Received December 30, 2019, accepted January 8, 2020, date of publication January 15, 2020, date of current version February 19, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2966694

Power Allocation for Multiple Transmitter-Receiver Pairs Under Frequency-Selective Fading Based on Convolutional Neural Network

MINGJUN DAI[1,2,3], QINGWEN HUANG[1,2,3], ZEXIN LU[4], BIN CHEN[1,2,3], HUI WANG[1,2,3], AND XUE QIN[5]

[1]Guangdong Provincial Engineering Center for Ubiquitous Computing and Intelligent Networking, College of Electronic and Information Engineering, Shenzhen University, Shenzhen 518060, China
[2]Shenzhen Key Lab of Advanced Communication and Information Processing, Shenzhen University, Shenzhen 518060, China
[3]Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), Shenzhen University, Shenzhen 518060, China
[4]Department of Computing, The Hong Kong Polytechnic University, Hong Kong
[5]Department of Computing Sciences, Texas A&M University at Corpus Christi, Corpus Christi, TX 78412, USA

Corresponding author: Bin Chen (bchen@szu.edu.cn)

This work was supported in part by the Natural Science Foundation of Guangdong under Grant 2018A0303130131 and Grant 905265797093, in part by the Key Project of Department of Education of Guangdong Province under Grant 2015KTSCX121, in part by the Natural Science Foundation of China under Grant 61575126, in part by the Basic Research Foundation of Shenzhen City under Grant JCYJ20160422093217170 and Grant JCYJ20170818091801577, and in part by the Natural Science Foundation of Shenzhen University under Grant 00002501.

ABSTRACT For multiple transmitter-receiver pairs communication in a frequency-selective environment, typical power allocation method is the Iterative-Waterfilling (IW) algorithm. Main drawback of IW is its poor convergence performance, including low convergence probability and slow convergence speed in certain scenarios, which lead to high computational load. Large-scale network significantly magnifies the above drawback by lowering the convergence probability and convergence speed, which is difficult to satisfy real-time requirements. In this work, we propose a power allocation scheme based on convolutional neural network (CNN). The design of loss function takes into account the Sum Rate (SR) of all users. The output layer of the CNN model is replaced by several Softmax blocks, and the output of each Softmax block is the ratio of the transmission power of each user on the sub-carrier to the total power. Numerical studies show the advantages of our proposed scheme over IW: with the constraint of not lowering SR, there is no convergence problem and the computational load is significantly reduced.

INDEX TERMS Power allocation, convolutional neural network, sum rate, iterative waterfilling.

I. INTRODUCTION

The form of future communication networks will become more diverse [1], [2], which extends from current human-to-human to machine-to-machine (M2M) [3], and to internet of things (IoT) communication [4]–[6]. In certain IoT networks, the number of IoT devices may be huge, which leads to fierce competition for scarece bandwidth. To reap relatively high data delivery rate, one typical solution is that multiple users share multiple sub-carriers in non-orthogonal manner. In this scenario, multiple users interfere with each other and are viewed as noise makers between each other [7], [8]. The corresponding effect is that the data delivery rate of one user in a sub-carrier decreases as the transmission power of other users in this sub-carrier increases. Therefore, an efficient power allocation algorithm is desired to ensure that all users can simultaneously achieve relatively high data deliver rate [9], [10].

On one hand, one representative power allocation scheme is the Iterative-Waterfilling (IW) algorithm [11], which is sub-optimal in terms of achieved sum rate over all users. In [12], the IW algorithm is utilized to solve the power allocation problem in digital subscriber line (DSL) systems, which algorithm provides a complete solution to the network power allocation problem. Typical drawback of IW is its poor convergence performance, including low convergence

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

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see http://creativecommons.org/licenses/by/4.0/

probability and slow convergence speed in certain scenarios which leads to high computational load. Large-scale networks, especially IoT networks with huge amount of communication terminals, magnify the above drawback by further lowering the convergence probability and convergence speed. This confines its deployment scope, including limited to small-scale networks and limited to poor real-time requirement scenarios. Therefore, a lot of works focus on studying the sufficient while non-necessary (no sufficient and necessary condition has been provided) conditions for the convergence [13]–[15].

On the other hand, due to excellent practical performance, deep learning has been deployed into diverse domains, including image classification, natural language processing, and speech recognition, etc [16]–[20]. In recent years, due to the convenience and timeliness of data acquisition in wireless communication networks, deep learning also shows its advantage in wireless communication networks [21]–[24]. The work in [25] employs deep learning to solve channel distortion and symbol detection problems in an end-to-end OFDM channel, and its performance is comparable to that of the minimum mean square error (MMSE) estimator. In [26], in order to reduce the complexity of the iterative algorithm, deep neural network (DNN) is utilized to approximate the power control strategy of the WMMSE algorithm with minor performance loss. For multi-channel cognitive radio networks, scheme based on numerical optimization [27] enables primary users to achieve higher throughput, while [28] proposes a resource allocation scheme based on DNN, which achieves higher spectral efficiency of secondary users without causing excessive influence on the communication of primary users. The work in [29] shows the advantage of deep learning in achieving higher spectral efficiency (SE) or energy efficiency when compared with traditional power control strategy. For the IoT system based on non-orthogonal multiple access, under the assumption of imperfect successive interference cancellation, [30] takes SE as an optimization objective and proposes an optimal resource allocation solution based on CNN, which has fast convergence and low computational load.

For multiple transmitter-receiver pairs communication network model in the frequency-selective environment, there are some shortcomings in the existing works. On the one hand, the traditional IW algorithm [11] has high computational load, poor convergence performance, and is difficult to be generalized to large-scale networks. On the other hand, existing deep learning-based approaches [29] do not consider the multi-carrier environment and the optimization of the Sum Rate (SR).

In this work, for multiple transmitter-receiver pairs in a multi-carrier environment, we propose a power allocation strategy based on a convolution neural network (CNN). The loss function takes into account the SR of all users. Numerical comparison with traditional power control scheme shows that with the constraint of not reducing the SR, our proposed scheme reduces the operation time significantly.

The rest of this paper is arranged as follows. In sec. II, preliminaries are given. In sec. III, the proposed power control scheme based on CNN is proposed. The simulation results are given in sec. IV and conclusions are drawn in sec. V.

II. PRELIMINARIES

A. SYSTEM MODEL

As shown in Fig. 1, M transmitter-receiver pairs, denoted by (S_i, D_i) for $i \in \{1, 2, \ldots, M\} \triangleq \mathcal{M}$, communicate simultaneously, under frequency-selective fading, where transmitter node S_i wants to deliver data to receiver D_i, and there are K orthogonal sub-carriers with each having different link gains. These M transmitter-receiver pairs share these K orthogonal sub-carriers. For $k \in \{1, 2, \ldots, K\} \triangleq \mathcal{K}$, $b_i^{(k)}$ denotes the direct link gain on sub-carrier k between transmitter i and receiver i for $i \in \mathcal{M}$, and $c_{i,j}^{(k)}$ denotes the interference link gain in sub-carrier k between transmitter i and receiver j for $i, j \in \mathcal{M}$, $i \neq j$. Transmitter S_i is subject to power constraint P_i, i.e., $\sum_{k\in\mathcal{K}} P_i^{(k)} = P_i$ for $i \in \mathcal{M}$, where $P_i^{(k)}$ denotes the transmit power allocated by transmitter i into subcarrier k. Let $n_{id}^{(k)}$ denote the noise power at receiver D_i in sub-carrier k.

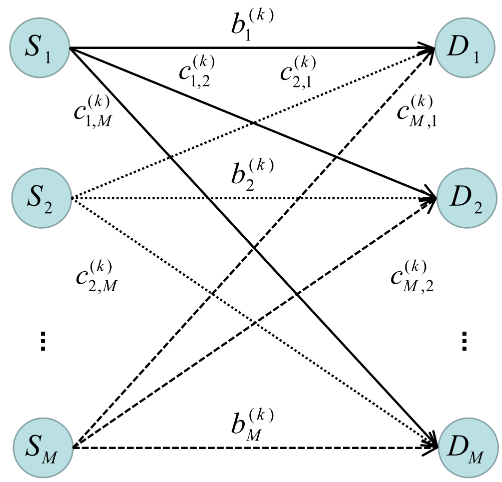


FIGURE 1. Definition of direct link gain and interference link gain in the transmitter and receiver. Superscript $^{(k)}$ indicates subcarrier k.

Except the additive noise at receiver D_i in sub-carrier k, there exists interferences from other nodes, which interferences are collectively viewed as another source of additive noise. For the sake of simple representation, let $n_i^{(k)}$ and $I_i^{(k)}$ denote the normalized noise power and normalized accummulated interference power at receiver D_i in sub-carrier k, respectively, where $n_i^{(k)} = \frac{n_{id}^{(k)}}{b_i^{(k)}}$, and normalized accummulated interference power $I_i^{(k)} = \sum_{i\neq j} \frac{P_j^{(k)} c_{i,j}^{(k)}}{b_i^{(k)}}$. The data rate of user i in sub-carrier k can be calculated by Shannon's formula:

$$R_{ii}^{(k)} = \frac{1}{2} \log_2 \left(1 + \frac{b_i^{(k)} P_i^{(k)}}{n_i^{(k)} + I_i^{(k)}}\right). \tag{1}$$

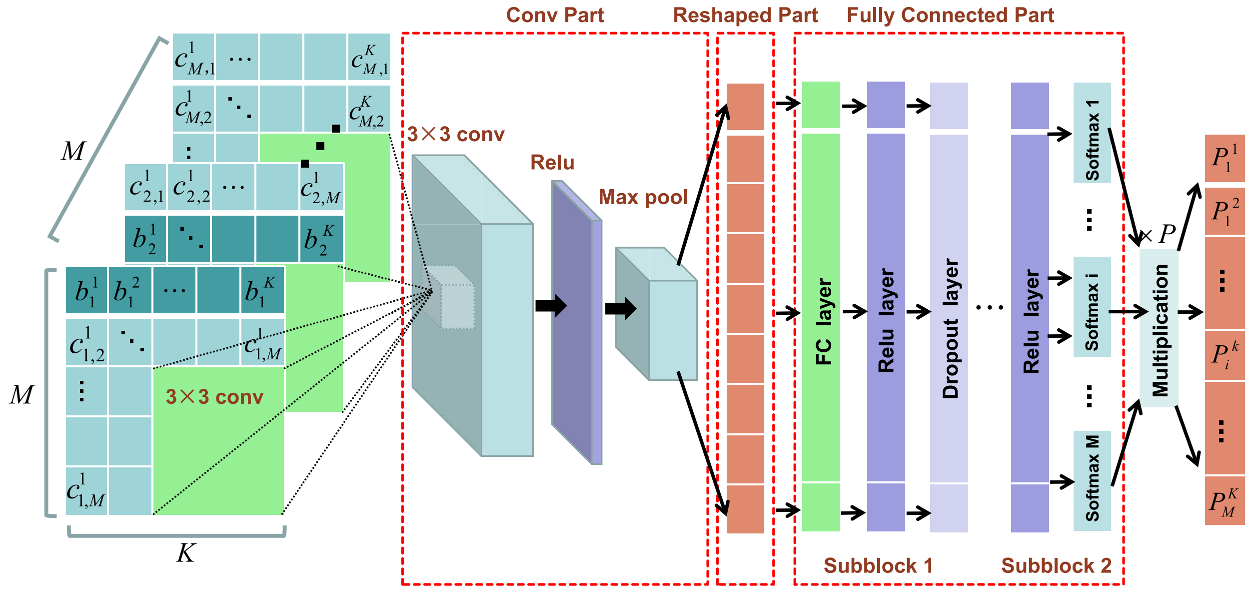


FIGURE 2. The proposed CNN model structure.

Correspondingly, the SR of user i can be expressed as:

$$R_i = \sum_{k=1}^{K} R_{ii}^{(k)}, i \in \mathcal{M}. \tag{2}$$

B. THE ITERATIVE-WATERFILLING (IW) ALGORITHM

The IW algorithm repeats the following two stages in each iteration. In the first stage, the waterfilling operation [31] is performed by the users either successively or in parallel. After the transmission powers are updated, the interference power caused to other users will also change. In the second stage, all users then iteratively update their power allocation using the waterfilling method. As the number of users and sub-carriers in a communication network increases, it is likely that the impact of interference noise on high-speed data transmission of users cannot be ignored after several iterations, which leads to the poor convergence performance of the IW algorithm.

The SR is evaluated after the power levels of all users converge. Note that the above two-stage IW algorithm may not converge. In this case, the system fails to find one equilibrium.

III. PROPOSED RESOURCE ALLOCATION SCHEMES

In this section, we propose a power allocation scheme based on CNN, which includes the following four subsections. In subsec. A, the structure of the proposed CNN scheme is described. The way of data generation is given in subsec. B. The training and testing of the model is described in subsec. C and subsec. D, respectively.

A. THE STRUCTURE OF THE NETWORK

As shown in Fig. 2, the proposed CNN scheme consists of three concatenated parts, including a convolutional part, a reshaped part, and a fully connected part. The input of the model is channel coefficients of each user in each sub-carrier, i.e., the direct link gain $\{b_i\}$ and interference link gain $\{c_{ij}\}$, while the output of the model is the transmission power allocated by each user into each sub-carrier. To facilitate the convolutional operation on the data matrix using CNN, we construct a three-dimensional data matrix for input data. The dimension size of the data matrix is $M \times K \times M$, which corresponds to M users and K orthogonal sub-carriers.

In the following, we describe each component of the proposed CNN scheme and the loss function design in detail.

1) CONVOLUTIONAL PART

The convolutional part consists of a convolutional layer, a Rectified linear unit (Relu) layer, and a Max pooling layer.

With the objective of finding the relationship between link gain (characteristic) of different users on different sub-carriers, the function of the convolutional layer is to perform three-dimensional spatial convolution on the input data, i.e., a convolution on the data matrix. Another advantage of adding a convolutional layer is that the number of training parameters of the model can be reduced. If we reshape the data matrix into a one-dimensional column vector, i.e., use the DNN model to train directly, the training parameters are too many to train when the number of users increases to a large value. The parameter sharing mechanism of the convolutional layer can effectively reduce the number of training parameters, and the relationship between features can be trained more quickly.

By introducing non-linearity into CNN, the function of the Relu layer is to zero the output of some neurons, so as to increase the sparsity of the whole network and reduce the dependence between parameters, which further alleviates the occurrence of overfitting.

As a kind of pooling operation, the purpose of adding Max pooling is to reduce both parameters and computational complexity through downsampling the data matrix as far as possible without losing data features. Moreover, other effective function of the Max pooling layer includes preventing overfitting, and improving the generalization ability of the model.

2) RESHAPED PART

The function of the reshaped part is to reshape the output of the convolutional part into a one-dimensional column vector, which will serve as the input of the succeeding fully connected part.

3) FULLY CONNECTED PART

The fully connected part consists of two concatenated sub-blocks, or two hidden layers. The first sub-block consists of a fully connected (FC) layer, a Relu layer, and a Dropout layer. The role of the Dropout layer is to temporarily discard neurons according to a certain probability from the network during the training process. This operation can effectively prevent the emergence of overfitting and increase the generalization capability of the training model.

The last sub-block of the fully connected part consists of an FC layer and a Relu layer. The number of neurons in the FC layer is $M \times K$, which corresponds to M users and K orthogonal sub-carriers. After passing through the Relu layer, these MK outputs are equally divided into M parts with each containing K consecutive components, and these M parts are treated as the input of M Softmax blocks, where the input and output sizes of each Softmax block are K. Softmax function, or normalized exponential function, is widely used in multi-category scenarios [32]. Its function is to map the input to real numbers between 0 and 1, subject to the constraint that the normalized sum is 1. Therefore, it can be used in multi-classification scenarios or meet the need for seeking probability in certain applications. In this work, we let the Softmax block i correspond to the user i, and correspondingly the output of the Softmax block i is $\frac{P_i^k}{P}$ for $k \in \mathcal{K}, i \in \mathcal{M}$, where P is the total power of the user. In other words, the k-th output of the Softmax block i is the ratio of the transmission power assigned on the sub-carrier k by the user i to the total power.

4) LOSS FUNCTION

Generally, in supervised learning, the loss function is set to be the error between the predicted value and the real value. In our proposed scheme, to maximize the optimization objective, we redesign the loss function in the training process of the model, as described below.

To maximize the SR of all users, the CNN model can be trained by minimizing the following loss function:

$$L_{\mathrm{SR}} = -\sum_{i \in \mathcal{M}} R_i = -\sum_{i \in \mathcal{M}} \sum_{k \in \mathcal{K}} R_{ii}^{(k)}. \tag{3}$$

B. DATA GENERATION

Channel coefficients are generated according to Rayleigh fading distribution [33], i.e., the direct link gain $\{b_i\}$ and the interference link gain $\{c_{i,j}\}$ obey the Rayleigh fading distribution, where $i, j \in \mathcal{M}$, $i \neq j$. We define b and c as the expectation of the direct link gain and the interference link gain, namely, $E[\{b_i\}] = b$, $E[\{c_{i,j}\}] = c$. In the communication network that we focus on, when the number of users is M and the number of sub-carriers is K, the number of the direct link gains $\{b_i\}$ and interference link gain $\{c_{i,j}\}$ of user i is K and $(M-1) \times K$ for $i, j \in \mathcal{M}$, respectively. Hence, the size of the data matrix dimension generated by user i is $M \times K$. For a given communication network parameter (M, K, b, c), we generate a data set with a sample size of 10000, and then divide the whole data set into training set, verification set, and testing set for training, validating, and testing, which takes up 60%, 20%, and 20%, respectively.

C. MODEL TRAINING

We use the training set to optimize the weight parameters of the CNN model, and use the validation set to evaluate the performance and generalization capability of the training model. The optimization algorithm is essentially a mathematical method. Typical optimization algorithms include the Stochastic Gradient Descent (SGD) [34], Adadelta [35], Nesterov Accelerated Gradient (NAG) [36], and Adam [37]. We adopt Adam, which is a first-order optimization algorithm that can replace the traditional SGD process. It can update the weights of neural networks in an iterative manner based on training data set. Moreover, it is suitable for solving optimization problems with large-scale data and parameters, as well as for problems with high noise or sparse gradients, which requires only a very small amount of tuning parameters.

Different initialization values of neural network parameters can affect the convergence speed in a certain extent. If the parameters are initialized to be too small, the output of each layer of the network will probablly be close to zero, which will drive the vanishing gradient problem occur during back propagation procedure. To make the CNN model find the optimal solution faster and to speed up the convergence of the model, we use the approach suggested in [38] to initialize the weight. More specifically, the weight of each neuron is divided by the square root of its input to be the variance of each neuron's output.

D. MODEL TESTING

Under the same testing data set, we validate the generalization capability of IW, DNN, and CNN scheme proposed in this work through several experiments. Besides, we calculate the SR of all uers in various network scenarios and the average testing time in 2000 testing samples. Due to the limitation of IW in terms of not always convergent, when comparing the average testing time performance, we only consider the convergent samples (ignore the samples that are not convergent) for IW.

IV. NUMERICAL RESULTS

In this section, we numerically compare the performance of the proposed CNN scheme, DNN scheme, and IW algorithm in different network scenarios. Part A describes the parameter settings of the CNN, DNN, and the communication network scenario, and part B analyzes and compares the performance of the schemes. Specifically, the average SR performance of all users and the average testing time in 2000 testing samples of each algorithm are taken as the evaluation criteria. Besides, we count the convergence times of the IW algorithm in different network scenarios to show the limitation of the algorithm in terms of convergence and the superiority of the proposed CNN scheme. In this numerical study, codes for implementing DNN and the proposed CNN network are implemented in Python 3.6 on one computer node with an Nvidia 1080ti Graphical Processing Unit (GPU).

A. PARAMETER SETTING

In the following, we describe the parameter setting of CNN, DNN, and communication network scenario in three parts, respectively.

1) PARAMETER SETTING OF THE PROPOSED CNN MODEL

In the convolutional layer, we choose a medium-sized convolutional core with size $3 \times 3 \times M$. Besides, the number of convolution cores is set to be 32, and the step size of convolution in both horizontal and vertical directions is set to be 2. With respect to the Max pooling layer, we let the sliding window size to be 2×2, and the sliding step length in both horizontal and vertical directions to be 2. In the fully connected part, the number of hidden layer neurons in the first sub-block is 1024, and the number of hidden layer neurons in the last sub-block is $M \times K$, which corresponds to M users and K orthogonal sub-carriers. In the training phase of the model, the learning rate of Adam optimization algorithm is set to be 0.001, while the others are set to be the default values. The value of keep_prob represents the probability of each neuron being retained in the Dropout layer. It is set to be 0.8 during the training process, and set to be 1 during the testing process since all neurons need to be used.

2) PARAMETER SETTING OF THE DNN MODEL

To verify that DNN is inferior to CNN, the setting of DNN is the same as that of CNN. In other words, we use the same optimization algorithm, parameter initialization method, hyperparameter setting, and the same number of network layers to train and test the DNN model. Specifically, a sub-block is added between the reshaped part and the fully connected part, which is the same as the first sub-block of the fully connected part, namely, it contains an FC layer, a Relu layer, and a Dropout layer. Similarly, the number of neurons in the FC layer is 1024. After training, the performance of the DNN model is tested, including the SR performance and average testing time in 2000 testing samples, under the same testing data set.

3) PARAMETER SETTING OF THE COMMUNICATION NETWORK SCENARIO

We define M, K, b, and c as the number of users, the number of sub-carriers, the expectation of direct link gain $\{b_i\}$, and the expectation of interference link gain $\{c_{i,j}\}$ in the communication network, respectively. We mainly consider four different network scenarios. The first three network scenarios are $(M, K, b) \in \{(2, 96, 1), (24, 24, 4), (48, 24, 4)\}$. They are mainly used to compare the SR performance between the proposed CNN scheme and the traditional IW algorithm, and used to compare the time complexity of the three schemes, including CNN, DNN, and IW. In network scenario 1, we let $b = 1$, $c \in (0 \sim 2)$, and take into account both high SNR regime and medium SNR regime, i.e., $P \in \{10000, 500\}$. In network scenarios 2 and 3, since IW has poor convergence performance, b is set to be 4 while other parameters are kept as constant. In the comparison of the SR performance of the proposed CNN scheme and the DNN scheme, we only consider network scenario 1 with a relatively small number of samples, i.e., $(M, K, b) = (2, 96, 1)$, where c ranges from 0.1 to 2.0. The stability between the two algorithms can be compared by a network scenario with fewer samples and more sub-carriers. To compare the average testing time, we consider the performance of these three schemes mentioned in this paper. Similarly, because of the poor convergence performance of IW, the convergent samples of IW are considered in the comparision of the computational load. In network scenario 4, we mainly compare the SR performance of CNN and IW with a different number of users when the number of sub-carriers, b, and c are constant, i.e., $(K, b, c) = (24, 4, 0.1)$, $M \in \{10, 20, 30, 40, 50\}$.

For the above network scenarios, we plot different figures and tables to show the superiority of the proposed CNN scheme.

B. PERFORMANCE ANALYSIS

We evaluate the SR and average testing time performance of the three schemes, including CNN, DNN, and IW.

Fig. 3 shows the SR performance comparison between CNN and DNN schemes under different values of c in

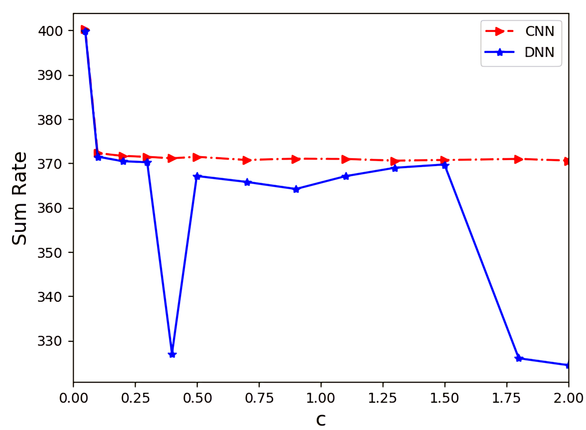


FIGURE 3. Performance comparison of CNN and DNN schemes for setting $(M, K, b) = (2, 96, 1)$.

TABLE 1. Number of convergence for IW algorithm in different network scenarios and different c.

(M, K, b, c)	$c = 0.01$	$c = 0.02$	$c = 0.05$	$c = 0.1$	$c = 0.3$	$c = 0.5$	$c >= 0.6$
(2,96,10000,1)	2000	2000	1998	1978	1116	130	0
(2,96,500,1)	2000	2000	2000	1999	1636	302	0
(24,24,10000,4)	2000	2000	1989	1575	0	0	0
(24,24,500,4)	2000	2000	2000	1780	0	0	0
(48,24,10000,4)	2000	1963	1283	1978	0	0	0
(48,24,500,4)	2000	1956	1322	0	0	0	0

network scenario 1. It can be observed that the SR curve for DNN scheme is unstable, with multiple undulating points. However, for the CNN scheme, the change of SR curve is smooth. Therefore, it can be concluded that in our network scenario, the SR performance of the proposed CNN scheme is better than that of the DNN scheme, and the specific advantages are as follows. On one hand, the CNN scheme is more stable than the DNN scheme. On the other hand, due to the parameter sharing mechanism of the convolutional layer, the parameters of CNN are less than DNN. Therefore, we don't consider the DNN scheme in the comparison of subsequent SR performance. In addition, because a convolution layer can basically learn the correlation between users and sub-carriers, increasing the number of layers of the network will only increase the testing time, which will not improve the SR performance of the CNN network. Therefore, in the subsequent experiments, the number of layers for convolution layer is set to be 1.

Fig. 4 shows the SR performance comparison between the CNN scheme and IW in different communication network scenarios, while table 1 shows the number of convergence of IW in 2000 testing samples. From Fig. 4a, 4b, and 4c, it can be observed that IW does not converge when the value of c is relatively large in every network scenarios. Specifically, we can observe that in 2000 testing samples, the number of convergence of IW decreases with c, which indicates the limitation of IW in terms of convergence when the interference from other users is large. However, there is no convergence problem in the proposed CNN scheme, because CNN does not need iterative operations in the testing of new samples, which indicates that the computational load can be significantly reduced. From Fig. 4a, we can observe that when the number of users is small and the SNR regime is high, the difference between the two schemes is small if c is small. As c increases, the achieved SR by the proposed CNN scheme can be higher than that of IW. In other scenarios, the SR performance of the proposed CNN scheme and IW are almost equal when IW converges. These simulation studies show the advantages of our proposed CNN scheme over IW: with the constraint of not lowering SR, there is no convergence problem.

Fig. 5 shows the SR performance comparison between CNN scheme and IW with a different number of users in network scenario 4. It can be observed that when the number of users is 30, IW does not converge, while the CNN scheme still has a good SR performance, even if the number of users continues to increase.

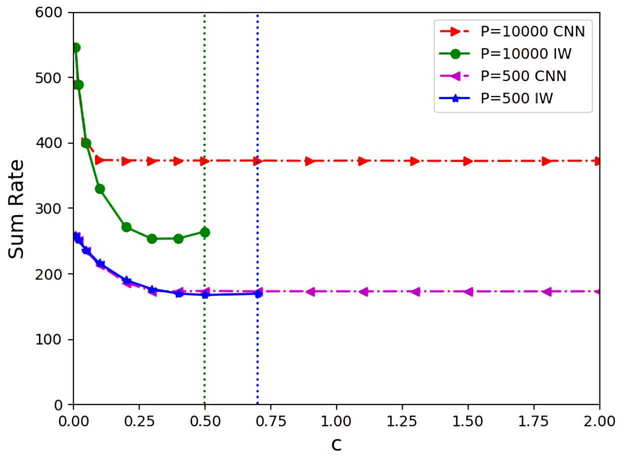
TABLE 2. Testing time of CNN, DNN, and IW schemes for single sample with different c under setting $(M, K, b) = (24, 24, 4)$ (unit: ms).

scheme	$c = 0.01$	$c = 0.02$	$c = 0.05$	$c = 0.1$
IW	122.25	158.2	291.2	753.95
DNN	0.92	0.94	0.94	0.94
CNN	0.88	0.86	0.88	0.88

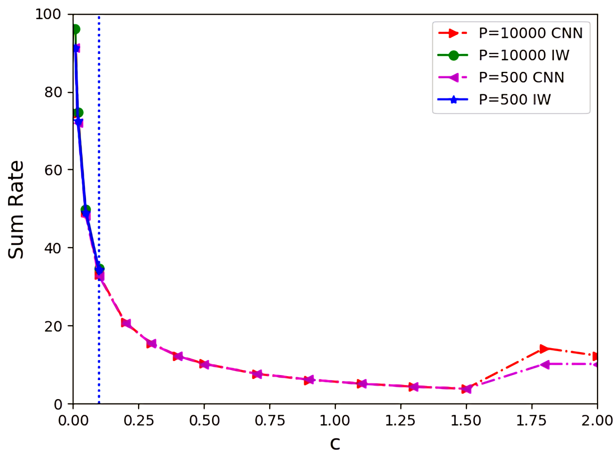
TABLE 3. Testing time of CNN, DNN, and IW schemes for single sample with different c under setting $(M, K, b) = (48, 24, 4)$ (unit: ms).

scheme	$c = 0.01$	$c = 0.02$	$c = 0.05$
IW	509.45	760.45	2284.52
DNN	4.01	3.73	3.44
CNN	3.42	3.57	3.50

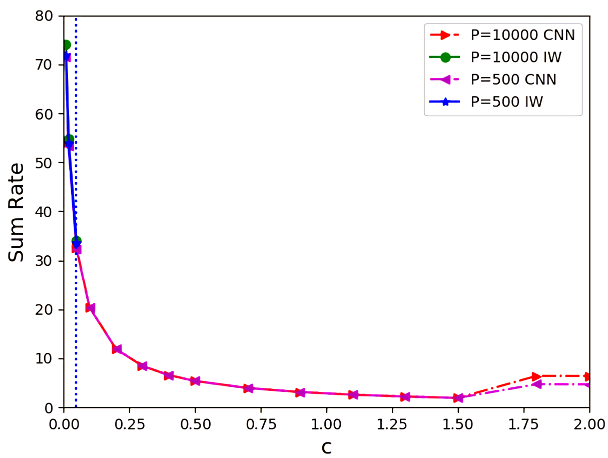
Table 2 and Table 3 show the computational load performance, namely the average testing time in 2000 testing samples, of the proposed CNN scheme, DNN scheme, and IW in network scenarios 2 and 3, respectively. Since the SNR level has less influence on the average testing time, the influence of SNR is neglected from the table. It can be observed that in each network scenario, the CNN scheme and DNN scheme perform better than IW in terms of average testing time. Between CNN and DNN schemes, the performance of the former is better than that of the latter in most cases. Meanwhile, in each network scenario, with respect to the average testing time, CNN scheme is more than 100 times faster than that of IW, and the average testing time of IW increases with the value of c. For example, in network scenario 3, the average testing time of the proposed CNN scheme is about 3.5 ms, while the larger c is, the longer time it takes to converge for IW. With c increases from 0.001 to 0.05, the average testing time of IW increased from 509.45 ms to 2284.52 ms. It indicates that the proposed CNN scheme is more stable in terms of average testing time.



(a) $(M, K, b) = (2, 96, 1)$



(b) $(M, K, b) = (24, 24, 4)$



(c) $(M, K, b) = (48, 24, 4)$

FIGURE 4. The change of the sum rate under different network scenarios. When the number of users is large, the IW algorithm does not converge when *c* is large. CNN scheme does not need to consider the problem of convergence, and is readily to obtain a larger sum rate.

The above experimental studies show the limitations of IW in terms of convergence. Besides, IW is restricted to a small-scale network with not so many users. It indicates that

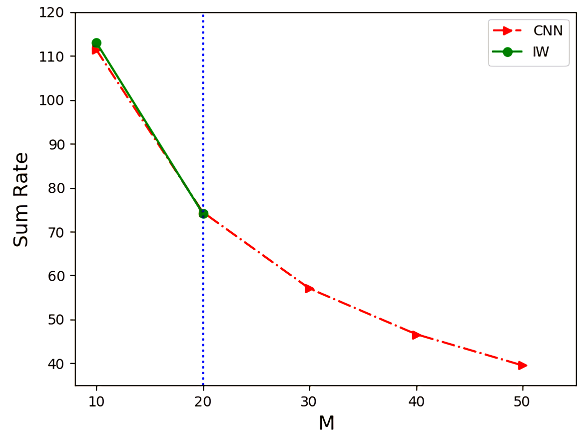


FIGURE 5. Performance of IW and the proposed CNN scheme under different number of users.

it is difficult to meet the real-time requirements in practical application scenarios. For the proposed CNN scheme, with the constraint of not lowering the SR in different network scenarios, it has better stability and shorter operation time than IW.

V. CONCLUSION

In this work, for multiple transmitter-receiver pairs communication network model in the frequency-selective environment, we propose a power allocation scheme based on CNN. The constructed loss function takes into account the SR of all users. The output layer of the CNN model is replaced by several Softmax blocks, and the output of each Softmax block is the ratio of the transmission power of each user on the sub-carriers to the total power. Numerical comparison with traditional power control scheme shows the advantages of our proposed CNN scheme over IW: with the constraint of not lowering the SR, there is no convergence problem and the computational load is significantly reduced.

REFERENCES

[1] T. X. Tran, A. Hajisami, P. Pandey, and D. Pompili, "Collaborative mobile edge computing in 5G networks: New paradigms, scenarios, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 4, pp. 54–61, Apr. 2017.

[2] J. Liu, H. Guo, H. Nishiyama, H. Ujikawa, K. Suzuki, and N. Kato, "New perspectives on future smart FiWi networks: Scalability, reliability, and energy efficiency," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1045–1072, Nov. 2016.

[3] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, and J. Wang, "A first look at cellular machine-to-machine traffic," *SIGMETRICS Perform. Eval. Rev.*, vol. 40, no. 1, p. 65, Jun. 2012.

[4] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013.

[5] J. Huang, Y. Sun, and Q. Chen, "GALLERY: A game-theoretic resource allocation scheme for multicell device-to-device communications underlaying cellular networks," *IEEE Internet Things J.*, vol. 2, no. 6, pp. 504–514, Dec. 2015.

[6] M. Swan, "Sensor Mania! The Internet of Things, wearable computing, objective metrics, and the quantified self 2.0," *J. Sens. Actuator Netw.*, vol. 1, no. 3, pp. 217–253, Nov. 2012.

[7] A. Goldsmith, S. Jafar, N. Jindal, and S. Vishwanath, "Capacity limits of MIMO channels," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 5, pp. 684–702, Jun. 2003.

[8] D. Zhang, Z. Chen, J. Ren, N. Zhang, M. K. Awad, H. Zhou, and X. S. Shen, "Energy-harvesting-aided spectrum sensing and data transmission in heterogeneous cognitive radio sensor network," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 831–843, Jan. 2017.

[9] N. Zhang, N. Lu, N. Cheng, J. W. Mark, and X. S. Shen, "Cooperative spectrum access towards secure information transfer for CRNs," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 11, pp. 2453–2464, Nov. 2013.

[10] N. Zhang, N. Cheng, N. Lu, X. Zhang, J. W. Mark, and X. Shen, "Partner selection and incentive mechanism for physical layer security," *IEEE Trans. Wireless Commun.*, vol. 14, no. 8, pp. 4265–4276, Aug. 2015.

[11] W. Yu, G. Ginis, and J. Cioffi, "Distributed multiuser power control for digital subscriber lines," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 5, pp. 1105–1115, Jun. 2002.

[12] T. Starr, M. Sorbara, J. M. Cioffi, and P. J. Silverman, *DSL Advances*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2003.

[13] M. Kobayashi and G. Caire, "An iterative water-filling algorithm for maximum weighted sum-rate of Gaussian MIMO-BC," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1640–1646, Aug. 2006.

[14] M. Hong and A. Garcia, "Averaged iterative water-filling algorithm: Robustness and convergence," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2448–2454, May 2011.

[15] B. Fallah, F. Wang, and W. Yin, "Synchronous and asynchronous game theoretic analysis for multiuser MIMO systems," in *Proc. 7th Int. Conf. Wireless Commun., Netw. Mobile Comput.*, Sep. 2011, pp. 1–5.

[16] J. Chen, E. Druhl, B. Polepalli Ramesh, T. K. Houston, C. A. Brandt, D. M. Zulman, V. G. Vimalananda, S. Malkani, and H. Yu, "A natural language processing system that links medical terms in electronic health record notes to lay definitions: System development using physician reviews," *J. Med. Internet Res.*, vol. 20, no. 1, p. e26, Jan. 2018.

[17] P. Jiang, H. Fu, H. Tao, P. Lei, and L. Zhao, "Parallelized convolutional recurrent neural network with spectral features for speech emotion recognition," *IEEE Access*, vol. 7, pp. 90368–90377, 2019.

[18] J. Shi, J. Wu, Y. Li, Q. Zhang, and S. Ying, "Histopathological image classification with color pattern random binary hashing-based PCANet and matrix-form classifier," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 5, pp. 1327–1337, Sep. 2017.

[19] X. Li and X. Wu, "Improving long short-term memory networks using maxout units for large vocabulary speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 4600–4604.

[20] G. Gui, F. Liu, J. Sun, J. Yang, Z. Zhou, and D. Zhao, "Flight delay prediction based on aviation big data and machine learning," *IEEE Trans. Veh. Technol.*, to be published, doi: 10.1109/tvt.2019.2954094.

[21] G. Gui, H. Huang, Y. Song, and H. Sari, "Deep learning for an effective nonorthogonal multiple access scheme," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8440–8450, Sep. 2018.

[22] Y. Xu, G. Li, Y. Yang, M. Liu, and G. Gui, "Robust resource allocation and power splitting in SWIPT enabled heterogeneous networks: A robust minimax approach," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10799–10811, Dec. 2019.

[23] H. Huang, Y. Peng, J. Yang, W. Xia, and G. Gui, "Fast beamforming design via deep learning," *IEEE Trans. Veh. Technol.*, to be published, doi: 10.1109/tvt.2019.2949122.

[24] M. Liu, T. Song, J. Hu, J. Yang, and G. Gui, "Deep learning-inspired message passing algorithm for efficient resource allocation in cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 641–653, Jan. 2019.

[25] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.

[26] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to optimize: Training deep neural networks for wireless resource management," in *Proc. IEEE 18th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jul. 2017, pp. 1–6.

[27] N. Zhang, N. Cheng, N. Lu, H. Zhou, J. W. Mark, and X. S. Shen, "Risk-aware cooperative spectrum access for multi-channel cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 3, pp. 516–527, Mar. 2014.

[28] W. Lee, "Resource allocation for multi-channel underlay cognitive radio network based on deep neural network," *IEEE Commun. Lett.*, vol. 22, no. 9, pp. 1942–1945, Sep. 2018.

[29] W. Lee, M. Kim, and D.-H. Cho, "Deep power control: Transmit power control scheme based on convolutional neural network," *IEEE Commun. Lett.*, vol. 22, no. 6, pp. 1276–1279, Jun. 2018.

[30] M. Liu, T. Song, and G. Gui, "Deep cognitive perspective: Resource allocation for NOMA-based heterogeneous IoT with imperfect SIC," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2885–2894, Apr. 2019.

[31] W. Yu and J. Cioffi, "Competitive equilibrium in the Gaussian interference channel," in *Proc. IEEE Int. Symp. Inf. Theory*, Nov. 2002, p. 431.

[32] N. Lu, T. Li, X. Ren, and H. Miao, "A deep learning scheme for motor imagery classification based on restricted Boltzmann machines," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 6, pp. 566–576, Jun. 2017.

[33] J. Winters, "On the capacity of radio communication systems with diversity in a Rayleigh fading environment," *IEEE J. Sel. Areas Commun.*, vol. 5, no. 5, pp. 871–878, Jun. 1987.

[34] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. COMPSTAT*, Jan. 2010, pp. 177–186.

[35] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," Dec. 2012, *arXiv:1212.5701*. [Online]. Available: https://arxiv.org/abs/1212.5701

[36] W. Su, S. Boyd, and E. J. Candes, "A differential equation for modeling Nesterov's accelerated gradient method: Theory and insights," in *Proc. Adv. Neural Inf. Process. Syst.*, Mar. 2015, vol. 3, no. 1, pp. 2510–2518.

[37] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015.

[38] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," *J. Mach. Learn. Res.*, vol. 9, pp. 249–256, May 2010.

MINGJUN DAI received the B.Eng. and the M.Eng. degrees from the Harbin Institute of Technology, Harbin, China, and the Ph.D. degree in electronic engineering from the City University of Hong Kong, in 2012. He joined the Faculty with the College of Information Engineering, Shenzhen University, Shenzhen, China, where he is currently an Associate Professor. His research interests include machine learning, network coding design, distributed computing, distributed storage, and so on.

QINGWEN HUANG is with Shenzhen University, Shenzhen, China. His research interests include machine learning and distributed computing.

ZEXIN LU is currently with The Hong Kong Polytechnic University, Hong Kong. His research interests include machine learning and distributed storage.

BIN CHEN is with Shenzhen University, Shenzhen, China. His research interests include machine learning and networks.

HUI WANG is with Shenzhen University, Shenzhen, China. His research interests include resource allocation, machine learning, and distributed storage.

XUE QIN is with the Department of Computing Sciences, Texas A&M University at Corpus Christi, USA. Her research interests include resource allocation and machine learning.