Distributed Learning-Based Beamforming Codebooks for Unevenly Distributed Users in mmWave Massive MIMO System

Zhiheng Chen‡, Pei Liu‡, Yun Chen✠, Wen Zhan◇, Giovanni Interdonato§, and Stefano Buzzi§
‡School of Information Engineering, Wuhan University of Technology,
Wuhan 430070, China (e-mail: leochenzhiheng@whut.edu.cn; pei.liu@ieee.org)
✠TDD Algorithm Design Department, Huawei Technologies CO., Ltd.,
Dongguan 523830, China (yunchen.hust@gmail.com)
◇School of Electronics and Communication Engineering, Sun Yat-sen University,
Shenzhen 518107, China (e-mail: zhanw6@mail.sysu.edu.cn)
§Department of Electrical and Information Engineering, University of Cassino and Southern Latium,
03043 Cassino, Italy (e-mail: {giovanni.interdonato, buzzi}@unicas.it)

***Abstract*—Millimeter wave (mmWave) massive multiple-input multiple-output (MIMO) technology represents a promising technology in wireless communication. This technology relies on beamforming codebooks for initial access and transmission. However, conventional codebooks comprise a multitude of single-lobe narrow beams, resulting in redundant beams that may never be utilized in beam training. While centralized machine learning methods can partially address the concern of redundancy, they tend to overlook the presence of minority users scattered across diverse regions. The equitable coverage of environmental adaptive codebooks depends on addressing this issue. Hence, we devise a distributed learning (DL) framework for codebook design, which is tailored for scenarios with uneven user distribution and fully exploits the decentralized and online learning features of DL. Our approach begins by segmenting the user channels into various subsets through a pre-classification process based on the power response of the featured combining vectors from different subregions. Then, we introduce a novel DL architecture designed to process the subsets that are assigned to individual user equipments (UEs). Each UE then generates a phase shift matrix that contributes to the concatenation-based global aggregation in the base station. The simulation results confirm the effectiveness of DL in improving the performance of mmWave massive MIMO systems in scenarios with unevenly distributed users.**

***Index Terms*—Beamforming codebook, distributed learning, massive MIMO, Beyond 5G, millimeter wave**

I. Introduction

Millimeter wave (mmWave) multiple-input multiple-output (MIMO) technology is pivotal for 5G and advancing beyond 5G. Large antenna arrays in these systems deliver significant beamforming gains and ample signal reception power. However, the adoption of fully digital transceiver architectures, which deploys an RF chain for every antenna, becomes impractical due to the power consumption linked to high-frequency mixed-signal circuits. Consequently, mmWave systems opt for analog-only or hybrid designs to perform combining or beamforming [1]. Massive MIMO systems pose significant challenges, particularly in channel estimation and feedback. As a result, the predefined single-lobe beamforming codebooks are often adopted, with the discrete Fourier transform (DFT) codebooks being a prominent example [3]. These traditional approaches are designed to explore each feasible direction for initial access or data transmission [2]. However, the classical beam-steering codebooks suffer from two primary issues. Primarily, they incur unnecessary costs in beam training by covering all possible directions, even those that may never be utilized. Secondly, they do not compatible with specific scenarios such as non-line-of-sight (NLOS) or uneven user distribution. Although some progress has been made in adapting to NLOS users in [6], [7], research on uneven and nonuniform user distribution, which is a crucial aspect in 5G [4], [5], remains largely overlooked.

Lately, centralized machine learning (CML) based neural network techniques have made strides in addressing challenges related to environmental adaptivity and cost-effective training for beam codebook learning or beam alignment in [6]-[8]. Additionally, the proliferation of federated learning has spurred interest in exploring the synergy between distributed learning (DL) and beamforming. Some studies have focused on the robustness of federated learning models during training or the convenience of collecting local data and protecting privacy [9]-[11]. Their claimed reduction in communication overhead only holds true when the number of communication rounds is relatively small. In comparison, few studies have specifically compared the environmental adaptivity of the distributed models with CML-based methods in the specialized uneven user distribution scenarios. In these scenarios, the decentralized nature of DL can be leveraged to achieve equity among different groups of users. In this work, we aim to devise a distributed neural network-based codebook for

The corresponding author is Pei Liu (e-mail: pei.liu@ieee.org). S. Buzzi is also affiliated with DEIB, Politecnico di Milano, I-20133 Milano, Italy. The work of S. Buzzi was supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 - program RESTART, Structural Project 6GWINET).

users distributed unevenly. The proposed method inherits the high adaptivity of neural network-enhanced approaches and employs a distributed learning framework to independently obtain local codebooks for both the minority and majority users in the uneven user distribution scenario. Moreover, it reduces the computation load of base station (BS) by sharing the burden with user equipments (UEs) and thus facilitates online learning.

In this paper, we focus on the design of codebooks aided by the DL framework in mmWave massive MIMO communication scenarios with unevenly distributed users. Firstly, we propose a method to classify the user channels based on the power responses of the featured combining vectors from different subregions, which enables the differentiation between majority and minority user channels. This classification approach results in the identification of various subsets. Subsequently, we conceive a novel DL architecture to process these pre-classified subsets. The architecture includes assigning different subsets to individual UEs, thereby generating a corresponding phase shift matrix for each subset. These matrices are then transmitted back to the BS for aggregation and global updating. For the aggregation process, we propose the distributed concatenation method (Discat) which is suitable for the possible non-identical sizes of different UE codebooks. Furthermore, DL also enables online learning, distributing the computational load to the UEs. Notably, the DL-based method effectively captures the channel characteristics of the minority, which surpasses the performance of the CML method as a baseline in terms of achievable rate.

Throughout this paper, lower-case (upper-case) boldface letters indicate vectors (matrices). $\mathbb{R}^{M\times K}$ and $\mathbb{C}^{M\times K}$ indicate the $M\times K$ real space and $M\times K$ complex space, respectively. $\mathbf{A}^\dagger$ and $\mathbf{A}^{\mathrm{T}}$ indicate the Hermitian transpose and pure transpose of matrix $\mathbf{A}$, respectively. $\mathbf{I}_M$ means an $M\times M$ identity matrix; $\mathbf{0}_{M\times N}$ means an $M\times N$ zero matrix. A complex Gaussian random vector $\mathbf{x}$ is defined as $\mathbf{x}\sim\mathcal{N}_{\mathbb{C}}(\bar{\mathbf{x}},\mathbf{\Sigma})$, where $\bar{\mathbf{x}}$ is the mean vector and $\mathbf{\Sigma}$ is the covariance matrix. $\|\cdot\|_n$ indicates the n-norm of a vector.

II. System Model

A. Channel Model

In this paper, we consider a single-cluster multipath mmWave massive MIMO communication system where K single-antenna users distributed across R subregions are served by a BS equipped with a uniform linear array (ULA) of M antennas [12], [13]. Also, time division duplexing is adopted in this paper, and it is assumed that both the BS and users are perfectly synchronized within each symbol. As the Saleh-Valenzuela channel model in [12] is utilized, the clustered channel vector between the kth user and a M-antenna BS is given by

$$\mathbf{h}_k=\sqrt{\frac{M}{L}}\sum_{l=1}^{L}\alpha_{k,l}\mathbf{a}(\psi_{k,l}), \tag{1}$$

where $\alpha_{k,l}$ indicates the complex gain on the lth ray, L denotes the paths in total, and $\psi_{k,l}$ represents the angle of arrival (AOA) of the lth path. Moreover, $\psi_{k,l}$ locates in the angular domain $\Psi\triangleq[\beta_1,\beta_2)$, where the lower bound β_1 and upper bound β_2 are assumed to be known as the basic characteristics of the scenario. In detail, the array response vector $\mathbf{a}(\psi_{k,l})\in\mathbb{C}^{M\times 1}$ can be modeled by the ULA geometric of the M-antennas whose m-th is given as $[\mathbf{a}(\psi_{k,l})]_m=\frac{1}{\sqrt{M}}e^{-j\frac{2\pi}{\lambda}d(m-1)\cos(\psi_{k,l})}$, where λ is the wavelength and d represents the array spacing between adjacent antennas. To conveniently model the channel states of all K users, the channel set is defined as $\mathcal{H}$, i.e., $\mathcal{H}\triangleq\bigcup_{k=1}^{K}\{\mathbf{h}_k\}$ and $|\mathcal{H}|=K$.

B. Transmission Model

In the context of an uplink data transmission procedure, the kth user transmits the uplink symbol $a_k\in\mathbb{C}$ to the BS. The BS combines the received signal, and hence the processed signal $z_k\in\mathbb{C}$ can be represented as

$$z_k=\mathbf{w}_{\upsilon_k}^\dagger\mathbf{h}_k a_k+\mathbf{w}_{\upsilon_k}^\dagger\mathbf{n}_k, \tag{2}$$

where the symbol a_k has average power $\mathbb{E}[|a_k|^2]=P_{a_k}$ and the noise vector $\mathbf{n}_k$ at the BS antennas obeys the distribution $\mathbf{n}_k\sim\mathcal{N}_{\mathbb{C}}(\mathbf{0},\sigma^2\mathbf{I}_M)$. Also, the pure analog combining vector $\mathbf{w}_{\upsilon_k}\in\mathbb{C}^{M\times 1}$ is the υ_kth beamformer of the N-beam codebook $\mathcal{W}$, i.e., $\mathcal{W}\triangleq\bigcup_{n=1}^{N}\{\mathbf{w}_n\}$, $|\mathcal{W}|=N$ and $\upsilon_k\in\{1,\ldots,n,\ldots,N\}$. In detail, the m-th element of $\mathbf{w}_{\upsilon_k}$ is constructed as $[\mathbf{w}_{\upsilon_k}]_m=\frac{1}{\sqrt{M}}e^{j\theta_{\upsilon_k,m}}$ where $\theta_{\upsilon_k,m}$ denotes the corresponding phase shift at the mth BS antenna. Furthermore, it is worth noting that $\mathbf{w}_{\upsilon_k}$ solely applies a phase shift and does not adjust the received signal power. Therefore, it is advisable to properly fine-tune $\mathbf{w}_{\upsilon_k}$.

C. User Distribution Model

To represent the user locations in a simplified manner, the channel set $\mathcal{H}$ is partitioned into R subsets and hence is rewritten as $\mathcal{H}=\bigcup_{r=1}^{R}\mathcal{H}_r$, where the user channel group $\mathcal{H}_r$ contains the designated users' channels that belonging to class r $(r=1,...,R)$. As elucidated in [13], every propagation path within a given cluster is characterized by a mean AOA and follows a specific probability density function (PDF), which may include Gaussian and Laplacian PDFs. To this end, by utilizing the mean AOA the user's single-clustered channel, the channel subset $\mathcal{H}_r$ contains those channels whose mean AOA falls within the interval $\Psi_r\triangleq\Big[\beta_1+(r-1)\frac{\beta_2-\beta_1}{R},\beta_1+r\frac{\beta_2-\beta_1}{R}\Big)$, where Ψ is divided into R non-overlapping equally-spaced subareas. Following the above definition, the uneven distribution of users can be attributed to a significant size disparity between one subset and the remaining subsets, i.e., $\exists n\in[1,R]$ $s.t.$ $\forall r\in[1,R]$ and $r\neq n$, $|\mathcal{H}_n|\gg|\mathcal{H}_r|$. Consequently, $\mathcal{H}_n$ can be interpreted as $\mathcal{H}_{\text{major}}$, representing the channel set of the majority of users. The remaining channels can be classified as $\mathcal{H}_{\text{minor}}$, which denotes the set of channels of the users belonging to the minority group. Therefore, it can be stated that $\mathcal{H}_{\text{major}}\bigcup\mathcal{H}_{\text{minor}}=\mathcal{H}$ and $|\mathcal{H}_{\text{major}}|\gg|\mathcal{H}_{\text{minor}}|$.

III. Problem Definition

In this paper, our objective is to formulate a design for an N-beam codebook denoted as $\mathcal{W}$ that can effectively adapt to the imbalanced user distribution described in Section II. To this end, the codebook $\mathcal{W}$ construction process should be transformed into an optimization problem.

In detail, the Signal-to-noise ratio (SNR) of user k after combining is given as

$$\mathrm{SNR}_k = \frac{\mathbb{E}\left\{\left|\mathbf{w}_{\upsilon_k}^{\dagger}\mathbf{h}_k a_k\right|^2\right\}}{\mathbb{E}\left\{\left|\mathbf{w}_{\upsilon_k}^{\dagger}\mathbf{n}_k\right|^2\right\}} = \frac{P_{a_k}}{\sigma^2}\left|\mathbf{w}_{\upsilon_k}^{\dagger}\mathbf{h}_k\right|^2, \tag{3}$$

where the second equation is achieved when satisfying the constrictions outlined in (1). Note that, the corresponding υ_k is arbitrary in this case and may not be the optimal choice to match the user and then maximize the SNR. To this end, the optimal υ_k is expected to be derived through an exhaustive searching through $\mathbf{w}_n$ from the beam codebook, i.e.,

$$\upsilon_k^* = \arg\max_n \left|\mathbf{w}_n^{\dagger}\mathbf{h}_k\right|^2, \quad \text{s.t. } n=1,\ldots,N, \tag{4}$$

where υ_k^* denotes the best matched beamformer chosen from the N-beam codebook $\mathcal{W}$. Then, the objective is clarified, i.e., the designed codebook should maximize SNR gains over the entire dataset. Consequently, formulation of the optimal codebook $\mathcal{W}_{\text{opt}}$ can be restated as

$$\mathcal{W}_{\text{opt}} = \arg\max_{\mathcal{W}} \sum_{k=1}^{K}\left|\mathbf{w}_{\upsilon_k^*}^{\dagger}\mathbf{h}_k\right|^2, \tag{5}$$
$$\text{s.t. } \left|[\mathbf{w}_{\upsilon_k^*}]_m\right| = \frac{1}{\sqrt{M}},\ \forall m = 1,...,M.$$

In the next section, we will analyze the unequal SNR elevation for the minority and majority caused by the uneven user distribution and the batch based gradient descent in CML. Then, we propose a novel DL-based solution that addresses the issue.

IV. Distributed Machine Learning Solution

In this section, we leverage the decentralized characteristic and online learning feature of DL to build an adaptive codebook in uneven user distribution. Our approach begins with the introduction of a pre-classification method, which distinguishes between $\mathcal{H}_{\text{minor}}$ and $\mathcal{H}_{\text{major}}$, and then allocates them to different UEs. Following the allocation, we propose a DL architecture to achieve the desired beam pattern for various subsets at UEs. Then, a novel aggregation method is proposed to equally integrate the local model into the global model.

A. Pre-classification for Dataset Allocation

Based on the user distribution description provided in Section II, the following objective is to categorize users and their channels into R distinct classes, thereby differentiating between $\mathcal{H}_{\text{minor}}$ and the corresponding $\mathcal{H}_{\text{major}}$. In essence, the feature of each class can be identified as the median angle γ_r of subregion r ($\gamma_r \triangleq \frac{2\beta_1+(2r-1)\frac{\beta_2-\beta_1}{R}}{2}$)[9]. Hence, by defining the median angle of each class as the feature, the pre-classified index $r' \in \{1,...,R\}$ can be expressed as

$$r' = \arg\max_i \left|\mathbf{a}(\gamma_i)^{\dagger}\mathbf{h}_k\right|^2, \text{s.t.} \quad i=1,\ldots,R, \tag{6}$$

where the operation $\mathbf{a}(\cdot)$ represents the array response (recall (1)) of a specific AOA and R exhaustive searches are conducted to determine the optimal index r' that maximizes the corresponding received power $\mathcal{P}_i = \left|\mathbf{a}(\gamma_i)^{\dagger}\mathbf{h}_k\right|^2$. This also indicates that the channels in $\mathcal{H}_{r'}$ share similar power responses to the featured combining vector in r'th subregion. Then, the classified channels are delivered to the r'th UE as part of $\mathcal{H}_{r'}$ for the subsequent DL tasks.

B. Model Architecture

This paper adopts the self-supervised method outlined in [6] as the backbone network which doesn't rely on explicit channel information. The details of the DL architecture will be elaborated in the subsequent parts.

1) Complex-valued Fully-connected and Power Computation Layer: The first layer aims to execute complex-valued multiplicative and additive operations. Here, the inner product is given by

$$\mathbf{y} = \mathbf{W}_{r'}^{\dagger}\mathbf{h}, \tag{7}$$

where $\mathbf{y} \in \mathbb{C}^{N\times 1}$ signifies the received signal after combining and $\mathbf{h} \in \mathbb{C}^{M\times 1}$ denotes a channel from the local subset $\mathcal{H}_{r'}$. The term $\mathbf{W}_{r'} \in \mathbb{C}^{M\times N}$ denotes the local codebook matrix in the r'th UE. This corresponding codebook is attained via the $\frac{1}{\sqrt{M}}$ scaled phase-to-complex operation as follows,

$$\mathbf{W}_{r'} = \frac{1}{\sqrt{M}}\left(\cos\left(\boldsymbol{\Theta}_{r'}\right) + j\sin\left(\boldsymbol{\Theta}_{r'}\right)\right), \tag{8}$$

where $\boldsymbol{\Theta}_{r'} = \left[\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_n,\ldots,\boldsymbol{\theta}_{N_{r'}}\right] \in \mathbb{R}^{M\times N_{r'}}$ is the phase shift matrix and $\boldsymbol{\theta}_n = \left[\theta_{1,n},...,\theta_{M,n}\right]^{\mathrm{T}}$, $n \in \{1,...,N_{r'}\}$ is the single phase shift vector. Then, the received power associated with each combining vector is derived by calculating the square modulus of every complex number in the received signal vector $\mathbf{y}$. Thus, the nth element of the received signal power vector $\mathbf{p} \in \mathbb{R}^{N_{r'}\times 1}$ is defined as $p_n = |[\mathbf{y}]_n|^2$.

2) Softmax and Argmax Layer: In this layer, the initial step involves the application of the softmax operation to infer the "likelihood" of how likely a combining vector suits the current channel $\mathbf{h}$ best, which reflects on the received power $\mathbf{p}$. Accordingly, the nth element of the softmax vector $\mathbf{s} \in \mathbb{R}^{N_{r'}\times 1}$ can be given as $s_n = e^{p_n}/\sum_{j=1}^{N_{r'}} e^{p_j}$. On a parallel track, the argmax layer then generates the one-hot vector $\mathbf{c} = [c_1,\ldots,c_n,\ldots,c_{N_{r'}}]^{\mathrm{T}} \in \mathbb{R}^{N_{r'}\times 1}$. The nth element $c_n = 1$ when $n = \arg\max_i s_i$, $i \in \{1,...,N_{r'}\}$, and it is set to 0 otherwise. The one-hot label $\mathbf{c}$ denotes the position of the best combining vector and help ensure the gain of the combining vector is still the largest when faced with similar channels via backpropagation. Stepping into how the DL-based codebook is learned, the next subsection will delve into the local backpropagation and the complete DL solution.

C. Learning the DL-based Codebooks

1) Local Backpropagation: Within a batch, the quality of the codebook is evaluated by quantifying the discrepancy

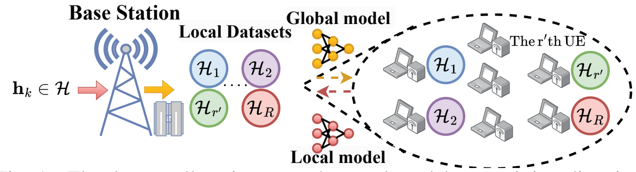


Fig. 1. The dataset allocation procedure and model transmitting direction.

between the corresponding $\mathbf{s}$ and $\mathbf{c}$ by the cross-entropy loss in a batch, i.e.,

$$\mathcal{L}_b = -\sum_{n=1}^{N_{r'}} c_{nb} \log_2 s_{nb},\ b \in \{1, ..., B\}, \tag{9}$$

where s_{nb} and c_{nb} represent the nth element of the bth data pair in a batch. The term B denotes the number of channels in a single batch. Notably, in the CML-based method, the channels in a single batch randomly select channel from both $\mathcal{H}_{\text{major}}$ and $\mathcal{H}_{\text{minor}}$ which constitute the global dataset $\mathcal{H}$. In contrast, the DL-based batch exclusively consists of $\mathcal{H}_{\text{major}}$ or $\mathcal{H}_{\text{minor}}$. Furthermore, the loss function above treats the one-hot encoded label $\mathbf{c}$ as the probability distribution's objective for the model. Its value is minimized by adjusting phase shift vectors $\boldsymbol{\theta}_n \in \boldsymbol{\Theta}_{r'}$ to diminish the dissimilarity between $\mathbf{s}$ and $\mathbf{c}$ as much as possible. Then, the backpropagation gradient of phase shift vectors $\mathbf{g}_B(\boldsymbol{\theta}_n)$ for a batch is computed through the chain rule [6], i.e.,

$$\mathbf{g}_B(\boldsymbol{\theta}_n) = \sum_{b=1}^{B} \left(\frac{\partial \mathcal{L}_b}{\partial \boldsymbol{\theta}_n}\right)^{\mathrm{T}} = \sum_{b=1}^{B} \left(\frac{\partial \mathcal{L}_b}{\partial \mathbf{s}}\right)^{\mathrm{T}} \cdot \frac{\partial \mathbf{s}}{\partial \mathbf{p}} \cdot \frac{\partial \mathbf{p}}{\partial \mathbf{y}} \cdot \frac{\partial \mathbf{y}}{\partial \boldsymbol{\theta}_n}. \tag{10}$$

Therefore, the local phase matrix $\boldsymbol{\Theta}_{r'}$ can be updated using the stochastic gradient descent (SGD) method, which is defined as

$$\boldsymbol{\theta}_n^{'} = \boldsymbol{\theta}_n - \eta \cdot \mathbf{g}_B(\boldsymbol{\theta}_n)^{\mathrm{T}}, \tag{11}$$

where η represents the learning rate of the SGD process. The terms $\boldsymbol{\theta}_n^{'}$ and $\boldsymbol{\theta}_n$ denote the phase shift vectors after and before updating in an updating cycle.

Remark 1: The main drawback of CML lies in the batch-based gradient. The gradient of channels from $\mathcal{H}_{\text{minor}}$ might be overlooked if there is an imbalance in number of the users in different regions, which means SNR of minority users won't be elevated significantly through the updating. To illustrate, consider a CML-based updating process of a phase shift matrix $\boldsymbol{\Theta} \in \mathbb{R}^{M\times N}$ based on the global dataset $\mathcal{H}$, the number of channels from $\mathcal{H}_{\text{major}}$ in a batch is assumed to be B_{major}, channels from $\mathcal{H}_{\text{minor}}$ is set as B_{minor} ($B_{\text{major}} \gg B_{\text{minor}}$, $B_{\text{major}} + B_{\text{minor}} = B$). The phase shifter $\boldsymbol{\theta}_n$ serves both the minority and majority, which is common when size of the codebook is relatively small. To this end, the gradient of CML with batch size B can be expressed as

$$\mathbf{g}_B(\boldsymbol{\theta}_n) = \sum_{b=1}^{B} \left(\frac{\partial \mathcal{L}_b}{\partial \boldsymbol{\theta}_n}\right)^{\mathrm{T}} = \sum_{b=1}^{B_{\text{major}}} \left(\frac{\partial \mathcal{L}_b}{\partial \boldsymbol{\theta}_n}\right)^{\mathrm{T}} + \sum_{b=1}^{B_{\text{minor}}} \left(\frac{\partial \mathcal{L}_b}{\partial \boldsymbol{\theta}_n}\right)^{\mathrm{T}}. \tag{12}$$

Accordingly, if $B_{\text{major}} \gg B_{\text{minor}}$, the CML-based method tends to overlook the gradient computed in certain regions when the number of channels in those regions is much smaller than in the region with the majority of users.

2) Distributed Learning: Fig. 1 illustrates the dataset allocation process and the direction of model transmission. Following the local epochs, the phase shift matrices $\boldsymbol{\Theta}_{r'}$ from various UEs are transmitted to the BS for synchronous aggregation. Naturally, we aim to aggregate the model across different $N_{r'}$ distributions evenly, thereby alleviating the issue of ignorance (recall *Remark 1*) in CML. Motivated by the objective above, Discat is proposed to concatenate phase shift matrices of varying sizes $N_{r'}$ to form a global model, as mathematically expressed by

$$\boldsymbol{\Theta}_{\text{cat}} = [\boldsymbol{\Theta}_1, ..., \boldsymbol{\Theta}_{r'}, ..., \boldsymbol{\Theta}_R], \tag{13}$$

where local models $\boldsymbol{\Theta}_{r'}$ from different UEs are combined along the second dimension. Therefore, $\boldsymbol{\Theta}_{\text{cat}} \in \mathbb{R}^{M\times \sum_{r'=1}^{R} N_{r'}}$ denotes the model after aggregation and the Discat codebook is represented by $\mathbf{W}_{\text{cat}}$. However, the concatenation process implies invalid information exchange between different local models for the next communication round. Hence, the global model is expected to be updated using the same manner as the local model. Nevertheless, instead of pursuing convergence like the local updating, minimal global updating cycles are required when extracting the feature of the global dataset $\mathcal{H}$. For the next communication round, $\boldsymbol{\Theta}_{\text{cat}}$ is split up at the columns where it was previously concatenated and returns to the original sizes before aggregation, which can be regarded as the inverse process of (13). Then, the partitioned updated subset $\boldsymbol{\Theta}_{r'}$ is broadcasted back to the corresponding UE to start the local updating of the next communication round. So far, as concluded in Fig. 2, the complete DL neural network architecture and aggregation strategies have been introduced. The next part will elucidate how DL enables online learning.

3) Online Learning: As for practicality, it has been stated in [6] that offline learning feature of supervised CML in BS is its main drawback. That is to say, it is impractical to wait for the time-consuming convergence of the model at BS and not utilize the evolving codebooks during updating. From a trade-off perspective, although DL introduces a higher communication overhead, it alleviates the BS's computational load. This, in return, allows the BS to deal with other communication tasks during the time-consuming UE update phases. Meanwhile, there is no need to collect all the channel data in advance to construct $\mathcal{H}$. Actually, the $\mathbf{h}_k$ can be collected, classified and sent to the corresponding UEs with local models in each communication round, thus realizing the online-classification. Furthermore, the new environmentally adapted codebook constructed in each communication round can be used for collecting more channel state information for the global and local datasets or performing other tasks in BS, the same as the concept of online learning as discussed in [6].

Remark 2: The DL-based methods effectively address the limitation of the batch-based gradient. When considering the updating process of a phase shift matrix $\boldsymbol{\Theta} \in \mathbb{R}^{M\times N}$, the

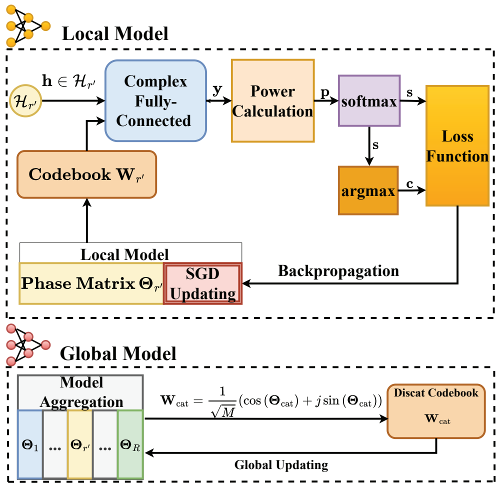


Fig. 2. The neural network architecture and the model aggregation strategies.

improvement of the minority users' SNR through updating can be quantified by the contribution of $\mathcal{H}_{\text{minor}}$ to the total gradient in a given model, i.e.,

$$\mathcal{G}_{\text{CML}} \triangleq \frac{\mathbf{g}_{B_{\text{minor}}}(\boldsymbol{\theta}_n)}{\mathbf{g}_B(\boldsymbol{\theta}_n)} = \frac{\sum_{b=1}^{B_{\text{minor}}} \left(\frac{\partial \mathcal{L}_b}{\partial \boldsymbol{\theta}_n}\right)^{\text{T}}}{\sum_{b=1}^{B} \left(\frac{\partial \mathcal{L}_b}{\partial \boldsymbol{\theta}_n}\right)^{\text{T}}}, \tag{14}$$

where $\mathcal{G}$ is the proportion of the gradient calculated from $\mathcal{H}_{\text{minor}}$ in the total gradient $\mathcal{L}_B$. For comparison, if we consider a UE model updating process for the minority subset, $\mathcal{G}$ of Discat is given as

$$\mathcal{G}_{\text{Discat}} = \frac{\mathbf{g}_{B_{\text{minor}}}(\boldsymbol{\theta}_n)}{\mathbf{g}_{B_{\text{minor}}}(\boldsymbol{\theta}_n)} = 1, \tag{15}$$

Obviously, $\mathcal{G}_{\text{Discat}} > \mathcal{G}_{\text{CML}}$ when $B \gg B_{\text{minor}}$ in the assumed scenario (recall *Remark 1*). Therefore, it is found that in the proposed methods, the minority of the users makes a greater contribution to the ultimate gradient. Hence, the SNR improvement of the minority is ensured, compared with the CML solution.

TABLE I
HYPER-PARAMETER FOR CHANNEL GENERATION AND MODEL TRAINING INSTANCE

Parameter	Value	Parameter	Value
Wavelength λ	[1mm, 10mm]	UE learning epoch	5
Number of antennas	64	BS learning epoch	1
Angle spread	$5°$	UE batch size	100
Type of PDF	Laplacian PDF	BS batch size	5000
Antenna spacing d	$\lambda/2$	UE learning rate schedule	0.1,0.01
Number of paths	5	BS learning rate schedule	0.01
Number of subareas	10	Validation rate of UEs	0.1
Active UE	4	Validation rate of BS	0.1

V. Simulation Result

A. Experimental Settings

In order to assess the effectiveness of the DL method in handling the imbalance of $\mathcal{H}$, a scenario with 10 subregions is selected. The AOAs of $\mathbf{h} \in \mathcal{H}$ are confined within the range $\Psi = [0, \pi)$, and Ψ is subdivided into 10 non-overlapping equally-spaced subregions. The clustered AOAs of the generated channels, within the nth subregion, follow a Laplacian distribution with the same mean angle. Specifically, the mean angle of AOA in area n falls in the interval $[\frac{\pi}{10}(n-1), \frac{\pi}{10}n)$. A total of 10000 realizations of $\mathbf{h}$ are generated in area 10, which represents the majority of users, while 50 realizations of $\mathbf{h}$ in each area of 1,3,4,5 stand for the minority groups.Table I lists the additional hyper-parameters used for channel generation. The averaged achievable rate for all candidate users is represented as $\mathcal{R} = \frac{1}{K}\sum_{k=1}^{K}\log_2\left(1 + \frac{P_{a_k}}{\sigma^2}\left|\mathbf{w}_{\upsilon_k}^{\dagger}\mathbf{h}_k\right|^2\right)$ and the term P_{a_k}/σ^2 is set as ρ (recall (3)). Furthermore, to evaluate the performance of the proposed method, it is compared with the N-beam DFT codebook constructed as $\mathbf{w}_n^{\text{DFT}} = \frac{1}{\sqrt{M}}[1, ..., e^{j\frac{2\pi n}{M}}, ..., e^{j\frac{2\pi n(M-1)}{M}}]^{\text{T}}, n = 0, 1, ..., N-1$ and the self-supervised CML solution in the prior work [6]. The equal-gain combining vector $\mathbf{w}_{\text{egc}}$ is modeled as $\mathbf{w}_{\text{egc}} = \frac{1}{\sqrt{M}}\left[e^{j\angle h_1}, e^{j\angle h_2}, \ldots, e^{j\angle h_M}\right]^{\text{T}}$, where the upper bound of received power for each user is given as $p = \left|\mathbf{w}_{\text{egc}}^{\dagger}\mathbf{h}\right|^2 = \frac{1}{M}\|\mathbf{h}\|_1^2$. A parameter example for learning is also shown in Table I. For the DL-based solution, 4 UEs are activated and the codebooks sizes of different UEs are kept consistent. Besides, the UE validation sets comprise the BS validation set .

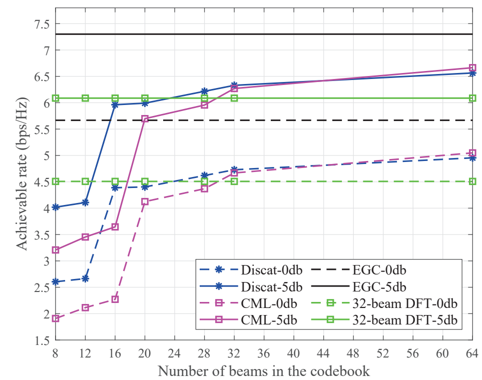


Fig. 3. Average achievable rate against the size of codebook utilizing different methods.

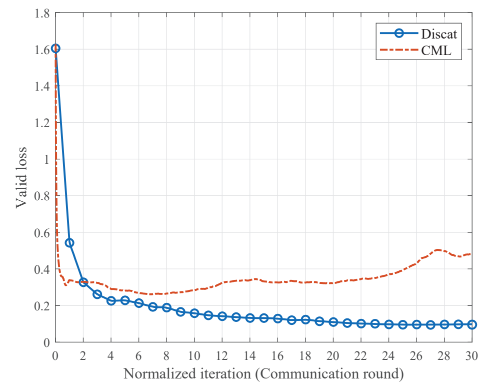
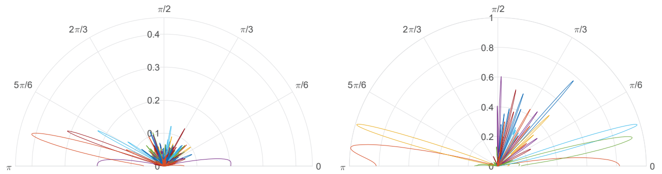


Fig. 4. Convergence behavior of 16-beam codebooks (loss vs. normalized iteration/communication round).



(a) 16-beam CML codebook (b) 16-beam Discat codebook

Fig. 5. Beam patterns of 16-beam codebook with learned by the CML and DL solutions in a single-cluster multipath setting.

B. Performance Analysis

1) Pre-classification Result: As it was previously noted in subsection A, when generating the dataset, area 10 contains 10,000 realizations, whereas the remaining regions have 200 realizations. After pre-classification simulation, the channel sets $\mathcal{H}_1$, $\mathcal{H}_3$, $\mathcal{H}_4$, and $\mathcal{H}_{10}$ contained 48, 69, 124, and 9959 channels, respectively. Therefore, 9959 of the channels are identified as $\mathcal{H}_{\text{major}}$ and 241 as $\mathcal{H}_{\text{minor}}$. This result indicates that none ideal conditions such as limited subarea division and angle spacing preclude accurate classification but the difference between $\mathcal{H}_{\text{major}}$ and $\mathcal{H}_{\text{minor}}$ can still be distinguished.

2) Achievable Rate Comparison: Fig. 3 presents the average achievable rate against the number of beams of the codebook when ρ is set as 0 dB and 5 dB, with the EGC receiver as the upper bound. From the experiment, larger codebook sizes correspond to higher gains and achievable rates, which suggests superior environmental adaptivity. As for the performance of the proposed Discat method, it achieves a higher achievable rate than the CML method (except for 64-beam). Particularly, when the codebook size is small, the method exhibits better adaptation to the uneven user distribution, which is consistent with the analysis in *Remark 1*. This result signifies a significant reduction in beam training overhead. For instance, when the codebook size is 16, the proposed method shows similar performance to the 32-beam DFT codebook, while the CML method falls behind.

3) Convergence of the DL Solution: Convergence is vital for distributed learning. Here, Fig. 4 plots the loss of CML versus the normalized iteration which contains 83 updating cycles (same as the number of UE cycles in one communication round) and the valid loss of DL versus the communication round. As for the CML solution, it almost reaches saturation during the first normalized iteration. After that, the valid loss even increases with the oscillatory behavior. In contrast, the Discat method achieves convergence slower, which is close to saturation in the 20th communication round. However, the performance is better than CML.

4) Beam Pattern Analysis: Fig. 5 illustrates the multiple beam patterns generated from two learned codebooks. These patterns depict the learning outcomes of the proposed and previous methods. The CML codebook beam magnitudes of minority groups of channels in areas 1, 3, 4, and 5 aren't elevated significantly after updating. In contrast, the DL-based method adapts a single-cluster multipath scenario with multi-lobe beams or directly aligns with the orientation of the minority with a magnitude greater than 0.8.

VI. Conclusion

In this paper, we proposed a DL framework to address the codebook design problem in mmWave massive MIMO communication systems with uneven user distribution. The proposed framework addressed the issue of neglecting minority users in the CML-based codebook design problem and enhanced the adaptivity of the neural network-based beam codebook design. Moreover, DL was capable of online learning by sharing the computation burden with UEs. In the initial part of our method, we classified user channel sets $\mathcal{H}$ based on the power responses of the featured combining vectors from different subregions, resulting in the attainment of R subsets. Then, we devised a novel DL architecture to process all the pre-classified subsets. In terms of model aggregation, we proposed Discat for cases with different UE codebook sizes. The DL framework captured channel characteristics of the minority and showed superiority over the traditional CML method in terms of achievable rate. In future work, we plan to further extend a more communication-efficient DL approach to encompass the terahertz communication system and apply it to hybrid precoding design.

References

[1] A. Alkhateeb, J. Mo, N. Gonzalez-Prelcic, and R. W. Heath, Jr, "MIMO precoding and combining solutions for millimeter-wave systems," *IEEE Commun. Mag.*, vol. 52, no. 12, pp. 122-131, Dec. 2014.

[2] M. Giordani, M. Polese, A. Roy, D. Castor, and M. Zorzi, "A tutorial on beam management for 3GPP NR at mmWave frequencies," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 173-196, 1st Quart., 2018.

[3] S. Hur, T. Kim, D. J. Love, J. V. Krogmeier, T. A. Thomas, and A. Ghosh, "Millimeter wave beamforming for wireless backhaul and access in small cell networks," *IEEE Trans. Commun.*, vol. 61, no. 10, pp. 4391-4403, Oct. 2013.

[4] C. Kong, C. Zhong, and Z. Zhang, "Performance of ZF precoder in downlink massive MIMO with non-uniform user distribution," *J. Commun. Netw.,* vol. 18, no. 5, pp. 688-698, Oct. 2016.

[5] J. Ye, X. Ge, G. Mao, and Y. Zhong, "5G ultradense networks with nonuniform distributed users," *IEEE Trans. Veh. Technol.*, vol. 67, no. 3, pp. 2660-2670, Mar. 2018.

[6] M. Alrabeiah, Y. Zhang, and A. Alkhateeb, "Neural networks based beam codebooks: Learning mmWave massive MIMO beams that adapt to deployment and hardware," *IEEE Trans. Commun.*, vol. 70, no. 6, pp. 3818-3833, Jun. 2022.

[7] Y. Zhang, M. Alrabeiah, and A. Alkhateeb, "Reinforcement learning of beam codebooks in millimeter wave and terahertz MIMO systems," *IEEE Trans. Commun.*, vol. 70, no. 2, pp. 904-919, Feb. 2022.

[8] Y. Heng, J. Mo, and J. G. Andrews, "Learning site-specific probing beams for fast mmWave beam alignment," *IEEE Trans. Wireless Commun.,* vol. 21, no. 8, pp. 5785-5800, Aug. 2022.

[9] A. M. Elbir and S. Coleri, "Federated learning for hybrid beamforming in mm-Wave massive MIMO," *IEEE Wireless Commun. Lett.*, vol. 24, no. 12, pp. 2795-2799, Dec. 2020.

[10] J. Wang, Q. Xue, Y. Sun, G. Feng, L. Tang, and S. Ma, "Beam management in ultra-dense mmWave network via federated reinforcement learning: An intelligent and secure approach," *IEEE Trans. Cogn. Commun. Netw.,* vol. 9, no. 1, pp. 185-197, Feb. 2023.

[11] M. Kim, A. L. Swindlehurst, and D. Park, "Beamforming vector design and device selection in over-the-air federated learning," *IEEE Trans. Wireless Commun.*, 2023, to appear.

[12] O. E. Ayach, S. Rajagopal, S. Abu-Surra, Z. Pi, and R. W. Heath, Jr, "Spatially sparse precoding in millimeter wave MIMO systems," *IEEE Trans. Wireless Commun.,* vol. 13, no. 3, pp. 1499-1513, Mar. 2014.

[13] A. Forenza, D. J. Love, and R. W. Heath, Jr, "Simplified spatial correlation models for clustered MIMO channels with different array configurations," *IEEE Trans. Veh. Technol.*, vol. 56, no. 4, pp. 1924-1934, Jul. 2007.