhongwei Tan.)

Danni Zhang, Zhongwei Tan, Shun Lu, Wenhua Ren, and Fengping Yan are with the Institute of Lightwave Technology, Key Lab of All Optical Network and Advanced Telecommunication Network of EMC and the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China (e-mail: dannizhang@bjtu.edu.cn; zhwtan@bjtu.edu.cn; shunlu@bjtu.edu.cn; whren@bjtu.edu.cn; fpyan@bjtu.edu.cn).

Xinyuan Ma is with the National Engineering Lab for Mobile Network Technologies, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: maxinyuan@bupt.edu.cn).

Digital Object Identifier 10.1109/JPHOT.2024.3382596

for a human expert to extract. The principles of machine learning are distinct from traditional algorithms in that it primarily uses computational methods to learn information directly from data, rather than relying on predetermined equations for modeling, as various types of traditional algorithms do. Its networks involve a great deal of nonlinear computation, which makes it much more versatile. MIMO detection techniques based on ML neural networks are already available and successfully implemented in MDM optical transmission systems [15], [16]. However, the performance of traditional shallow neural networks may not improve even if more data is provided, instead they may create more problems such as local optimum or overfitting. Recurrent Neural Network (RNN) is a kind of neural network specialized in processing sequential information. RNN can be extended to longer sequences and it was created to address the limitations of traditional neural networks in processing sequence information [17], [18].

In this paper, a MIMO detection algorithm based on a RNN is proposed and successfully implemented in a simulated WDM-MDM optical transmission system with five wavelengths and three modes. In the system of this paper, we use a step FMF and perform MIMO processing on the signal. For MIMO detection, we designed and trained a supervised adaptive cycle learning rate RNN (CLR-RNN) in this paper. In this method, 15 channels of Quadrature Phase Shift Keying (QPSK) signals are transmitted over WDM-MDM optical transmission and successfully detected with the help of CLR-RNN. The results of the study show that this method is a highly improved algorithm in terms of complexity and detection, and the accuracy of the training can reach 100% with a bit error rate (BER) of 0.

II. FUNDAMENTAL PRINCIPLES

A. MIMO Received Model

When FMFs are used for long-distance transmission, problems with fiber materials and production processes cause coupling, dispersion, and loss between FMF modes that are supposed to be orthogonal to each other, resulting in significant degradation of the received signal quality. To recover the transmitted signal as much as possible at the receiver side, it is necessary to select an appropriate MIMO-DSP technique to recover the transmitted signal in the transmission system. The following section focuses on the WDM-MDM-MIMO system model and channel matrix parameters, and the system model is shown below in Fig. 1:

The WDM-MDM optical communication system in Fig. 1 contains N wavelengths and M modes. In the transmission process, the transmitter sends N optical signals with different wavelengths through a continuous wave (CW) laser, and modulates the signals into the signals required for the test through an optical modulator, and the modulated optical signals first pass through a Wavelength Selective Switch (WSS) to create different wavelengths. After that, the mode multiplexer multiplexes the signals of different modes into one optical fiber. In this paper, we choose FMF as the transmission medium, and then it goes through a mode division decomposition multiplexer and WSS for demultiplexing. Finally, the signal is received by the coherent

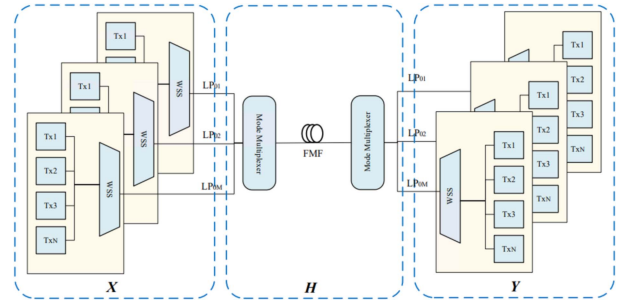


Fig. 1. Schematic diagram of WDM-MDM optical communication system. The multiplexing structure of WDM and MDM is represented in it.

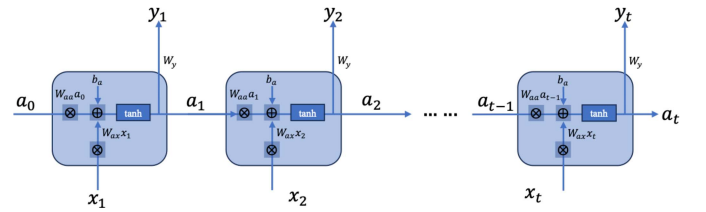


Fig. 2. RNN model. a_t is state vector, x_t is the input vector and y_t is the output. W represents the weight matrix.

optical detector at the receiving end, which normally processes the received signal to restore it to the initial signal. The equation of this MIMO system can be expressed as [19]:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{b} \quad (1)$$

where, \mathbf{b} is Gaussian white noise. To eliminate the interfering signals from other transmitting antennas, the MIMO-DSP technique should be used at the receiver side, and it is particularly important to determine the channel matrix \mathbf{H} .

B. Recurrent Neural Network and Cyclical Learning Rate

After determining the channel matrix \mathbf{H} for MDM-MIMO, there is a need to formulate the problem of MIMO detection in a machine-learning framework. Recurrent neural networks (RNNs) are a type of learning model with internal memory that enables them to capture sequential dependencies. Unlike traditional neural networks where inputs are independent of each other, RNNs consider the time sequence of inputs, making them suitable for tasks involving sequential information. By using recurrences, RNNs apply the same operation to each element in the sequence, with the current computation depending on the current input and previous computations.

RNNs are designed for modeling sequential data where there is a serial correlation between samples. In each time step, they produce outputs through recurrent connections between the hidden layer units, as shown in Fig. 2. The input vector x_t and the previous neuron's state vector a_{t-1} generate the current state vector a_t after a matrix transformation W :

$$\mathbf{a}_t = \mathbf{f}(\mathbf{W}_a [\mathbf{a}_{t-1}, \mathbf{x}_t] + \mathbf{b}_a) \quad (2)$$

where, W is the weight matrix, \mathbf{b} is the bias vector, \mathbf{f} is an activation function through which each element of the input

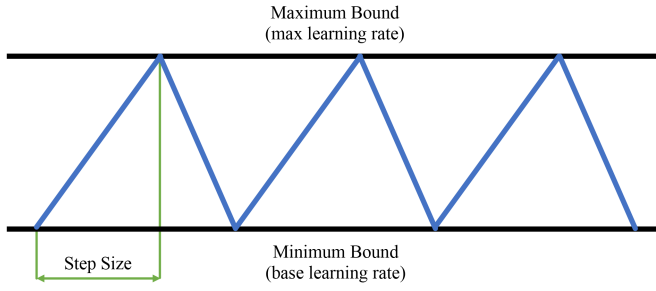


Fig. 3. Cyclical learning rate schematic. The learning rate increases and decreases according to the blue line, with two step lengths forming a cycle.

vector needs to pass. a_t then passes through a matrix change to generate the current prediction $\hat{y}(t)$ in (3):

$$\hat{y}(t) = f(W_y a_t + b_y). \quad (3)$$

The output $y(t)$ of the RNN can be obtained by iterating through the iteration chain of the two equations, as shown in (4):

$$y(t) = f(W_y f(W_a a_t + W_y f(W_a a_{t-1} + W_y f(W_a a_{t-2} + \dots))) + b_y). \quad (4)$$

It can be seen that the output at the current point in time contains historical information, which indicates that RNN saves historical information. Unlike traditional neural networks that use different parameters at each layer, RNNs share the same parameters. This reflects the fact that the same task is performed at each step, just using different inputs, which greatly reduces the total number of parameters for training.

The activation function introduces nonlinear factors into the neural network, through which the neural network can fit various curves. In this system the tanh function is used in the training process with the equation $f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$. [20] From its equation, it can be concluded that the use of the tanh activation function can limit the output values between -1 and 1. Other activation functions for RNN include the ReLU function. [21] ReLU is a non-negative output, only the parts above or equal to 0 will be retained in the recursive multiplication. Instead, the tanh, an activation function that is symmetric with a center of 0, can decide what information to keep and what to remove, which is more compatible with the data form of the channel matrix H in this system.

During the training process, the learning rate is probably the most important parameter. The learning rate determines how much the model is adjusted in each parameter update step. There are many ways to adaptively adjust the learning rate, such as Step Learning Rate and Cosine Annealing Learning Rate. After our testing and research, we decided to use Cyclical Learning Rate (CLR) in this system. CLR is a method that dynamically adjusts the learning rate based on the state of the neural network training, and works with the neural network to reach the best-fit point faster during training. [22], [23] As shown in Fig. 3 below, there are three variables in CLR, which are base learning rate, max learning rate, and step size. The CLR needs to set a minimum

bound (base learning rate, $base_lr$) and maximum bound (max learning rate, max_lr) to achieve the best fit of the network by adjusting the learning rate to vary in the maximum and minimum bound instead of simply decreasing the learning rate. In a cycle, the learning rate step size needs to be increased and decreased similar to the uphill and downhill slopes, and two-step sizes form a cycle. The step size is set based on the iterations required for network training.

The cyclic variation strategy enables the model to avoid local minima and saddle points that are encountered during the training process. Saddle points are more of a hindrance to convergence than local minima. If a saddle point happens to occur at a clever equilibrium point, a small learning rate usually does not produce a large enough gradient change to make it jump over the point, and even if it does, it takes a long time to do so. This is where a periodically high learning rate can be useful, by jumping over the saddle point faster. What is more, assuming that the CLR must fall between the minimum bound and maximum bound, periodic tuning is equivalent to constantly iterating to find the optimal solution. The CLR-RNN proposed in this paper utilizes the CLR approach which plays a vital role in MIMO detection in WDM-MDM optical transmission systems.

III. SYSTEM ESTABLISHMENT

In this paper, we design a mixed-multiplexed fiber optic transmission system with five wavelengths, three modes, and 15 channels as shown in Fig. 4, which is a 50 Gbps transmission system via Quadrature Phase Shift Keying (QPSK) modulation. In the simulation experiment, five arrays of CW lasers with emission wavelengths of 193.1–193.5 THz are selected at the transmitter side, and each array contains three lasers corresponding to three different fiber modes. Each laser has a linewidth of 0.1 MHz and a power of 10 dBm. Afterward, QPSK signal is performed by IQ modulation (IQM), and using WSS for wavelength selection. Thereafter, the transmission is split into two paths with polarization orthogonal to each other by a 3 dB delay, one of which is decorrelated by a delay line and multiplexed with the other through a Polarization Beam Combiner (PBC). Signals are amplified by the erbium-doped fiber amplifier (EDFA) after signal modulation and polarization multiplexing. Before entering the mode multiplexer, the signals are mode-stimulated to have three different modes, LP_{01} , LP_{11x} , and LP_{11y} . After wavelength-division multiplexing of the channels of these five MDMs through a WDM, these mixed-multiplexed optical signals are coupled into a step-type FMF, which is 60 km long and has a dispersion of 17 ps/nm/km and a dispersion slope of 0.12 ps/nm²/km.

At the output of the three-mode fiber, five different wavelengths of the mode mixing channel are first separated by a wavelength demultiplexer. Afterwards, five independent branches of different wavelengths are re-separated the three modes signals through the mode demultiplexer of the current channel. The demultiplexer consists of the mode demultiplexer and the depolarization multiplexer. After wavelength decomposition multiplexing, the LP_{01} and LP_{11} modes are first separated by a mode demultiplexer. It also needs to go through a depolarization

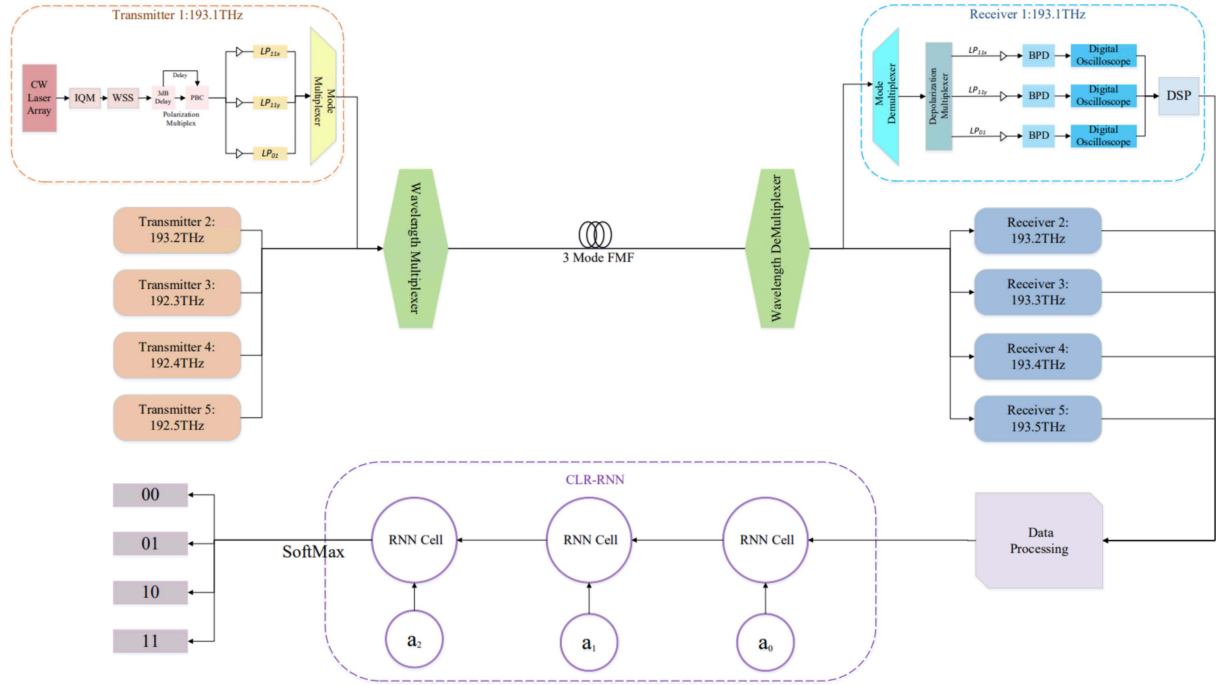


Fig. 4. System setup. The data is first transmitted through the WDM-MDM system followed by data reception, and then passed into the CLR-RNN for processing. Finally, the classification results are obtained.

multiplexer to split the LP_{11} modes into LP_{11x} and LP_{11y} for LP_{11} modes and amplified by EDFA. Then the optical signals are received and detected by multiple coherent optical receivers, and the signals are demodulated. To focus on the MIMO algorithm, since the power entering the fiber in this system becomes much smaller with modulation and multiplexing, the nonlinear crosstalk generated during transmission is negligible, and the channels can be considered independent of each other. Finally, the generated electrical signals are processed by a MIMO digital signal processing (DSP) to eliminate various effects in the link. In order to confirm that the correct mode is stimulated, we examined the output power of each mode. The results show that the signal information of each channel is smoothly reconstructed. The system has taken into account spontaneous radiation noise of EDFA and the dark current noise of the detector is set to 10nA in this simulation.

In this simulation system, there are various losses and interferences in the signal transmission process, which will have a great impact on the quality of the transmitted signal, and it is necessary to process the signal to improve the overall quality of the communication system. The introduction of MIMO technology can improve the overall signal quality of the communication system. Therefore, this system combines the unique characteristics of neural networks and optical communications to design an algorithm for optical communications. Since neural networks can generally only process real data, complex variables need to be divided into real and imaginary parts for processing:

$$\mathbf{y}(t) = \begin{bmatrix} \Re(\mathbf{y}(t)) \\ \Im(\mathbf{y}(t)) \end{bmatrix} \quad \mathbf{x}(t) = \begin{bmatrix} \Re(\mathbf{x}(t)) \\ \Im(\mathbf{x}(t)) \end{bmatrix}$$

$$\mathbf{H} = \begin{bmatrix} \Re(\mathbf{H}) & -\Im(\mathbf{H}) \\ \Im(\mathbf{H}) & \Re(\mathbf{H}) \end{bmatrix} \quad (5)$$

where, \Re and \Im are operations that take the real and imaginary parts of a complex vector. Considering that the content of the dataset used for training and testing is too large to satisfy the demand by directly feeding the data into the neural network, this design is introduced with a CLR-RNN learning model as shown in the bottom half of Fig. 4, which makes it possible to extract the features of the data and expects to obtain better results.

The RNN network model shown in Fig. 4 has 3 layers, where the number of recurrent neurons in each layer is 20. The last output layer is a fully connected layer consisting of 4 neurons, each representing a single category. The activation function used in the final output layer is the SoftMax function, which is used for the output of the multi-categorization problem. It normalizes a vector of values into a vector of probability distributions and the sum of the individual probabilities is equal to one. The SoftMax layer is often used in conjunction with the cross-entropy loss function. In this system, the cross-entropy loss function is mainly used to determine the closeness of the actual output and the desired output. It is used to measure the difference between the output of the network and the label, and this difference is utilized to update the network parameters after backpropagation. The tanh function is used as the activation function of the neurons for the other 3 layers apart from the output layer which uses the SoftMax function.

Based on the data collected from the WDM-SDM optical communication transmission system model proposed in this paper, we have produced the dataset, which is divided into

TABLE I
OVERALL PARAMETERS IN CLR-RNN

Parameters in CLR-RNN	
Number of layers	1, 2 or 3
Number of neurons in each layer	20
CLR <i>base_lr</i>	0.05
CLR <i>max_lr</i>	0.1
CLR <i>step size</i>	20
Batch size	50

the training set and the test set, and they are independent of each other. The total data contents of the dataset are 3.3×10^5 groups of symbols, the training set contains 2.8×10^5 groups of symbols and the test set contain 5×10^4 groups of symbols. Although using a larger database would provide greater insight, at this stage we used this WDM-MDM dataset for comparative testing. The data content of the dataset consists of the received signal data as $x(t)$ with the joint matrix $J(t)$ of the channel matrix H , and the transmitted signal data $y(t)$ at the output data. The input data is the joint matrix $J(t)$ of $x(t)$ with H , which can be written as follows (6) [19]:

$$\begin{aligned}
 J(t) &= [J_{re}(t) \quad J_{im}(t)] \\
 J_{re}(t) &= \begin{bmatrix} \Re(H_1) & \Re(x_1) \\ \vdots & \vdots \\ \Re(H_N) & \Re(x_N) \end{bmatrix} \\
 J_{im}(t) &= \begin{bmatrix} \Im(H_1) & \Im(x_1) \\ \vdots & \vdots \\ \Im(H_N) & \Im(x_N) \end{bmatrix} \quad (6)
 \end{aligned}$$

where, N is the number of transmission signals. In each signal of this project, the dimensions of $J_{re}(t)$ and $J_{im}(t)$ correspond to sequence length and input size, which are 4 and 6, respectively. During the training process, the parameters of the RNN are the weights W and bias b of individual neurons, which can be optimized by the backpropagation process.

For getting a perfect model, the learning rate, batch size, and epoch need to be modified according to the actual situation of training, and these three parameters together determine the performance of the system model. In the training process of this system, the learning rate is adaptively adjusted using CLR to make the training more efficient, where *base_lr* = 0.05, *max_lr* = 0.1, and *step size* = 20. A high value of epoch setting may lead to overfitting of the system and a low value may lead to the potential problem of the system not being able to train the model adequately. In this system, we used different epoch sizes for different RNN networks in order to compare the superiority of CLR. In addition to this, the batch size is uniformly set to 50. The parameters in CLR-RNN has shown in Table I. Overall, changing parameters and iteratively training models is an extremely important part of machine learning.

Nowadays, many tools and algorithms are available for building and training large neural networks. Among many existing machine learning frameworks, we use PyTorch to design and train our neural network. [24] PyTorch is a mathematical library

that performs efficient computations and automatic differentiation on various models. For hardware, we used a CPU Core I9-10900X and a GPU Nvidia GTX-3070 for training and testing the neural network.

IV. RESULTS AND COMPARISON

The details of the designed neural network have been described in the previous section. In this section, the machine learning model and its parameter settings are tested differently and its simulation results are analyzed and compared with traditional signal detection algorithms and fully connected neural networks.

In training, the loss function needs to be utilized to evaluate how well the model predicts the dataset. As mentioned in Section III, this system uses the cross-entropy loss function to measure the robustness of the model. The loss function plays a very important role in machine learning by calculating the deviation between the forward computed value and the correct value in each iteration of the neural network, thus guiding the next training step in the right direction. By reducing the loss between the true value and the predicted value, the predicted value generated by the model is made to be closer in the direction of the true value. Therefore, observing the value of the loss function at each iteration can determine whether the results are converging or not.

Shown in Fig. 5 are the loss curves and training accuracy curves for both RNN and CLR-RNN when the number of hidden layers is 3. The data set used at this point is the one at signal-to-noise ratio (SNR) equal to 35. For the RNN, there is fast convergence in the first 9 trainings, and the accuracy is less than 100% until the training is carried out 27 times. However, for our proposed CLR-RNN, the accuracy reaches 100% by the fifth time of training and converges faster. In addition to that, the loss of CLR-RNN is also smaller than that of RNN. At the same time, the increasing trend of the accuracy of the training set is extremely similar to the decreasing trend of the loss function value. For the RNN, the initial value of the loss function is about 0.006567, which starts to decrease rapidly by the 9th iteration, slows down by the 15th iteration, and stabilizes thereafter. When Epoch = 30, the loss function value drops to 3.58E-05. Turning to CLR-RNN, the initial value of its loss function is about 0.000219, which is an order of magnitude smaller than that of RNN, and when Epoch = 10, the loss function value drops to 1.57E-05. it can be seen that with the performance of the CLR, it is possible to avoid retuning and to achieve optimal accuracy with fewer iterations.

In order to find the optimal number of hidden layers and neurons required for RNNs, we developed a number of neural networks with different numbers of hidden layers for training and compared RNNs with traditional fully connected neural networks (FNN). FNN is one of the basic artificial neural network structures and is the most widely used type of neural network. [25] In an FNN, each neuron is connected to all neurons in the previous and following layers, forming a densely connected structure. FNN can learn the features of the input data and perform tasks such as classification and regression.

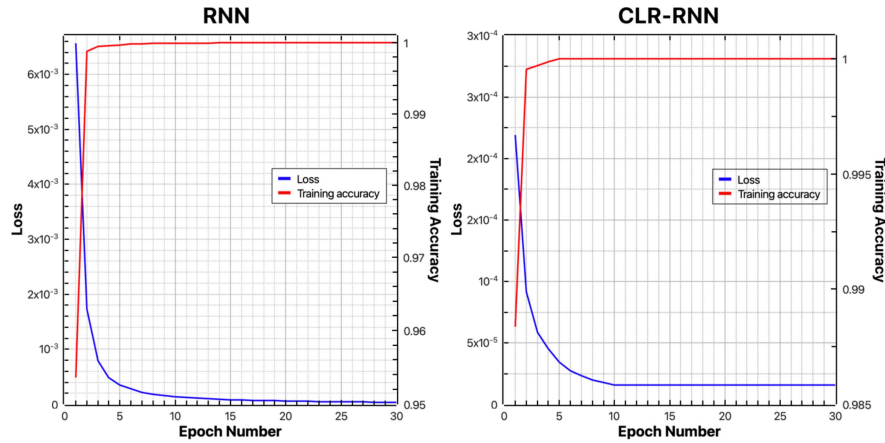


Fig. 5. Comparison of RNN and CLR-RNN training states when the hidden layer is 3. (a) Loss and training accuracy for RNN. (b) Loss and training accuracy for CLR-RNN.

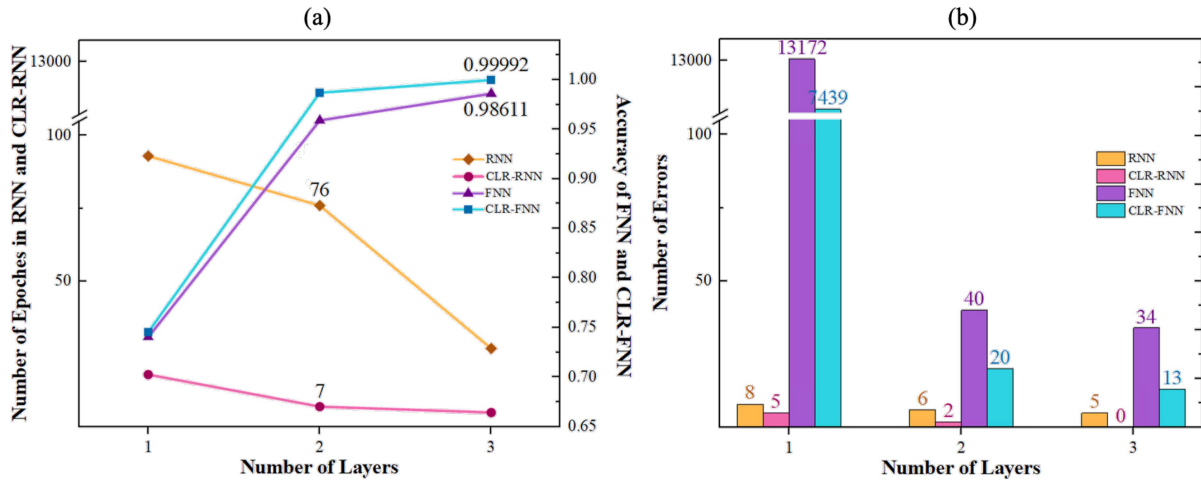


Fig. 6. Comparison of results between RNN and FNN with various layers and using CLR or not. (a) The left axis represents the epoch that reaches the highest accuracy for RNN and CLR-RNN. The right axis represents the highest accuracy for FNN and CLR-FNN. The horizontal coordinate is the number of layers of each neural network. (b) The number of errors in different neural network with different number of layers.

Another competitive neural network model, the convolutional neural network (CNN), has also been considered in our work. In previous simulation experiments [26], we tested the performance of CNN for MIMO detectors in MDM optical transmission systems. It is well known that CNNs are inconsistent with NNs and RNNs in the overall framework structure, hence in the following comparisons of the performance of different layers will be performed only between NNs and RNNs.

The performance of different neural networks using different numbers of hidden layers and network designs is shown in Fig. 6, when the dataset used is the one at SNR = 35. Since the behavior of fiber is nonlinear, the bit error rate (BER) is improved with the increase of hidden neurons. The ability to learn potential patterns increases with the number of hidden neurons increased as shown in Fig. 6(a). When the neural network structure is RNN, the accuracy of training is always 100%. For RNN, the number of epochs needed to reach the highest training accuracy is 93 for 1 layer, 76 for 2 layers, and 27 for 3 layers, however, after adding the adaptive algorithm CLR, the number of epochs

needed to reach the highest training accuracy is 18, 7, and 5, respectively, which greatly accelerates the speed of convergence and improves the efficiency significantly. The number of epochs needed to reach the highest training accuracy after adding Step Learning Rate and Cosine Annealing Learning Rate is 21 and 19, respectively, which are much higher than that for CLR. On the contrary, comparing FNN, their training number is 100, however, their highest training accuracy is 99.92% for CLR-FNN with three layers, which is much worse than CLR-RNN.

Comparing the BER can better show the superiority of CLR-RNN as shown in Fig. 6(b). When the neural network has only one layer, the number of errors for FNN is as high as 13172, however, the number of errors for CLR-RNN is only 5, with a BER equal to 10^{-4} . When the neural network has three layers, the CLR-RNN gives the best results, with a BER of 0, as compared to the number of errors for RNN with 5, for FNN with 34, and for CLR-FNN with 13. For a CLR-RNN with 3 hidden layers and 20 neurons in each hidden layer, the accuracy is 100% for both training and testing, which fully meets the requirements

of the MIMO detection algorithm. Thus, it seems that the RNN is much better than the traditional FNN, and the addition of the algorithm of adaptive learning rate CLR plays a crucial role in the MIMO detection of WDM-MDM optical transmission systems. Due to the drastic reduction in the number of epochs needed to reach the highest training accuracy, it can greatly reduce the training time and improve the convergence speed.

The model was trained several times at different SNRs to test its generality and its BER was compared with those of several detection algorithms. Specifically, we used white noise with additive noise of 0 mean and SNR ranging from 5–35. We used not only the previously mentioned FNN and CNN but also two common conventional detection algorithms, Zero Force (ZF) [27] and Minimum Mean Square Error (MMSE) [28], in our comparison tests. ZF is a linear signal detection algorithm that treats the bit stream emitted by the target transmitting antenna as useful and the bit stream emitted by other transmitting antennas as disturbing. The core concept is to use matrix operations to demodulate the signal and remove interference. Since ZF detection only considers the removal of interference between antennas and does not take into account the effect of additive noise, the recovered signal contains a large amount of enhancement noise. When the SNR is low, the detection performance decreases sharply. In order to make up for the shortcomings of the ZF algorithm, the MMSE detection algorithm is improved on the basis of the ZF algorithm, and the MMSE algorithm also considers the additive noise interference in the transmission process when decoupling the signal.

To perform a performance comparison between ZF, MMSE, FNN, CNN, RNN, and our CLR-RNN, we used the same input vectors and received vectors for constructing the test dataset to comparatively test the performance of CLR-RNN. As the SNR increases, the performance difference between the algorithms gradually comes out. The algorithm performance is ranked, and the CLR-RNN algorithm has the best performance, RNN is second best, CNN and FNN follows, and finally, MMSE is slightly better than ZF. The results are shown in Fig. 7.

The results show that the CLR-RNN is theoretically optimal and has better performance than the traditional algorithm. It can be found that the neural networks detection algorithm is sensitive to the SNR and cannot maintain a very low BER in high noise channels, but it has a very low BER or even no BER in low noise channels, and its performance is much better than the traditional detection algorithm. At SNR of 30–35, the CLR-RNN has a BER of 0, which achieves the ideal BER for QPSK. When the SNR is 5, the BER of ZF and MMSE detection algorithms are the same, and the BER of FNN, CNN, RNN, and CLR-RNN are lower than them. When the SNR is more than 10, the BER of the machine learning algorithm starts to decrease and gradually distances itself from the ZF and MMSE detection algorithms. When the SNR is about 5, the advantage of the CLR-RNN is not yet reflected, because the noise is large and the neural networks algorithm learns the features incorrectly. When the SNR is greater than 10, the neural networks algorithm outperforms the traditional algorithm in all aspects and the BER of the CNN, RNN and CLR-RNN algorithms decreases sharply. When the SNR is greater than 26 and less than 35, the CLR-RNN detection

BER versus SNR of Different MIMO Detection Algorithms

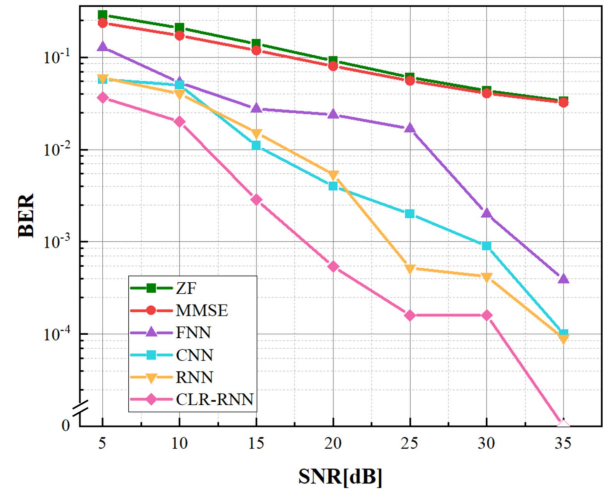


Fig. 7. BER versus SNR of different MIMO detection algorithms. In the plot line of the CLR-CNN, the BER measured at SNR = 35 is 0. Since this is a logarithmic scale, it is set as a null point.

Different MIMO Detection Algorithms Time Cost

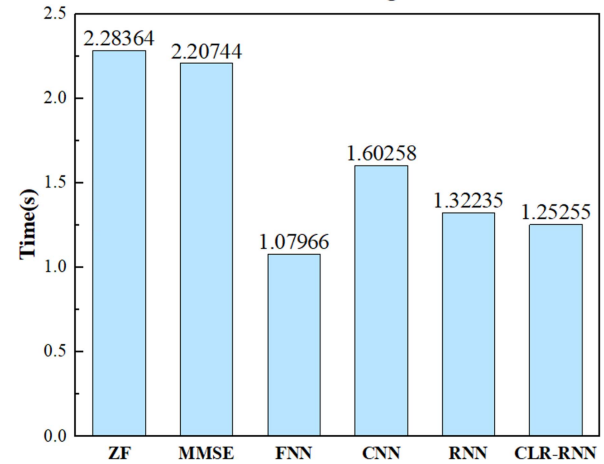


Fig. 8. Average time cost of different MIMO detection algorithms. FNN spends the least time because there are fewer parameters than CLR-RNN.

algorithm still outperforms the RNN detection algorithm. It is worth noting that the difference in BER between the ZF and the MMSE is always small, but it can also be seen that the MMSE outperforms the ZF at lower SNRs. Thus, it seems that the MIMO detection algorithm combined with neural networks shows high performance, although this performance requires a better channel environment.

In terms of complexity, the time consumed by the model for each test was calculated, and the results are shown in Fig. 8. It can be found that the average time consumed by the CLR-RNN algorithm is 1.293s, which is slightly lower than the average time consumed by the RNN, which is 1.322s. What is more, the time cost of CNN was 1.603s. The time complexity is lower compared to other traditional algorithms, while the average time consumed by the FNN algorithm, which has the lowest time complexity among the machine algorithms, is 1.079s. As for computation

complexity among these machine algorithms, their computational parameters are as follows: $num_{paras}(FNN) = 1004$, $num_{paras}(RNN) = 1460$, $num_{paras}(CNN) = 208180$. It can be seen that the total number of computational parameters of FNN is less than that of RNN, so the computation takes less time. The computation time of CLR-RNN is less than that of RNN, however, the computational parameters of their networks are the same. The comparison of Figs. 6 and 7 shows that the computational power of FNN is inferior to that of RNN and CLR-RNN. Therefore, it is acceptable to get higher computational accuracy at the expense of a small computational time. After calculation, the time complexity of this algorithm is 56.6% lower than the MMSE algorithm. The time complexity of deep learning depends on the batch size at the time of testing, and the value of the batch size can be appropriately adjusted to a larger size if a higher processing speed is required.

From the above simulation results, it is easy to see that the algorithm is superior to other algorithms in all tests even under poor channel conditions, and its BER performance is superior to other algorithms we mentioned under better channel conditions. In addition to these advantages, the end-to-end properties of the MIMO detection algorithm based on adaptive learning CLR-RNN significantly reduce the signal detection complexity and enhance the robustness while improving the system performance. In conclusion, the machine learning-based MIMO detection algorithm proposed in this paper can be applied to optical communication systems and has the advantages of high performance and low complexity that other existing algorithms do not have.

V. CONCLUSION

In this paper, we study the application of machine learning-based MIMO signal detection technology in hybrid WDM and MDM optical communication systems, propose an RNN based on an adaptive learning rate optimization algorithm, and introduce the basic principle of the algorithm and the simulation and testing process in detail. The results shown in this paper provide a new method for novel MIMO detection in WDM-MDM optical transmission systems. Our results can be considered as an achievement in the use of neural networks for MIMO optical communication systems.

Considering the many defects of traditional detection algorithms, it is difficult to meet the requirements of high precision and low complexity for optical communication. In this paper, some existing traditional MIMO detection algorithms are briefly summarized and applied to the WDM-MDM optical communication system proposed here. The performance and complexity of these algorithms are analyzed and compared, and the algorithm proposed in this paper can solve the problems of traditional MIMO detection algorithms. For the trained and tested signals of the WDM-MDM optical transmission system, the detection accuracy can reach 100% and the BER can reach 0, and its performance reaches the level of ideal QPSK. The above results clearly show that the performance of CLR-RNN can reach very high levels and provide better results compared to existing systems.

REFERENCES

- [1] Z. Pan, C. Yu, and A. E. Willner, "Optical performance monitoring for the next generation optical communication networks," *Opt. Fiber Technol.*, vol. 16, no. 1, pp. 20–45, 2010, doi: [10.1016/j.yofte.2009.09.007](https://doi.org/10.1016/j.yofte.2009.09.007).
- [2] J. Yu, "Photonics-assisted millimeter-wave wireless communication," *IEEE J. Quantum Electron.*, vol. 53, no. 6, Dec. 2017, Art. no. 8000517, doi: [10.1109/jqe.2017.2765742](https://doi.org/10.1109/jqe.2017.2765742).
- [3] H. J. Kim, H.-N. Kim, J. Bae, and N. Hur, "Performance analysis of wideband time division multiplexing for low power broadcasting transmission," in *Proc. 25th Asia-Pacific Conf. Commun.*, 2019, pp. 28–31, doi: [10.1109/apcc47188.2019.9026450](https://doi.org/10.1109/apcc47188.2019.9026450).
- [4] T. Gissibl, M. Schmid, and H. Giessen, "Spatial beam intensity shaping using phase masks on single-mode optical fibers fabricated by femtosecond direct laser writing," *Optica*, vol. 3, no. 4, pp. 448–451, 2016, doi: [10.1364/optica.3.000448](https://doi.org/10.1364/optica.3.000448).
- [5] C. Caucheteur, T. Guo, and J. Albert, "Polarization-assisted fiber Bragg grating sensors: Tutorial and review," *J. Lightw. Technol.*, vol. 35, no. 16, pp. 3311–3322, Aug. 2017, doi: [10.1109/jlt.2016.2585738](https://doi.org/10.1109/jlt.2016.2585738).
- [6] J. M. Kahn and D. A. Miller, "Communications expands its Space," *Nature Photon.*, vol. 11, no. 1, pp. 5–8, 2017, doi: [10.1038/nphoton.2016.256](https://doi.org/10.1038/nphoton.2016.256).
- [7] K. Igarashi et al., "114 space-division-multiplexed transmission over 9.8-km weakly-coupled-6-mode uncoupled-19-core fibers," in *Proc. Opt. Fiber Commun. Conf. Post Deadline Papers*, 2015, Paper Th5C.4, doi: [10.1364/ofc.2015.th5c.4](https://doi.org/10.1364/ofc.2015.th5c.4).
- [8] K. Saitoh, "Multi-core fiber technology for SDM: Coupling mechanisms and design," *J. Lightw. Technol.*, vol. 40, no. 5, pp. 1527–1543, Mar. 2022, doi: [10.1109/jlt.2022.3145052](https://doi.org/10.1109/jlt.2022.3145052).
- [9] M. Mazur et al., "Characterization of long multi-mode fiber links using digital holography," in *Proc. Opt. Fiber Commun. Conf.*, 2019, Paper W4C.5, doi: [10.1364/ofc.2019.w4c.5](https://doi.org/10.1364/ofc.2019.w4c.5).
- [10] G. Rademacher et al., "High capacity transmission with few-mode fibers," *J. Lightw. Technol.*, vol. 37, no. 2, pp. 425–432, Jan. 2019, doi: [10.1109/jlt.2018.2870038](https://doi.org/10.1109/jlt.2018.2870038).
- [11] P. Sillard et al., "Few-mode fiber technology, deployments, and systems," *Proc. IEEE*, vol. 110, no. 11, pp. 1804–1820, Nov. 2022, doi: [10.1109/jproc.2022.3207012](https://doi.org/10.1109/jproc.2022.3207012).
- [12] T. Sakamoto et al., "Spatial density and splicing characteristic optimized few-mode multi-core fiber," *J. Lightw. Technol.*, vol. 38, no. 16, pp. 4490–4496, Aug. 2020, doi: [10.1109/jlt.2020.2987351](https://doi.org/10.1109/jlt.2020.2987351).
- [13] F. Liu, G. Hu, C. Chen, W. Chen, and C. Song, "Significant dynamic range and precision improvements for FMF mode-coupling measurements by utilizing adaptive wavelet threshold denoising," *Opt. Commun.*, vol. 426, pp. 287–294, 2018, doi: [10.1016/j.optcom.2018.05.053](https://doi.org/10.1016/j.optcom.2018.05.053).
- [14] Z.-H. Zhou and L. Shaowu, *Machine Learning*. Berlin, Germany: Springer, 2021.
- [15] B. Poudel, J. Oshima, H. Kobayashi, and K. Iwashita, "MIMO detection using a deep learning neural network in a mode division multiplexing optical transmission system," *Opt. Commun.*, vol. 440, pp. 41–48, 2019, doi: [10.1016/j.optcom.2019.02.016](https://doi.org/10.1016/j.optcom.2019.02.016).
- [16] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2224–2287, Third Quart. 2019, doi: [10.1109/comst.2019.2904897](https://doi.org/10.1109/comst.2019.2904897).
- [17] S. Li, W. Li, C. Cook, C. Zhu, and Y. Gao, "Independently recurrent neural network (IndRNN): Building a longer and deeper RNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5457–5466, doi: [10.1109/cvpr.2018.00572](https://doi.org/10.1109/cvpr.2018.00572).
- [18] Z. Zhou, L. Liu, S. Jere, J. Zhang, and Y. Yi, "RCNet: Incorporating structural information into deep RNN for online MIMO-OFDM symbol detection with limited training," *IEEE Trans. Wireless Commun.*, vol. 20, no. 6, pp. 3524–3537, Jun. 2021, doi: [10.1109/twc.2021.3051317](https://doi.org/10.1109/twc.2021.3051317).
- [19] Q. Chen, S. Zhang, S. Xu, and S. Cao, "Efficient MIMO detection with imperfect channel knowledge—a deep learning approach," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2019, pp. 1–6, doi: [10.1109/wenc.2019.8885582](https://doi.org/10.1109/wenc.2019.8885582).
- [20] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in Deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, Sep. 2022, doi: [10.1016/j.neucom.2022.06.111](https://doi.org/10.1016/j.neucom.2022.06.111).
- [21] Y. Bodyanskiy, A. Deineko, V. Skorik, and F. Brodetskiy, "Deep neural network with adaptive parametric rectified linear units and its fast learning," *Int. J. Comput.*, pp. 11–18, Mar. 2022, doi: [10.47839/ijc.21.1.2512](https://doi.org/10.47839/ijc.21.1.2512).
- [22] L. N. Smith, "Cyclical learning rates for training neural networks," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2017, pp. 464–472, doi: [10.1109/WACV.2017.58](https://doi.org/10.1109/WACV.2017.58).

- [23] J. Li and X. Yang, "A cyclical learning rate method in deep learning training," in *Proc. Int. Conf. Comput., Inf. Telecommun. Syst.*, 2020, pp. 1–5, doi: [10.1109/cits49457.2020.9232482](https://doi.org/10.1109/cits49457.2020.9232482).
- [24] P. Mishra, "Introduction to neural networks using PyTorch," in *PyTorch Recipes: A Problem-Solution Approach to Build, Train and Deploy Neural Network Models*. New York, NY, USA: Apress, 2019, pp. 111–126, doi: [10.1007/978-1-4842-4258-2_4](https://doi.org/10.1007/978-1-4842-4258-2_4).
- [25] Y. Xue, Y. Wang, and J. Liang, "A self-adaptive gradient descent search algorithm for fully-connected neural networks," *Neurocomputing*, vol. 478, pp. 70–80, 2022, doi: [10.1016/j.neucom.2022.01.001](https://doi.org/10.1016/j.neucom.2022.01.001).
- [26] D. Zhang, X. Ma, S. Lu, and Z. Tan, "MIMO detection using an ALR-CNN in SDM Transmission Systems," *Appl. Opt.*, vol. 62, no. 34, pp. 8978–8986, 2023, doi: [10.1364/ao.496453](https://doi.org/10.1364/ao.496453).
- [27] X. Chen et al., "Equalization of two-mode fiber based MIMO signals with larger receiver sets," *Opt. Exp.*, vol. 20, no. 26, pp. B413–B418, 2012, doi: [10.1364/oe.20.00b413](https://doi.org/10.1364/oe.20.00b413).
- [28] L. Fang, L. Xu, and D. D. Huang, "Low complexity iterative MMSE-pic detection for medium-size massive MIMO," *IEEE Wireless Commun. Lett.*, vol. 5, no. 1, pp. 108–111, Feb. 2016, doi: [10.1109/lwc.2015.2504366](https://doi.org/10.1109/lwc.2015.2504366).