Deep Neural Network for Resource Management in NOMA Networks

Ning Yang, Haijun Zhang, *Senior Member, IEEE*, Keping Long, *Senior Member, IEEE*, Hung-Yun Hsieh, and Jiangchuan Liu, *Fellow, IEEE*

***Abstract*—Resource management plays a crucial role in improving sum rate of non-orthogonal multiple access (NOMA) networks. However, the traditional resource management methods have considerable complexity, creating a huge challenge in computational efficiency. To handle this challenge, a resource management method is proposed based on a deep neural network (DNN). The key advantage of the method is that the DNN can perform in almost real-time resource allocation because it requires a very simple operation. In this paper, the resource management problem of the NOMA system adopting imperfect successive interference cancellation (SIC) technology at the receivers is studied, including the power allocation stage and the user scheduling stage. For power allocation stage, the generic fully-connected DNN is trained to approximate the power allocation of interior point method (IPM), which not only greatly improves the computational efficiency but also increases the sum rate of the system. Based on the allocated power, the user scheduling algorithm is performed to further increase the system sum rate. Finally, simulation results verify some performances of the proposed algorithms.**

***Index Terms*—Deep neural network (DNN), non-orthogonal multiple access (NOMA), resource management.**

I. Introduction

As THE demand for traffic increases rapidly, the research and development of resource management has received much attention [1]–[4]. Resource management has a critical effect on promoting the sum rate, such as user admission control [5], channel assignment [6], and transmit power control [7]. However, there are still some problems on data transfer rate, network latency, computational efficiency, etc., which call for efficient resource allocation and management technologies.

As a promising network to improve data rate, energy efficiency, capacity and network coverage, non-orthogonal multiple access (NOMA) technology has been widely used in wireless networks [8]–[10]. By employing multiple users in the same subchannel, the energy efficiency and spectral efficiency can be significantly improved in wireless networks [11]–[14].

Meanwhile, in academia, there are many ways to increase system transfer rate and reduce latency, such as optimization theory and machine learning. Optimization-based algorithms have been studied for resource management including gradient descent, heuristic optimization, Lagrangian multiplier and alternating direction method of multipliers (ADMM). A descent method was investigated to achieve an equilibrium between the convergence rate and step size of gradient descent [15]. A profit aggregator-based demand response scheme was utilized to solve the problem of intelligent community resource allocation, which used a novel heuristic framework and executed in the form of the genetic algorithm [16]. The Lagrangian method was used to solve the optimal resources allocation in wireless communication, such as subchannels, radios and time slots [17]. Based on previous ADMM algorithms, a new distributed algorithm was developed that achieved faster convergence and overcame the shortcomings of dual decomposition [18]. In addition, machine learning is known as the driving force of the future technological revolution. It can solve the regression problem at the receiver because of the super nonlinear modeling ability. The stronger computing rate and lower price of deep neural network (DNN) make its more practical in wider scenarios, and there are plenty of works that use DNN to handle various communication in recent years. For example, [19]–[22] investigated promising performance of applying DNN in several tasks, such as, decoding, anomaly detection, signal recovery and resource management.

Although several researches have been investigated for wireless resource management in the NOMA systems [23]–[26], most papers focused on data sum rate of resource allocation in the wireless network. However, the computational efficiency of wireless resource allocation has not been well researched. The artificial intelligence-based algorithms can reduce computational cost and improve the speed of resource allocation. Therefore, we adopt machine learning approach that trains the DNN model to approximate the power allocation of IPM algorithm in the NOMA system, which greatly improves computational efficiency.

Manuscript received July 9, 2019; revised September 15, 2019 and September 20, 2019; accepted September 21, 2019. Date of publication November 6, 2019; date of current version January 15, 2020. This work is supported in part by the National Natural Science Foundation of China under Grant 61822104, Grant 61771044, in part by Beijing Natural Science Foundation under Grant L172025, Grant L172049, in part by the 111 Project No. B170003, and in part by the Fundamental Research Funds for the Central Universities under Grant FRF-TP-19-002C1, RC1631. The review of this article was coordinated by Dr. B. Mao. *(Corresponding author: Haijun Zhang.)*

N. Yang, H. Zhang, and K. Long are with the Beijing Advanced Innovation Center for Materials Genome Engineering, Institute of Artificial Intelligence, Beijing Engineering and Technology Research Center for Convergence Networks and Ubiquitous Services, University of Science and Technology Beijing, Beijing 100083, China (e-mail: b20170322@xs.ustb.edu.cn; haijunzhang@ieee.org; longkeping@ustb.edu.cn).

H.-Y. Hsieh is with the Department of Electrical Engineering, National Taiwan University, Taipei 10617, Taiwan (e-mail: hungyun@ntu.edu.tw).

J. Liu is with the Beijing Advanced Innovation Center for Materials Genome Engineering, University of Science and Technology Beijing, Beijing 100083, China, and also with the School of Computing Science, Simon Fraser University, Burnaby, BC V5A 1S6, Canada (e-mail: jcliu@cs.sfu.ca).

Digital Object Identifier 10.1109/TVT.2019.2951822

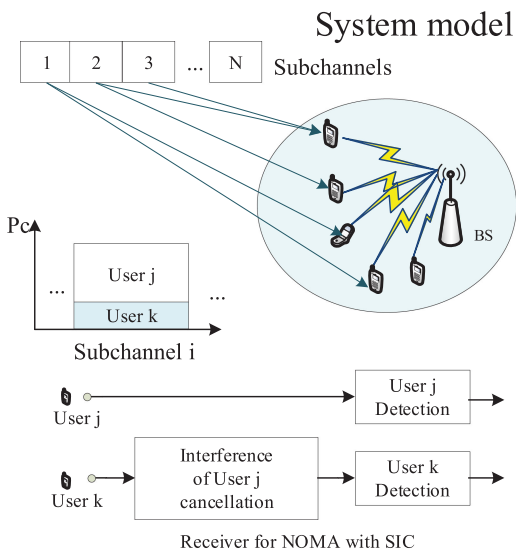

Fig. 1. System model.

The innovation of this paper mainly has the following aspects:

- The resource management is investigated in the NOMA system, including power allocation and user scheduling.
- For the power allocation stage, we use the IPM algorithm to generate the optimal power allocation that can be regard as the set of the label powers. Then the DNN model is trained to approximate the label powers. The trained adaptive model can directly perform power allocation, so the computational efficiency of DNN is much higher than that of traditional algorithms. In addition, the DNN-based power allocation algorithm also improves the sum rate of the NOMA system.
- Based on the power optimization scheme, the user scheduling scheme is treated as a many-to-many two-side matching game problem between the users and the subchannels. The user scheduling algorithm is performed to explore the preferred matching subchannel that further increases the sum rate.

The rest of this paper is organized as follows. The system model of NOMA is discussed in Section II. Section III describes power allocation with DNN for the NOMA system. Section IV describes the user scheduling algorithm for the NOMA system. The simulation results are presented in Section V, and finally the paper is concluded in Section VI.

II. Problem Formulation

A. System Model

We investigate a resource management mechanism of the downlink NOMA system and the system model includes a base station (BS), multiple subchannnels and multiple users. The system model is presented in Fig. 1 and the resource management mechanism is shown in Fig. 2. The set of users represented as $\mathcal{M} = \{1, 2, \ldots, M\}$ and the bandwidth is divided to the set of subchannels, represented as $\mathcal{N} = \{1, 2, \ldots, N\}$. The user j and the user k are multiplexed in the same subchannel i. The BS transmits two signals in the subchannel i with different powers. The user k is chosen to perform imperfect SIC, while user j is not selected.

Assuming that each user can obtain complete channel state information (CSI). Therefore, the SINR in subchannel i for user j is

$$SINR_{ij} = \frac{p_{ij}c_{ij}|g_{ij}|^2}{I_{ij}+\sigma_0^2}, \tag{1}$$

The channel gain in subchannel i for user j is g_{ij} and $|g_{ij}|^2$ is the square of the absolute value of g_{ij}. The power in subchannel i for user j to transmission signal is p_{ij}, and $i \in \mathcal{N}, j \in \mathcal{M}$. We introduce an $N \times M$ user-channel matrix $\mathbf{c}$ where the binary element c_{ij} denotes whether user j is allocated to subchannel i. The number of rows and columns corresponds to the number of subchannels and the number of users, respectively. The noise variance of the system is σ_0^2.

Orthogonal frequency division multiple access (OFDMA) technology allows each subchannel to access only one user, whereas in the NOMA system each subchannel can be assigned to multiple users. The imperfect SIC technology is performed that causes considerable complexity $\mathrm{O}(H^3)$ [27] at the receivers. Therefore, assume that each user can share at most $V = 4$ channels, and each channel can be occupied by at most $H = 3$ users. The key method of imperfect SIC technology is to reduce the interferences caused by the superimposed signals of multiple users. At the receivers, the channel gains are sorted in descending order in subchannel i for all users, which are shown as

$$g_{i1} \geq g_{i2} \geq g_{i3} \geq \cdots \geq g_{ik} \geq \cdots \geq g_{ij} \geq \cdots \geq g_{iM}, \tag{2}$$

If $g_{ik} \geq g_{ij}$, user k can remove the interference signal from user j and successfully decode with imperfect SIC technology. However, the user j cannot be decoded the interference signal from user k, which regarded as noise. The interference in subchannel i for user j is given by

$$I_{ij} = \sum_{l=j+1, l\neq j}^{M} \varepsilon_{il} p_{il} c_{il} |g_{il}|^2 + \sum_{k=1, k\neq j}^{j-1} p_{ik} c_{ik} |g_{ik}|^2, \tag{3}$$

where $k \in \mathcal{M}$, g_{ik} is the channel gain in subchannel i for user k and $|g_{ik}|^2$ are the square of the absolute value of g_{ik}. ε_{il} is the imperfect SIC long-term statistics obtained by user j decoding user l, $g_{il} \leq g_{ij}$ in subchannel i. p_{ik} is the transmission power from the BS to user k in subchannel i. The sum rate in subchannel i for user j is given by

$$R_{ij} = \log_2 \left(1 + \frac{p_{ij}c_{ij}|g_{ij}|^2}{\sum_{l=j+1, l\neq j}^{M} \varepsilon_{il} p_{il} c_{il} |g_{il}|^2 + \sum_{k=1, k\neq j}^{j-1} p_{ik} c_{ik} |g_{ik}|^2 + \sigma_0^2} \right). \tag{4}$$

Assuming there are two users in the channel i, the sum rate in the channel i is

$$R_i = \log_2 \left(1 + \frac{p_{i1}c_{i1}|g_{i1}|^2}{\varepsilon_{i2} p_{i2} c_{i2} |g_{i2}|^2 + \sigma_0^2} \right) + \log_2 \left(1 + \frac{p_{i2}c_{i2}|g_{i2}|^2}{p_{i1}c_{i1}|g_{i1}|^2 + \sigma_0^2} \right). \tag{5}$$

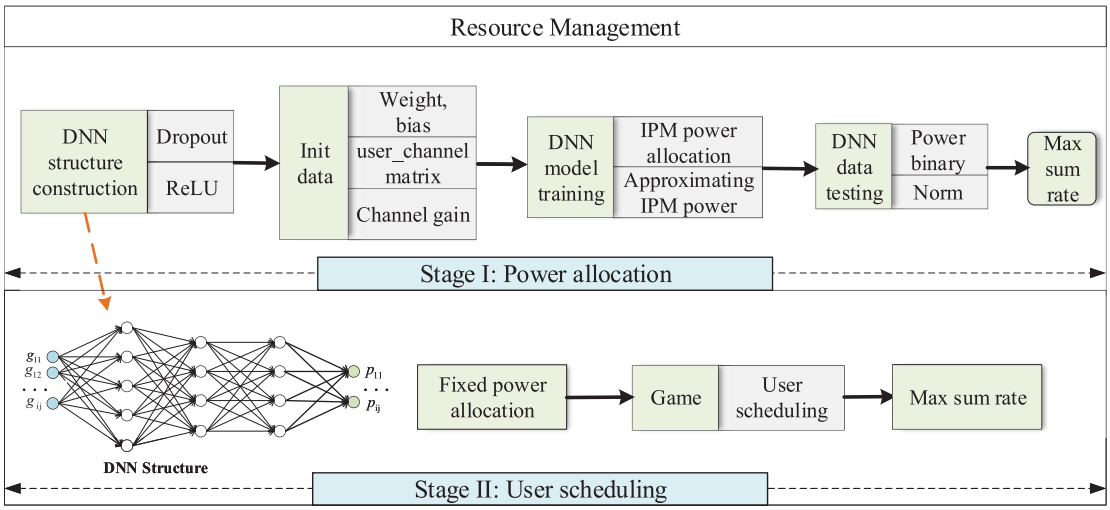

Fig. 2. Resource management mechanism.

where $c_{i1}=1$, $c_{i2}=1$ and $g_{i1}>g_{i2}$. Assuming there are three users in the channel i, the sum rate in the channel i is

$$\begin{aligned} R_i &= \log_2\left(1+\frac{p_{i1}c_{i1}|g_{i1}|^2}{\varepsilon_{i2}p_{i2}c_{i2}|g_{i2}|^2+\varepsilon_{i3}p_{i3}c_{i3}|g_{i3}|^2+\sigma_0^2}\right) \\ &\quad+\log_2\left(1+\frac{p_{i2}c_{i2}|g_{i2}|^2}{p_{i1}c_{i1}|g_{i1}|^2+\varepsilon_{i3}p_{i3}c_{i3}|g_{i3}|^2+\sigma_0^2}\right) \\ &\quad+\log_2\left(1+\frac{p_{i3}c_{i3}|g_{i3}|^2}{p_{i1}c_{i1}|g_{i1}|^2+p_{i2}c_{i2}|g_{i2}|^2+\sigma_0^2}\right) \end{aligned} \tag{6}$$

where $c_{i1}=1$, $c_{i2}=1$, $c_{i3}=1$ and $g_{i1}>g_{i2}>g_{i3}$. Therefore, the total sum rate with imperfect SIC technology in the NOMA system is

$$\begin{aligned} &R_{total} \\ &=\sum_{i=1}^{N}\sum_{j=1}^{M}\log_2\left(1+\frac{p_{ij}c_{ij}|g_{ij}|^2}{\sum_{l=j+1,l\neq j}^{M}\varepsilon_{il}p_{il}c_{il}|g_{il}|^2+\sum_{k=1,k\neq j}^{j-1}p_{ik}c_{ik}|g_{ik}|^2+\sigma_0^2}\right) \end{aligned} \tag{7}$$

B. Problem Formulation

Our goal is to improve the total sum rate and computational efficiency of the system. The complete optimization problem is given by:

$$\begin{aligned} &R_{total} \\ &=\sum_{i=1}^{N}\sum_{j=1}^{M}\log_2\left(1+\frac{p_{ij}c_{ij}|g_{ij}|^2}{\sum_{l=j+1,l\neq j}^{M}\varepsilon_{il}p_{il}c_{il}|g_{il}|^2+\sum_{k=1,k\neq j}^{j-1}p_{ik}c_{ik}|g_{ik}|^2+\sigma_0^2}\right), \end{aligned} \tag{8}$$

$$\begin{aligned} \text{subject to}\quad & C1:\sum_{i=1}^{N}c_{ij}\le H,\quad \forall j\in\mathcal{M}, \\ & C2:\sum_{j=1}^{M}c_{ij}\le V,\quad \forall i\in\mathcal{N}, \\ & C3: c_{ij}\in\{0,1\},\quad \forall i\in\mathcal{N},\ \forall j\in\mathcal{M}, \\ & C4: R_{ij}\ge R_{\min},\quad \forall i\in\mathcal{N},\ \forall j\in\mathcal{M}, \\ & C5:\sum_{i=1}^{N}\sum_{j=1}^{M}p_{ij}=P_c,\quad \forall i\in\mathcal{N},\forall j\in\mathcal{M}, \end{aligned} \tag{9}$$

where H, V and $R_{\min}$ are given constants. The power of BS is P_c. The set of power allocation for all users is $\mathbf{p}$, $\mathbf{p}=\{p_{11},p_{12},\ldots,p_{NM}\}$.

Constraints (C1)–(C2) represent that each user only occupies at most H subchannels and at most V users can share one same subchannel respectively; Constraints C3 means that user scheduling variables are binary; Constraint C4 ensures the minimum QoS requirements for users, where the date rate of any user needs to be larger than the minimum sum rate $R_{\min}$; Constraint C5 restricts that the powers are supposed to satisfy transmission power of the BS.

The resource management is divided into two stages: power allocation and user scheduling, which is shown in Fig. 2. The starting of the resource management is power allocation, where the aim is to improve computing performance and total sum rate of the NOMA network. The objective function of power allocation in the first stage can be obtained as follow

$$\min_{\mathbf{p}}\quad -R_{total}, \tag{10}$$

$$\begin{aligned} \text{subject to}\quad & C1: R_{ij}\ge R_{\min},\forall i\in\mathcal{N},\ \forall j\in\mathcal{M}, \\ & C2:\sum_{i=1}^{N}\sum_{j=1}^{M}p_{ij}=P_c,\ \forall i\in\mathcal{N},\forall j\in\mathcal{M}, \end{aligned} \tag{11}$$

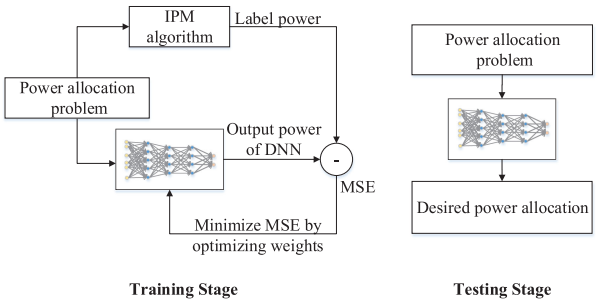

Fig. 3. The training and testing stage of DNN.

where c_{ij} and c_{ik} are given constants, while p_{ij} and p_{ik} are power variables. Constraint C1 ensures the minimum date rate for users, which needs to be larger than the minimum sum rate $R_{\min}$; Constraint C2 ensures that the powers should satisfy transmit power of the BS.

Next, a DNN is trained to approximate the power allocation using the IPM algorithm. The loss function is shown as

$$\min_{\widehat{\mathbf{p}}} ||\widehat{\mathbf{p}} - \arg\max_{\mathbf{p}} R_{total}||^2, \tag{12}$$

where $\widehat{\mathbf{p}}$ is the set of the output powers of DNN. $\mathbf{p}$ is the set of the label powers, which is calculated by the IPM algorithm.

In the second stage, the total sum rate of the NOMA network is further improved by user scheduling algorithm. The objective function of user scheduling is expressed as

$$\begin{aligned} &\max_{\mathbf{c}} \quad R_{total}, \\ \text{subject to} \quad &C1: \sum_{i=1}^{N} c_{ij} \le H, \forall j \in \mathcal{M}, \\ &C2: \sum_{j=1}^{M} c_{ij} \le V, \forall i \in \mathcal{N}, \\ &C3: c_{ij} \in \{0,1\}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}, \end{aligned} \tag{13, 14}$$

where p_{ij} and p_{ik} are given constants, and g_{ij} and g_{ik} are channel gain variables.

III. DNN-Based Power Allocation

The IPM [35] has the advantages in terms of convergence and calculation speed. we use the IPM algorithm to generate the optimal power allocation that can be regard as the set of the label powers. Then the DNN model is trained to approximate the label powers. The trained adaptive model can directly perform power allocation. Therefore, the computing performance of power allocation rate far exceeds traditional algorithms.

A. Deep Neural Network

Deep learning [29]–[31] is a method based on representational learning of data in machine learning. The deep learning structure has multi-layer perceptrons which means it has multiple hidden layers. In this paper, a deep learning scheme is shown in Fig. 3, which consists of the training stage and the testing stage.

Network Structure. The network contains one input layer, three hidden layers and one output layer. The input is the set of channel gains and the output is the set of power allocation, the dimension of which depend on the number of users and the number of subchannels. The three hidden layers have 200, 80 and 80 neurons, and the neuron receives information from the neuron of the previous layer is described as follow.

$$p_u^{(l+1)} = \sum_{v=1}^{L} w_{uv}^{(l)} x_v^{(l)} + b_u^{(l)}, \tag{15}$$

the weights of the neuron is $w_{uv}^{(l)}$, which reflects the relationship between the uth neuron in lth layer and vth neuron in the $(l+1)$th layer, where $l = 1, 2, \ldots, L$. The bias of the neuron is $b_u^{(l)}$, which associates with the uth neuron in the $(l+1)$th layer. In the lth layer, the number of neurons is L.

Activation functions, such as ReLU function, are used in each hidden layer and it can preserve and map the features of the activated neurons. The ReLU function can mitigate gradient dispersion, which is denoted as

$$ReLU(s) = \begin{cases} s & s > 0 \\ 0 & s \le 0. \end{cases} \tag{16}$$

Data generation stage. The input data of the network is generated according to the following manners. The channel gains g_{ij}^t are generated by a standard normal distribution, where t represents tth training sample. The optimized power allocations p_{ij}^t is generated by using IPM for each tuple (g_{ij}^t) with the total sum rate of the system $R_{new} - R_{old} < 10^{-5}$ or the number of iteration > 1000 as termination conditions. The ith training sample is the tuple (g_{ij}^t, p_{ij}^t). We then execute the these process many times to generate the validation set and the training set. The number of the validation set is smaller than the training set.

Training stage. The training process is the process of continuously optimizing the weight of the DNN. In the process, we use RMSprop algorithm and the decay rate is fixed to 0.9. The batch size and learning rate are selected using cross-validation. The goal is to minimize the loss function which reflects square error (MSE) between the label power and the network output power. The training and testing stage are shown in Fig. 3.

Testing stage. The robustness of the trained network is verified. Firstly, the IPM algorithm to generate the optimal power allocation that can be regard as the set of the label powers. we pass the validation set to the network and then we get the output powers of the DNN model. Then, the sum rate of the system using the label powers and the output powers are calculated. Finally, the sum rates of the two algorithms are compared.

B. Approximating Power of IPM by DNN

Suppose channel assignment is fixed, i.e., user-channel matrix $\mathbf{c}$. The IPM is used to allocate power for the NOMA system. The characteristic of the IPM is that it always iterates inside the feasible region and tends to the optimal solution. The IPM method can solve the objective function (10). Proof: Please refer to Appendix

Next, the set of label power is $\mathbf{p}$, which is obtained by using IPM. The set of the network output powers of the tth iteration and $(t+1)$th iteration are $\widehat{\mathbf{p}}^t$ and $\widehat{\mathbf{p}}^{(t+1)}$, respectively. The relationship between $\widehat{\mathbf{p}}^t$ and $\widehat{\mathbf{p}}^{(t+1)}$ can be shown as

$$\widehat{\mathbf{p}}^{(t+1)} = \xi^t(\widehat{\mathbf{p}}^t, \mathbf{p}). \tag{17}$$

The number of iteration is t, and ξ^t is a mapping.

Proposition 3: (Universal approximation proposition for iterative algorithm) The mapping from the label power $\mathbf{p}$ and initialization $\widehat{\mathbf{p}}^0$ to final output $\widehat{\mathbf{p}}^t$.

$$\widehat{\mathbf{p}}^t = f^t(f^{t-1}(\cdots f^1(f^0(\widehat{\mathbf{p}}^0, \mathbf{p}), \mathbf{p})\ldots, \mathbf{p}), \mathbf{p}), \tag{18}$$

be accurately approximated by activation function, which can be expressed as

$$\max|\widehat{\mathbf{p}}^t - \mathbf{p}|^2 \leq \delta, \tag{19}$$

where δ is a very small value.

The algorithm behavior is learned by taking advantage of a trained DNN. The problem of power allocation is nonconvex that may result in the algorithm converges to multiple isolated solutions. Thus, the initialization as input feature of DNN to define the $\widehat{\mathbf{p}}^t$ is necessary. In summary, each iteration of IPM shows continuous mapping that IPM can be approximated arbitrarily well.

The main steps of the approximation algorithm is summarized as bellow: 1) Construct DNN that consists of activation function of ReLU function, normalization to approximate division and multiplication operations; 2) Select a reasonable parameter for the DNN to approximate the power allocation of IPM; 3) Utilize the DNN to approximate the power allocation of IPM by bounding the error propagated.

The DNN power allocation algorithm is formulated that can closely approximate the power allocation of the IPM algorithm for power allocation. The algorithm is described in detail in Algorithm 1.

Algorithm 1: DNN-Based Power Allocation.

Stage I: Data Generation

1: Initialization of the user-channel matrix $\mathbf{c}$ for all the SCs $\forall i \in \{1, 2, \ldots, N\}$ and all the users $\forall j \in \{1, 2, \ldots, M\}$ with $H = 3$ and $V = 4$.
2: Initialize the same power for each user in the different subchannels.
3: Initialize the maximum tolerance ε, the maximum number of iteration L_m and the number of iterations ℓ.
4: **while** $|F_{l+1}(\mathbf{p}) - F_l(\mathbf{p})| > \varepsilon$ or $\ell \leq L_m$ **do**
5: Update p_{ij} according to solve the objective function (10) using IPM.
6: $\ell = \ell + 1$.
7: **end while**

Stage II: DNN approaching IPM power allocation

1: Initialize DNN structure, the weights w and bias b.
2: **for** $m = 1$ to training epoch **do**
3: **for** $n = 1$ to num batch **do**
4: DNN model training: DNN approaching IPM power allocation by minimizing loss function (12).
5: **end for**
6: **end for**
7: DNN data testing: Normalized network output power p_{ij} with $p_{ij} \in \mathbf{p}$ and then $P_c * p_{ij}$.

Algorithm 2: Power Allocation and User Scheduling Algorithm.

1: **repeat**
2: In the first stage: DNN power allocation
3: Update power allocation using a DNN-based power allocation method.
4: In the second stage: User scheduling
5: **for** $i = 1$ to N **do**
6: **for** $j = 1$ to M **do**
7: Select two users (j_1, j_2) and two subchannels (i_1, i_2), where $i_1 \in X(j_1), i_2 \in X(j_2)$, $i_1 \notin X(j_2), i_2 \notin X(j_1)$.
8: **while** $F_{i_2,j_1}^{i_1,j_2}(X') > F_{i_2,j_2}^{i_1,j_1}(X)$ **do**
9: Update user-channel matrix $\mathbf{c}$.
10: **end while**
11: **end for**
12: **end for**
13: **until** Convergence

IV. User Scheduling Algorithm

Each user has different transmission quality on the different subchannels since the multipath fading of the channel. Based on the allocated powers by DNN, the BS uses user scheduling algorithm to select preferred matching subchannel for the users with high channel gain. The user scheduling algorithm can reduce the impact of multipath fading on data sum rate in the NOMA system.

A. User Scheduling Problem

The proposed user scheduling solution is expressed as an optimization problem (13) and solved by maximizing the total sum rate in the NOMA system.

To describe the dynamic scheduling, we first initialize the user-channel matrix $\mathbf{c}$ and the channel gain $\mathbf{g}$, and they both have the same dimension. If user j is assigned to subchannel i, $c_{ij} = 1$, otherwise $c_{ij} = 0$. Given any two users j_1, j_2, any two subchannels i_1, i_2, and two matchings X, X', as well as two function values corresponding to two matching schemes, i.e., $F_{i_2,j_1}^{i_1,j_2}(X')$, $F_{i_2,j_2}^{i_1,j_1}(X)$. If $F_{i_2,j_1}^{i_1,j_2}(X') > F_{i_2,j_2}^{i_1,j_1}(X)$, the user j_2 is scheduled to subchannel i_1 and user j_1 should be assigned to subchannel i_2.

B. User Scheduling Algorithm

Based on the DNN power optimization, the user scheduling algorithm is performed by the BS to explore the preferred matching subchannel for the users.

The proposed power allocation and user scheduling (PAUS) algorithm is shown in Algorithm 2. In the first stage, the power allocation for each user in different subchannels is updated by using a DNN-based power allocation method. In the second stage,

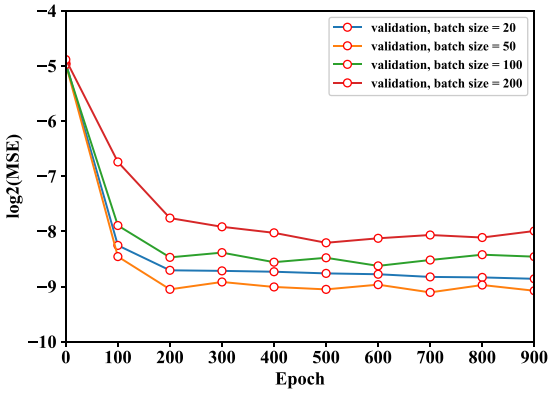

Fig. 4. MSE vs. epoch for different batch sizes.

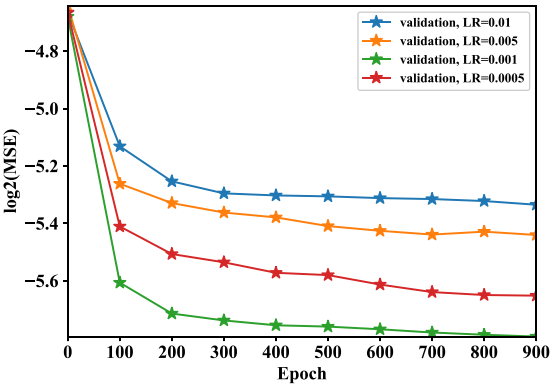

Fig. 5. MSE vs. epoch for different learning rates.

the user is scheduled to the corresponding subchannel according to the preferred matching, and then the user-channel matrix $\mathbf{c}$ is updated. When user scheduling happens, the user-channel matrix will change. The scheduling process will terminate when there is no user scheduling to increase the total sum rate. After some iterations, the total sum rate of Algorithm 2 no longer grows and converges to a fixed value [33]. The convergence of the proposed Algorithm 2 is verified in the next subsection.

C. Convergence Analysis

The convergence performance of Algorithm 2 depends on DNN-based power allocation (which is Algorithm 1) and user scheduling [34].

1) The convergence performance of the DNN-based power allocation
   It can be seen from Fig. 4 and Fig. 5 in Section V that after reaching a certain number of iterations, the MSE tends to be flat, and the DNN power allocation algorithm finally converges to a fixed value. The convergence of the proposed Algorithm 1 is proved.
2) The convergence performance of user scheduling.

After multiple user scheduling operations, the structure of the preferred matching changes is represented as

$$F_0 \to F_1 \to F_2 \to \cdots. \tag{20}$$

We ensure that the sum rate of subchannel i_1 and subchannel i_2 are supposed to satisfy $F_{i_1,j_2}^{i_1,j_2}(X') > F_{i_2,j_2}^{i_1,j_1}(X)$ after executing preferred matching for each user. The total sum rate of the system increases after each match operation ℓ.

$$\begin{aligned}\Delta_{l-1}^{l} &= R_{total}(F_l) - R_{total}(F_{l-1}) \\ &= \sum_{i\in N} R_{total}(F_l) - \sum_{i\in N} R_{total}(F_{l-1}) > 0\end{aligned} \tag{21}$$

The number of potential scheduling user is limited in the many-to-many game. In addition, the total sum rate has an upper bound in the NOMA system since spectrum resources are limited. Hence, the total sum rate convergence after a preferred matching. In summary, the both Algorithm 1 and Algorithm 2 have convergence.

D. Complexity Analysis

1) The Complexity Analysis of Algorithm 1: The complexity performance of Algorithm 1 depends on the DNN-based power allocation. In general, the floating-point operations (FLOPs) can be used to present the time complexity of the DNN. For each fully connected layer of DNN without bias, the number of FLOPs is given by:

$$\mathrm{FLOPs} = (2\mathrm{I_l} - 1)\mathrm{O_l}. \tag{22}$$

where l is the index of the network layer, I_l is the input dimension of the lth layer and O_l is the output dimension of the lth layer. For each fully connected layer of DNN with bias, the number of FLOPs is represented as:

$$\mathrm{FLOPs} = 2\mathrm{I_l}\mathrm{O_l}. \tag{23}$$

Therefore, for all fully connected layers of DNN in our model, the number of FLOPs is:

$$\begin{aligned}FLOPs &= 2\sum_{l=1}^{L+1} \mathrm{I_l}\mathrm{O_l} \\ &= 2(TNMh_1 + h_1h_2 + h_2h_3 + h_3TNM)\end{aligned} \tag{24}$$

where L is the number of network layers and the T is the number of validation samples. The input layer and output layer dimensions of the network are both $N * M$. h_1, h_2, and h_3 are the number of neurons in the three hidden layers.

The complexity performance of IPM is discussed in [36]. Any method of solving an equation requires a precision of $O(L')$ bits for each arithmetic operation, e.g., $Ax = b$ if $\det(A) = O(2^{L'})$. The IPM needs $O((n)^{3.5}L')$ arithmetic operations on $O(L')$ bit numbers, where n is the dimension of power variable $n = NM$, L' is the encoding size of the matrix(E, P_c) and L' is not lager than n. All elements in a matrix E are 1, because E is a coefficient matrix with IPM inequality constraints $(\sum_{i=1}^{N}\sum_{j=1}^{M} p_{ij} = P_c)$. The complexity comparison between DNN-based power allocation and IPM power allocation is as follows:

$$\begin{aligned}&O(2(TNMh_1 + h_1h_2 + h_2h_3 + h_3TNM)) \\ &< O((NM)^{3.5}\mathrm{L}')\end{aligned} \tag{25}$$

Therefore, the proposed DNN model has very low algorithm complexity than IPM.

2) The Complexity Performance of Algorithm 2: The complexity performance of the PAUS algorithm is $O(2\sum_{l=1}^{L+1} \mathrm{I_l O_l} + SMVH(N-H))$. The number of user and the number of subchannel are M and N, respectively. Besides, at most H subchannels can be occupied for each user and at most V users can share one same subchannel. If $MH = NV$, each player keeps fully matched after every user scheduling. The matching $F_{i_2,j_2}^{i_1,j_1}$ has two users and two subchannels. For user i_1, there exists $H(N-H)$ possible combinations of j_1 and j_2 in $F_{i_2,j_2}^{i_1,j_1}$. For the subchannel j_1, the preferred matching $F_{i_2,j_2}^{i_1,j_1}$ with i_1 has $VH(N-H)$ possible combinations. There are $\frac{1}{2}MVH(N-H)$ preferred matchings for i_1 user in each user scheduling. The number of user scheduling is S. The complexity performance of PAUS is expressed by $O(2\sum_{l=1}^{L+1} \mathrm{I_l O_l} + SMVH(N-H))$.

E. Deployment Analysis

Deep learning deployment is becoming essential for the wireless resource management. The DNN provides excellent accuracy, but it also brings heavy computing and storage burden to the system. When deploying deep learning models on network systems, researchers face the following challenges: 1) Due to the communication bandwidth limitation of mobile terminals, downloading large DNN models is still challenging, and even offline downloading of DNNs is difficult to achieve. 2) The large model of DL also imposes strict requirements on the memory size and the computing resources of network system.

Driven by these challenges, there has been a preliminary exploration of DL deployment [30], [31]. The fast fourier transform (FFT)-based DNN was proposed for the training model which can lower asymptotic complexity for computing and storing [38]. The greedy two-dimensional partition was studied, and the workload was allocated to heterogeneous mobile devices to achieve maximum execution parallelism. A pruning scheme with a specific set of lasso normalization was also proposed, which allowed DNN to be deployed more efficiently and compactly on the mobile network [39].

V. Simulation Results

A. Simulation Setup

In this section, simulations have been executed to evaluate the performance of the PAUS algorithm and the DNN-based power allocation scheme with IPM applied, by comparing their performance with pure IPM, maximum power allocation (Max scheme), and random power allocation scheme (Rand scheme). The power of the BS is equally distributed to users and randomly assigned to users in Max scheme and Rand scheme, respectively. For the simulation, the peak power Pc of BS is 43 dbm. We set the minimum QoS of users to 0.01 bps/Hz, the number of subchannels to 10 and the number of users to 10 if there are no special instructions. There exists a Rayleigh fading in the model and the channel gains satisfy the standard normal distribution of zero mean and unit variance. The parameters of DNN are set as shown in Table I. The validation set is utilized to measure the computing performance and sum rate performance of DNN and the training set is used for model training.

TABLE I
Summary of Symbols and Notation

Symbols	Values
Training epoch	300
Batch size	50
Learning rate	0.001
Decay rate	0.9
The number of training samples	20000
The number of validation samples	5000
The dimension of input	$N*M$
The dimension of output	$N*M$
The number of neurons in the 1 th layer	200
The number of neurons in the 2 th layer	80
The number of neurons in the 3 th layer	80

The proposed scheme is performed in Python 3.5 with TensorFlow 1.8.0 with Intel(R) Core(TM)i7-7700@3.6 GHz, NVIDIA GeForce GTX 1050. It is worth noting that the IPM uses the MATLAB, DNN and PAUS use the Python in the paper.

B. Parameter Selection

The learning rate and batch size of the DNN is selected in this section. When MSE tends to 0, it is hard to distinguish them in the real domain. The MSE performance is shown in logarithmic domain. Fig. 4 shows the MSE of the system vs. the number of iterations with different batch sizes for the DNN to convergence. The batch size is adopted 20, 50, 100 and 200 on the validation set to evaluate the MSE performance. The proposed DNN converges within 200 iteration operations. Note that larger batch size makes the convergence rate become slower on the validation set. The smallest MSE achieves with batch size 50 for the validation set. Therefore, the batch size 50 is chosen in this DNN model.

Fig. 5 demonstrates the MSE of the system on the training set and the validation set vs. the number of iterations with different learning rates for the DNN to convergence. The MSE performance is shown in logarithmic domain. The learning rate affects the convergence speed of the model. Normally the higher the learning rate becomes, the faster the neural network learns. When the learning rate is very small, the network may fall into a local optimum. However, the loss will stop falling and it will oscillate repeatedly at a certain position when it is too large and exceeds the extreme value. From Fig. 5, when the number of iterations exceeds 200, the MSE is close to a steady level. when the learning rate is 0.001, the MSE becomes the smallest. Therefore, the learning rate 0.001 is adopted in the model.

C. Computing Performance

The computing performance of the DNN scheme are investigated with 5000 validation samples in the testing stage and the batch size is 100. The average sum rate for each validation sample for DNN and IPM algorithm are shown in the Table II and the Table III. The operation time of DNN on CPU and GPU is calculated for all validation samples because when the validation samples are large enough, the network input dimension has impact on the operation time. However, the operation time of the IPM algorithm in table is just for one sample since we only need to calculate the power allocation of the current scene in

TABLE II
COMPUTATIONAL PERFORMANCE AND SUM RATE FOR DIFFERENT NUMBERS OF USERS

Number	Average sum rate (bps/Hz)			Total testing time (s)				
(N, M)	DNN	IPM	DNN/IPM	DNN (CPU)	DNN (GPU)	IPM	DNN/IPM (CPU)	DNN/IPM (GPU)
(10, 5)	6.799	6.867	99.02%	0.023	0.004	0.161	14.286%	6.558%
(10, 10)	9.217	9.386	98.20%	0.028	0.008	0.430	6.511%	1.860%
(10, 20)	10.934	11.429	95.67%	0.038	0.011	1.815	2.093%	0.606%
(10, 30)	10.641	11.867	89.67%	0.050	0.016	3.878	1.289%	0.413%

TABLE III
COMPUTATIONAL PERFORMANCE AND SUM RATE FOR DIFFERENT NUMBERS OF SUBCHANNELS

Number	Average sum rate (bps/Hz)			Total testing time(s)				
(N, M)	DNN	IPM	DNN/IPM	DNN (CPU)	DNN (GPU)	IPM	DNN/IPM (CPU)	DNN/IPM (GPU)
(10, 20)	10.934	11.429	95.67%	0.038	0.011	1.815	2.093%	0.606%
(15, 20)	11.425	12.284	93.01%	0.043	0.014	9.391	0.458%	0.149%
(20, 20)	11.818	12.832	92.10%	0.054	0.019	26.807	0.201%	0.071%
(25, 20)	12.540	13.910	90.15%	0.066	0.025	82.925	0.080%	0.030%

TABLE IV
COMPUTATIONAL PERFORMANCE FOR DIFFERENT VARIABLES

Variables	Total training time (s)	
(A, B, N, M)	CPU	GPU
(5000, 500, 10, 10)	22.37	9.77
(10000, 500, 10, 10)	43.96	14.57
(15000, 500, 10, 10)	68.24	33.38
(10000, 500, 10, 10)	43.96	14.57
(10000, 2000, 10, 10)	30.05	13.61
(10000, 4000, 10, 10)	25.30	9.84
(10000, 500, 10, 10)	43.96	14.57
(10000, 500, 20, 10)	58.93	31.83
(10000, 500, 30, 10)	75.70	47.66
(10000, 500, 10, 20)	43.96	14.57
(10000, 500, 10, 20)	58.90	31.81
(10000, 500, 10, 30)	75.72	47.68

practice. The operation time of DNN on CPU and GPU grows with an increase of the number of users and the number of subchannels. When the batch size bigger enough, the GPU has better performance than the CPU in the operation time. As expected, the GPU has higher computational efficiency than CPU. It is especially worth noting that the DNN achieves a higher computational efficiency and approximation accuracy. For example, the DNN achieves 89.67% sum rate compares with the IPM algorithm with $M = 30$ and $N = 10$ in the Table II, while obtaining more than one hundred times for computational efficiency. The DNN has reached 90.15%, the total sum rate of IPM algorithm with $M = 20$ and $N = 25$, which only takes up three-thousandth of the IPM operation time on GPU.

In this model, the training of DNN is performed offline. The Table IV shows that total training time of DNN is related to the number of users and subchannel, the number of training samples and batch size. The network training time rises with the grow of the number of users or channels and the number of training samples. As the batch size increases, the training time declines.

D. Sum Rate Performance

The impact of the minimum QoS of users $\mathrm{R}_{\min}$ is demonstrated with different subchannels and users in the IPM algorithm. From Fig. 6 and Fig. 7, the value of $\mathrm{R}_{\min}$ is set from 0 bps/Hz to 0.06 bps/Hz. The total sum rate drops sharply with a grow of the minimum QoS of users $\mathrm{R}_{\min}$ since the users with poor channel conditions need the BS to send more power to meet the minimum QoS requirement.

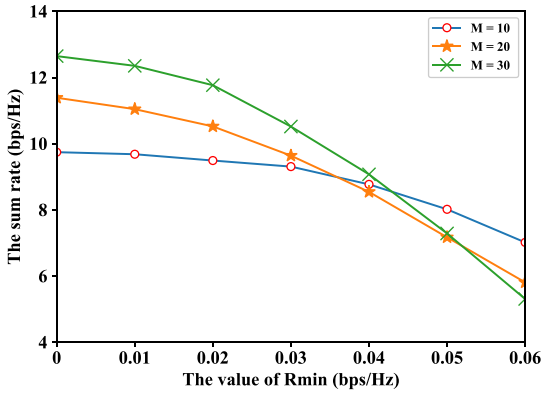

Fig. 6. The sum rate vs. The value of Rmin.

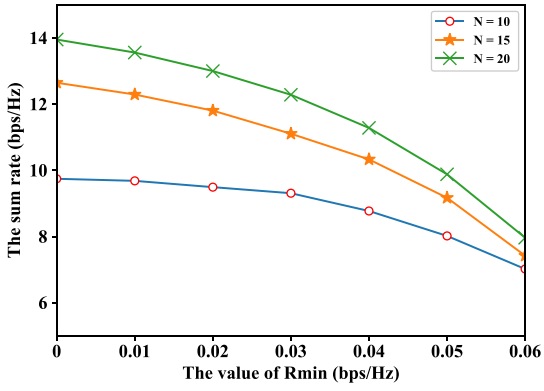

Fig. 7. The sum rate vs. The value of Rmin.

Fig. 8 shows the total sum rate vs. the number of users with different algorithms. The total sum rate of the PAUS and DNN-based methods is evaluated that compared to the following schemes: 1) the IPM; 2) the Max scheme; 3) the Rand scheme.

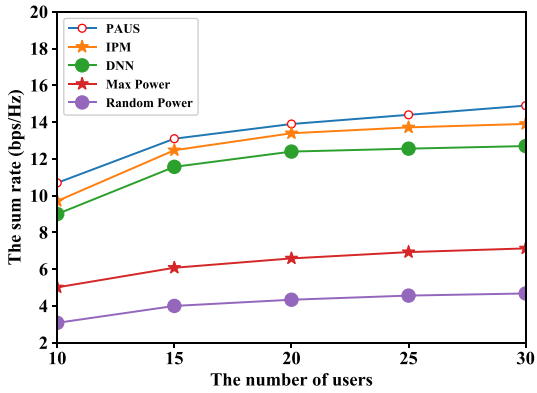

Fig. 8. The sum rate vs. the number of users.

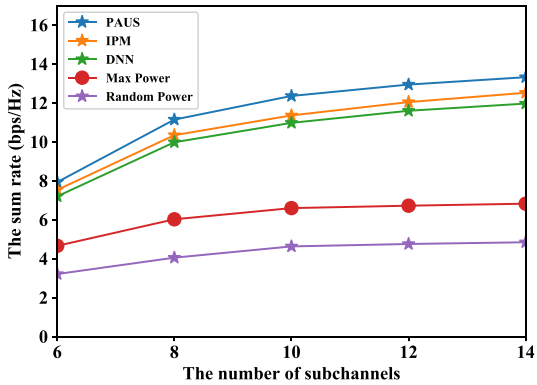

Fig. 9. The sum rate vs. the number of subchannels.

As observed in Fig. 8, The PAUS scheme achieves higher sum rate than that of DNN scheme as well as the three comparison schemes. Besides, DNN has a high approximation proportion for IPM, and it takes much less time than IPM. Fig. 9 shows the total sum rate vs. the number of subchannels with different algorithms. In Fig. 9 that the larger the number of subchannels in the network is, the higher the total sum rate can be obtained. The main reason is that, as the number of subchannels increases, more users can be assigned to achieve more diversity channel gains in subchannels.

VI. Conclusion

In this work, a framework with DNN for downlink resource management in NOMA system was studied and the imperfect SIC technology was adopted at the receivers. The advantage of the proposed DNN-based power allocation and PUSA algorithm is to improve the system sum rate and computing performance. The BS performed power allocation first. To guarantee the higher computational efficiency, DNN was trained to well-approximate the IPM algorithm for power allocation. Based on the issued power, the user scheduling was performed to improve the total sum rate of the system. However, there are many significant issues that need to be handled, which are listed as follows:

1) Exploring unsupervised learning resource management methods for scenarios in which the number of users dynamically changes in real time.
2) How to cache network structure parameters for distributed computing and parallel computing, reducing computation time?

Appendix

Proof: The objective function (10) can be solved by utilizing IPM. The IPM can solve the convex optimization function with inequality constraints and/or equality constraints. The objective and constraint functions need to satisfy twice differentiable [37]. ■

The first order partial derivative of R_{ij} for power p_{ik} is

$$\frac{\partial R_{ij}}{\partial p_{ik}} = \frac{c_{ik}|g_{ik}|^2}{\ln(2)} \cdot \frac{\sum_{k=1,k\neq j}^{j-1} p_{ik}c_{ik}|g_{ik}|^2 + \sigma_0^2}{p_{ij}c_{ij}|g_{ij}|^2 + \sum_{k=1,k\neq j}^{j-1} p_{ik}c_{ik}|g_{ik}|^2 + \sigma_0^2}, \tag{26}$$

The second order partial derivative of R_{ij} for power p_{ik} and power p_{ij} is given by.

$$\frac{\partial^2 R_{ij}}{\partial p_{ik}\partial p_{ij}} = \frac{-c_{ik}c_{ij}|g_{ik}|^2|g_{ij}|^2\left(\sum_{k=1,k\neq j}^{j-1} p_{ik}c_{ik}|g_{ik}|^2 + \sigma_0^2\right)}{\ln(2)\left(p_{ij}c_{ij}|g_{ij}|^2 + \sum_{k=1,k\neq j}^{j-1} p_{ik}c_{ik}|g_{ik}|^2 + \sigma_0^2\right)^2} < 0, \tag{27}$$

where

$$\frac{\partial^2(R_{ij})}{\partial p_{ik}\partial p_{ij}} < 0. \tag{28}$$

Therefore,

$$W\frac{\partial^2(-R_{total})}{\partial p_{ik}\partial p_{ij}} > 0. \tag{29}$$

The objective (10) and constraint function $C1$ in (11) are second-order continuous derivable convex functions. The objective function (10) can be solved by utilizing IPM.

References

[1] B. Di, L. Song, and Y. Li, "Sub-channel assignment, power allocation and user scheduling for non-orthogonal multiple access networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7686–7698, Nov. 2016.
[2] F. Tang *et al.*, "On removing routing protocol from future wireless networks: A real-time deep learning approach for intelligent traffic control," *IEEE Wireless Commun. Mag.*, vol. 25, no. 1, pp. 154–160, Feb. 2018.
[3] Y. Zhou, Z. M. Fadlullah, B. Mao, and N. Kato, "A deep learning based radio resource assignment technique for 5 G ultra dense networks," *IEEE Netw. Mag.*, vol. 32, no. 6, pp. 28–34, Nov. 2018.
[4] B. Xu, Y. Chen, J. R. Carrion, and T. Zhang, "Resource allocation in energy-cooperation enabled two-tier NOMA HetNets towards green 5G," *IEEE Commun. Lett.*, vol. 35, no. 12, pp. 2758–2770, Dec. 2017.
[5] N. Yang, H. Zhang, K. Long, C. Jiang, and Y. Yang, "Spectrum management scheme in fog IoT networks," *IEEE Commun. Mag.*, vol. 56, no. 10, pp. 101–107, Oct. 2018.
[6] H. Zhang, N. Liu, K. Long, J. Cheng, V. C. M. Leung, and L. Hanzo, "Energy efficient subchannel and power allocation for software-defined heterogeneous VLC and RF networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 658–670, Mar. 2018.
[7] H. Zhang, S. Huang, C. Jiang, K. Long, V. C. M. Leung, and H. V. Poor, "Energy efficient user association and power allocation in millimeter-wave-based ultra dense networks with energy harvesting base stations," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 9, pp. 1936–1947, Sep. 2017.

[8] B. Di, L. Song, Y. Li, and G. Li, "Non-orthogonal multiple access for high-reliable and low-latency V2X communications in 5 G systems," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 10, pp. 2383–2397, Oct. 2017.
[9] L. Song, Y. Li, Z. Ding, and H. V. Poor, "Resource management in non-orthogonal multiple access networks for 5G and beyond," *IEEE Netw.*, vol. 31, no. 4, pp. 8–14, Jul. 2017.
[10] N. Kato *et al.*, "The deep learning vision for heterogeneous network traffic control proposal, challenges, and future perspective," *IEEE Wireless Commun.*, vol. 24, no. 3, pp. 146–153, Dec. 2016.
[11] J. Choi, "Effective capacity of NOMA and a suboptimal power control policy with delay QoS," *IEEE Trans. Commun.*, vol. 65, no. 4, pp. 1849–1858, Apr. 2017.
[12] Z. Ding, Z. Yang, P. Fan, and H. V. Poor, "On the performance of non-orthogonal multiple access in 5 G systems with randomly deployed users," *IEEE Signal Process. Lett.*, vol. 21, no. 12, pp. 1501–1505, Dec. 2014.
[13] M. Jia, Z. Yin, D. Li, Q. Guo, and X. Gu, "Toward improved offloading efficiency of data transmission in the IoT-cloud by leveraging secure truncating OFDM," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4252–4261, Jun. 2019.
[14] M. Jia, Z. Yin, Q. Guo, G. Liu, and X. Gu, "Downlink design for spectrum efficient IoT network," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3397–3404, Oct. 2018.
[15] A. S. Bedi and K. Rajawat, "Network resource allocation via stochastic subgradient descent: Convergence rate," *IEEE Trans. Commun.*, vol. 66, no. 5, pp. 2107–2121, May 2018.
[16] T. M. Hansen, R. Roche, S. Suryanarayanan, A. A. Maciejewski, and H. J. Siegel, "Heuristic optimization for an aggregator-based resource allocation in the smart grid," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, 2016, pp. 1785–1794.
[17] A. Ibrahim and A. S. Alfa, "Using Lagrangian relaxation for radio resource allocation in high altitude platforms," *IEEE Trans. Wireless Commun.*, vol. 14, no. 10, pp. 5823–5835, Oct. 2015.
[18] C. Feng, H. Xu, and B. Li, "An alternating direction method approach to cloud traffic management," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 8, pp. 2145–2158, Aug. 2017.
[19] N. Farsad and A. Goldsmith, "Neural network detection of data sequences in communication systems," *IEEE Trans. Signal Process.*, vol. 66, no. 21, pp. 5663–5678, Nov. 2018.
[20] H. Huang, Y. Song, J. Yang, G. Gui, and F. Adachi, "Deep learning based millimeter-wave massive MIMO for hybrid precoding," *IEEE Trans. Veh. Tech.*, vol. 68, no. 3, pp. 3027–3032, Mar. 2019.
[21] Q. Zhou *et al.*, "Enhanced multi-level signal recovery in mobile fronthaul network using DNN decoder," *IEEE Photon. Veh. Lett.*, vol. 30, no. 17, pp. 1511–1514, Sep. 1, 2018.
[22] B. Mao *et al.*, "Routing or computing? The paradigm shift towards intelligent computer network packet transmission based on deep learning," *IEEE Trans. Comput.*, vol. 66, no. 11, pp. 1946–1960, Nov. 2017.
[23] B. Di, L. Song, Y. Li, and Z. Han, "V2X meets NOMA: Non-orthogonal multiple access for 5G enabled vehicular networks," *IEEE Wireless Commun.*, vol. 24, no. 6, pp. 14–21, Dec. 2017.
[24] N. Zhou, X. Zhu, and Y. Huang, "Optimal asymmetric resource allocation and analysis for OFDM-based multidestination relay systems in the downlink," *IEEE Trans. Veh. Technol.*, vol. 60, no. 3, pp. 1307–1312, Mar. 2011.
[25] H. Zhang, H. Liu, J. Cheng, and V. C. M. Leung, "Downlink energy efficiency of power allocation and wireless backhaul bandwidth allocation in heterogeneous small cell networks," *IEEE Trans. Commun.*, vol. 66, no. 4, pp. 1705–1716, Apr. 2018.
[26] H. Xu, Y. Cao, S. Yang, and X. Zhou, "Differential game based link resource management for next generation optical network," *China Commun.*, vol. 14, no. 9, pp. 72–79, Sep. 2017.
[27] L. Dai, B. Wang, Y. Yuan, S. Han, C. I, and Z. Wang, "Non-orthogonal multiple access for 5G: Solutions, challenges, opportunities, and future research trends," *IEEE Commun. Mag.*, vol. 53, no. 9, pp. 74–81, Sep. 2015.
[28] R. Mason and A. Valentinyi, "The existence and uniqueness of monotone pure strategy equilibrium in Bayesian games," Univ. Southampton, Southampton, U.K., 2010.
[29] L. Shao, D. Wu, and X. Li, "Learning deep and wide: A spectral method for learning deep networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 12, pp. 2303–2308, Dec. 2014.
[30] Z. M. Fadlullah, F. Tang, B. Mao, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "State-of-the-art deep learning: Evolving machine intelligence toward tomorrows intelligent network traffic control systems," *IEEE Commun. Surv. Tut.*, vol. 19, no. 4, pp. 2432–2455, May 2017.
[31] 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. Tech.*, vol. 68, no. 1, pp. 641–653, Jan. 2019.
[32] L. Wolsey and G. Nemhauser, "Integer and combinatorial optimization," Hoboken, NJ, USA: Wiley 2014.
[33] S. Bayat, R. H. Y. Louie, Z. Han, B. Vucetic, and Y. Li, "Distributed user association and femtocell allocation in heterogeneous wireless networks," *IEEE Trans. Commun.*, vol. 62, no. 8, pp. 3027–3043, Aug. 2014.
[34] H. Zhang, N. Yang, K. Long, M. Pan, G. K. Karagiannidis, and V. C. M. Leung, "Secure communications in NOMA system: Subcarrier assignment and power allocation," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 7, pp. 1441–1452, Jul. 2018.
[35] W. Lu, M. Liu, S. Lin, and L. Li, "Fully decentralized optimal power flow of multi-area interconnected power systems based on distributed interior point method," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 901–910, Jan. 2018.
[36] N. Karmarkar, "A new polynomial-time algorithm for linear programming," *Proc. 16th Annu. ACM Symp. Theory Comput.*, Nov. 1984, vol. 4, no. 4, pp. 373–395.
[37] S. Boyd and L. Vandenberghe, "Convex optimization," Cambridge, U.K.: Cambridge Univ. Press, 2004.
[38] S. Lin *et al.*, "FFT-based deep learning deployment in embedded systems," in *Proc. Autom. Test Eur. Conf. Exhib.*, 2018, pp. 1045–1050.
[39] J. Mao *et al.*, "MeDNN: A distributed mobile system with enhanced partition and deployment for large-scale DNNs," *Proc. IEEE/ACM Int. Conf. Comput.-Aided Des.*, 2017, pp. 751–756.

Ning Yang is currently working toward the Ph.D. degree with the University of Science and Technology, Beijing, China. Since September 2019, she has been visiting Communications and Networking Laboratory, Northwestern University, Chicago, IL, USA, as a Visiting Research Associate. Her research interests include deep learning, resource allocation, energy efficiency, and spectrum sharing in wireless communications.

Haijun Zhang (M'13–SM'17) is currently a Full Professor with the University of Science and Technology, Beijing, China. He is an Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE TRANSACTIONS ON GREEN COMMUNICATIONS NETWORKING, and IEEE COMMUNICATIONS LETTERS. He is the recipient of the IEEE CSIM Technical Committee Best Journal Paper Award in 2018 and IEEE ComSoc Young Author Best Paper Award in 2017.

Keping Long (SM'06) received the M.S. and Ph.D. degrees from the University of Electronic Science and Technology of China, Chengdu, China, in 1995 and 1998, respectively. From July 2001 to November 2002, he was a Research Fellow with the ARC Special Research Centre for Ultra Broadband Information Networks, University of Melbourne, Parkville, VIC, Australia. He is currently a Professor and the Dean of the School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing, China. He is/was a member of the Editorial Committee of Sciences in China Series F and China Communications. He is/was also a TPC and ISC member for COIN, IEEE IWCN, ICON, and APOC, and Organizing Co-Chair of IWCMC'06, TPC Chair of COIN'05/'08, and TPC Co-Chair of COIN'08/'10. He was awarded the National Science Fund Award for Distinguished Young Scholars of China in 2007 and selected as the Chang Jiang Scholars Program Professor of China in 2008.

Hung-Yun Hsieh received the Ph.D. degree in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA. He is currently a Professor with the Department of Electrical Engineering and the Graduate Institute of Communication Engineering, National Taiwan University, Taipei, Taiwan. His research interest includes wireless communications, mobile computing, and network science.

Jiangchuan Liu (S'01–M'03–SM'08–F'17) received the B.Eng. degree (cum laude) in computer science from Tsinghua University, Beijing, China, in 1999, and the Ph.D. degree in computer science from The Hong Kong University of Science and Technology, Hong Kong, in 2003. He is currently a University Professor with the School of Computing Science, Simon Fraser University, Burnaby, BC, Canada, and a Visiting Professor of Beijing Advanced Innovation Center for Materials Genome Engineering, University of Science and Technology Beijing, Beijing, China.

His research interests include multimedia systems and networks, cloud computing, social networking, online gaming, big data computing, RFID, and Internet of things. He has served on the editorial boards of IEEE/ACM TRANSACTIONS ON NETWORKING, IEEE TRANSACTIONS ON BIG DATA, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, and IEEE INTERNET OF THINGS Journal. He is a Steering Committee member of the IEEE TRANSACTIONS ON MOBILE COMPUTING and Steering Committee Chair of IEEE/ACM IWQoS during 2015–2017.

He is a Fellow of Canadian Academy of Engineering.