This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2019.2923314, IEEE Transactions on Vehicular Technology

Machine Learning Inspired Codeword Selection for Dual Connectivity in 5G User-centric Ultra-dense Networks

Yang Yang, *Member*, *IEEE*, Xingyi Deng, Dazhong He, Yanan You, *Member*, *IEEE*, and Ruoning Song

***Abstract*—In future 5G user-centric ultra-dense networks (U-UDN), demands of high data rate and high spectrum efficiency (SE) are effectively met by dual connectivity (DC) technology. However, due to huge increase of base stations (BSs) and mobile users (MUs), it becomes difficult for BSs to quickly and precisely select the codeword and provide DC to MUs. Hence, different from some traditional methods, our work aims to improve the network performance using the method of machine learning (ML). First, we model the random distribution of BSs by homogeneous Poisson point processes (HPPPs), where each MU is served by millimeter-wave (mmWave) channel. Second, the probabilities that macro cell BS (MBS) or small cell BS (SBS) serves the MU are further derived to get the average sum rate (ASR) in UUDN. Third, inspired by ML, we utilize an iterative support vector machine (SVM) classifier to select the codewords of BSs, with sequential minimal optimization (SMO) algorithm used for training all link samples in UUDN. Then, an iterative SVM-SMO classification (ISSC) algorithm is proposed to achieve a highly efficient performance of DC, where the convergence and complexity are also discussed. The sample training and simulation at last are evaluted by Google TensorFlow. The simulation verified that our proposed algorithm gets a higher ASR than the traditional channel estimation based (CE-based) algorithm. In addition, the results also show a lower computational complexity can be achieved by the proposed algorithm as well.**

***Index Terms*—User-centric ultra-dense networks, dual connectivity, codeword selection, machine learning, support vector machine**

I. INTRODUCTION

In the next-generation 5G user-centric ultra-dense networks (UUDN), the mobile user (MU) could use dual connectivity (DC) to improve the transmission rate. Both macro cell base station (MBS) and small cell base station (SBS) simultaneously transmit signals to one MU with the help of millimeter-wave (mmWave) massive MIMO technology. In DC, each base station (BS) selects a codeword from a pre-defined codebook to form one directional analog beam to MU, which brings a great performance improvement in downlink transmission of UUDN [1].

DC has already been investigated under different SBS densities before [2], while dynamic DC is further discussed in [3], which leads to a seamless handover of MUs. Those prior works achieve a good enhancement of DC in UUDN. However, with the increasing number of BSs and MUs, it becomes more and more difficulty to use traditional methods [4] to improve the performance, especially for the large random distributed MUs and ultra-dense SBS deployment.

Fortunately, AI-based technologies such as machine learning (ML), deep learning (DL), show a very promising way to enhance the wireless performance. Those technologies utilize a large amount of network state information and extract the network features [5]. Based on those features, BSs are able to perform precise judgement such as smart modulation [6], smart channel state estimation [7], and so on. Those previous works significantly lower the calculation complexity, which also show a great potential performance improving for future explosive MUs and SBSs.

Hence, in this paper, we utilize a novel ML-inspired method in UUDN, where a low complexity and effective codeword selection is proposed for both MBSs and SBSs under DC. First, the large random distributions of MBSs and SBSs are modeled by homogeneous Poisson point processes (HPPPs), then, we derive the average sum rate (ASR) of each MU. Second, all BSs transmit powers and the channel parameters are established as the big database for ML training, then, an iterative support vector machine (SVM) classifier is proposed for codeword selection of each BS. Third, based on the traditional sequential minimal optimization (SMO) algorithm, we proposed an iterative SVM-SMO classification (ISSC) algorithm, where both MBS and SBS can perform a high efficient and low complexity codeword selection in DC. Finally, we use Google TensorFlow to train the data samples and simulate the network. The results verified that our proposed algorithm is very close to the theoretical performance boundary but with a very substantial reduction of calculation complexity. In our simulation, the impact of SBS density and the number of MU to the ASR is discussed, and we also demonstrated the proposed ISSC algorithm can achieve a high ASR of DC than the traditional algorithm based on channel estimation (CE).

The rest of the paper is organized as follows. Section II describes the system model. Section III derives the ASR of DC. In section IV, we analyze the big data iterative SVM classifier and design the ISSC algorithm for DC. Then, simulation is discussed in section V and section VI concludes the paper.

II. SYSTEM MODEL

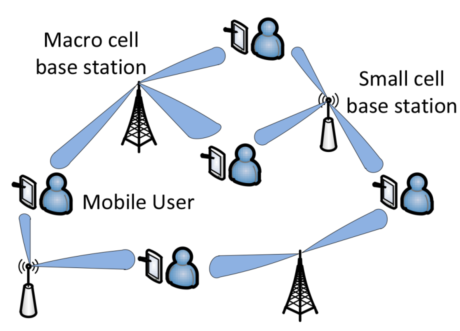

Fig. 1. Scenario of UUDN with dual connectivity

This work is supported by the basic BUPT scientific research project (500418763), and partly supported by Beijing Natural Science Foundation (L182024), National Natural Science Foundation of China (61801035).

Y. Yang, X. Deng, D. He, Y. You and R. Song are with the Center of Data Science, Beijing University of Posts & Telecommunications, Beijing, 100876, China (e-mails: buptyy@bupt.edu.cn, ikachan@126.com, {hdz, youyanan, songrn}@bupt.edu.cn).

As shown in Fig. 1, we consider a 5G downlink UUDN with the random distribution of MBSs and SBSs. The deployments of both MBSs and SBSs satisfy the HPPPs $\Pi_{\rm M}$ and $\Pi_{\rm S}$ on two-dimensional plane $\Re$ with the density $\lambda_{\rm M}$ and $\lambda_{\rm S}$, respectively [8], [9]. Every MU, MBS or SBS is equipped with mmWave massive MIMO, and the number of antenna are defined as $N_{\rm M,BS}$, $N_{\rm S,BS}$ and $N_{\rm UE}$, respectively. In our UUDN, the MU supports DC that the downlink transmission can be served by one MBS and one SBS to provide data transmission at the same time [10]. The perfect channel state information (CSI) is supposed to be obtained since we only focus on the process of the codeword selection. Each MU adopts the zero forcing (ZF) technique in the downlink baseband process. The wireless traffic is assumed as full buffer. According to Slivnyak's theory, a typical MU can be defined as a downlink receiver in the origin on $\Re$, which does not impact the statistical property of the PPP. With the help of phase shifters before the antennas on each MBS/SBS, a directional analog beam can be transmitted to each MU as Fig. 1 shown. The maximum communication radius of the MU is denoted as R. Consider one snapshot of HPPP, we have $N_k = \lfloor \lambda_k \pi R^2 \rfloor$, $k \in \{\rm M, S\}$, where N_k means the number of MBSs (k is M) or SBSs (k is S) in this circular area around the MU. $\lfloor \cdot \rfloor$ is the floor function for a practical UUDN. Different with some selection strategies such as choosing the BS which provides the best transmission quality, or with the minimum attenuation, etc., for simplicity, we define this typical MU can randomly select one MBS and one SBS for DC, with the MBS and SBS powers $P_{\rm M}$ and $P_{\rm S}$, respectively. Each data symbol from the base station (BS) to the MU is d_k, ($k \in \{\rm M, S\}$), which the power of the symbol is normalized as ${\rm E}\left(\|d_k\|^2\right) = 1$, then we get the downlink signal of MU as

$$s_k = c_k d_k, \quad \left(c_k \in \mathbb{C}^{N_k^{\rm B} \times 1}, k \in \{\rm M, S\}\right), \tag{1}$$

where c_k is the analog beam of the BS. Then, we consider the mmWave channel propagation in the UUDN, which is a narrow band clustered channel model based on the extended Saleh-Valenzudela model with following form:

$$\mathrm{H}_k = \gamma_k \sum_{l=1}^{L} \alpha_{k,l} \mathrm{a}_{\rm UE}\left(\phi_{{\rm UE},l}\right) [\mathrm{a}_k\left(\phi_{k,l}\right)]^H, \quad k \in \{\rm M, S\}, \tag{2}$$

where $\gamma_k = \sqrt{\frac{N_k N_{UE}}{L}}$, L is the number of propagation paths, $\alpha_{k,l}$ is the complex gain of the lth path with $\alpha_{k,l}\ \mathcal{CN}(0,1)$. H_k satisfies $\|\mathrm{H}_k\|_F^2 = N_k N_{UE}$, $\|\cdot\|_F$ is the Frobenius norm of the matrix. $\mathrm{a}_{\rm UE}\left(\phi_{{\rm UE},l}\right)$ and $\mathrm{a}_k\left(\phi_{k,l}\right)$ is the antenna array response of MU and BS, respectively. $\phi_{{\rm UE},l}$ and $\phi_{k,l}$ represent the azimuth angle of AoA and AoD in the lth path. The receive signal as

$$\begin{aligned} y_{\rm UE} &= g_{\rm UE}\mathrm{H}_{\rm M}c_{\rm M}d_{\rm M} + g_{\rm UE}\mathrm{H}_{\rm S}c_{\rm S}d_{\rm S} + g_{\rm UE}n \\ &= g_{\rm UE}\begin{bmatrix} \mathrm{H}_{\rm M}c_{\rm M} & 0 \\ 0 & \mathrm{H}_{\rm S}c_{\rm S} \end{bmatrix}\begin{bmatrix} d_{\rm M} \\ d_{\rm S} \end{bmatrix} + g_{\rm UE}n \end{aligned}, \tag{3}$$

where $g_{\rm UE} \in \mathbb{C}^{1 \times N_{\rm UE}}$ is the receive phase vector of the MU, which satisfies $g_{\rm UE} = \left[e^{j\theta_1}, ..., e^{j\theta_{N_{\rm UE}}}\right]$, and each element in this vector represents the phase shifter value of the antenna. Define $\mathrm{G} = g_{\rm UE}\begin{bmatrix} \mathrm{H}_{\rm M}c_{\rm M} & 0 \\ 0 & \mathrm{H}_{\rm S}c_{\rm S} \end{bmatrix}$, then the signal after ZF receiver becomes

$$y_{\rm UE,Z} = [d_{\rm M}, d_{\rm S}]^{\rm T} + \left(\mathrm{G}^{\rm H}\mathrm{G}\right)^{-1}\mathrm{G}^{\rm H} g_{\rm UE} n. \tag{4}$$

Define the codebook $\mathcal{C}$ of each BS in downlink transmission as $\mathcal{C} = \left\{\mathrm{c}^1, \mathrm{c}^2, ..., \mathrm{c}^{N^{\mathcal{C}}}\right\}$, where $N^{\mathcal{C}}$, $\left(N^{\mathcal{C}} > 2\right)$ is the number of the candidate codewords, $\mathrm{c}^i \in \mathbb{C}^{N_{k,\rm BS} \times 1}$ is the analog beam which MBS/SBS ($k \in \{\rm M, S\}$) may use. According to the statistical characteristic of PPP, we get the existence probability of the MBS/SBS number in the R-radius circular area as:

$$\Pr_k\left(N_k = \tau\right) = \frac{\left(\lambda_k \pi R^2\right)^\tau}{\tau!} e^{-\lambda_k \pi R^2}, \quad \tau = 0, 1, ... \ . \tag{5}$$

From (4) and the expression of G, the ASR of MU [11] is:

$$\begin{aligned} R_{\rm UE} &= \sum_{k \in \{\rm M,S\}} \Pr_k\left(N_k \geq 1\right) \log_2\left(1 + {\rm SNR}_k\right) \\ &= \sum_{k \in \{\rm M,S\}} \left(1 - e^{-\lambda_k \pi R^2}\right) \log_2\left(1 + \frac{P_k \|\mathrm{H}_k c_k\|^2}{N_{\rm BS}\sigma^2}\right) \end{aligned}, \tag{6}$$

where ${\rm SNR}_k = P_k\|\mathrm{H}_k c_k\|^2 \big/ \left(N_{\rm BS}\sigma^2\right)$ is the MU signal-to-noise (SNR) of the receive signal from MBS/SBS.

III. Big Data Iterative SVM Classifier in 5G UUDN

A. *Big data iterative SVM classifier with SMO algorithm*

The big data training samples can be obtained based on multiple HPPP snapshots by many times. As we have L propagation paths, there are $2 + 4L$ random real values as the elements in each training sample, which includes the transmit power, the path loss, $2L$ azimuth angles of AoA and AoD, $2L$ real and imaginary parts of the complex gain. Last, each sample in the database is a $1 \times (4L + 2)$ vector x_j, $j \in \{1, 2, ..., J\}$, where J is the total number of the samples in the database. In downlink DC, BS selects one analog beam from $N^{\mathcal{C}}$ candidate codewords in the codebook. Our target is mapping every training sample to an optimal analog beam c^{i*}, $i* \in \left\{1, 2, ..., N^{\mathcal{C}}\right\}$. When c^{i*} is chosen for downlink transmission, the ${\rm SNR}_k$ can reach a maximum value ($k \in \{\rm M, S\}$). Based on the codebook $\mathcal{C}$, we classify the training samples as $N^{\mathcal{C}}$ kinds. With the help of SVM classifier, we can predict the optimal analog beam of the MBS/SBS once the downlink channel condition changed. To classify all the feature vectors, a separating hyperplane should be found between one feature vector and the other vectors. However, as the unbalance of the data, the number of the samples belong to one certain kind will be much smaller than the samples of the rest, which brings a very big bias to the final results and cause the inaccurate prediction. Therefore, we proposed the following big data iterative SVM classifier.

In our proposed SVM classifier, we classify the training samples based on a sub set U of the codebook $\mathcal{C}$. The sub set always contains two candidate codewords. After the classification, the training samples are classified by those two kinds. Then, we take a new candidate codeword from $\mathcal{C}$ and replace the unselected one in U. With the updated U, we continue to classify the part of training samples which were classified to the selected vector in the previous round.

The SVM classifier continues the iteration until all $N^{\mathcal{C}}-1$ hyperplanes are found. In each iteration, we adopt the SMO algorithm for training the samples, where the SVM classifier can achieve a good classification with this algorithm [12]. The detail of one iteration are as follows:

Without loss of generality, we define $U=\{c^m,c^n\}\subset\mathcal{C}$, $(c^m,c^n\in\mathcal{C})$, $(m\neq n)$. Denote all the feature vectors which classified to c^m are labeled as -1, while classified to c^n are labeled as 1. Then, we have the following hyperplane optimization problem:

$$\begin{array}{ll} \min & \frac{1}{2}\|w\|^2+C\sum\limits_{j=1}^{J}\xi_j \\ s.t. & y_j\left[w^T\phi(x_j)+b\right]\geq 1-\xi_j \quad (j\in\{1,2,...,J\}) \\ & \xi_j\geq 0 \quad (j\in\{1,2,...,J\}) \end{array}, \tag{7}$$

where w is the vector of separating hyperplane coefficients, $\phi(\cdot)$ is the map of x_j to the transformed feature space, y_j is the class label, b is the constant item of the hyperplane formula, the feature point $\xi_j\geq 0$, the weight C controls the margin between the support vectors and the hyperplane. The above optimization problem satisfies KKT (Karush-Kuhn-Tucker) condition [13]. Then, we introduce the Lagrange multiplier $\alpha_j\geq 0$, $\beta_j\geq 0$, j=1, 2, ..., J, choose the kernel function $\phi(\cdot)$ as linear kernel, the Lagrange function of the original problem as:

$$\begin{array}{l} \mathcal{L}(w,b,\xi_j,\alpha_j,\beta_j)=\frac{1}{2}\|w\|^2+C\sum\limits_{j=1}^{J}\xi_j \\ -\sum\limits_{j=1}^{J}\alpha_j\left\{y_j\left[w^Tx_j+b\right]-1+\xi_j\right\}-\sum\limits_{j=1}^{J}\beta_j\xi_j \end{array}. \tag{8}$$

Take the partial derivation of (8) with w, b, and ξ_j, then, take the results back to $\mathcal{L}(w,b,\xi_j,\alpha_j,\beta_j)$, then we have the dual problem as:

$$\begin{array}{ll} \min\limits_{\alpha_j,\alpha_k} & \frac{1}{2}\sum\limits_{j=1}^{J}\sum\limits_{k=1}^{J}\alpha_j\alpha_k y_j y_k\langle x_j,x_k\rangle-\sum\limits_{j=1}^{J}\alpha_j \\ s.t. & C\geq\alpha_j\geq 0,(j\in\{1,2,...,J\}) \\ & \sum\limits_{j=1}^{J}\alpha_j y_j=0 \end{array}, \tag{9}$$

where $\langle x_j,x_k\rangle$ is the linear kernel function, which is the scalar product of vector x_j and x_k. For all the samples, we denote the output function of the feature as

$$\mu_j=w^Tx_j+b,\quad j=1,2,...,J\ . \tag{10}$$

From $\frac{\partial\mathcal{L}}{\partial w}=0$, we can get $w=\sum\limits_{j=1}^{J}\alpha_j y_j x_j$, take it back to (10), we have $\mu_j=\sum\limits_{j=1}^{J}\alpha_j y_j x_j^T x_j+b,$, $j=1,2,...,J$. Then, we get the following three conditions:

1) When $\alpha_j=0$, the sample belongs to a certain candidate codeword, it locates at one side of the support plane, and we have $y_i\mu_j\geq 1$;
2) When $0<\alpha_j<C$, the sample is the support vector, it locates on the support plane, and we have $y_i\mu_j=1$;
3) When $\alpha_j=C$, the support vector locates between the separated hyperplane and support plane, $y_i\mu_j\leq 1$.

α_j should also satisfies $\sum\limits_{j=1}^{J}\alpha_j y_j$=0, when the three conditions is not established, we need to simultaneously update two α_j values. Suppose we update α_{j_1} and α_{j_2}, $j_1\neq j_2$, $j_1,j_2\in\{1,2,...,J\}$. We have

$$\alpha_{j_1}^{\mathrm{new}}y_1+\alpha_{j_2}^{\mathrm{new}}y_2=\alpha_{j_1}^{\mathrm{old}}y_1+\alpha_{j_2}^{\mathrm{old}}y_2=\rho, \tag{11}$$

where the subscript 'new' and 'old' mean the value before and after update, ρ is constant. From the first constraint of (9), define $\alpha_{j_2}^{\mathrm{new}}\in[\alpha_L,\alpha_H]$, set $\Xi=\{1,2,...,J\}\setminus\{j_1,j_2\}$, we have

1) When $y_1y_2<0$, $\alpha_{j_1}^{\mathrm{old}}-\alpha_{j_2}^{\mathrm{old}}=\rho$. So $\alpha_L=\max(0,-\rho)$, $\alpha_H=\min(C,C-\rho)$;
2) When $y_1y_2>0$, $\alpha_{j_1}^{\mathrm{old}}+\alpha_{j_2}^{\mathrm{old}}=\rho$. So $\alpha_L=\max(0,\rho-C)$, $\alpha_H=\min(C,\rho)$.

From $\sum\limits_{j=1}^{J}\alpha_j y_j$=0 we know $\alpha_{j_1}y_{j_1}$=$\alpha_{j_2}y_{j_2}+\sum\limits_{j\in\Xi}\alpha_j y_j$, multiply y_{j_1} at both sides, we have $\alpha_{j_1}=-t\alpha_{j_2}+A$, where t=$y_{j_1}y_{j_2}$, $A=y_{j_1}\sum\limits_{j\in\Xi}\alpha_j y_j$. Then, the updated α_{j_2} as

$$\alpha_{j_2}^{\mathrm{new}\prime}=\frac{\left(-y_{j_1}+y_{j_2}+v_{j_1}-v_{j_2}-y_{j_1}Ax_{j_1}^{\mathrm{T}}x_{j_2}+y_{j_1}Ax_{j_1}^{\mathrm{T}}x_{j_1}\right)y_{j_2}}{x_{j_1}^{\mathrm{T}}x_{j_1}+x_{j_2}^{\mathrm{T}}x_{j_2}-2x_{j_1}^{\mathrm{T}}x_{j_2}}. \tag{12}$$

Make $E_{j_i}=\mu_{j_i}-y_{j_i}$, $(i=1,2)$, ζ=$x_{j_1}^{\mathrm{T}}x_{j_1}+x_{j_2}^{\mathrm{T}}x_{j_2}-2x_{j_1}^{\mathrm{T}}x_{j_2}$ and consider the expression of t, A, v_{j_1} and v_{j_2}, and bring all those equations back to (12), we have

$$\alpha_{j_2}^{\mathrm{new}\prime}=\alpha_{j_2}^{old}+(y_{j_2}/\zeta)(E_{j_1}-E_{j_2}). \tag{13}$$

According to the constraint of $0<\alpha_j<C$, we can get

$$\alpha_{j_2}^{\mathrm{new}}=\begin{cases} \alpha_H & \alpha_{j_2}^{\mathrm{new}\prime}<\alpha_L \\ \alpha_{j_2}^{\mathrm{new}\prime} & \alpha_L\leq\alpha_{j_2}^{\mathrm{new}\prime}\leq\alpha_H \\ \alpha_L & \alpha_{j_2}^{\mathrm{new}\prime}>\alpha_H \end{cases} \tag{14}$$

From (11), we get the updated $\alpha_{j_1}^{\mathrm{new}}$ as

$$\alpha_{j_1}^{\mathrm{new}}=\alpha_{j_1}^{old}+y_{j_1}y_{j_2}\left(\alpha_{j_2}^{old}-\alpha_{j_2}^{new}\right) \tag{15}$$

B. Iterative SVM-SMO classification (ISSC) algorithm for codeword selection

Based on the analysis before, an iteration algorithm is used for classifying all training samples belong to the codebook $\mathcal{C}$, which the output will be the coefficients of all separating hyperplanes. The detail steps of our algorithm are shown as **Algorithm 1**. In addition, there are $N^{\mathcal{C}}$ codewords in the codebook $\mathcal{C}$, so $\frac{1}{2}\left(N^{\mathcal{C}}\right)^2$ separating hyperplanes can be obtained with the proposed ISSC algorithm. Then, the MBS/SBS can select the optimal codeword by directly referring those separating hyperplanes instead of some traditional methods such as calculating and comparing SNRs or ASRs for all codewords.

The codeword selection relies on the separating hyperplanes, which are generated by the model of our proposed ISSC algorithm. The big data samples are required for our training procedure, those samples are obtained by the collection of all BSs in a long observation time.

Algorithm 1 Iterative SVM-SMO classification (ISSC) algorithm

Initialization:

Initialize λ_{M}, λ_{S}, N_{UE}, $N_{\mathrm{M,BS}}$, $N_{\mathrm{S,BS}}$, R, $\mathcal{C}$, $U=\{c^m, c^n\}$, $(m=1, n=2)$, J.

Loop:

while $k \leftarrow \{\mathrm{M,S}\}$ **do**
  Initialize training samples x_j, labels y_j according to SNR_k, and α_j, $(j \in \{1, 2, ..., J\})$.
  for all $m, n \leq N^{\mathcal{C}}$, $(m \neq n)$ **do**
    Select α_{j_1} that not satisfies the above three conditions;
    Select α_{j_2} which has the maximum $E_{j_i} - E_{j_2}$;
    Fixed all α_j, $j \in J\backslash\{j_1, j_2\}$, calculate ρ, α_L, α_H, and ζ;
    Calculate $\alpha_{j_2}^{\mathrm{new}}$ in (14), then update E_{j_1} and E_{j_2};
    Update $\alpha_{j_1}^{\mathrm{new}}$ in (15).
    if all α_j satisfies the above three conditions **then**
      Store all α_j as coefficients of separating hyperplane $SP^{m,n}$ between c^m and c^n;
      Update i_U^1 or i_U^2 in U.
    end if
  end for
end while

Output: The coefficients α_j, $(j \in \{1, 2, ..., J\})$ of all separating hyperplanes $SP^{m,n}$ between each two codewords c^m and c^n, $\left(m, n \leq N^{\mathcal{C}}, m \neq n\right)$.

C. Convergence and Complexity Analysis

As the proposed ISSC algorithm is based on the two Lagrange multipliers α_{j_1} and α_{j_2} at every two steps, which α_{j_1} that not satisfies the above three conditions of (10). Then, according to Osuna's theorem, each iteration will decrease the target function (9), i.e., the convergence is guaranteed [14].

Then, we compare the computational complexity of our proposed algorithm with CE-based algorithm and ASR-based algorithm. The CE-based algorithm can select the best codeword in the codebook according to the perfect CSI of each channel. Since each codeword is a $N_{k,\mathrm{BS}}$-dimensional vector and the antenna numbers of each BS and UE are $N_{k,\mathrm{BS}}$ and N_{UE}, respectively, the calculation complexity is $\mathcal{O}\left(N_{k,BS}^2 N_{UE}\right)$, $(k \in \{\mathrm{M,S}\})$. The ASR-based algorithm calculates the ASR caused by every codeword in the codebook. Then, the algorithm selects the best one for signal transmission. As the dimension of channel H_k is $N_{k,\mathrm{BS}} \times N_{\mathrm{UE}}$, the calculation complexity is $\mathcal{O}\left(N_{k,BS}^3 N_{UE}^2\right)$. Consider the number of BSs and codewords are N_k and $N^{\mathcal{C}}$, respectively. The complexity of ASR-based algorithm is $\mathcal{O}\left[N^{\mathcal{C}} \sum\limits_{k\in\{\mathrm{M,S}\}} \left(\left({N_k}^2 + N_k\right) N_{k,BS}^3 N_{UE}^2\right) \Big/ 2\right]$. The computational complexity of CE-based algorithm is $\mathcal{O}\left[N^{\mathcal{C}} \sum\limits_{k\in\{\mathrm{M,S}\}} \left(\left({N_k}^2 + N_k\right) N_{k,BS}^2 N_{UE}\right) \Big/ 2\right]$.

Based on the proposed algorithm, $\frac{1}{2}\left(N^{\mathcal{C}}\right)^2$ separating hyperplanes are generated for beam selection. The training of ISSC algorithm is usually completed beforehand, so the beam selection is performed by codewords in pairwise in $\mathcal{C}$. The number of separating hyperplanes decrease a half after each comparison, the complexity at every BS is $\mathcal{O}\left[\left(1 - \frac{1}{2^{N^{\mathcal{C}}-1}}\right)\left(N^{\mathcal{C}}\right)^2 N_k\right]$, $(k \in \{\mathrm{M,S}\})$. The number of BSs is N_k, $(k \in \{\mathrm{M,S}\})$. Hence, the complexity of codeword selection based on the separating hyperplane satisfies

$$\begin{aligned}&\mathcal{O}\left[\sum_{k\in\{\mathrm{M,S}\}} \sum\nolimits_1^{N_k} \left(\frac{1}{2} - \frac{1}{2^{N^{\mathcal{C}}}}\right)\left(N^{\mathcal{C}}\right)^2 N_{k,\mathrm{BS}}\right] \\ &= \mathcal{O}\left[\left(N^{\mathcal{C}}\right)^2 \left(\frac{1}{4} - \frac{1}{2^{N^{\mathcal{C}}+1}}\right) \sum_{k\in\{\mathrm{M,S}\}} \left({N_k}^2 + N_k\right) N_{k,\mathrm{BS}}\right]\end{aligned}. \tag{16}$$

IV. Simulation Discussion

In this section, the performance of our proposed iterative SVM classifier is evaluated by the simulation. To accelerate the training procedure of the proposed ISSC algorithm, we use Google TensorFlow as the basic simulation structure, which is a very popular ML frame to provide many powerful libraries and functions in AI area. The hardware acceleration platform is based on the computing cluster which combines 4 Nvidia Geforce GTX graphics cards within a workstation. The scenario is based on DC, only when there is no SBS under the communication radius of one MU, the MBS serves the MU independently. The initial number of training samples are set as 20000.

TABLE I
SIMULATION PARAMETERS

Parameter	Connotation	Value
λ_M	MBS density	$1 \times 10^{-5} m^{-2}$
λ_S	SBS density	$1 \times 10^{-4} m^{-2}$
P_M	Maximum MBS power	24dBm
P_S	Maximum SBS power	20dBm
L	Propagation path number	2
N_{UE}	Antenna number of MU	2
$N_{\mathrm{M,BS}}$	Antenna number of MBS	64
$N_{\mathrm{S,BS}}$	Antenna number of SBS	32
R	Maximum MU communication radius	100m
$N^{\mathcal{C}}$	Codebook size	8

Fig. 2 shows the relationship between ASR and the number of MUs. The ASR keeps increasing as the MU number increases. This is due to more MUs can receive downlink signals in the same area. The ASR-based algorithm calculates the ASR caused by every codeword in the codebook. Then, the algorithm selects the best one for signal transmission. The CE-based algorithm selects the best codeword in the codebook according to the perfect CSI of each channel. We can see the performance of our proposed ISSC algorithm is very closed to the ASR-based algorithm which can be seen as the theoretical performance boundary. This is due to the proposed algorithm selects the codeword according to the SVM classifier, which is based on the data sample training to full 'learned' the system features. So, the codeword selection can adaptively fit for the change of the network. The CE-based algorithm performs

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2019.2923314, IEEE Transactions on Vehicular Technology

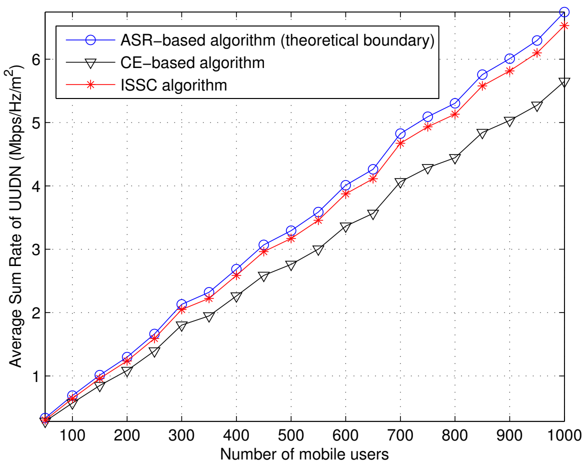

Fig. 2. ASR of UUDN vs. Number of MUs

worse because it is based on the perfect CSI, which can be seen as only part of the system features.

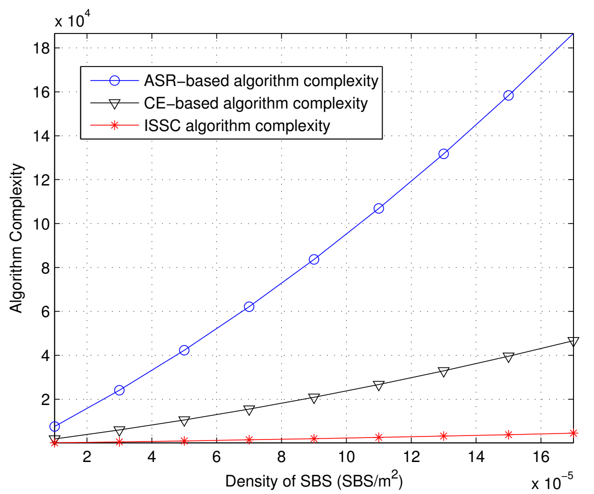

Fig. 3. Complexity comparison between three algorithms

Last, Fig. 3 compares our proposed ISSC algorithm with the ASR-based algorithm and CE-based algorithm. We can see the complexity is significantly reduced because the large computation overhead and interaction between BSs and MUs can be avoided. The transmitter can directly select the best codeword to form the optimal transmit beam according to its well-trained classification model without the complicated traversal operation of all the codewords in the codebook. In addition, with more BSs involving the codeword selection, the computational complexity increases as the BS number increases.

V. Conclusions

In this paper, we consider an ML inspired codeword selection for downlink DC to improve the ASR of MUs in 5G UUDN. By modeling the random distributions of MBSs and SBSs as HPPPs, we get the expression of receive signal in downlink transmission and further derive the ASR of each MU under DC. Then, we propose an iterative SVM classifier with the data samples include the BS transmit power and channel parameters. With the help of the proposed ISSC algorithm, the separating hyperplanes for code selection in DC can be obtained by data sample training, where both MBS and SBS get a high efficient ASR with very low complexity. In our simulation, the data samples are trained by Google TensorFlow and the results verify that our ISSC algorithm achieves a very closed performance to the theoretical boundary with a significant reduction of calculation complexity.

References

[1] U. S. Hashmi, S. A. R. Zaidi, and A. Imran, "User-centric cloud RAN: An analytical framework for optimizing area spectral and energy efficiency," *IEEE Access*, vol. 6, pp. 19 859–19 875, 2018.
[2] 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, 2017.
[3] 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. 21 359–21 368, 2018.
[4] J. Cao, T. Peng, Z. Qi, R. Duan, Y. Yuan, and W. Wang, "Interference management in ultra-dense networks: A user-centric coalition formation game approach," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, 2018.
[5] 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.
[6] 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.
[7] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in ofdm systems," *IEEE Wireless Commun. Letters*, vol. 7, no. 1, pp. 114–117, Feb. 2018.
[8] 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, 2018.
[9] M. Haenggi, *Stochastic geometry for wireless networks*. Cambridge University Press, 2012.
[10] M. A. Lema, E. Pardo, O. Galinina, S. Andreev, and M. Dohler, "Flexible dual-connectivity spectrum aggregation for decoupled uplink and downlink access in 5G heterogeneous systems," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 11, pp. 2851–2865, 2016.
[11] S. Shalmashi, E. Björnson, M. Kountouris, K. W. Sung, and M. Debbah, "Energy efficiency and sum rate when massive MIMO meets device-to-device communication," in *Proc. IEEE International Conference on Communication Workshop (ICC 2015 Workshop)*, Jun. 2015.
[12] J. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," *Advances in Kernel Methods-Support Vector Learning*, vol. 208, Jul. 1998.
[13] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
[14] N. Takahashi and T. Nishi, "Global convergence of decomposition learning methods for support vector machines," *IEEE Trans. Neural Networks*, vol. 17, no. 6, pp. 1362–1369, 2006.