DropNet

Please check the document version of this publication:

DropNet: An Improved Dropping Algorithm Based On Neural Networks for Line-of-Sight Massive MIMO

AMIRASHKAN FARSAEI[1], ALIREZA SHEIKH[1], (Member, IEEE), ULF GUSTAVSSON[2], ALEX ALVARADO[1], (Senior Member, IEEE), FRANS M. J. WILLEMS[1], (Fellow, IEEE)

[1]Eindhoven University of Technology, Eindhoven, The Netherlands (e-mail: a.farsaee@tue.nl; a.sheikh@tue.nl; a.alvarado@tue.nl; f.m.j.willems@tue.nl)
[2]Ericsson Research, Gothenburg, Sweden (e-mail: u.gustavsson@ericsson.com)

Corresponding author: A. Farsaei (e-mail: a.farsaee@tue.nl).

ABSTRACT In line-of-sight massive MIMO, the downlink channel vectors of few users may become highly correlated. This high correlation limits the sum-rates of systems employing linear precoders. To constrain the reduction of the sum-rate, few users can be dropped and served in the next coherence intervals. The optimal strategy for selecting the dropped users can be obtained by an exhaustive search at the cost of high computational complexity. To alleviate the computational complexity of the exhaustive search, a correlation-based dropping algorithm (CDA) is conventionally used, incurring a sum-rate loss with respect to the optimal scheme. In this paper, we propose a dropping algorithm based on neural networks (DropNet) to find the set of dropped users. We use appropriate input features required for the user dropping problem to limit the complexity of DropNet. DropNet is evaluated using two known linear precoders: conjugate beamforming (CB) and zero-forcing (ZF). Simulation results show that DropNet provides a trade-off between complexity and sum-rate performance. In particular, for a 64-antenna base station and 10 single-antenna users: (i) DropNet reduces the computational complexity of the exhaustive search by a factor of 46 and 3 for CB and ZF, respectively, (ii) DropNet improves the 5th percentile sum-rate of CDA by 0.86 and 2.33 bits/s/Hz for CB and ZF, respectively.

INDEX TERMS Correlated scenarios, dropping algorithm, line-of-sight massive MIMO, neural network.

I. INTRODUCTION

THE mutual orthogonality of the channel vectors from the base station (BS) to the users is known as favorable propagation (FP) [1]. Line-of-sight (LOS) environments exhibit FP both in theory and practice [2], [3]. There are important use cases (e.g., stadiums or exhibitions), in which the channel vectors of some users become highly correlated [4], which in turn results in a non-FP environment. In these non-FP environments, the high correlation yields a reduction in the achievable sum-rates of linear precoders [3], [5]. In particular, this reduction is non-negligible when the max-min power control is employed due to the fairness criterion, as shown in our previous papers [6], [7].

To limit the reduction in the achievable sum-rates of linear precoders, a correlation-based dropping algorithm (CDA) for LOS environments with max-min power control is proposed in [2]. In CDA, the BS drops a few users to constrain

the spatial correlations between the remaining users up to a predefined threshold, which is optimized using extensive simulations. We previously derived this threshold for two known linear precoders: (i) conjugate beamforming (CB) and (ii) zero-forcing (ZF) in [6], and for a known non-linear precoder, i.e., Tomlinson-Harashima precoding [8] in [7]. Employing CDA with the thresholds given in [6], [7] for channels with only one pair of correlated users (any other pairs are orthogonal) yields the optimal dropping strategy. However, when there are more than one pair of correlated users, the CDA approach is suboptimal. The optimal strategy in such a scenario can be found via an exhaustive search at the cost of significantly high computational complexity. Therefore, a low-complex yet near-optimal dropping strategy is required when there are more than one pair of correlated users. To the best of our knowledge, this problem has not been studied in the literature.

In this paper, we propose a dropping algorithm based on neural networks (DropNet) to find the set of dropped users that maximizes the achievable sum-rate of the remaining users with max-min power control. DropNet is inspired by the universal function approximation property of neural networks (NNs) [9]–[11]. By employing NN, we study the user dropping problem in a general scenario for LOS massive MIMO, when there might be more than one pair of correlated users. We find appropriate input features for the NN by studying the signal to noise plus interference ratio (SINR) of CB and ZF. We treat the user dropping as a classification problem, where each output class of the NN represents a possible set of dropped users. To achieve a near-optimal 5th percentile achievable sum-rate with low complexity compared to the exhaustive search, we adjust the hyperparameters of the NN. DropNet lifts the need for employing a predefined threshold to find the set of dropped users as opposed to [2], [6], [7]. Simulation results for two known linear precoders with max-min power control show that DropNet provides a good trade-off between performance in terms of achievable sum-rate and computational complexity.

II. SYSTEM MODEL

The schematic of the massive MIMO downlink channel with linear precoding is shown in Fig. 1, where an M-antenna BS serves K single-antenna users in a time division duplexing manner. The symbol of the users is $\boldsymbol{s} = (s_1, s_2, ..., s_K)^T \in \mathbb{C}^{K \times 1}$, where the components of $\boldsymbol{s}$ are assumed to be zero-mean, uncorrelated, and unit variance. [1] The diagonal power control matrix $\boldsymbol{D} = \text{diag}(\boldsymbol{d})$ and a linear precoding matrix $\boldsymbol{U} = (\boldsymbol{u}_1, \boldsymbol{u}_2, ..., \boldsymbol{u}_K) \in \mathbb{C}^{M \times K}$ (with unit-norm column vectors $\boldsymbol{u}_i$) precode $\boldsymbol{s}$ to $\boldsymbol{x} \in \mathbb{C}^{M \times 1}$. The power control vector $\boldsymbol{d} = (\sqrt{d_1}, \sqrt{d_2}, ..., \sqrt{d_K})^T$ has the coefficients $d_i \in \mathbb{R}^+$ with $i = 1, 2, ..., K$ with the total power constraint $\sum_{i=1}^{K} d_i = P$. The transmit vector $\boldsymbol{x}$ is found by

$$\boldsymbol{x} = \boldsymbol{U} \boldsymbol{D} \boldsymbol{s}. \tag{1}$$

Then, $\boldsymbol{x}$ is transmitted through the propagation channel denoted by $\boldsymbol{H} = (\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_K)^T \in \mathbb{C}^{K \times M}$, where $\boldsymbol{h}_i$ is the channel vector from the BS antennas to user i.

The received signal at user i is given as

$$y_i = \boldsymbol{h}_i^T \boldsymbol{x} + n_i = \boldsymbol{h}_i^T \boldsymbol{u}_i \sqrt{d_i} s_i + \sum_{\substack{j=1 \\ j \neq i}}^{K} \boldsymbol{h}_i^T \boldsymbol{u}_j \sqrt{d_j} s_j + n_i, \tag{2}$$

where n_i is zero mean complex Gaussian noise with the variance of N_0. Assuming perfect channel state information at the BS, the SINR for user i denoted by γ_i is given as

$$\gamma_i = \frac{|\boldsymbol{h}_i^T \boldsymbol{u}_i|^2 d_i}{\sum_{j=1, j \neq i}^{K} |\boldsymbol{h}_i^T \boldsymbol{u}_j|^2 d_j + N_0}. \tag{3}$$

[1]**Notation**: Lowercase, bold lowercase, and bold uppercase letters denote scalars, column vectors, and matrices, respectively. $|\cdot|$ and $\|\cdot\|$ denote the absolute value and l^2-norm operators. The superscripts $*$, T, and H denote complex conjugate, un-conjugated transpose, and conjugated transpose, respectively. $\text{diag}(\boldsymbol{p})$ denotes a diagonal matrix with diagonal entries taken from $\boldsymbol{p}$. The operator $\otimes$ denotes the kronecker product.
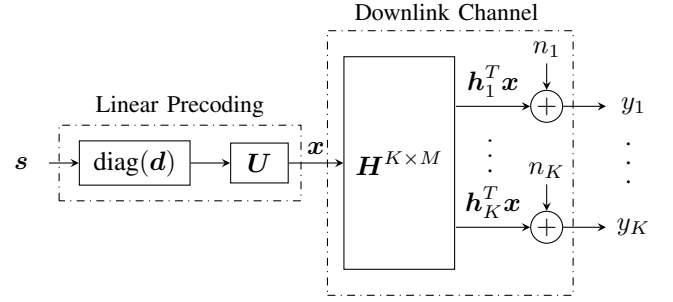

FIGURE 1. The model of the downlink channel with linear precoding.

In this paper, we consider two known linear precoders: CB and ZF. To find the precoding matrix $\boldsymbol{U}$ for CB and ZF, we first find $\boldsymbol{G} = \boldsymbol{H}^H$ and $\boldsymbol{G} = \boldsymbol{H}^\dagger = \boldsymbol{H}^H(\boldsymbol{H}\boldsymbol{H}^H)^{-1}$, respectively. The precoding matrix $\boldsymbol{U}$ is then found for each precoder by normalizing $\boldsymbol{G}$ to have unit-norm column vectors, i.e., $\boldsymbol{u}_i = \boldsymbol{g}_i / \|\boldsymbol{g}_i\|$. By replacing CB and ZF filters, the following SINR is obtained for each user:

$$\gamma_i^{\text{CB}} = \frac{\|\boldsymbol{h}_i\|^2 d_i}{\|\boldsymbol{h}_i\|^2 \sum_{j=1, j \neq i}^{K} |\rho_{ij}|^2 d_j + N_0}, \tag{4}$$

$$\gamma_i^{\text{ZF}} = \frac{|\boldsymbol{h}_i^T \boldsymbol{u}_i|^2 d_i}{N_0} = \frac{d_i}{\|\boldsymbol{g}_i\|^2 N_0}. \tag{5}$$

For a given set of filters $\boldsymbol{u}_i, i = 1, 2, ..., K$, we are interested in finding the coefficients $d_i, i = 1, 2, ..., K$, that maximize the minimum γ_i among the users, which is referred to as max-min power control [12, Sec. 7.1]. Employing the max-min power control equalizes the throughput of all users [2], i.e., $\gamma_i^{\text{CB}} = \gamma^{\text{CB}}$ and $\gamma_i^{\text{ZF}} = \gamma^{\text{ZF}}$ for $i = 1, 2, ..., K$. The power control vector $\boldsymbol{d}^*$ is found by solving

$$\boldsymbol{d}^* = \underset{d_1, d_2, ..., d_K \in \mathbb{R}^+}{\text{argmax}} \ \underset{i \in \{1, 2, ..., K\}}{\min} \ \gamma_i, \tag{6}$$

where γ_i is given by (3). To solve (6), we use the bisection method (see [2, Algorithm 2]) for CB, and we use the Lagrangian multiplier for ZF.

III. DROPNET: PROPOSED DROPPING ALGORITHM BASED ON NEURAL NETWORKS

In this section, we present details of DropNet. DropNet is designed to find the set of users that shall be dropped such that the achievable sum-rate of the remaining users is maximized. At the end of this section, a complexity analysis is given to compare the complexity of DropNet with the exhaustive search and the previous CDA.

A. DESIGN METHODOLOGY

We model the user dropping as a classification problem. In the classification problem, we consider one class representing the case where no user is dropped, $\binom{K}{i}$ classes representing the cases where i users out of K users are dropped. We assume $1 \leq i \leq n_{\text{max}}$, where n_{max} is the maximum number of

users that we allow to be dropped. Overall, the total number of classes is

$$n_{\text{out}} = 1 + \binom{K}{1} + \binom{K}{2} + ... + \binom{K}{n_{\text{max}}}. \qquad (7)$$

Each class denotes a neuron in the output layer. Thus, the number of neurons corresponding to the output layer is n_{out}.

We choose the inputs of the NN as follows assuming a real-valued NN. For a given channel realization $\boldsymbol{H}$, the chosen inputs should correspond to a meaningful metric related to the users to be dropped. Moreover, as will be shown in Sec. III-B, the computational complexity of the resulting NN is directly related to the number of input and output neurons. Therefore, the number of inputs should be as small as possible. In general, for a fixed transmit power P, one can employ the elements of $\boldsymbol{H}$ as the inputs of the NN, as $\boldsymbol{H}$ contains all the information required for the dropping algorithm. However, the number of elements of $\boldsymbol{H}$ is $2MK$,[2] which scales linearly with both the number of antennas M and the number of users K. This is not desirable because in massive MIMO $M >> K$. Thus, it is beneficial to find appropriate input features for which the number of input nodes does not scale with M. Previous dropping algorithms [2], [6], [7] use the absolute value of the pair-wise normalized spatial correlation of the users ρ_{ij}

$$\rho_{ij} = \frac{\boldsymbol{h}_j^H \boldsymbol{h}_i}{\|\boldsymbol{h}_i\| \|\boldsymbol{h}_j\|}, \quad i, j \neq i \in \{1, 2, ..., K\}, \qquad (8)$$

to drop some of the users. There are $\binom{K}{2}$ values of $|\rho_{ij}|$, i.e., $(K^2 - K)/2$, which is much less than $2MK$ values of $\boldsymbol{H}$. Thus, $|\rho_{ij}|$ values are possible candidates for the inputs of the NN. By studying (4) and (5), we propose to use $|\rho_{ij}|$ and $\|\boldsymbol{h}_i\|^2$ as the input features of the NN as explained in the following.

To find the SINR for CB as in (4), we need to use bisection method to find the power control coefficients, for which $\|\boldsymbol{h}_i\|$ and $|\rho_{ij}|$ are required. Therefore, $\|\boldsymbol{h}_i\|$ and $|\rho_{ij}|$ provide enough information to find the set of dropped users for CB. To find the SINR for ZF as in (5), we need to use Lagrangian multiplier to find the power control coefficients for which $\|\boldsymbol{g}_i\|^2$ the diagonal elements of $(\boldsymbol{HH}^H)^{-1}$ are required. To compute the diagonal elements of $(\boldsymbol{HH}^H)^{-1}$, we need $\|\boldsymbol{h}_i\|$ and ρ_{ij}. Thus, for ZF, we need the complex values of ρ_{ij} rather than $|\rho_{ij}|$ as for CB. However, by using $|\rho_{ij}|$ instead of ρ_{ij}, we can further reduce the number of input nodes for ZF. Thus, in this paper, we use $|\rho_{ij}|$ and $\|\boldsymbol{h}_i\|^2$ as the input features for both CB and ZF. Overall, the number of input nodes becomes:

$$n_{\text{in}} = \binom{K}{2} + K = \frac{K^2 + K}{2}, \qquad (9)$$

which is much lower than $2MK$. For instance, assuming $K = 10$ and $M = 100$, $\binom{K}{2} + K = 55$, while $2MK = 2000$. We emphasize that by using $|\rho_{ij}|$ and $\|\boldsymbol{h}_i\|^2$ values,

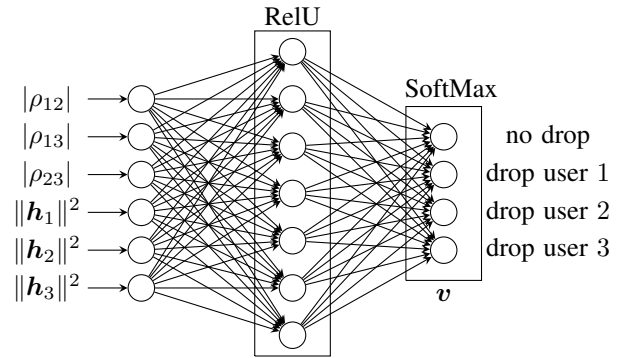**FIGURE 2.** The schematic of a NN for DropNet when $K = 3$ and $n_{\text{max}} = 1$ with inputs $|\rho_{12}|, |\rho_{13}|, |\rho_{23}|, \|\boldsymbol{h}_1\|^2, \|\boldsymbol{h}_2\|^2, \|\boldsymbol{h}_3\|^2$ and output one-hot vector $\boldsymbol{v}$.

we remove the dependency of the number of inputs to M and therefore, reduce the complexity of NN considerably. Hence, such input selection is practical for massive MIMO systems. We investigated different NNs with more than one hidden layer, however, to limit the computational complexity of the designed NN we use only one hidden layer instead. The number of neurons in the hidden layer is the design parameter, which provides a performance-complexity trade-off in DropNet.

As an example, the NN of DropNet for $K = 3$, $n_{\text{max}} = 1$ is illustrated in Fig. 2. There are $\binom{3}{2} + 3 = 6$ input nodes and there are $1 + \binom{3}{1} = 4$ output nodes with 7 nodes in the hidden layer. In DropNet, we employ "Relu" as the activation function for the hidden layer and "Softmax" as the activation function for the output layer (see Fig. 2). The output of Softmax represents the probability of each class for a given set of input features. The output of NN is represented by a one-hot vector $\boldsymbol{v}$ of size $n_{\text{out}} \times 1$, where the component corresponding to the class of dropped users is "1" and all the other components are "0". We employ cross-entropy as the cost function, as NN is used to find the set of dropped users with a high probability. The standard back-propagation algorithm [13] is used for optimizing the parameters of NN.

To train (test) the NN, we generate the training (test) set for a given precoder as follows. We generate a large number of realizations of $\boldsymbol{H}$ for the training (test) set. For each realization of $\boldsymbol{H}$, we compute $\binom{K}{2}$ values of $|\rho_{ij}|$ and K values of $\|\boldsymbol{h}_i\|^2$ associated with $\boldsymbol{H}$. We find the optimal set of dropped users corresponding to $\boldsymbol{H}$ with an exhaustive search. The solution of the exhaustive search is stored as a one-hot vector $\boldsymbol{v}$ of size $n_{\text{out}} \times 1$, where the component corresponding to the class of dropped users is "1" and all the other components are zero. The vector $\boldsymbol{v}$ serves as the NN output corresponding to the computed input nodes. After the training phase, the trained NN is evaluated using the test set. We evaluate the complexity of the designed NN in the sequel.

---

[2]Note that the factor 2 is due to considering a real-valued NN.

B. COMPLEXITY ANALYSIS

We first explain the computational complexity of the exhaustive search for CB and ZF precoding, then, we compute the corresponding complexity of DropNet. For the complexity analysis, it is assumed that each complex addition costs 2 floating point operations (FLOPS) and each complex multiplication costs 6 FLOPS [14]. To drop i out of K users, there are $\binom{K}{i}$ possibilities. In the exhaustive search, one requires to check all possible sets of dropped users. For each set of dropped users, we need to find the SINR with max-min power control of the remaining users to compute the corresponding sum-rate. For CB, it is required to use the bi-section method to compute the max-min SINR denoted by γ^{CB} for the users. At each iteration of the bi-section method, an inverse of a $K \times K$ matrix is required, which entails $K^3/2 + 3K^2/2$ multiplications and $K^3/2 - K^2/2$ additions and K square roots operations[3] [15]. For a given $n_{\max}$, the complexity of the exhaustive search for CB in FLOPS is given as

$$C^{\mathrm{CB}} = \sum_{i=0}^{n_{\max}} \binom{K}{i} I_i (4(K-i)^3 + 8(K-i)^2 + (K-i)), \quad (10)$$

where I_i is the number of iterations used to run the bi-section method, which depends on the search interval for γ_{CB} and the required accuracy for γ_{CB} [16, Th. 2.1]. For instance, for $n_{\max} = 2$ and $I_0 = I_1 = I_2 = K$ with accuracy of 0.01, C^{CB} has complexity of $\mathcal{O}(K^6)$.
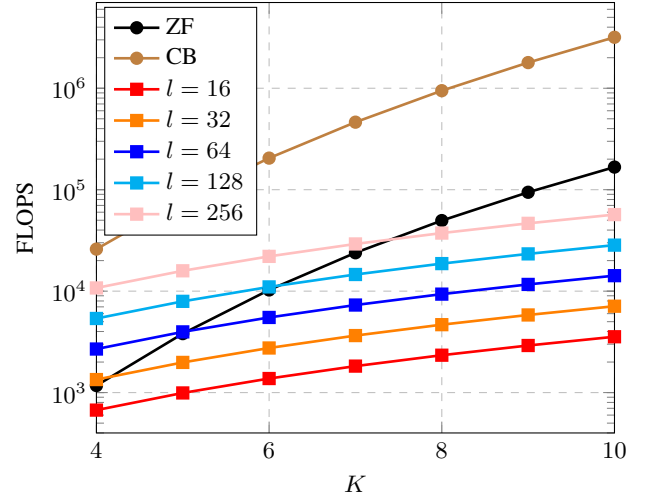
The max-min SINR for ZF is given as [17, eq. (14)]:

$$\gamma^{\mathrm{ZF}} = \frac{P}{N_0 \, \mathrm{tr}(\boldsymbol{H}\boldsymbol{H}^H)^{-1}}. \quad (11)$$

As can be seen from (11), the trace of $(\boldsymbol{H}\boldsymbol{H}^H)^{-1}$ for computing γ_{ZF} needs to be calculated. It is known that the trace of $(\boldsymbol{H}\boldsymbol{H}^H)^{-1}$ is equal to the sum of the eigenvalues of $(\boldsymbol{H}\boldsymbol{H}^H)^{-1}$ [18, Ch. 4]. To find the eigenvalues of $(\boldsymbol{H}\boldsymbol{H}^H)^{-1}$, it is enough to find the eigenvalues of $\boldsymbol{H}\boldsymbol{H}^H$ and then inverse them. Consequently, the computational complexity of evaluating γ^{ZF} is equal to the complexity of finding the eigenvalues of a $K \times K$ symmetric matrix, which is $16/3K^3$ [14]. Overall, the complexity of the exhaustive search for ZF in FLOPS is given as

$$C^{\mathrm{ZF}} = \sum_{i=0}^{n_{\max}} \binom{K}{i} \frac{16}{3}(K-i)^3. \quad (12)$$

For instance, for $n_{\max} = 2$, C^{ZF} has complexity of $\mathcal{O}(K^5)$.

The complexity of DropNet depends on the number of multiplications and additions in the forward propagation of the NN at the test phase.[4] We consider real-valued NN, where the multiplications and additions are all real operations. Recall that the input and output layers of NN have n_{in} (see (9)) and n_{out} (see (7)) neurons. Let us assume that the hidden layer contains l neurons. The number of multiplications for computing the value of a given neuron in the hidden and

---

[3]We assume that each square root operation costs 1 FLOP [15].

[4]Back-propagation is performed offline at the training phase, thus, only the complexity of the forward propagation is considered at the test phase.

FIGURE 3. The computational complexity of the exhaustive search for CB and ZF compared to the DropNet with different number of nodes (16 to 256) as a function of the number of users K with $n_{\max} = 2$.

output layers are n_{in} and l, respectively, rendering a total of $n_{\mathrm{in}}l + ln_{\mathrm{out}}$ multiplications in the forward propagation. Assuming a bias term for the hidden and output layers, the total number of additions in the forward propagation becomes the same as the number of multiplications, i.e., $n_{\mathrm{in}}l + ln_{\mathrm{out}}$. As each real-value multiplication and addition require one FLOP, the total number of FLOPs for the forward propagation is given as

$$C^{\mathrm{DropNet}} = 2n_{\mathrm{in}}l + 2ln_{\mathrm{out}},$$
$$= 2l \left(\frac{K^2 + K}{2} + \sum_{i=0}^{n_{\max}} \binom{K}{i} \right) \quad (13)$$

For instance, for $n_{\max} = 2$ and $l = K^2$, C^{DropNet} has complexity of $\mathcal{O}(K^4)$, which is lower than that of CB and ZF for the same $n_{\max}$.

In Fig. 3, the complexity (FLOPS) of DropNet as a function of K is compared with that of the exhaustive search for $n_{\max} = 2$ for both CB and ZF. The number of nodes in the hidden layer for DropNet changes from 16 to 256. Although by increasing the number of nodes in the hidden layer, the complexity of DropNet increases, the slope of DropNet is much lower than the exhaustive search for both CB and ZF. To compute the complexity of CB in Fig. 3 for a given number of users, we use the average I_i found by running a large number of simulations. For instance for $K = 10$ and accuracy of 0.01, $I_0 = 24.2$, $I_1 = 21.4$ and $I_2 = 19.9$. We use the results in Fig. 3 to find an appropriate number of neurons for the hidden layer in DropNet that considerably reduces the computational complexity of the exhaustive search.
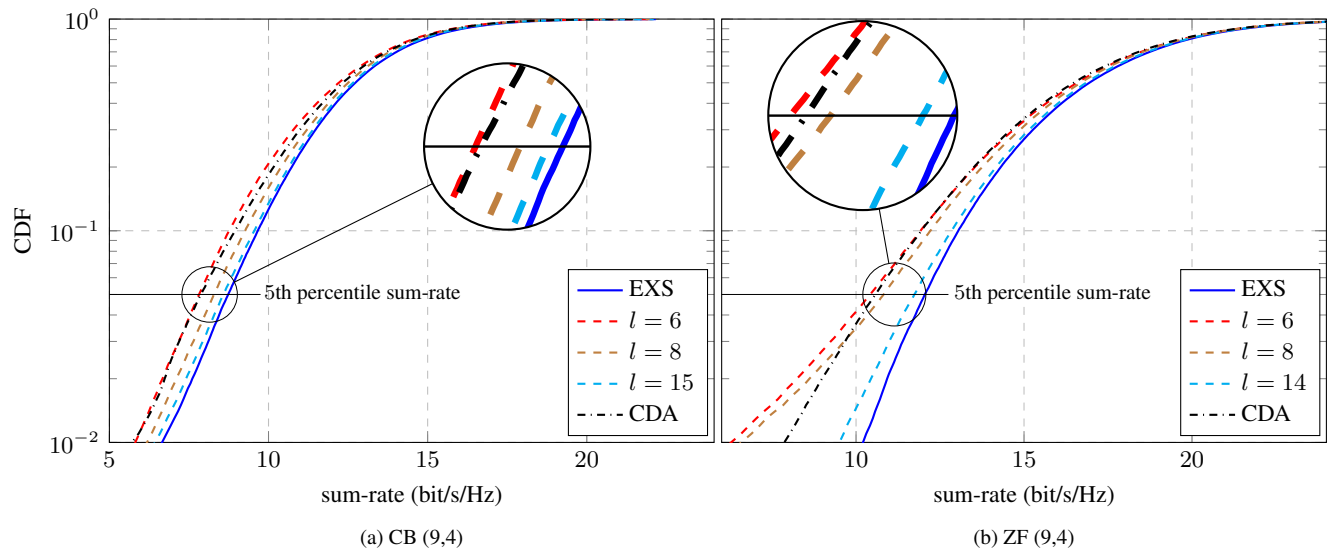
IV. SIMULATION RESULTS

We consider a single-cell massive MIMO, where a BS with a uniform planar array (UPA) of $\sqrt{M} \times \sqrt{M}$ antennas located at x-y plane with half-wavelength spacing (carrier frequency

FIGURE 4. The CDF plots of the sum-rates for the exhaustive search (EXS), CDA and DropNet with three different l when a 3×3 UPA serves 4 users ($M = 9, K = 4$) with CB and ZF.

of 30 GHz) serving K single-antenna users. The users are uniformly distributed in the cell (10-200 m). The channel matrix $\boldsymbol{H}$ is computed using the LOS model given in [19, eq. (5)]. Moreover, shadowing effect (log-normal shadow fading with the variance of 12 dB) is considered. The azimuth and elevation angles of the users are uniformly distributed in the intervals $(0, 2\pi)$ and $(0, \pi/2)$, respectively. The minimum distance between two users is set to a wavelength. We set the transmit power (without loss of generality) at the BS such that in FP, $\gamma_{\text{CB}} = \gamma_{\text{ZF}} = 15$ is achieved and we consider $n_{\max} = 2$ for all the dropping algorithms. This means for each dropping algorithm, the maximum number of users that is allowed to be dropped is 2. For the training set 3.9M (97.5% of dataset) and for the test set 100K realizations (2.5% of dataset) of the channels are used. We present the cumulative distribution function (CDF) of CB and ZF achievable sum-rate for DropNet compared to the exhaustive search and previous CDA [6, Algorithm 1].

In Fig. 4, the CDF of the sum-rate is shown for CB and ZF for a BS with a 3×3 UPA serving 4 users ($M = 9, K = 4$) employing the exhaustive search (blue solid line), CDA (black dash-dotted) and DropNet with three different l (dashed lines) to drop some of the users. The 5th percentile sum-rate is magnified for a better comparison for all the scenarios. In addition, in Fig. 5, the same curves are presented for a BS with a 8×8 UPA serving 10 users ($M = 64, K = 10$). The following conclusions are inferred from Fig 4 and Fig. 5. First, there is a gap between the CDF of the sum-rate of the exhaustive search and CDA. Second, by employing DropNet with an appropriate l, one can improve the CDF of CDA and reduce the gap to the exhaustive search. Third, for a given l, the designed NN for CB has a performance much closer to the exhaustive search compared to ZF. For ZF, the gap to the exhaustive search can

be further reduced by using ρ_{ij} instead of $|\rho_{ij}|$ as the input features, however, with extra complexity.

We compare the 5th percentile sum-rate of DropNet with a given l with the exhaustive search and CDA in Table 1. We use simulation scenarios in Fig. 4 and Fig. 5. The improvement of DropNet for ZF is more than that of CB for both MIMO systems. In terms of the 5th percentile sum-rate, by employing DropNet, the gap (loss) to the exhaustive search is smaller than the gap (improvement) to CDA. We further compare the computational complexity (FLOPS) of DropNet for a given l with the exhaustive search and CDA in Table 2 for the same scenarios as in Table 1. Note that computing $|\rho_{ij}|$ for CDA and DropNet costs $8MK^2$ FLOPS. The complexity reduction of DropNet for CB is much more than that of ZF for both MIMO systems. For both CB and ZF, by employing DropNet, the complexity of the exhaustive search is reduced.

The results in Table 1 and Table 2 show that DropNet provides a 5th percentile sum-rate close to that of the exhaustive search while its complexity is close to that of CDA. Therefore, DropNet provides an interesting trade-off between complexity and sum-rate performance.

We further present the CDF of sum-rates for CB and ZF in Fig. 6 for $M = 64$ and $K = 10$ when there is no shadowing. In this case, the gap between CDA and the exhaustive search is smaller. Similar to the shadowing scenarios, by employing DropNet, we can approach the exhaustive search performance.

V. CONCLUSIONS

In this paper, a dropping algorithm based on neural networks is proposed for LOS massive MIMO. We show that the proposed dropping algorithm provides a performance-complexity trade-off between conventional correlation-based
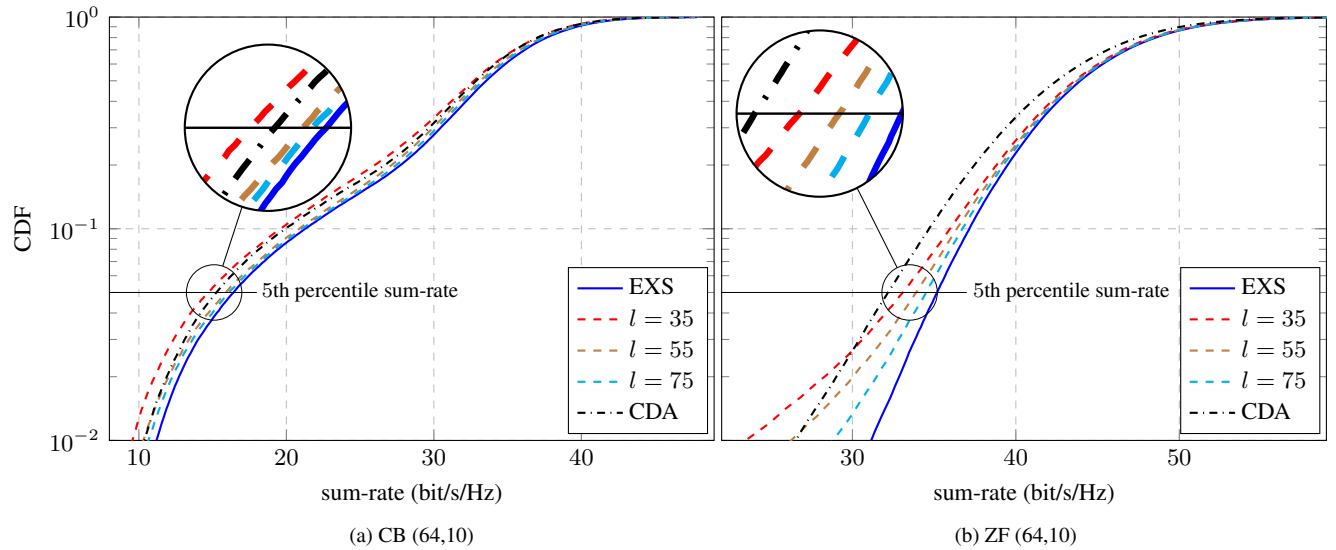
FIGURE 5. The CDF plots of the sum-rates for the exhaustive search (EXS), CDA and DropNet with three different l when a 8×8 UPA serves 10 users ($M = 64$, $K = 10$) with CB and ZF.

TABLE 1. 5th percentile sum-rate comparison between DropNet, exhaustive search, and CDA dropping schemes for the simulation scenarios of Fig. 4 and Fig. 5.

Case study	Number of neurons in hidden layer (l)	5th percentile sum rate gap w.r.t. exhaustive search (bit/s/Hz)	5th percentile sum rate improvement w.r.t. CDA (bit/s/Hz)
CB ($M = 9$, $K = 4$)	15	0.15	0.70
CB ($M = 64$, $K = 10$)	75	0.30	0.86
ZF ($M = 9$, $K = 4$)	14	0.31	1.16
ZF ($M = 64$, $K = 10$)	75	0.67	2.33

TABLE 2. Complexity (FLOPS) comparison between DropNet, exhaustive search, and CDA dropping schemes for the simulation scenarios of Fig. 4 and Fig. 5. To find the complexity reduction (increase) ratio, we compute (10) for CB, (12) for ZF and (13) for DropNet.

Case study	Number of neurons in hidden layer (l)	Complexity reduction ratio w.r.t. exhaustive search	Complexity increase ratio w.r.t. CDA
CB ($M = 9$, $K = 4$)	15	14.57	1.55
CB ($M = 64$, $K = 10$)	75	46.89	1.33
ZF ($M = 9$, $K = 4$)	14	1.34	1.51
ZF ($M = 64$, $K = 10$)	75	3.22	1.33

dropping algorithms and the optimal dropping strategy found by an exhaustive search. The proposed dropping algorithm outperforms the correlation-based dropping algorithm and achieves a 5th percentile sum-rate close to that of the exhaustive search with up to a factor of 46 and 3 lower computational complexity compared to the exhaustive search for CB and ZF, respectively.

REFERENCES

[1] H. Q. Ngo, E. G. Larsson, and T. L. Marzetta, "Aspects of favorable propagation in massive MIMO," in 2014 22nd European Signal Processing Conference (EUSIPCO), pp. 76–80, Sep. 2014.

[2] H. Yang and T. L. Marzetta, "Massive MIMO with max-min power control in line-of-sight propagation environment," IEEE Transactions on Communications, vol. 65, pp. 4685–4693, Nov. 2017.

[3] X. Gao, O. Edfors, F. Rusek, and F. Tufvesson, "Massive MIMO performance evaluation based on measured propagation data," IEEE Transactions on Wireless Communications, vol. 14, pp. 3899–3911, Jul. 2015.

[4] A. Osseiran et al., "Scenarios for 5G mobile and wireless communications: the vision of the METIS project," IEEE Communications Magazine, vol. 52, pp. 26–35, May 2014.

[5] X. Gao, O. Edfors, F. Rusek, and F. Tufvesson, "Linear pre-coding performance in measured very-large MIMO channels," in 2011 IEEE Vehicular Technology Conference (VTC Fall), pp. 1–5, Sep. 2011.

[6] A. Farsaei, A. Alvarado, F. M. J. Willems, and U. Gustavsson, "An improved dropping algorithm for line-of-sight massive MIMO with max-min power control," IEEE Communications Letters, vol. 23, pp. 1109–1112, Jun. 2019.

[7] A. Farsaei, A. Alvarado, F. M. J. Willems, and U. Gustavsson, "An improved dropping algorithm for line-of-sight massive MIMO with Tomlinson-Harashima Precoding," IEEE Communications Letters, vol. 23, pp. 2099–2103, Nov. 2019.

[8] R. F. H. Fischer, C. Windpassinger, A. Lampe, and J. B. Huber, "Tomlinson-Harashima precoding in space-time transmission for low-rate backward channel," in 2002 International Zurich Seminar on Broadband Communications Access - Transmission - Networking (Cat. No.02TH8599), pp. 7–7, 2002.

[9] G. Cybenko, "Approximation by superpositions of a sigmoidal function," Math. Control Signal Systems, vol. 2, p. 303–314, 1989.
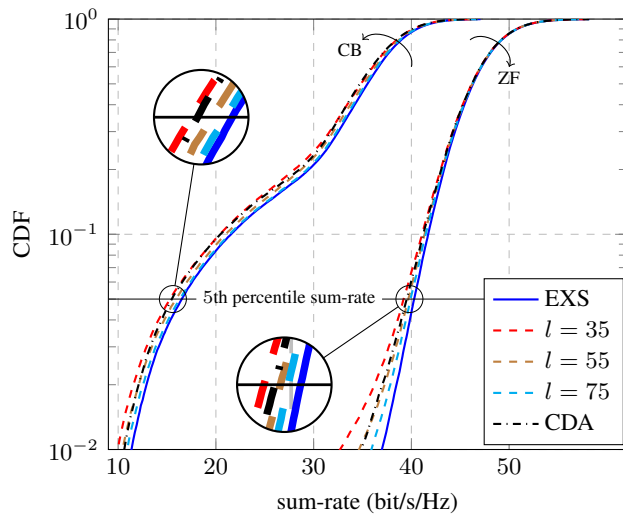
FIGURE 6. The CDF plots of the sum-rates for the exhaustive search (EXS), CDA and DropNet with $l = 35, 55, 75$ when a 8×8 UPA serves 10 users ($M = 64$, $K = 10$) with CB (left curves) and ZF (right curves) with no shadowing effect.

[10] A. Pinkus, "Approximation theory of the MLP model in neural networks," Acta Numer, vol. 8, p. 143–195, 1999.

[11] T. L. P. Kidger, "Universal approximation with deep narrow networks," arXiv preprint arXiv:1905.08539v2, May 2019.

[12] E. Björnson, J. Hoydis, L. Sanguinetti, et al., Massive MIMO Networks: Spectral, Energy, and Hardware Efficiency. Now Publishers, Inc., Jan. 2017.

[13] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," nature, vol. 323, no. 6088, pp. 533–536, 1986.

[14] M. Arakawa, "Computational workloads for commonly used signal processing kernels," tech. rep., MIT, Cambridge, MA, USA, DTIC Document, 2006.

[15] R. Hunger, "Floating point operations in matrix-vector calculus," Tech. Rep. TUM-LNS-TR-05-05, Technische Universität München, München, Germany, Oct. 2005.

[16] R. L. Burden and J. D. Faires, Numerical Analysis. Boston: PWS-Kent Publishing Company, fourth ed., 1989.

[17] H. Yang and T. L. Marzetta, "Max-min SINR dependence on channel correlation in line-of-sight massive MIMO," in 2017 IEEE Global Communications Conference (GLOBECOM), pp. 1–6, Dec. 2017.

[18] G. Hartman, Fundamentals of Matrix Algebra. Virginia Military Institute, 3rd ed., 2011.

[19] W. Tan, S. D. Assimonis, M. Matthaiou, Y. Han, X. Li, and S. Jin, "Analysis of different planar antenna arrays for mmwave massive mimo systems," in 2017 IEEE 85th Vehicular Technology Conference (VTC Spring), pp. 1–5, 2017.

• • •