IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 4, 2020, accepted January 17, 2020, date of publication January 21, 2020, date of current version February 12, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2968533

Codeword Selection for Concurrent Transmissions in UAV Networks: A Machine Learning Approach

YANG YANG[1,2], (Member, IEEE), ZHEN GAO[3], (Member, IEEE), YUE ZHANG[4], QING YAN[1,2], AND DAZHONG HE[1,2], (Member, IEEE)

[1]School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]Center for Data Science, Beijing University of Posts and Telecommunications, Beijing 100876, China
[3]Advanced Research Institute of Multidisciplinary Science, Beijing Institute of Technology, Beijing 100081, China
[4]HAOHAN Data Technology Company Ltd., Beijing 100142, China

Corresponding author: Dazhong He (hdz@bupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61801035 and Grant 61701027, in part by the Beijing Natural Science Foundation under Grant 4182055 and Grant L182024, in part by the Young Elite Scientists Sponsorship Program by China Association for Science and Technology (CAST) and Chinese Institute of Electronics (CIE), and in part by the Talent Innovation Project of Beijing Institute of Technology (BIT).

ABSTRACT The unmanned aerial vehicles (UAVs) have been widely considered as one of the key applications for future wireless communication systems, where UAVs can be used as aerial base stations (BSs) for coverage extension, transmission improvement, emergency communication, and etc. Against this background, each UAV BS is expected to select the optimal codeword to form directional analog beams, and it is capable of achieving concurrent transmissions from multiple other UAV BSs simultaneously. However, in such a kind of UAV networks, due to the vast number of connected mobile users (MUEs), UAV BSs cannot timely and preciously select the codeword from the pre-defined codebook. Fortunately, machine learning (ML) is suitable for decreasing complexity in codeword selection, because ML could extract features from the data samples acquired in real environments. In this paper, we propose an ML approach to achieve an efficient and low complexity codeword selection for UAV networks. Specifically, we first derive the probabilities that multiple UAV BSs serve one MUE to obtain the average sum rate (ASR) in UAV networks. On that basis, we develop an ML approach to maximize the ASR, where we design a classifier based on support vector machine (SVM), where our ML approach is used for selecting the optimal codeword and maximizing the ASR in UAV networks. Third, we proposed an iterative sequential minimal optimization (SMO) training algorithm to train the data of all links between UAV BSs and MUEs, where the algorithm convergence is also discussed. Finally, we show the comparison between our proposed algorithm and the traditional methods by the simulation results. The simulation at last demonstrate our method is a more efficient solution for obtain a higher performance, where a much lower computational complexity can achieved than the traditional algorithm based on channel estimation.

INDEX TERMS Machine learning, UAV networks, concurrent transmissions, codeword selection, support vector machine.

I. INTRODUCTION

Unmanned aerial vehicles (UAVs) networks play an important role for future wireless communication system [1]. By densely deploying a large number of UAV base stations (BSs), thousands of connections and high transmission rate are supported to provide a variety of local wireless services [2]. By leveraging mmWave large-scale arrays in UAV networks for improved spatial spectrum efficiency, multiple UAV BSs can select different codewords to form directional analog beams aligned to the same target mobile user (MUE) for providing concurrent transmissions simultaneously [3], [4]. The UAV BS can substantially reduce the propagation loss because the mmWave transmission is used in a very short distance to each MUE.

Concurrent transmissions of UAV BSs have already been under investigation as a critical issue in previous works [5]–[8]. In [5], the downlink performance of concurrent

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

transmissions is analyzed under different UAV BS deployments, where the impact of different fading channels is also considered. Further, dynamic concurrent transmissions are analyzed to evaluate the communication qualities in both line of sight (LoS) and Non LoS (NLoS) scenarios [6]. In addition, the UAV BS handover strategy is also discussed in [7] and [8], which leads to seamless concurrent transmissions for MUE. The pervious works list above make contributions for the research of UAV networks. However, those traditional methods meet a great obstacle for performance improvement with the evolution of the UAV networks, especially with the large number of UAV and MUEs.

Recently, machine learning (ML) becomes popular in solving problems for wireless communication. ML is the artificial intelligence (AI) theory for finding features hiding behind the data [9], [10], which means ML is data driven. Utilizing the ML theory, the system could extract the features form UAV networks through the data training of the historical network information [11], [12]. After the data training, an intelligent system can be established and make automatic decisions such as smart throughput improvement [13], smart UAV deployment [14], computing offloading [15], etc. The works above bring a huge performance gain and deeply improve the effectiveness of UAV networks.

Hence, in this paper, based on our previous research work on codeword selection, we aims to show a novel ML approach for concurrent transmissions in UAV networks, where UAV BSs can use a very low computational overhead to effectively select the optimal codeword. In our UAV networks, first, the entire distribution of UAV BSs can be modeled as a heterogeneous Poisson point process (HPPP). Then, the average sum rate (ASR) of MUE under concurrent transmissions can be obtained. Second, we collect data related with all downlink UAV BSs. The data are further prepared for extracting the features by ML, where the support vector machine (SVM) classifier is used. Then, we proposed an iterative sequential minimal optimization (SMO) training algorithm for UAV BSs. During the concurrent transmission for each MUE, the codeword is selected effectively in a low complexity. Finally, we used Google TensorFlow to evaluation the proposed SVM classifier, all the data of the UAV networks is iteratively trained during our simulation. The performance results obtained demonstrate that the iterative SVM classifier gets a approximate performance to the theoretical bound of codeword selection. In addition, compare with the traditional algorithm based on channel estimation (CE), our method can greatly improve the ASR of the UAV networks with a much lower computational complexity.

The remaining of this paper is organized as follows. To begin with, we describe the basic scenario UAV networks, in which the UAV BS are modeled by the HPPP. Then, in section III, the ASR of concurrent transmission is further derived. For section IV, both SVM classifier and the iterative SMO training algorithm are demonstrated in detail. In section V, simulation results for assessing our proposed method are provided. Last, our conclusions are shown in section VI.

II. SCENARIO DESCRIPTION OF UAV NETWORKS

We consider a downlink UAV networks with a lot of UAV BSs randomly distributed. The deployment of UAV BSs is modeled as points of an HPPP Π_{U} with density λ_{U} on two dimensional plane $\Re$ [16]. All UAV BSs are in charge of providing wireless services to MUEs, which include smart phones, tablets, and so on. Each UAV BS or MUE is equipped with massive MIMO to support mmWave transmission. The antenna numbers of UAV BS and MUE are denoted as N_{UAV} and N_{MUE}, respectively. Utilizing beamforming technology, the MUE can be simultaneously served by several UAV BSs through the directional analog beams as shown in Fig. 1, i.e., a scheme of concurrent transmissions is adopted to improve the spectrum efficiency. Then, it is assumed that the MUE could estimate the channel and process the receive signal from each UAV BS by zero forcing (ZF). Further, we can place a typical MUE at the origin on 2-dimensional plane $\Re$. Based on Slivnyak's theory [17], the system performance can be evaluated by this typical MUE because the statistics of HPPP will not be changed. Denote R as the maximum radius of the communication service, the average number of UAV BS in this circular area around the MUE can be written as

$$N_{\mathrm{U}} = \left\lfloor \lambda_{\mathrm{U}} \pi R^2 \right\rfloor, \tag{1}$$

FIGURE 1. Scenario of concurrent transmissions in UAV networks.

where $\lfloor \cdot \rfloor$ is the floor function resulted from the practical communication system. Define the data stream from the kth UAV BS to MUE is $d_{\mathrm{U},k}$, $(1 \le k \le N_{\mathrm{U}})$, and the transmit power of UAV BS is defined as $P_{\mathrm{U},k}$. The downlink signal of UAV BS can be written as

$$\mathbf{s}_{\mathrm{UAV},k} = \mathbf{c}_{\mathrm{U},k} d_{\mathrm{U},k},. \tag{2}$$

where $\mathbf{c}_{\mathrm{U},k} \in \mathbb{C}^{N_{\mathrm{UAV}} \times 1}$ is the analog beam of the kth UAV BS. This beam can directionally point to the MUE by using the radio frequency phase shifters. We use the Saleh-Valenzudela channel model [18] to build the channel model between UAV BS and MUE. The form of the channel satisfies as follows:

$$\mathbf{H}_{\mathrm{U},k} = \gamma \sum_{l=1}^{L} \alpha_{\mathrm{U},k,l} \mathbf{a}_{\mathrm{MUE}} \left(\phi_{\mathrm{MUE},k,l}\right) \left[\mathbf{a}_{\mathrm{U},k} \left(\phi_{\mathrm{U},k,l}\right)\right]^{H} \quad (3)$$

where $\gamma = \sqrt{\frac{N_{\mathrm{UAV}} N_{\mathrm{MUE}}}{L}}$, L is the number of propagation paths, $\alpha_{\mathrm{U},k,l}$ is the complex gain of the lth path with $\alpha_{\mathrm{U},k,l} \sim \mathcal{CN}(0,1)$. $\mathbf{H}_{\mathrm{U},k}$ satisfies $\left\|\mathbf{H}_{\mathrm{U},k}\right\|_F^2 = N_{\mathrm{MUE}} N_{\mathrm{UAV}}$, where $\|\cdot\|_F$ is the Frobenius norm of the matrix. $\mathbf{a}_{\mathrm{MUE}} \left(\phi_{\mathrm{MUE},k,l}\right)$ is the array response of MUE's antenna, while $\mathbf{a}_{\mathrm{U},k} \left(\phi_{\mathrm{U},k,l}\right)$ is the array response of UAV BS's antenna. For lth path, $\phi_{\mathrm{MUE},l}$ is the azimuth angle of arrival (AoA) and $\phi_{\mathrm{U},l}$ is the azimuth angle of departure (AoD). For the uniform linear array (ULA), both the elevation angle of AoA and AoD are set as $\frac{\pi}{2}$, which can be ignored. We also define that the antennas of ULA are deployed along y-axis at each MUE and UAV BS. Hence, the array steering vectors $\mathbf{a}_{\mathrm{MUE}} \left(\phi_{\mathrm{MUE},k,l}\right)$ and $\mathbf{a}_{\mathrm{U},k} \left(\phi_{\mathrm{U},k,l}\right)$ can be written as:

$$\begin{aligned}&\mathbf{a}_{\mathrm{MUE}} \left(\phi_{\mathrm{MUE},k,l}\right)\\ &= \frac{\left[1, e^{j\sigma D_{\mathrm{MUE}} \sin\left(\phi_{\mathrm{MUE},k,l}\right)}, \ldots, e^{j\sigma D_{\mathrm{MUE}} (N_{\mathrm{MUE}}-1) \sin\left(\phi_{\mathrm{MUE},k,l}\right)}\right]^{\mathrm{T}}}{\sqrt{N_{\mathrm{MUE}}}},\end{aligned} \quad (4)$$

$$\begin{aligned}&\mathbf{a}_{\mathrm{U}} \left(\phi_{\mathrm{U},k,l}\right)\\ &= \frac{1}{\sqrt{N_{\mathrm{UAV}}}} \left[1, e^{j\sigma D_{\mathrm{U}} \sin\left(\phi_{\mathrm{U},k,l}\right)}, \ldots, e^{j\sigma D_{\mathrm{U}} (N_{\mathrm{UAV}}-1) \sin\left(\phi_{\mathrm{U},k,l}\right)}\right]^{\mathrm{T}},\end{aligned} \quad (5)$$

where $\sigma = \frac{2\pi}{\lambda}$, λ is the signal wavelength, D_{MUE} and D_{U} are the spacing of two adjacent ULA elements at the MUE and the UAV BS. Then, we get the received signal of MUE as follows:

$$\begin{aligned}&\mathbf{y}_{\mathrm{MUE}}\\ &= \mathbf{g}_{\mathrm{MUE}} \sum_{k=1}^{N_{\mathrm{U}}} \mathbf{H}_{\mathrm{U},k} \mathbf{c}_{\mathrm{U},k} d_{\mathrm{U},k} + \mathbf{g}_{\mathrm{MUE}} \mathbf{n}\\ &= \mathbf{g}_{\mathrm{MUE}} \left[\mathbf{H}_{\mathrm{U},1} \mathbf{c}_{\mathrm{U},1}, \ldots, \mathbf{H}_{\mathrm{U},N_{\mathrm{U}}} \mathbf{c}_{\mathrm{U},N_{\mathrm{U}}}\right] \begin{bmatrix} d_{\mathrm{U},1} \\ \vdots \\ d_{\mathrm{U},N_{\mathrm{U}}} \end{bmatrix} + \mathbf{g}_{\mathrm{MUE}} \mathbf{n},\end{aligned} \quad (6)$$

where $\mathbf{g}_{\mathrm{MUE}} = \begin{bmatrix} g_{\mathrm{MUE},1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & g_{\mathrm{MUE},N_{\mathrm{MUE}}} \end{bmatrix}$, each element in this vector represents the phase shifter value of the antenna. We define $\mathbf{G} = \mathbf{g}_{\mathrm{MUE}} \left[\mathbf{H}_{\mathrm{U},1} \mathbf{c}_{\mathrm{U},1}, \ldots, \mathbf{H}_{\mathrm{U},N_{\mathrm{U}}} \mathbf{c}_{\mathrm{U},N_{\mathrm{U}}}\right]$, after ZF receiver, the signal becomes

$$\mathbf{y}_{\mathrm{MUE,ZF}} = \left[d_{\mathrm{U},1}, \cdots, d_{\mathrm{U},N_{\mathrm{U}}}\right]^{\mathrm{T}} + \left(\mathbf{G}^{\mathrm{H}} \mathbf{G}\right)^{-1} \mathbf{G}^{\mathrm{H}} \mathbf{g}_{\mathrm{MUE}} \mathbf{n}. \quad (7)$$

For the downlink concurrent transmissions, all UAV BSs must choose their own codeword to form the directional analog beams based on the pre-defined codebook $\mathcal{C} = \left\{\mathbf{c}_{\mathrm{U}}^{1}, \mathbf{c}_{\mathrm{U}}^{2}, \ldots, \mathbf{c}_{\mathrm{U}}^{N_{\mathcal{C}}}\right\}$, where $\mathbf{c}_{\mathrm{U}}^{i} \in \mathbb{C}^{N_{\mathrm{UAV}} \times 1}$, $i = 1, 2, \ldots, N_{\mathcal{C}}$, $(N_{\mathcal{C}} > 2)$, where $N_{\mathcal{C}}$ is the number of the codewords.

III. AVERAGE SUM RATE (ASR) OF UAV NETWORKS

The MUE can be simultaneously served by multiple UAV BSs, where the UAV BS density is λ_{U}. Based on the statistical property of HPPP, the probability of UAV BS number in the R-radius circular area satisfies

$$\Pr_{\mathrm{U}} \left(N_{\mathrm{U}} = \tau\right) = \frac{\left(\lambda_{\mathrm{U}} \pi R^2\right)^{\tau}}{\tau!} e^{-\lambda_{\mathrm{U}} \pi R^2}, \quad (\tau = 0, 1, \ldots). \quad (8)$$

The received noise power of MUE satisfies

$$\mathrm{E}\left[\left(\mathbf{G}^{\mathrm{H}} \mathbf{G}\right)^{-1} \mathbf{G}^{\mathrm{H}} \mathbf{g}_{\mathrm{MUE}} \mathbf{n}\right] = \delta^2 N_{\mathrm{MUE}} \left(\mathbf{G}^{\mathrm{H}} \mathbf{G}\right)^{-1}. \quad (9)$$

According to the expression of $\mathbf{G}$, we can get

$$\begin{aligned}&\left(\mathbf{G}^{\mathrm{H}} \mathbf{G}\right)^{-1}\\ &= \frac{1}{N_{\mathrm{MUE}}} \begin{bmatrix} \left\|\mathbf{H}_{\mathrm{U},1} \mathbf{c}_{\mathrm{U},1}\right\|^2 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & \left\|\mathbf{H}_{\mathrm{U},N_{\mathrm{U}}} \mathbf{c}_{\mathrm{U},N_{\mathrm{U}}}\right\|^2 \end{bmatrix}.\end{aligned} \quad (10)$$

Consider the concurrent transmissions of UAV BSs, the ASR of UAV networks can be written as:

$$\begin{aligned}R_{\mathrm{U}} = \lim_{\tau \to \infty} \sum_{k=1}^{\tau} &\left[\frac{\left(\lambda_{\mathrm{U}} \pi R^2\right)^{\tau}}{\tau!} e^{-\lambda_{\mathrm{U}} \pi R^2}\right]\\ &\times \log_2 \left(1 + \frac{P_{\mathrm{U},k} \left\|\mathbf{H}_{\mathrm{U},k} \mathbf{c}_{\mathrm{U},k}\right\|^2}{N_{\mathrm{UAV}} \sigma^2}\right),\end{aligned} \quad (11)$$

where

$$\mathrm{SNR}_{\mathrm{U},k} = \frac{P_{\mathrm{U},k} \left\|\mathbf{H}_{\mathrm{U},k} \mathbf{c}_{\mathrm{U},k}\right\|^2}{\left(N_{\mathrm{UAV}} \delta^2\right)}, \quad (12)$$

it represents the signal to noise ratio (SNR) between the MUE and kth UAV BS.

IV. DATA DRIVEN ITERATIVE SVM CLASSIFIER FOR UAV NETWORKS

According to ML theory, we propose an iterative SVM classifier for UAV networks, which is used to select the codeword from the codebook. The training of our SVM classifier is based on the data collected from the UAV networks beforehand. Then, we propose an iterative SMO algorithm for data training, where the convergence and complexity are also discussed at last.

A. DATA TRAINING SAMPLES OF CONCURRENT TRANSMISSIONS FOR UAV NETWORKS

Generally, SVM classifier is a kind of ML classifier based on big data training samples. For UAV networks, the UAV BSs are densely deployed with the density λ_{U}. All the data samples for training can be obtained through multiple HPPP snapshots with the same UAV BS density. The distribution of each UAV BS is changed in each snapshot. Hence, the transmit power of UAV BS should be chosen in a value range. From the description before, we know the number of the path is L. Hence, the element number of each data sample is $4L+2$. Each element has one path loss value, one power value of UAV BS. There are $2 \times L$ azimuth angles for both AoD and AoA. In addition, the gain is the complex value, which consists of L real part and L imaginary part, respectively. Because different elements in the sample are chosen in different value range (e.g., dBm of power and angle value in $[0, 2\pi)$), we get rid of the difference between each element in the sample with normalization. Define $\Lambda = \{1, 2, \ldots, J\}$. Therefore, we get a database contains each sample as a $1 \times (4L+2)$ vector $\mathbf{x}_j$, $j \in \Lambda$, where J is the number of data samples in the database.

In downlink concurrent transmissions, each UAV BS should choose one analog beam from $N_{\mathcal{C}}$ candidate vectors in $\mathcal{C}$. Each training sample should be mapped to its own optimal analog beams $\mathbf{c}^{i*}$, $i* \in \{1, 2, \ldots, N_{\mathcal{C}}\}$, i.e., if $\mathbf{c}^{i*}$ is chosen, the $\mathrm{SNR}_{\mathrm{U},k}$ can reach a maximum value. Therefore, classify the samples as $N_{\mathcal{C}}$ kinds for ML training and the jth sample is $\mathbf{x}_j$, and it is the feature vector which will be labeled as one kind of codewords in $\mathcal{C}$. If the channel variation happed, UAV BS could judge which codeword is fit for transmission based on the SVM classifier, i.e., the hyperplane for classification between different kinds of data samples. In order to classify all data samples, an easy way is to use $N_{\mathcal{C}}$ traditional SVM classifiers. Nevertheless, consider the impact of the amount of data samples, the samples belong to one kind must be too small compared with the rest of the data. If we directly use those data for training, there may cause a serious bias to our SVM classifier. In order to avoid this situation, we propose an iterative SVM classifier for ML training.

B. DATA DRIVEN ITERATIVE SVM CLASSIFIER WITH SMO ALGORITHM

Before the training, first we divide the data samples into two different kinds based on a sub set $\mathbf{U}$ of $\mathcal{C}$, which has two different codewords. After each classification, a part of samples are classified into one of the two kinds. Next, a new codeword is taken from the codebook and replace the classified one to update $\mathbf{U}$. Further, we classify the part of training data based on this updated $\mathbf{U}$. The SVM classifier iteratively keeps the data training until the rest $N_{\mathcal{C}} - 1$ codewords are taken. During each iteration, the data training is based on the SMO algorithm, where the details of an iteration are shown below.

We take the sub set $\mathbf{U} = \left\{\mathbf{c}_{\mathrm{U}}^1, \mathbf{c}_{\mathrm{U}}^2\right\} \subset \mathcal{C}$ as an example, actually, the classification for any two different vectors of $\mathcal{C}$ are the same. We also label the data samples which classified to $\mathbf{c}_{\mathrm{U}}^1$ as -1, while classified to $\mathbf{c}_{\mathrm{U}}^2$ as 1. The minimization problem of the SVM classifier is shown below:

$$\begin{aligned} &\min \frac{1}{2}\|\mathbf{w}\|^2 \\ &s.t.\ y_j\left[\mathbf{w}^T\phi\left(\mathbf{x}_j\right)+b\right] \geq 1 (j \in \Lambda), \end{aligned} \tag{13}$$

where $\mathbf{w}$ is the vector for separated hyperplane coefficients, $\phi(\cdot)$ is the map of $\mathbf{x}_j$ to the transformed feature space, y_j is the class label, b is the constant item of the hyperplane formula. Then, due to the impact of fading, shadowing, and noise in the practical networks, the training samples must contain the outliers with the noise and deteriorate the selection of support vectors. So we introduce the slack variable, which is the tolerable value of the function margin for the feature point $\xi_j \geq 0$. The hyperplane optimization problem can be rewritten as

$$\begin{aligned} &\min \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{j=1}^{J}\xi_j \\ &s.t.\ y_j\left[\mathbf{w}^{\mathrm{T}}\phi\left(\mathbf{x}_j\right)+b\right] \geq 1-\xi_j\ (j \in \Lambda) \\ &\quad\ \xi_j \geq 0, \end{aligned} \tag{14}$$

where the weight C controls the margin between the support vectors and the hyperplane. According to the Karush-Kuhn-Tucker condition [19]. We further introduce the Lagrange multiplier $\alpha_j \geq 0$, $\beta_j \geq 0$, $j = 1, 2, \ldots, J$, choose the kernel function $\phi(\cdot)$ as linear kernel, then we get the Lagrange function of the original problem as:

$$\begin{aligned} &\mathcal{L}\left(\mathbf{w}, b, \xi_j, \alpha_j, \beta_j\right) \\ &\quad = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{j=1}^{J}\xi_j \\ &\qquad - \sum_{j=1}^{J}\alpha_j\left\{y_j\left[\mathbf{w}^{\mathrm{T}}\mathbf{x}_j+b\right]-1+\xi_j\right\} - \sum_{j=1}^{J}\beta_j\xi_j. \end{aligned} \tag{15}$$

Take the partial derivation of (15) with $\mathbf{w}$, b, and ξ_j, respectively, we can get

$$\frac{\partial\mathcal{L}}{\partial\mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{k=1}^{J}\alpha_k y_k \mathbf{x}_k, \tag{16}$$

$$\frac{\partial\mathcal{L}}{\partial b} = 0 \Rightarrow \sum_{j=1}^{J}\alpha_j y_j = 0, \tag{17}$$

$$\frac{\partial\mathcal{L}}{\partial\xi_j} = 0 \Rightarrow C - \alpha_j - \beta_j = 0. \tag{18}$$

Take the results back to $\mathcal{L}\left(\mathbf{w}, b, \xi_j, \alpha_j, \beta_j\right)$, we have the dual problem as:

$$\begin{aligned} &\min_{\alpha_j,\alpha_k} \frac{1}{2}\sum_{j=1}^{J}\sum_{k=1}^{J}\alpha_j\alpha_k y_j y_k\left\langle\mathbf{x}_j, \mathbf{x}_k\right\rangle - \sum_{j=1}^{J}\alpha_j \\ &s.t.\ C \geq \alpha_j \geq 0, \quad (j \in \Lambda) \\ &\quad \sum_{j=1}^{J}\alpha_j y_j = 0, \end{aligned} \tag{19}$$

where $\left\langle\mathbf{x}_j, \mathbf{x}_k\right\rangle$ is the linear kernel function, which is the scalar product of vector $\mathbf{x}_j$ and $\mathbf{x}_k$. For all the samples, we denote the output function of the feature as

$$\mu_j = \mathbf{w}^{\mathrm{T}}\mathbf{x}_j + b, \quad (j \in \Lambda). \tag{20}$$

From $\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = 0$, we can get $\mathbf{w} = \sum_{k=1}^{J} \alpha_k y_k \mathbf{x}_k$, take it back to (20), we have

$$\mu_j = \sum_{k=1}^{J} \alpha_k y_k \mathbf{x}_k^{\mathrm{T}} \mathbf{x}_j + b, \quad (j \in \Lambda). \tag{21}$$

Consider the constraints of (21) and the outliers, we get α_j as follows:

1) When $\alpha_j = 0$, the data samples for training must belong to one kind of codeword, they are at one side of the support hyperplane, and we have $y_i \mu_j \geq 1$;
2) When $0 < \alpha_j < C$, the data samples are support vectors, all the data are on the support hyperplane, and $y_i \mu_j = 1$ is established;
3) When $\alpha_j = C$, the support vectors locate between the support hyperplane and separated hyperplane, and $y_i \mu_j \leq 1$ is established.

α_j should also satisfy $\sum_{j=1}^{J} \alpha_j y_j = 0$, when the three conditions are not established, we need to simultaneously update two α_j values. Suppose we update α_{j_1} and α_{j_2}, $j_1 \neq j_2$, $j_1, j_2 \in \Lambda$. We have

$$\alpha_{j_1}^{\text{new}} y_1 + \alpha_{j_2}^{\text{new}} y_2 = \alpha_{j_1}^{\text{old}} y_1 + \alpha_{j_2}^{\text{old}} y_2 = \rho, \tag{22}$$

the word 'new' or 'old' in equation (22) represents the value is updated or not. ρ has the constant value. Define $\alpha_{j_2}^{\text{new}} \in [\alpha_L, \alpha_H]$, set $\Xi = \Lambda \backslash \{j_1, j_2\}$, we have

1) If $y_1 y_2 < 0$, $\alpha_{j_1}^{\text{old}} - \alpha_{j_2}^{\text{old}} = \rho$. We can get $\alpha_L = \max(0, -\rho)$, $\alpha_H = \min(C, C - \rho)$;
2) If $y_1 y_2 > 0$, $\alpha_{j_1}^{\text{old}} + \alpha_{j_2}^{\text{old}} = \rho$. So $\alpha_L = \max(0, \rho - C)$, $\alpha_H = \min(C, \rho)$.

Because $\sum_{j=1}^{J} \alpha_j y_j = 0$ needs to be established. Hence, we can get $\alpha_{j_1} y_{j_1} = \alpha_{j_2} y_{j_2} + \sum_{j \in \Xi} \alpha_j y_j$, transpose the equation, then we can get

$$\alpha_{j_1} = -t\alpha_{j_2} + A, \tag{23}$$

where $t = y_{j_1} y_{j_2}$, $A = y_{j_1} \sum_{j \in \Xi} \alpha_j y_j$.

Define $v_{j_1} = \sum_{j \in \Xi} \alpha_j y_j \mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_j$, $v_{j_2} = \sum_{j \in \Xi} \alpha_j y_j \mathbf{x}_{j_2}^{\mathrm{T}} \mathbf{x}_j$, the target function changes as

$$\begin{aligned} f\left(\alpha_{j_1}, \alpha_{j_2}\right) &= \alpha_{j_1} + \alpha_{j_2} - \frac{1}{2}\alpha_{j_1}^2 \mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_1} - \frac{1}{2}\alpha_{j_2}^2 \mathbf{x}_{j_2}^{\mathrm{T}} \mathbf{x}_{j_2} \\ &\quad - y_{j_1} y_{j_2} \alpha_{j_1} \alpha_{j_2} \mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_2} - y_{j_1} \alpha_{j_1} v_{j_1} - y_{j_2} \alpha_{j_2} v_{j_2} + D, \end{aligned} \tag{24}$$

where D represents all the rest item without α_{j_1} and α_{j_2}. Take (23) take back to (24), and notice that $t^2 = 1$, we have

$$\begin{aligned} f\left(\alpha_{j_2}\right) &= A - t\alpha_{j_2} + \alpha_{j_2} - \frac{1}{2}\left(A - t\alpha_{j_2}\right)^2 \mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_1} \\ &\quad - \frac{1}{2}\alpha_{j_2}^2 \mathbf{x}_{j_2}^{\mathrm{T}} \mathbf{x}_{j_2} - t\left(A - t\alpha_{j_2}\right) \alpha_{j_2} \mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_2} \\ &\quad - y_{j_1}\left(A - t\alpha_{j_2}\right) v_{j_1} - y_{j_2} \alpha_{j_2} v_{j_2} + D. \end{aligned} \tag{25}$$

Then,

$$\begin{aligned} \frac{\partial f}{\partial \alpha_{j_2}} &= -t + 1 + tA\mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_1} - \alpha_{j_2} \mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_1} - \alpha_{j_2} \mathbf{x}_{j_2}^{\mathrm{T}} \mathbf{x}_{j_2} \\ &\quad - tA\mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_2} + 2\alpha_{j_2} \mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_2} + y_{j_2} v_{j_1} - y_{j_2} v_{j_2} = 0. \end{aligned} \tag{26}$$

The updated α_{j_2} is

$$\alpha_{j_2}^{\text{new}\prime} = \frac{\left(-y_{j_1} + y_{j_2} + v_{j_1} - v_{j_2} - y_{j_1} A \mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_2} + y_{j_1} A \mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_1}\right) y_{j_2}}{\mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_1} + \mathbf{x}_{j_2}^{\mathrm{T}} \mathbf{x}_{j_2} - 2\mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_2}}. \tag{27}$$

From (20) and $j_1 \neq j_2$, $j_1, j_2 \in \{1, 2, \ldots, J\}$, we know $\mu_{j_2} = \mathbf{w}^{\mathrm{T}} \mathbf{x}_{j_2} + b$.

Then, make $E_{j_i} = \mu_{j_i} - y_{j_i}$, $(i = 1, 2)$, $\zeta = \mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_1} + \mathbf{x}_{j_2}^{\mathrm{T}} \mathbf{x}_{j_2} - 2\mathbf{x}_{j_1}^{\mathrm{T}} \mathbf{x}_{j_2}$. Take v_{j_1}, v_{j_2}, t, and A into (27), it has the form shown below

$$\alpha_{j_2}^{\text{new}\prime} = \alpha_{j_2}^{old} + \left(y_{j_2}/\zeta\right)\left(E_{j_1} - E_{j_2}\right). \tag{28}$$

Further, we combine the constraint of $0 < \alpha_j < C$, we can get

$$\alpha_{j_2}^{\text{new}} = \begin{cases} \alpha_H & \alpha_{j_2}^{\text{new}\prime} < \alpha_L \\ \alpha_{j_2}^{\text{new}\prime} & \alpha_L \leq \alpha_{j_2}^{\text{new}\prime} \leq \alpha_H \\ \alpha_L & \alpha_{j_2}^{\text{new}\prime} > \alpha_H. \end{cases} \tag{29}$$

From (22), we get the updated $\alpha_{j_1}^{\text{new}}$ as

$$\alpha_{j_1}^{\text{new}} = \alpha_{j_1}^{old} + y_{j_1} y_{j_2}\left(\alpha_{j_2}^{old} - \alpha_{j_2}^{new}\right). \tag{30}$$

C. ITERATIVE SMO TRAINING ALGORITHM FOR CODEWORD SELECTION IN UAV NETWORKS

All data samples need to be divided by several separated hyperplanes, so, an iterative SMO training algorithm is proposed for selecting the codeword. The detail of the algorithm is shown in **Algorithm 1**. The convergence of this proposed algorithm is ensured by Osuna's theorem [20], which means the function shown in equation (19) will keep decreasing in each round of the iteration.

In the following, the complexity of the proposed algorithm is analyzed. The traditional codeword selection algorithm judges the ASR caused by every analog beam to select the optimal one for signal transmission, which is the rate-based algorithm. Since each candidate vector is an N_{UAV}-dimensional vector and the dimension of channel $\mathbf{H}_{\text{U},k}$ is $N_{\text{UAV}} \times N_{\text{MUE}}$, the calculation complexity of each receive signal is $\mathcal{O}\left(N_{\text{UAV}}^3 N_{\text{MUE}}^2\right)$. Because there are $N_{\mathcal{C}}$ candidate vectors. The computational complexity is $\mathcal{O}\left[N_{\mathcal{C}} N_{\text{UAV}}^3 N_{\text{MUE}}^2 \left(N_{\text{U}}^2 + N_{\text{U}}\right)/2\right]$.

Based on the proposed algorithm, $\frac{1}{2}(N_{\mathcal{C}})^2$ separated hyperplanes can be obtained by ML training. As the iterative SMO training algorithm can be performed offline, and the number of separated hyperplanes is reduced by half after every iteration, e.g., when classifies them to two different codewords $\mathbf{c}^m$ and $\mathbf{c}^n$, $(m, n \leq N_{\mathcal{C}}, m \neq n)$. If the codeword $\mathbf{c}^m$ is selected, all the separated hyperplanes related to $\mathbf{c}^n$ are removed in the future round of classification. Each round of comparison

Algorithm 1 Iterative SMO Training Algorithm

Initialization:

Set the initial values of λ_{U}, R, $\mathcal{C}$, $\mathbf{U}$, N_{MUE}, N_{UAV}, N_{U}.

Iteration:

while $k \leftarrow \{1, 2, \ldots, N_{\mathrm{U}}\}$ **do**
  Based on $\mathrm{SNR}_{\mathrm{U},k}$, and α_j, set the initial values of data samples $\mathbf{x}_j$, labels y_j, $(j \in \Lambda)$.
  for all $m, n \leq N_{\mathcal{C}}$, $(m \neq n)$ **do**
    Chose α_{j_1} that do not meet the above three conditions of (21);
    Chose α_{j_2} which could make $E_{j_i} - E_{j_2}$ get the maximum value;
    Fixed all $\alpha_j,, j \in J \setminus \{j_1, j_2\}$, calculate ρ, α_L, α_H, and ζ;
    Calculate $\alpha_{j_2}^{\mathrm{new}}$ according to (29), further update E_{j_1} and E_{j_2};
    Update $\alpha_{j_1}^{\mathrm{new}}$ as (30).
    if all α_j meet the above three conditions of (21) **then**
      Store all α_j as coefficients for separated hyperplane $SP^{m,n}$ between $\mathbf{c}^m$ and $\mathbf{c}^n$;
      Update $\mathbf{c}^m$ or $\mathbf{c}^n$ in $\mathbf{U}$.
    end if
  end for
end while

includes N_{UAV} times of multiplication and addition to decide which side the testing sample locates and the complexity of each round is $\mathcal{O}(2N_{\mathrm{UAV}})$. Consider the number change of all separated hyperplanes, the complexity at every UAV BS is $\mathcal{O}\left[\left(1 - \frac{1}{2^{N_{\mathcal{C}}-1}}\right) N_{\mathcal{C}}^2 N_{\mathrm{UAV}}\right]$. Because there are $N_{\mathrm{U}} = \lfloor \lambda_{\mathrm{U}} \pi R^2 \rfloor$ UAV BSs for current transmissions, the complexity based on the separated hyperplane can be further calculated as follows $\mathcal{O}\left[\left(\frac{1}{4} - \frac{1}{2^{N_{\mathcal{C}}+1}}\right) \left(N_{\mathrm{U}}^2 + N_{\mathrm{U}}\right) N_{\mathrm{UAV}} N_{\mathcal{C}}^2\right]$.

V. ANALYSIS OF SIMULATION RESULTS

In order to assess the performance of the iterative SVM classifier, simulation of UAV networks is analyzed in this section. The detail simulation parameters are list in Table 1.

TABLE 1. Simulation parameters.

Parameter	Connotation	Value
λ_{U}	UAV BS density	$2 \times 10^{-5} m^{-2}$
P_{U}	Maximum UAV BS power	24dBm
L	Propagation path number	3
N_{MUE}	Antenna number of MUE	2
N_{UAV}	Antenna number of UAV BS	32
R	Maximum MUE communication radius	50m
$N_{\mathcal{C}}$	Number of all candidate vectors	8

In Fig. 2, we can see as the increase of UAV BS density, the ASR increases. This is due to more UAV BSs can offer more options for user to transmit signals, which increase the probabilities of transmission with small propagation loss. The increased tendency of ASR becomes slow when UAV BS density continues to increase, all MUEs can be served by UAV BSs. The CE algorithm selects the codeword based on the estimated channels. We can see our proposed algorithm achieve a closed ASR the rate-based algorithm which is the theoretical bound, and the proposed algorithm performs better than the CE algorithm. This is due to the reason that more communication links can be used for sample training when the density of UAV BS becomes larger, the feature extraction is much more accurate, which makes the classification much better.

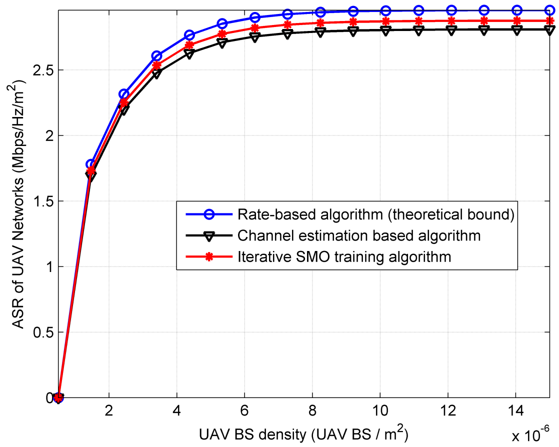

FIGURE 2. ASR of UAV networks vs. UAV BS density.

Fig. 3 illustrates the relationship between ASR of small cell networks and power of UAV BS. We can see as the power of UAV BS increases, the ASR becomes larger because the SNR of receiver increases. The iterative SMO training algorithm outperforms the classical CE algorithm since more features of channel can be 'learned' by the iterative SVM classifier.

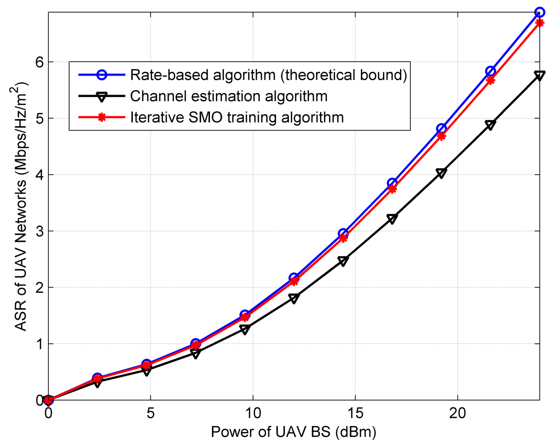

FIGURE 3. ASR of UAV networks vs. power of UAV BS.

Fig. 4 depicts the variation of ASR with different amounts of MUEs. It can be obviously seen that the ASR increases as the MUE number increases. Similar with the analysis above, this is due to features of data samples are fully 'learned' by the training of our proposed algorithm, which causes the SVM

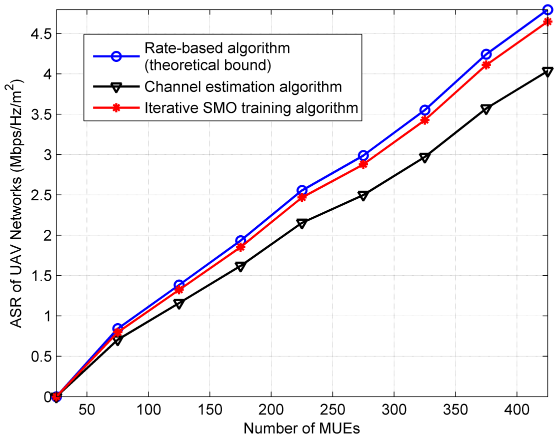

FIGURE 4. ASR of UAV networks vs. number of MUEs.

classifier dynamically adjust to perform codeword selection according to the variation of the UAV networks. Last, CE algorithm has the worst performance, due to it estimates the channel change based on the previous channel states, which only contain a part features of data samples.

Last, Fig. 5 shows the computational complexity comparison between rate-based algorithm and our proposed iterative SMO training algorithm. With the help of SVM classification, the UAV BS can directly select the optimal codeword with the help of the well-trained SVM classifier without traversing the entire codebook. So, the complexity of the proposed algorithm is significantly reduced.

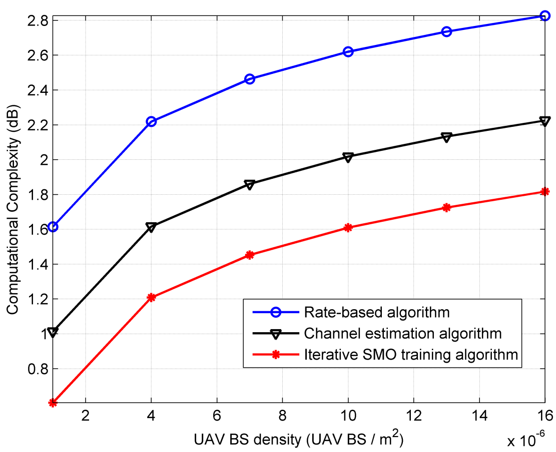

FIGURE 5. Computational complexity comparison.

VI. CONCLUSION

In this paper, we design an ML based codeword selection for concurrent transmissions in UAV networks. By modeling the distributions of UAV BSs as an HPPP, we derive the ASR of UAV networks. Further, we propose an iterative SVM classifier based on the data collected beforehand, which include the parameters of channel and the power of UAV BS. For data training, an iterative SMO training algorithm is used to obtain the separated hyperplanes, which makes UAV BS quickly select the optimal codeword under a low computational complexity. Last, Google TensorFlow is adopted as the ML framework in our simulation. The results show that the proposed algorithm can get a very closed ASR to the performance bound. In addition, the results demonstrate our method is a more efficient solution for obtain a higher performance, where a much lower computational complexity can achieved than the traditional algorithm based on CE.

REFERENCES

[1] M. Liu, G. Gui, N. Zhao, J. Sun, H. Gacanin, and H. Sari, "UAV-aided air-to-ground cooperative non-orthogonal multiple access," *IEEE Internet Things J.*, to be published.

[2] Z. Feng, L. Ji, Q. Zhang, and W. Li, "Spectrum management for MmWave enabled UAV swarm networks: Challenges and opportunities," *IEEE Commun. Mag.*, vol. 57, no. 1, pp. 146–153, Jan. 2019.

[3] C. Zhang, W. Zhang, W. Wang, L. Yang, and W. Zhang, "Research challenges and opportunities of UAV millimeter-wave communications," *IEEE Wireless Commun.*, vol. 26, no. 1, pp. 58–62, Feb. 2019.

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

[5] M. Kamel, W. Hamouda, and A. Youssef, "Performance analysis of multiple association in ultra-dense networks," *IEEE Trans. Commun.*, vol. 65, no. 9, pp. 3818–3831, Sep. 2017.

[6] V. Petrov, D. Solomitckii, A. Samuylov, M. A. Lema, M. Gapeyenko, D. Moltchanov, S. Andreev, V. Naumov, K. Samouylov, M. Dohler, and Y. Koucheryavy, "Dynamic multi-connectivity performance in ultra-dense urban mmWave deployments," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 9, pp. 2038–2055, Sep. 2017.

[7] X. Huang, S. Tang, Q. Zheng, D. Zhang, and Q. Chen, "Dynamic femtocell GNB on/off strategies and seamless dual connectivity in 5G heterogeneous cellular networks," *IEEE Access*, vol. 6, pp. 21359–21368, 2018.

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

[9] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3039–3071, Jul. 2019.

[10] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, "Applications of deep reinforcement learning in communications and networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3133–3174, May 2019.

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

[12] Y. Wang, M. Liu, J. Yang, and G. Gui, "Data-driven deep learning for automatic modulation recognition in cognitive radios," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 4074–4077, Apr. 2019.

[13] A. Koushik, F. Hu, and S. Kumar, "Deep *Q*-learning-based node positioning for throughput-optimal communications in dynamic UAV swarm network," *IEEE Trans. Cogn. Commun. Netw.*, vol. 5, no. 3, pp. 554–566, Sep. 2019.

[14] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 5, pp. 1046–1061, May 2017.

[15] X. Cheng, F. Lyu, W. Quan, C. Zhou, H. He, W. Shi, and X. Shen, "Space/aerial-assisted computing offloading for IoT applications: A learning-based approach," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 5, pp. 1117–1129, May 2019.

[16] M. Haenggi, *Stochastic Geometry for Wireless Networks*. Cambridge, U.K.: Cambridge Univ. Press, 2012.

[17] M. G. Kibria, K. Nguyen, G. P. Villardi, W.-S. Liao, K. Ishizu, and F. Kojima, "A stochastic geometry analysis of multiconnectivity in heterogeneous wireless networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 9734–9746, Oct. 2018.

[18] Z. Wan, Z. Gao, B. Shim, K. Yang, G. Mao, and M.-S. Alouini, "Compressive sensing based channel estimation for millimeter-wave full-dimensional MIMO with lens-array," *IEEE Trans. Veh. Technol.*, to be published.
[19] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2014.
[20] N. Takahashi and T. Nishi, "Global convergence of decomposition learning methods for support vector machines," *IEEE Trans. Neural Netw.*, vol. 17, no. 6, pp. 1362–1369, Nov. 2006.

YANG YANG (Member, IEEE) received the master's and Ph.D. degrees from the School of Telecommunications Engineering, Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2012 and 2015, respectively. From 2015 to 2017, he was a Postdoctoral Researcher with the Tsinghua National Laboratory for Information Science and Technology (TNList), Department of Electronic Engineering, Tsinghua University, Beijing. He is currently an Assistant Professor with the Center of Data Science, BUPT. His research interests include AI-based wireless networks, machine learning, and ultra-dense networks. He was a recipient of the Exemplary Reviewer of the IEEE TRANSACTIONS ON COMMUNICATIONS, in 2018.

ZHEN GAO (Member, IEEE) received the B.S. degree in information engineering from the Beijing Institute of Technology, Beijing, China, in 2011, and the Ph.D. degree in communication and signal processing from the Tsinghua National Laboratory for Information Science and Technology, Department of Electronic Engineering, Tsinghua University, China, in 2016. He is currently an Assistant Professor with the Beijing Institute of Technology. His research interests are in wireless communications, with a focus on multicarrier modulations, multiple antenna systems, and sparse signal processing. He was a recipient of the IEEE Broadcast Technology Society 2016 Scott Helt Memorial Award (Best Paper), the Exemplary Reviewer of the IEEE COMMUNICATIONS LETTERS, in 2016, the *IET Electronics Letters* Premium Award (Best Paper), in 2016, and the Young Elite Scientists Sponsorship Program from the China Association for Science and Technology, from 2018 to 2021.

YUE ZHANG received the B.E. and M.E. degrees from the Beijing University of Posts and Telecommunications (BUPT), in 1983 and 1995, respectively. He is currently a Professor-Level Senior Engineer and has been working in the information and communication technology area for 35 years. As the Chairman and the General Manager of the HAOHAN Data Technology Company Ltd., Beijing, he has been leading the company in growing as a professional supplier of network traffic monitoring and control systems. He is also a part-time Professor with BUPT.

QING YAN is currently a Lecturer with the Beijing University of Posts and Telecommunications (BUPT). She has been with BUPT, since 1996. Her current research interests focus on broadband network traffic monitoring, network traffic identification, mobile internet traffic analysis, and user behavior analysis.

DAZHONG HE (Member, IEEE) received the B.E. and M.E. degrees in information engineering from the Beijing University of Posts and Telecommunications (BUPT), in 1994 and 1997, respectively. He has been working as a Teacher with BUPT, since 1997. His current research interests are in network traffic research, big data analysis, and hardware-based deep learning.