Distributed Machine Learning Based Downlink Channel Estimation for RIS Assisted Wireless Communications

Linglong Dai, *Fellow, IEEE*, and Xiuhong Wei

***Abstract*— The downlink channel estimation requires a huge pilot overhead in the reconfigurable intelligent surface (RIS) assisted communication system. By exploiting the powerful learning ability of the neural network, the machine learning (ML) technique can be used to estimate the high-dimensional channel from a few received pilot signals at the user. However, since the training dataset collected by the single user only contains the information of part of the channel scenarios of a cell, the neural network trained by the single user is not able to work when the user moves from one channel scenario to another. To solve this challenge, we propose to leverage the distributed machine learning (DML) technique to enable the reliable downlink channel estimation. Specifically, we firstly build a downlink channel estimation neural network shared by all users, which can be collaboratively trained by the BS and the users with the help of the DML technique. Then, we further propose a hierarchical neural network architecture to improve the channel estimation accuracy, which can extract different channel features for different channel scenarios. Simulation results show that compared with the neural network trained by the single user, the proposed DML based neural networks can achieve better estimation performance with the reduced pilot overhead for all users from different scenarios.**

***Index Terms*— Distributed machine learning, reconfigurable intelligent surface (RIS), channel estimation.**

I. Introduction

RECENTLY, reconfigurable intelligent surface (RIS) is proposed as a potential technology for future 6G wireless communications [1]. Specifically, RIS is a surface composed of a large number of controllable elements made of low-cost passive components, where each RIS element can adjust the incident signal independently [2]. With the help of the RIS, the communication environment can be controlled intelligently by optimizing the coefficients of RIS elements to satisfy different communication requirements [3]. Thanks to its low-cost and configurability, RIS has been used to assist many communication applications. For example, by reasonably deploying RIS in the cell, RIS is able to overcome the blockage of the line of sight (LoS) path, particularly for millimeter-wave (mmWave) and terahertz (THz) communications [4], [5].

In the RIS assisted communication system, the base station (BS) sends downlink data to the user via the RIS [6]. However, the downlink channel estimation is a big challenge due to the introduction of the RIS. Specifically, since all RIS elements are passive, the channel between the BS and the RIS and the channel between the RIS and the user are difficult to be estimated separately. Generally, the user can only estimate the cascaded channel from the BS to the RIS and then from the RIS to the user. The cascaded channel estimation problem can be solved by some classical algorithms, such as least square (LS) [7], or minimum mean squared error (MMSE) [8]. However, the number of elements of the cascaded channel is equal to the product of the number of antennas at the BS and the number of RIS elements. As a result, compared with the conventional massive multiple-input multiple-output (MIMO) channel estimation without RIS, the required pilot overhead of the cascaded channel estimation for the above classical algorithms increases significantly in the RIS assisted communication system.

A. Prior Works

There have been some schemes recently proposed to reduce the required pilot overhead of the cascaded channel estimation for the RIS assisted communication system. Specifically, a multi-user correlation based cascaded channel estimation scheme was proposed in [9], where the total number of elements to be estimated for all users can be reduced at the BS. Moreover, by considering the double-structured sparsity of the multi-user angular cascaded channels, a multi-user joint cascaded channel estimation scheme with the compressive sensing (CS) tools was proposed to further reduce the pilot overhead [10]. However, most of existing overhead-reduced channel estimation schemes are proposed for the uplink channel estimation, where cascaded channels for all users can be estimated jointly to reduce the pilot overhead [9], [10]. By contrast, the downlink channel estimation is performed by each user independently, since the received pilot signals cannot be shared among different users. Therefore, these overhead-reduced multi-user joint uplink channel estimation

Manuscript received 10 July 2021; revised 2 December 2021 and 15 March 2022; accepted 8 May 2022. Date of publication 16 May 2022; date of current version 15 July 2022. This work was supported in part by the National Key Research and Development Program of China (Grant No. 2020YFB1807201), in part by the National Natural Science Foundation of China (Grant No. 62031019), and in part by the European Commission through the H2020-MSCA-ITN META WIRELESS Research Project under Grant 956256. The associate editor coordinating the review of this article and approving it for publication was A. Alkhateeb. *(Corresponding author: Linglong Dai.)*

The authors are with the Beijing National Research Center for Information Science and Technology (BNRist) and the Department of Electronic Engineering, Tsinghua University, Beijing 100084, China (e-mail: daill@tsinghua.edu.cn; weixh19@mails.tsinghua.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCOMM.2022.3175175.

Digital Object Identifier 10.1109/TCOMM.2022.3175175

schemes are no longer suitable for the downlink channel estimation.

Some machine learning (ML) techniques can be used to reduce the required pilot overhead of the downlink channel estimation. Specifically, the user can deploy the deep neural network (DNN) to map a nonlinear relationship between the low-dimensional signals and the high-dimensional downlink channel. By considering the correlation of adjacent RIS elements, an ordinary differential equation (ODE) based convolution neural network (CNN) was proposed to construct the entire high-dimensional cascaded channel from the low-dimensional sampled cascaded channel. The sampled cascaded channel is only corresponding to a small part of RIS elements, which can be estimated by the LS algorithm with low pilot overhead [11]. In order to further improve the estimation accuracy with low pilot overhead, a denoising CNN based cascaded channel estimation scheme was proposed by combing the CS algorithms with the denoising neural network [12]. However, the communication environment of different locations is different for the entire cell, which can be divided into different channel scenarios. The training dataset collected by the single user is limited, which only contains the information of part of the channel scenarios in the entire cell. Therefore, when the user moves from one channel scenario to another, the trained DNN by the single user cannot work [11], [12].

B. Our Contributions

In this paper, for the RIS assisted communication system, we leverage the distributed machine learning (DML) technique to enable the reliable downlink channel estimation with the reduced pilot overhead.[1] Our main contributions are summarized as follows.

1) We leverage the DML technique to train a downlink channel estimation neural network to enable it to work when the user moves from one channel scenario to another in a cell. Specifically, we build a global neural network shared by the BS and all users, which can be jointly trained based on the local training datasets available at all users. During the training process, each user firstly calculates the gradient vector of the weights of the global neural network on its own local training dataset. Then, all gradient vectors from all users are uploaded to the BS. The BS updates the weights of the global neural network based on all uploaded gradient vectors. Finally, the BS broadcasts the updated weights to all users for the next training iteration. By leveraging this DML based training method, the downlink channel estimation neural network can be trained by different training datasets from different channel scenarios. When the user moves from one channel scenario to another, the trained neural network can still work. However, since different channel scenarios have different channel features, the single neural network architecture cannot achieve the accurate channel estimation in different channel scenarios.
2) In order to further improve the channel estimation accuracy, we propose a DML based hierarchical neural network architecture to extract different channel features in different channel scenarios. Specifically, the proposed hierarchical neural network consists of three parts: the scenario classifier, the feature extractors, the feature mapper. Firstly, the received pilot signals are input to the scenario classifier to predict which scenario the channel to be estimated belongs to. Then, we design a corresponding feature extractor for each channel scenario. Based on the output of the scenario classifier, the received pilot signals are input to the corresponding feature extractor to extract the channel feature. Finally, the extracted channel feature is input to the feature mapper to construct the downlink cascaded channel. Moreover, we propose the DML based hierarchical training method to train the proposed hierarchical neural network, where the scenario classifier is trained separately, while the feature extractors and the feature mapper are jointly trained.
3) We provide the simulation results to verify the channel estimation performance of the proposed schemes. Compared with the channel estimation neural network only trained by the single user, the proposed DML based neural networks can work well when the users from one scenario to another in a cell. Further, the proposed hierarchical neural network architecture can achieve better channel estimation accuracy than the single neural network architecture.

C. Organization and Notation

The rest of the paper is organized as follows. In Section II, we introduce the system model in the RIS assisted communication system and review the basic principle of the downlink channel estimation neural network. In Section III, we first propose to leverage the DML technique to train the downlink channel estimation neural network, and then further propose the hierarchical neural network architecture to improve the estimation accuracy. The computational complexity of the two proposed schemes is also analyzed in Section III. Simulation results and conclusions are given in Section IV and Section V, respectively.

Notation: Lower-case and upper-case boldface letters $\mathbf{a}$ and $\mathbf{A}$ denote a vector and a matrix, respectively; $\mathbf{A}^H$ and $\mathbf{A}^T$ denote the conjugate transpose and transpose of matrix $\mathbf{A}$, respectively; $\mathrm{diag}\,(\mathbf{a})$ denotes the diagonal matrix with the vector $\mathbf{a}$ on its diagonal; $\|\mathbf{a}\|_2$ denotes the l_2-norm of vector $\mathbf{a}$; $\mathbf{A}\otimes\mathbf{B}$ denotes the Kronecker product of $\mathbf{A}$ and $\mathbf{B}$.

II. System Model

In this section, we first introduce the RIS assisted wireless communication system. Then, the downlink channel estimation problem at the user is formulated. Finally, the basic principle of the downlink channel estimation neural network is presented.

[1]Simulation codes are provided to reproduce the results presented in this article: http://oa.ee.tsinghua.edu.cn/dailinglong/publications/publications.html.

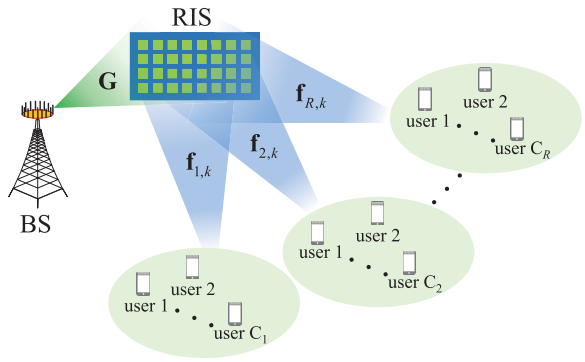


Fig. 1. The RIS assisted wireless communication system.

A. Signal Model

We consider a RIS assisted single-cell narrowband wireless communication system, as shown in Fig. 1, where the BS employing the M-antenna uniform planer array (UPA) communicates with multiple single-antenna users. An RIS with N reflecting elements is deployed between the BS and the users to enhance communication. The RIS can be controlled by the BS via a separate wireless link [3], [5], [8]. The entire cell can be divided into R regions, where the corresponding channel features of the users located in the same region are similar, while the channel features of the users located in different regions are different. In this paper, the different regions also refer to the different channel scenarios. We can define different channel scenarios based on different factors, such as the angles of the channels, the distance from the current location to BS and so on. When the user moves from one channel scenario to another, the channel features may change greatly. We denote that the number of users located in rth region is C_r $(r = 1, 2, \cdots, R)$.

In this paper, the frequency division duplex (FDD) mode is considered in the RIS assisted system [13], [14], where the uplink and downlink channels are not reciprocal[2] It is noted that this paper only focuses on how to obtain the channel state information of the reflecting link, while the direct channels between the BS and the users can be estimated as in conventional massive MIMO systems only by turning off all RIS elements.[3]

Let $\mathbf{G} \in \mathbb{C}^{N\times M}$ denote the channel from the BS to the RIS, and $\mathbf{f}_{r,k} \in \mathbb{C}^{1\times N}$ denote the channel from the RIS to the kth user $(k = 1, 2, \cdots, C_r)$ located in the rth region. Assuming the BS transmits the signal to the users via the reflecting link with the RIS, the downlink transmission signal model after removing the impact of the direct channel is given by

$$y_{r,k} = \mathbf{f}_{r,k}\mathbf{\Phi}\mathbf{G}\mathbf{w}_{r,k}x_{r,k} + n_{r,k}, \tag{1}$$

[2]Note that even in the time division duplex (TDD) mode, the uplink and downlink channels cannot be guaranteed to be reciprocal in some hardware implementations of RIS [15]. In order to achieve effective beamforming for the BS and the RIS, the accurate downlink channel estimation is required [16].

[3]Note that "turn off" is a widely used expression in the literature but inaccurate, since an RIS with all elements turned off is also a scatterer to reflect the incident electromagnetic wave. An implementation method with a special setting of RIS elements proposed in [17] can realize the perfect "turn off" for the incident electromagnetic wave [18].

where $y_{r,k}$ is the corresponding received signal at the user, $\mathbf{\Phi} \in \mathbb{C}^{N\times N}$ is the reflecting matrix at the RIS, $\mathbf{w}_{r,k} \in \mathbb{C}^{M\times 1}$ is the precoding vector at the BS, $x_{r,k} \in \mathbb{C}$ is the transmitted signal at the BS, and $n_{r,k} \in \mathbb{C}$ is the additive noise, respectively. It is noted that the reflecting matrix $\mathbf{\Phi} = \mathrm{diag}(\phi_1, \phi_2, \cdots, \phi_N)$ is a diagonal matrix, where ϕ_n represents the reflecting coefficient of the nth RIS element.

The channels $\mathbf{G}$ and $\mathbf{f}_{r,k}$ can be represented by the widely used Saleh-Valenzuela channel model. Specifically, $\mathbf{G}$ is given by

$$\mathbf{G} = \sqrt{\frac{MN}{L_G}} \sum_{l_1=1}^{L_G} \alpha_{l_1}^G \mathbf{a}\left(\vartheta_{l_1}^{G_\mathrm{r}}, \psi_{l_1}^{G_\mathrm{r}}\right) \mathbf{b}\left(\vartheta_{l_1}^{G_\mathrm{t}}, \psi_{l_1}^{G_\mathrm{t}}\right)^T, \tag{2}$$

where L_G represents the number of paths between the BS and the RIS, $\alpha_{l_1}^G$, $\vartheta_{l_1}^{G_\mathrm{r}}$ $(\psi_{l_1}^{G_\mathrm{r}})$, and $\vartheta_{l_1}^{G_\mathrm{t}}$ $(\psi_{l_1}^{G_\mathrm{t}})$ represent the complex gain, the azimuth (elevation) angle at the RIS, and the azimuth (elevation) angle at the BS for the l_1th path. We can find that $\mathbf{G}$ is common for all users. Once the RIS is deployed in the cell, $\mathbf{G}$ is almost unchanged.

Similarly, the channel $\mathbf{f}_{r,k}$ can be represented by

$$\mathbf{f}_{r,k} = \sqrt{\frac{N}{L_{r,k}}} \sum_{l_2=1}^{L_{r,k}} \alpha_{l_2}^{r,k} \mathbf{a}\left(\vartheta_{l_2}^{r,k}, \psi_{l_2}^{r,k}\right)^T, \tag{3}$$

where $L_{r,k}$ represents the number of paths between the RIS and the kth user located in the rth region, $\alpha_{l_2}^{r,k}$, $\vartheta_{l_2}^{r,k}$, and $\psi_{l_2}^{r,k}$ represent the complex gain, the azimuth angle and the elevation angle at the RIS for the l_2th path. $\mathbf{b}(\vartheta, \psi) \in \mathbb{C}^{M\times 1}$ and $\mathbf{a}(\vartheta, \psi) \in \mathbb{C}^{N\times 1}$ represent the normalized array steering vector associated to the BS and the RIS, respectively. For a typical $N_1 \times N_2$ $(N = N_1 \times N_2)$ UPA, $\mathbf{a}(\vartheta, \psi)$ can be represented by [19]

$$\mathbf{a}(\vartheta, \psi) = \frac{1}{\sqrt{N}}\left[e^{-j2\pi d\cos(\psi)\mathbf{n}_1/\lambda}\right] \otimes \left[e^{-j2\pi d\sin(\psi)\cos(\theta)\mathbf{n}_2/\lambda}\right], \tag{4}$$

where $\mathbf{n}_1 = [0, 1, \cdots, N_1 - 1]^T$ and $\mathbf{n}_2 = [0, 1, \cdots, N_2 - 1]^T$, λ is the carrier wavelength, and d is the antenna spacing usually satisfying $d = \lambda/2$.

By denoting $\boldsymbol{\phi} = [\phi_1, \phi_2, \cdots, \phi_N]$, (1) can be further represented by

$$\begin{aligned} y_{r,k} &= \boldsymbol{\phi}\mathrm{diag}(\mathbf{f}_{r,k})\mathbf{G}\mathbf{w}_{r,k}x_{r,k} + n_{r,k}, \\ &= \boldsymbol{\phi}\mathbf{H}_{r,k}\mathbf{w}_{r,k}x_{r,k} + n_{r,k}, \end{aligned} \tag{5}$$

where $\mathbf{H}_{r,k} = \mathrm{diag}(\mathbf{f}_{r,k})\mathbf{G}$ represents the downlink cascaded channel corresponding to the kth user located in rth region. Since the RIS has no ability to process signals independently, the cascaded channel $\mathbf{H}_{r,k}$ is usually estimated instead of the two separate channels $\mathbf{G}$ and $\mathbf{f}_{r,k}$ [7], [8].

B. Problem Formulation

In order to estimate the downlink cascaded channel $\mathbf{H}_{r,k}$, the BS is required to send the known pilot signals to the users via the RIS over Q time slots. According to (5), in the qth $(q = 1, 2, \cdots, Q)$ time slot, the received pilot signal $y_{r,k,q}^p \in \mathbb{C}$ at the kth user located in rth region can be represented as

$$y_{r,k,q}^p = \boldsymbol{\phi}_q\mathbf{H}_{r,k}\mathbf{w}_{r,k}p_{r,k,q} + n_{r,k,q}, \tag{6}$$

where $p_{r,k,q}$ represents the pilot signal sent by the BS, $\boldsymbol{\phi}_q$ represents the reflecting vector of the qth time slot at the RIS, and $n_{r,k,q}$ is the received noise of the qth time slot at the user, which follows the complex Gaussian distribution with the mean 0 and the variance σ_n^2.

After Q time slots of pilot transmission, we can obtain the $Q \times 1$ overall received pilot vector $\mathbf{y}_{r,k}^p = [y_{r,k,1}^p, y_{r,k,2}^p, \cdots, y_{r,k,Q}^p]^T$ by assuming $p_{r,k,q} = 1$ as

$$\mathbf{y}_{r,k}^p = \boldsymbol{\Theta}\mathbf{H}_{r,k}\mathbf{w}_{r,k} + \mathbf{n}_{r,k}, \tag{7}$$

where $\boldsymbol{\Theta} = [\boldsymbol{\phi}_1^T, \boldsymbol{\phi}_2^T, \cdots, \boldsymbol{\phi}_Q^T]^T$ and $\mathbf{n}_{r,k} = [n_{r,k,1}, n_{r,k,2}, \cdots, n_{r,k,Q}]^T$. According to $\mathrm{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A})\mathrm{vec}(\mathbf{B})$, (7) can be represented by

$$\mathbf{y}_{r,k}^p = ({\mathbf{w}_{r,k}}^T \otimes \boldsymbol{\Theta})\mathrm{vec}(\mathbf{H}_{r,k}) + \mathbf{n}_{r,k}. \tag{8}$$

By denoting $\boldsymbol{\Psi}_{r,k} = ({\mathbf{w}_{r,k}}^T \otimes \boldsymbol{\Theta})$ and $\mathbf{h}_{r,k} = \mathrm{vec}(\mathbf{H}_{r,k})$, (8) can be further represented by

$$\mathbf{y}_{r,k}^p = \boldsymbol{\Psi}_{r,k}\mathbf{h}_{r,k} + \mathbf{n}_{r,k}. \tag{9}$$

It is noted that $\mathbf{w}_{r,k}$ and $\boldsymbol{\Theta}$ will be pre-designed to be fixed values for the channel estimation. Thus, like pilot signals, $\boldsymbol{\Psi}_{r,k}$ is known for both the BS and the user during the channel estimation. The downlink cascaded channel estimation is to estimate $\mathbf{h}_{r,k}$ based on the known $\mathbf{y}_{r,k}^p$ and $\boldsymbol{\Psi}_{r,k}$. After that, the estimated downlink cascaded channel can be fed back to the BS for effective beamforming by carefully designing the precoding vector $\mathbf{w}_{r,k}$ at the BS and the reflecting matrix $\boldsymbol{\Theta}$ at the RIS.

Since the dimension of the cascaded channel is N times larger than that of the conventional massive MIMO channel, the required pilot overhead for the downlink cascaded channel estimation is very high. For example, the required time slots Q for pilot transmission of the conventional LS algorithm should satisfy $Q \geq NM$. In order to reduce the pilot overhead, some ML techniques can be used to solve the channel estimation problem thanks to the powerful learning ability of DNNs. Like many existing ML based works, DNNs can be deployed at the users to enable communications, such as channel feedback [20], receiver design [21] and end-to-end communications [22]. Next, we will introduce the basic principle of the downlink cascaded channel estimation neural network.

C. The Basic Principle of DNNs

The DNN is deployed at the user side for the downlink cascaded channel estimation. Specifically, the channel estimation neural network builds the non-linear mapping relationship from the received pilot signal to the downlink cascaded channel, which can be represented by

$$\hat{\mathbf{h}}_{r,k} = \boldsymbol{f}_{\boldsymbol{\theta}}(\mathbf{y}_{r,k}^p), \tag{10}$$

where $\boldsymbol{f}_{\boldsymbol{\theta}} : \mathbb{C}^Q \longmapsto \mathbb{C}^{MN}$ represents the non-linear mapping function with the weights $\boldsymbol{\theta}$. In order to train the DNN, the user need to collect enough training data in advance. The training dataset is defined by $\{\mathbf{y}_{r,k}^{p,d}, \mathbf{h}_{r,k}^d\}_{d=1}^{D_{r,k}}$, where $D_{r,k}$ represents the size of training dataset for the kth user located in rth region. The loss function can be represented by

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{D_{r,k}} \sum_{d=1}^{D_{r,k}} \parallel \hat{\mathbf{h}}_{r,k}^d - \mathbf{h}_{r,k}^d \parallel_2^2, \tag{11}$$

where $\hat{\mathbf{h}}_{r,k}^d = \boldsymbol{f}_{\boldsymbol{\theta}}(\mathbf{y}_{r,k}^{p,d})$ represents the output of the neural network corresponding to the input $\mathbf{y}_{r,k}^{p,d}$, and the label $\mathbf{h}_{r,k}^d$ can be obtained by the conventional LS based channel estimation scheme. The aim of training the DNN is to minimize the above loss function by optimizing the weights $\boldsymbol{\theta}$, i.e.,

$$\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) = \frac{1}{D_{r,k}} \sum_{d=1}^{D_{r,k}} \parallel \boldsymbol{f}_{\boldsymbol{\theta}}(\mathbf{y}_{r,k}^{p,d}) - \mathbf{h}_{r,k}^d \parallel_2^2. \tag{12}$$

After determining the objective function, the DNN is trained on the training dataset by an iterative process. In each iteration t, the weights $\boldsymbol{\theta}$ are updated by the gradient descent, i.e.,

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t \mathbf{g}(\boldsymbol{\theta}_t), \tag{13}$$

where $\boldsymbol{\theta}_t$ and $\boldsymbol{\theta}_{t+1}$ are the weights for the tth iteration and $t+1$th iteration, respectively. $\mathbf{g}(\boldsymbol{\theta}_t)$ is the gradient vector for $\boldsymbol{\theta}_t$, and η_t is the learning rate. After training the DNN, the user can estimate the channel directly based on the trained DNN.

However, the training dataset of each user is limited, which can only cover the part of the channel scenarios of the entire cell. When the user moves from one channel scenario to another, the trained DNN trained by the single user cannot work. In order to solve this problem, we will propose the DML based channel estimation neural network in the next Section III.

III. DML for the Downlink Channel Estimation

In this section, we first leverage the DML technique to enable the reliable downlink channel estimation when the user moves from one channel scenario to another. Then, we further propose a DML based hierarchical neural network architecture to improve the channel estimation accuracy in different channel scenarios. Finally, we analyze the computational complexity of the two proposed DML based channel estimation neural networks.

A. The DML Based Channel Estimation Neural Network

In order to work steadily when the user moves from one channel scenario to another, the neural network should be trained based on sufficient training samples. However, the single user can only obtain the limited training samples. Therefore, we propose to leverage the DML technique for cooperative training between the BS and all users in a cell.

Specifically, the DML technique is a novel training mechanism, where the BS and all users can collaboratively train a global DNN by using the local training dataset collected by the user [23], [24]. By leveraging the DML based training, the trained DNN can have access to different training datasets associated with different channel scenarios in the entire cell. Moreover, this DML technique can protect the privacy of

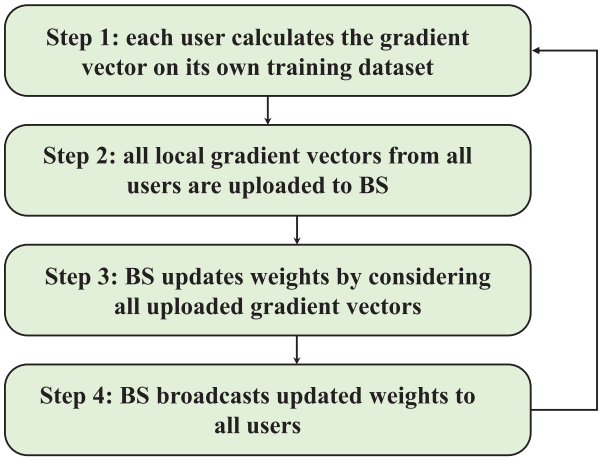


Fig. 2. The DML based training flow.

users, since only the gradient vectors calculated on the local data instead of the local data of the user are required to be uploaded to the BS [25]. There have been some works that employ the DML technique to wireless communications, such as resource allocation [26], power control [27], hybrid beamforming in the massive MIMO system [28] and the RIS beamforming in the RIS assisted communication system [29].

In this subsection, we first build a global DNN called the downlink channel estimation (DCE) network shared by the BS and all users. Then, with the help of DML, the global DCE network is jointly trained based on the local training datasets available at all users. Next, we will first introduce the training processing in detail, and then present the architecture of the DCE network.

1) The DML Based Training: Different from the training process at the single user mentioned in Subsection II-C, the weights of the DCE network are updated in each iteration t by the following four steps for DML, as shown in Fig. 2.

Firstly, each user calculates the gradient vector $\mathbf{g}_{r,k}(\boldsymbol{\theta}_t)$ on its own training dataset $\{\mathbf{y}_{r,k}^{p,d}, \mathbf{h}_{r,k}^{d}\}_{d=1}^{D_{r,k}}$. Secondly, all local gradient vectors $\{\{\mathbf{g}_{r,k}(\boldsymbol{\theta}_t)\}_{k=1}^{C_r}\}_{r=1}^{R}$ from all users are uploaded to the BS. Thirdly, The BS updates the weights by considering all uploaded gradient vectors as

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t \frac{1}{\sum_{r=1}^{R} C_r} \sum_{r=1}^{R} \sum_{k=1}^{C_r} \mathbf{g}_{r,k}(\boldsymbol{\theta}_t). \tag{14}$$

Finally, the BS broadcasts the updated weights $\boldsymbol{\theta}_{t+1}$ to all users for the next iteration.[4]

Based on this DML based training method, all users will obtain the shared trained global DCE network. Compared with the channel estimation neural network trained by the local training dataset of the single user [11], the DML based trained DCE network is expected to learn the entire channel features in the cell. When the user moves from one channel scenario to another, the DCE network is also still able to work steadily.

2) The DCE Network Architecture: The adopted DCE network in this paper consists of three convolution layers and one linear layer.[5]

By considering the received pilot signal $\mathbf{y}_{r,k}^{p}$ and the cascaded channel to be estimated $\mathbf{h}_{r,k}$ are complex-valued, the input and the output of the DCE network are converted into the real-valued by the following processing. $\mathbf{y}_{r,k}^{p}$ is first reshaped as $Q_1 \times Q_2$ 2-dimensional matrix. Then the real part and the imaginary part of this 2-dimensional matrix are extracted as the two input feature maps of the first convolution layer. The output of the DCE network is expected as a vector with $2NM$ elements, where the first NM elements represent the real part of the downlink cascaded channel vector $\mathbf{h}_{r,k}$, while the last NM elements represent its imaginary part.

The first three convolution layers both consist of 32 filters with 3×3 kernel, a batch normalization operation, and a rectified linear unit (ReLU) for activation. The final linear layer consisting of $2NM$ neurons is used to construct the downlink cascaded channel vector.

However, for the entire cell, different channel scenarios have different channel features. The DCE network cannot achieve the accurate channel estimation for different channel scenarios with the single neural network architecture. In order to further improve the channel estimation accuracy, we will propose a hierarchical neural network architecture in the next subsection.

B. The DML Based Hierarchical Channel Estimation Neural Network

In order to achieve better channel estimation performance for different channel scenarios, we propose a DML based hierarchical channel estimation neural network. The basic idea of the hierarchical architecture is that we first determine which channel scenario the channel to be estimated belongs to, then extract different channel features for different channel scenarios, and finally construct the entire cascaded channel. Next, we will first introduce the hierarchical DCE (HDCE) network, and then present the hierarchical training for the proposed HDCE network.

1) The Hierarchical DCE Network Architecture: As shown in Fig. 3, the proposed HDCE network consists of the three parts: the scenario classifier, the feature extractors, the feature mapper.

Firstly, in the scenario classifier, the received pilot signals are input to predict the channel scenario index $\hat{r}$ ($\hat{r} \in \{1, 2, \cdots, R\}$). Then, in the feature extractors, we design a feature extractor for each channel scenario. There are R feature extractors in total. Based on the predicted channel scenario index $\hat{r}$, the received pilot signals are input to the corresponding feature extractor, where the corresponding channel feature can be extracted. For example, if the channel scenario index predicted by the scenario classifier is 2, the received pilot signals will be input to the feature extractor 2, where the output

[4]It is noted that during training the neural network, the BS can communicate with the user via the RIS based on the conventional schemes. After training, the trained neural network is used to replace the original downlink channel estimation module without changing other modules of the entire communication system.

[5]Although the neural network architecture presented in this paper takes the classical CNN architecture as an example, the DML based training idea can be extended to any more advanced neurual network architecture.

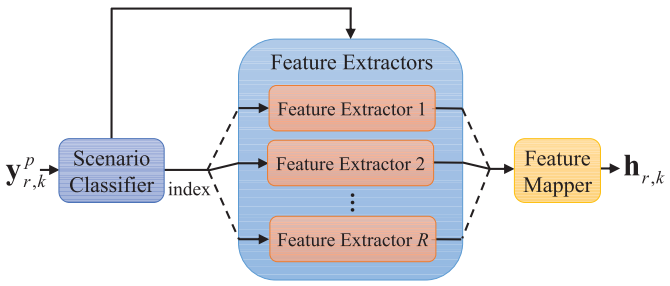


Fig. 3. The proposed hierarchical downlink channel estimation network.

of the feature extractors will be only generated by the feature extractor 2, as shown in Fig. 3. Finally, the extracted channel feature is input to the feature mapper to construct the entire downlink cascaded channel.

In order to obtain the real-valued input and output for the HDCE network, the received pilot signal $\mathbf{y}_{r,k}^{p}$ and the cascaded channel to be estimated $\mathbf{h}_{r,k}$ are also processed in the same way as that of the DCE network. Compared with the DCE network, the additional output, i.e., the channel scenario index $\hat{r}$, will be generated by the scenario classifier of the HDCE network.

The adopted scenario classifier consists of five layers. The first and third layers are both the convolution layers consisting of 32 filters with 3×3 kernel followed by the ReLu activation function. The second and the fourth layers are the max pooling layer with 2×2 kernel. The fifth layer is a linear layer, which maps the output of the previous layer into a vector with R elements. The predicted channel scenario index $\hat{r}$ is determined by the location of the largest element among R elements.

The feature extractor for each user has the same architecture, which consists of the three same convolution layers. Each convolution layer consists of 32 filters with 3×3 kernel, a batch normalization operation, and a ReLu for activation. The feature mapper is a linear layer, which maps the output of the feature extractor as a downlink cascaded channel vector with $2NM$ elements.

2) The DML Based Hierarchical Training: Based on the above hierarchical architecture, we can find that the proposed HDCE network should be trained on different datasets from different channel scenarios, which also needs to leverage the DML technique. We further propose the DML based hierarchical training to train the proposed HDCE network. Specifically, the HDCE network is not trained as a whole, where the scenario classifier is trained separately, while the feature extractors and the feature mapper are trained jointly. The two training processes are decoupled.

For training the scenario classifier, the users should obtain the corresponding channel scenario index label r in advance. In this paper, the different channel scenarios are divided based on the angles of the downlink channel, of which the channel scenario index can be obtained by the conventional angle estimation algorithms. We adopt the cross entropy between the output of the scenario classifier and the channel scenario index label as the loss function to train the scenario classifier. Similar to the DML based training of the DCE network, the gradient vectors of weights of the scenario classifier are transmitted among the BS and the users for the update of the weights.

When training the feature extractors and the feature mapper, we assumed that the scenario classifier can achieve the perfect channel scenario index prediction. Therefore, the two training processes can be decoupled, which can avoid converging to a bad local optimization. It is noted that although there are R feature extractors in the HDCE network, each user only uses one feature extractor to extract the corresponding channel feature. Therefore, the gradients of weights of other $R-1$ feature extractors for each user are zero.

Based on this DML based hierarchical training method, besides the received pilot signals and the downlink cascaded channel labels, the prior information of the channel scenario index can also be used to train the neural network. Thanks to this prior information, the trained HDCE network can perform better than the DCE network in different channel scenarios.

C. Computational Complexity Analysis

In this subsection, we provide a comparison of the computational complexity of the two proposed downlink channel estimation neural networks.

For a convolution layer, its computational complexity can be represented by $\mathcal{O}(E_1 E_2 F^2 C_{in} C_{out})$ [30], where E_1 and E_2 denote the number of rows and columns of each output feature map, F denotes the side length of the used filter, C_{in} and C_{out} denote the numbers of input and output feature maps, respectively. For a linear layer, its computational complexity is $\mathcal{O}(L_1 L_2)$, where L_1 and L_2 denote the dimension of input and output, respectively.

For the DCE network, the computational complexity of the first convolution layer is $\mathcal{O}(Q_1 Q_2 \cdot 3^2 \cdot 2 \cdot 32)$, and the computational complexity of the second and the third convolution layer is the same, i.e., $\mathcal{O}(Q_1 Q_2 \cdot 3^2 \cdot 32 \cdot 32)$. The computational complexity of the final linear layer is $\mathcal{O}(32 Q_1 Q_2 \cdot 2NM)$. Therefore, the total computational complexity of the DCE network is $\mathcal{O}(19008 Q_1 Q_2 + 64 Q_1 Q_2 NM)$. Although the HDCE network consists of R feature extractors, each user only uses one for extracting its channel feature. We can find the sum of the computation complexity of one feature extractor and the feature mapper is just equal to the computational complexity of the DCE network. Therefore, compared with the DCE network, the HDCE network only adds the computational complexity of the scenario classifier. The computational complexity of the scenario classifier is $\mathcal{O}(Q_1 Q_2 \cdot 3^2 \cdot 2 \cdot 32 + \frac{1}{2} Q_1 \frac{1}{2} Q_2 \cdot 3^2 \cdot 32 \cdot 32 + \frac{1}{4} Q_1 \frac{1}{4} Q_2 \cdot 32 \cdot 3)$, i.e., $\mathcal{O}(2886 Q_1 Q_2)$, which is small compared with that of the feature extractor and the feature mapper. Since the training process of the neural networks is offline, the above analysis only considers the computational complexity of the proposed neural networks during the online test process.

From the above analysis, we can find that the HDCE network further considers the channel scenario classification on the basis of the DCE network, which is expected to achieve better performance with a low computational complexity

increment. This hierarchical architecture can be extended to classify other factors except for the channel scenario, such as the SNR level. By considering the additional prior information about the scenario labels, the neural network can be better adapted to a variety of different scenarios.

IV. Simulation Results

In this section, the simulation results are provided to verify the effectiveness of the two proposed DML based channel estimation neural networks, i.e., the DCE network and the HDCE network.

In our simulation, we consider that the number of antennas at the BS and the number of the RIS elements are $M = 16$ (4×4) and $N = 64$ (8×8), respectively. Each element of $\mathbf{\Psi}_{r,k}$ in (9) is selected from $\{-1/\sqrt{Q}, +1/\sqrt{Q}\}$ by considering discrete phase shifts of the BS and the RIS, and $\mathbf{\Psi}_{r,k}$ is assumed to be same for $\forall k$ and $\forall r$. The signal to noise ratio (SNR) is defined as $1/\sigma_n^2$. The entire cell is divided into $R = 3$ regions (i.e., 3 channel scenarios), where each region contains $C_r = 3$ users for $r = 1, 2, 3$. The channel $\mathbf{G}$ between the BS and the RIS is generated according to (2), where the number of paths between the BS and the RIS is $L_G = 3$, the complex gains are generated by following the complex Gaussian distribution with the mean 0 and the variance 1, and the angles are generated by following the uniform distribution on $(-\pi/2, \pi/2)$. The channel $\mathbf{f}_{r,k}$ from the RIS and the kth user located in the rth region is generated according to the Saleh-Valenzuela channel model in (3) and the publicly-available DeepMIMO dataset based on ray-tracing, respectively. The operating frequency is considered as 28 GHz.

For the Saleh-Valenzuela channel model in (3), the number of paths between the RIS and the user is set as $L_{r,k} = 3$ for $\forall k$ and $\forall r$, the complex gains of $\mathbf{f}_{r,k}$ are generated by following the complex Gaussian distribution with the mean 0 and the variance 1. The angles of $\mathbf{f}_{r,k}$ are generated by following the uniform distribution on $(-\pi/2, \pi/2)$, where the elevation angles are divided into three equal parts, i.e., $(-\pi/2, -\pi/6)$, $(-\pi/6, \pi/6)$ and $(\pi/6, \pi/2)$. Each part is corresponding to each region, that is also each channel scenario.

For the DeepMIMO dataset, we generate the channels $\mathbf{f}_{r,k}$ from the RIS to the user by referring to [31]. Specifically, the DeepMIMO dataset is a parameterized dataset, where the channel samples can be generated based on the ray-tracing. It can capture the dependence on key environmental factors such as the environment geometry, operating frequency and so on [31]. In this paper, we consider the outdoor ray-tracing scenario 'O1' working at 28 GHz, where the adopted parameters are shown in Table I. We select BS 3 in the 'O1' scenario to be the RIS. The users belonging to channel scenario 1 are assumed to be distributed in rows R401-R800, the users belonging to channel scenario 2 are assumed to be distributed in rows R801-R1200, and the users belong to channel scenario 3 are assumed to be distributed in rows R1201-R1600. It is noted that each obtained channel sample based on DeepMIMO dataset will be normalized to $||\mathbf{f}_{r,k}||_2^2 = N$.

We assume that the single user can collect 20000 samples, 90% of which are used as the training dataset while 10% of which are used as the validation dataset. It is noted that there are 180000 samples in total for the DML based training, since all samples from all users can be effectively used to train the neural network. In each iteration, each user uses 256 local training samples to calculate the gradient vector of the weights. The Adam optimizer is used to update the weights. The training process consists of 100 epochs in total. For the DML based training, after 100 epochs, the latest trained network is used to test the performance. The learning rate is initialized to $1\mathrm{e}-3$, and then is reduced to the half of the original level every 30 epochs. It is noted that, for each user, the LS algorithm is used to obtain the downlink cascaded channel label with SNR 10 dB. After training the neural network, we use 10000 samples to test the performance of the trained networks.

TABLE I
The Adopted DeepMIMO Dataset Parameters

Parameter	Value
Frequency band	28 GHz
Active BSs (RIS)	3
Number of BS antennas (RIS elements)	$(N_x, N_y, N_z) = (1, 8, 8)$
Active users (belongs to scenario 1)	From row R401 to row R800
Active users (belongs to scenario 2)	From row R801 to row R1200
Active users (belongs to scenario 3)	From row R1201 to row R1600
Antenna spacing	0.5
System bandwidth	100 MHz
Number of OFDM subcarriers	512
OFDM sampling factor	1
OFDM limit	1
Number of channel paths	3

We compare the six downlink channel estimation schemes. In the first scheme, we consider the conventional training method. The DCE network is deployed at the user, which is only trained by the training dataset of the single user. In the second scheme, the DCE network is deployed at the BS and all users, where the DML technique is used for training this network, as described in Subsection III-A. In the third scheme, the proposed HDCE network is trained by the DML based hierarchical training method, as described in Subsection III-B. For the fourth scheme, we consider the specialized DCE (SDCE) network to show the advantage of the proposed HDCE network. The only difference between the two neural networks is that the feature mapper of the SDCE network, like the feature extractor, is also specialized for each channel scenario. For the above four schemes, the required pilot overhead is set as $Q = \frac{NM}{8} = 128$ ($Q_1 = 16$, $Q_2 = 8$). Finally, the classical LS algorithm and MMSE algorithm are considered to estimate the downlink cascaded channel with

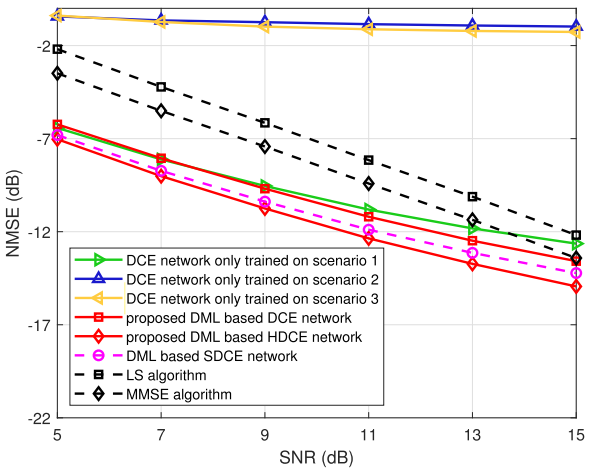


Fig. 4. NMSE performance comparison for the channel scenario 1 on the Saleh-Valenzuela channel model.

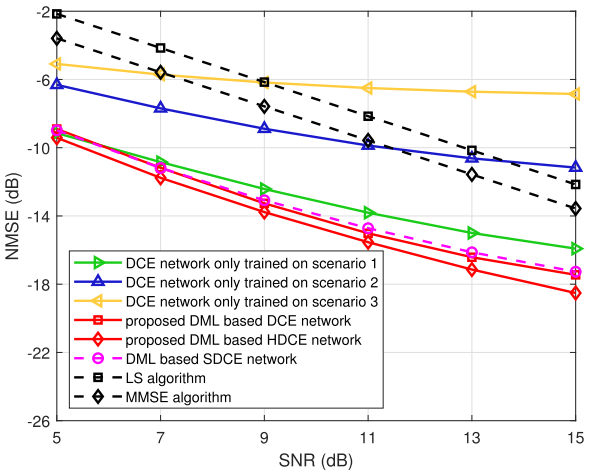


Fig. 6. NMSE performance comparison for the channel scenario 1 on the DeepMIMO dataset.

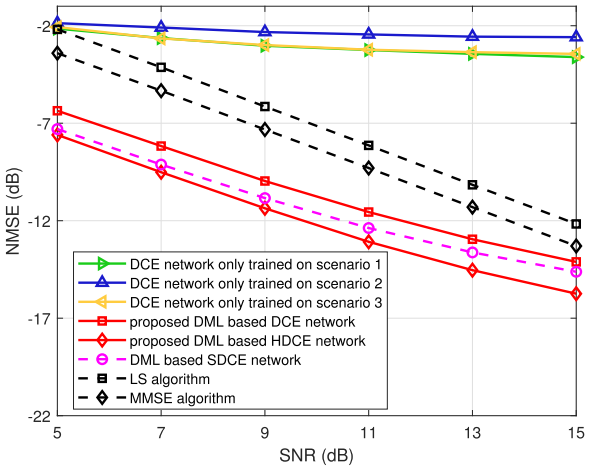


Fig. 5. NMSE performance comparison for the entire cell on the Saleh-Valenzuela channel model.

the required pilot overhead $Q = NM = 1024$. The prior channel autocorrelation matrix for the MMSE algorithm is obtained based on the LS estimated channels. It is noted that based on the above settings, both the DCE network and the HDCE network should occupy about 32 MB memory while need about 1.1×10^7 multiplications. The simulation results on the Saleh-Valenzuela channel model and the DeepMIMO dataset are respectively given as follows.

Fig. 4 and Fig. 5 show the normalized mean square error (NMSE) performance comparison of six different schemes mentioned above against different SNRs on the Saleh-Valenzuela channel model. In Fig. 4, all test samples are only from the channel scenario 1. It is noted that, for the first scheme, we provide the three DCE networks, which are trained by the single user from the channel scenario 1, the channel scenario 2, and the channel scenario 3, respectively. From Fig. 4, we can find that based on the conventional training method, the DCE networks trained on the channel scenario 2 and the channel scenario 3 are not able to work for the channel scenario 1. By contrast, the two proposed DML based trained networks can work well, which even also outperform the DCE network trained on the channel scenario 1. This is because the DML based neural networks can be trained with more training data from all users. Specifically, for the DCE network trained on the channel scenario 1, the training dataset can only be collected by one user, and thus the number of samples is small, which is only 20000 in our simulations. For the DML based trained networks, they can be trained by different training datasets from different users by leveraging the DML technique. The number of samples from scenario 1 can be regarded as 60000 in our simulations.

In Fig. 5, the test samples are randomly generated for all channel scenarios of the entire cell. From Fig. 5, we can find that although the required pilot overhead is reduced, the conventional training based schemes cannot achieve the reliable channel estimation in the entire cell. The LS based and MMSE based channel estimation schemes can achieve good estimation accuracy in the high SNR range, but requires a huge pilot overhead. The two proposed DML based schemes can achieve better channel estimation performance with the reduced pilot overhead. It is noted that the proposed HDCE network can outperform the SDCE network. This is because that the feature mapper of the SDCE network can only be trained with the less training data from the corresponding one channel scenario, while the feature mapper of the HDCE network can be trained with all training data from all channel scenarios.

As shown in Fig. 6 and Fig. 7, we further provide the NMSE performance comparison on the DeepMIMO dataset. In Fig. 6, all test samples are from the channel scenario 1 (i.e., from row R401 to R800). In Fig. 7, the test samples are randomly generated for the three considered channel scenarios (i.e., from row R401 to R1600). We can find since the channel scenario 2 is somewhat similar to the channel scenario 1 and the channel scenario 3 for the DeepMIMO dataset, the DCE network trained on the channel scenario 2 can outperform than those trained on the channel scenario 1 and the channel scenario 3 in Fig. 7. But on the whole, the proposed DML based networks can always achieve better channel estimation accuracy on the DeepMIMO dataset.

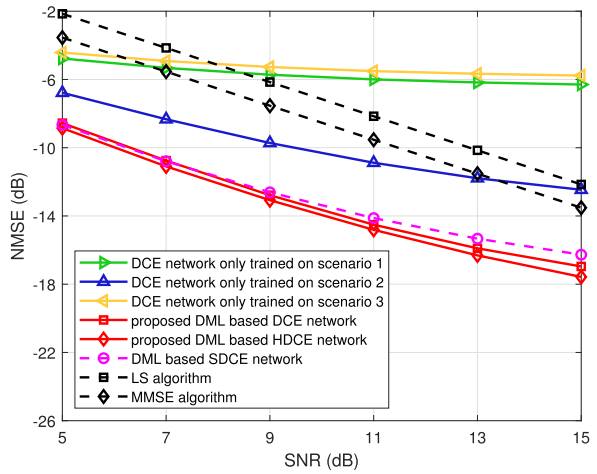


Fig. 7. NMSE performance comparison for the entire cell on the DeepMIMO dataset.

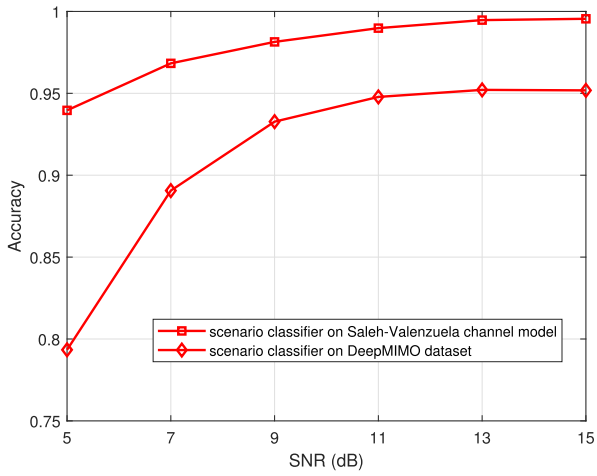


Fig. 8. The accuracy of the channel scenario prediction.

Finally, we provide the accuracy of the channel scenario prediction against different SNRs for the scenario classifier of the HDCE network, as shown in Fig. 8. We can find for the Saleh-Valenzuela channel model, the adopted scenario classifier can always achieve more than 93% accuracy, which ensures good performance of the proposed HDCE network. For the DeepMIMO dataset, the accuracy is slightly reduced, since the considered channel scenario 2 is somewhat similar to the channel scenario 1 and the channel scenario 3. Fortunately, the scenario classifier can still achieve 95% accuracy in the high SNR range.

V. Conclusion

In this paper, we proposed to leverage the DML technique to enable the reliable downlink cascaded channel estimation. Specifically, we first built a global DCE network shared by all users, which can be collaboratively trained by the BS and all users with the help of DML. Then, we further proposed a hierarchical neural network architecture to improve the estimation accuracy in the different channel scenarios. In order to verify the effectiveness of our work, we have considered the widely used channel model for theoretical analysis, and the publicly-available channel dataset based on ray-tracing in our simulations. Simulation results showed that compared with the conventional training method, the two proposed DML based neural networks can work well when the user moves from one channel scenario to another. Compared with the conventional algorithms, the proposed neural networks, particularly the hierarchical neural network, can achieve better estimation accuracy when the pilot overhead is reduced to $1/8$. For future work, we will design a light-weight hierarchical neural network for reducing the transmission overhead for DML.

References

[1] E. Basar, M. Di Renzo, J. De Rosny, M. Debbah, M. Alouini, and R. Zhang, "Wireless communications through reconfigurable intelligent surfaces," *IEEE Access*, vol. 7, pp. 116753–116773, 2019.

[2] L. Dai *et al.*, "Reconfigurable intelligent surface-based wireless communications: Antenna design, prototyping, and experimental results," *IEEE Access*, vol. 8, pp. 45913–45923, 2020.

[3] Q. Wu and R. Zhang, "Intelligent reflecting surface enhanced wireless network via joint active and passive beamforming," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5394–5409, Nov. 2019.

[4] M. Di Renzo *et al.*, "Smart radio environments empowered by reconfigurable intelligent surfaces: How it works, state of research, and the road ahead," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2450–2525, Jul. 2020.

[5] B. Ning, Z. Chen, W. Chen, Y. Du, and J. Fang, "Terahertz multi-user massive MIMO with intelligent reflecting surface: Beam training and hybrid beamforming," *IEEE Trans. Veh. Technol.*, vol. 70, no. 2, pp. 1376–1393, Feb. 2021.

[6] C. Huang, A. Zappone, G. C. Alexandropoulos, M. Debbah, and C. Yuen, "Reconfigurable intelligent surfaces for energy efficiency in wireless communication," *IEEE Trans. Wireless Commun.*, vol. 18, no. 8, pp. 4157–4170, Jun. 2019.

[7] D. Mishra and H. Johansson, "Channel estimation and low-complexity beamforming design for passive intelligent surface assisted MISO wireless energy transfer," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Brighton, U.K., May 2019, pp. 4659–4663.

[8] Q.-U.-A. Nadeem, H. Alwazani, A. Kammoun, A. Chaaban, M. Debbah, and M.-S. Alouini, "Intelligent reflecting surface-assisted multi-user MISO communication: Channel estimation and beamforming design," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 661–680, 2020.

[9] Z. Wang, L. Liu, and S. Cui, "Channel estimation for intelligent reflecting surface assisted multiuser communications: Framework, algorithms, and analysis," *IEEE Trans. Wireless Commun.*, vol. 19, no. 10, pp. 6607–6620, Oct. 2020.

[10] X. Wei, D. Shen, and L. Dai, "Channel estimation for RIS assisted wireless communications—Part II: An improved solution based on double-structured sparsity," *IEEE Commun. Lett.*, vol. 25, no. 5, pp. 1403–1407, May 2021.

[11] M. Xu, S. Zhang, C. Zhong, J. Ma, and O. A. Dobre, "Ordinary differential equation-based CNN for channel extrapolation over RIS-assisted communication," 2020, *arXiv:2012.11794*.

[12] S. Liu, Z. Gao, J. Zhang, M. D. Renzo, and M.-S. Alouini, "Deep denoising neural network assisted compressive channel estimation for mmWave intelligent reflecting surfaces," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 9223–9228, Aug. 2020.

[13] W. Chen, C.-K. Wen, X. Li, and S. Jin, "Adaptive bit partitioning for reconfigurable intelligent surface assisted FDD systems with limited feedback," *IEEE Trans. Wireless Commun.*, vol. 21, no. 4, pp. 2488–2505, Sep. 2021.

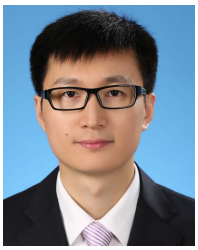
[14] B. Guo, C. Sun, and M. Tao, "Two-way passive beamforming design for RIS-aided FDD communication systems," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, May 2021, pp. 1–6.

[15] W. Tang *et al.*, "On channel reciprocity in reconfigurable intelligent surface assisted wireless network," 2021, *arXiv:2103.03753*.

[16] P. Wang, J. Fang, H. Duan, and H. Li, "Compressed channel estimation for intelligent reflecting surface-assisted millimeter wave systems," *IEEE Signal Process. Lett.*, vol. 27, pp. 905–909, 2020.

[17] M. F. Imani, D. R. Smith, and P. del Hougne, "Perfect absorption in a metasurface-programmable complex scattering enclosure," 2020, *arXiv:2003.01766*.

[18] X. Wei, D. Shen, and L. Dai, "Channel estimation for RIS assisted wireless communications—Part I: Fundamentals, solutions, and future opportunities," *IEEE Commun. Lett.*, vol. 25, no. 5, pp. 1398–1402, May 2021.
[19] C. Hu, L. Dai, T. Mir, Z. Gao, and J. Fang, "Super-resolution channel estimation for mmWave massive MIMO with hybrid precoding," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8954–8958, Sep. 2018.
[20] C.-K. Wen, W.-T. Shih, and S. Jin, "Deep learning for massive MIMO CSI feedback," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 748–751, Oct. 2018.
[21] 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. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.
[22] H. Ye, L. Liang, G. Y. Li, and B.-H. F. Juang, "Deep learning-based end-to-end wireless communication systems with conditional GANs as unknown channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3133–3143, May 2020.
[23] S. Niknam, H. S. Dhillon, and J. H. Reed, "Federated learning for wireless communications: Motivation, opportunities, and challenges," *IEEE Commun. Mag.*, vol. 58, no. 6, pp. 46–51, Jul. 2020.
[24] M. M. Amiri and D. Gündüz, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," *IEEE Trans. Signal Process.*, vol. 68, pp. 2155–2169, 2020.
[25] Y. Liu, X. Yuan, Z. Xiong, J. Kang, X. Wang, and D. Niyato, "Federated learning for 6G communications: Challenges, methods, and future directions," *China Commun.*, vol. 17, no. 9, pp. 105–118, Sep. 2020.
[26] M. Yan, B. Chen, G. Feng, and S. Qin, "Federated cooperation and augmentation for power allocation in decentralized wireless networks," *IEEE Access*, vol. 8, pp. 48088–48100, 2020.
[27] T. T. Vu, D. T. Ngo, N. H. Tran, H. Q. Ngo, M. N. Dao, and R. H. Middleton, "Cell-free massive MIMO for wireless federated learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 10, pp. 6377–6392, Oct. 2020.
[28] A. M. Elbir and S. Coleri, "Federated learning for hybrid beamforming in mm-wave massive MIMO," *IEEE Commun. Lett.*, vol. 24, no. 12, pp. 2795–2799, Aug. 2020.
[29] D. Ma, L. Li, H. Ren, D. Wang, X. Li, and Z. Han, "Distributed rate optimization for intelligent reflecting surface with federated learning," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Dublin, Ireland, Jun. 2020, pp. 1–6.
[30] P. Dong, H. Zhang, G. Y. Li, I. Gaspar, and N. NaderiAlizadeh, "Deep CNN-based channel estimation for mmWave massive MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 5, pp. 989–1000, Jul. 2019.
[31] A. Alkhateeb, "DeepMIMO: A generic deep learning dataset for millimeter wave and massive MIMO applications," in *Proc. Inf. Theory Appl. Workshop (ITA)*, San Diego, CA, USA, Feb. 2019, pp. 1–8.



Linglong Dai (Fellow, IEEE) received the B.S. degree from Zhejiang University, Hangzhou, China, in 2003, the M.S. degree (Hons.) from the China Academy of Telecommunications Technology, Beijing, China, in 2006, and the Ph.D. degree (Hons.) from Tsinghua University, Beijing, in 2011.

From 2011 to 2013, he was a Post-Doctoral Research Fellow with the Department of Electronic Engineering, Tsinghua University, where he was an Assistant Professor from 2013 to 2016 and has been an Associate Professor since 2016. He has coauthored the book *MmWave Massive MIMO: A Paradigm for 5G* (Academic Press, 2016). He has authored or coauthored over 70 IEEE journal articles and over 40 IEEE conference papers. He also holds 19 granted patents. His current research interests include reconfigurable intelligent surface (RIS), massive MIMO, millimeter-wave and terahertz communications, electromagnetic information theory, and machine learning for wireless communications. He has received the five IEEE Best Paper Awards at the IEEE ICC 2013, the IEEE ICC 2014, the IEEE ICC 2017, the IEEE VTC 2017-Fall, and the IEEE ICC 2018. He has also received the Tsinghua University Outstanding Ph.D. Graduate Award in 2011, the Beijing Excellent Doctoral Dissertation Award in 2012, the China National Excellent Doctoral Dissertation Nomination Award in 2013, the URSI Young Scientist Award in 2014, the IEEE TRANSACTIONS ON BROADCASTING Best Paper Award in 2015, the *Electronics Letters* Best Paper Award in 2016, the National Natural Science Foundation of China for Outstanding Young Scholars in 2017, the IEEE ComSoc Asia–Pacific Outstanding Young Researcher Award in 2017, the IEEE ComSoc Asia–Pacific Outstanding Paper Award in 2018, the *China Communications* Best Paper Award in 2019, the IEEE ACCESS Best Multimedia Award in 2020, the IEEE Communications Society Leonard G. Abraham Prize in 2020, and the IEEE ComSoc Stephen O. Rice Prize in 2022. He was listed as a Highly Cited Researcher by Clarivate Analytics in 2020 and 2021.

Xiuhong Wei received the B.E. degree from the School of Mechanical, Electrical and Information Engineering, Shandong University, Weihai, China, in 2019. She is currently pursuing the M.S. degree in electronic engineering with Tsinghua University, Beijing, China. Her research interests are massive MIMO, AI-based wireless communication, and RIS assisted wireless communications. She has received the National Scholarship for Postgraduates in 2021.