Learning-Based Fronthaul Compression for Uplink Cloud Radio Access Networks

Ruihua Qiao, Tao Jiang, and Wei Yu
Department of Electrical and Computer Engineering
University of Toronto, Toronto, ON, M5S 3G4, Canada
Emails: {ruihua.qiao,taoca.jiang}@mail.utoronto.ca, weiyu@ece.utoronto.ca

***Abstract*—This paper investigates the uplink signal dimension reduction problem for a user-centric cloud radio access network, in which each single-antenna user communicates with the central processor (CP) through a cluster of remote radio heads (RRHs). To reduce the fronthaul traffic, each RRH applies a compression matrix to reduce the dimension of the received signal before relaying it to the CP. However, the optimal design of the compression matrices requires significant communication overhead for transmitting the high-dimensional channel state information (CSI) matrices from the RRHs to the CP. To address this issue, this paper proposes a deep learning framework to first learn a sub-optimal compression matrix at each RRH based on the local CSI, then iteratively refine the learned compression matrix using a meta-learning-based gradient method. To reduce the communication cost for CSI sharing and gradients transmission, this paper proposes an efficient signaling scheme that only requires the transmission of low-dimensional effective CSI and its gradient between the CP and each RRH. Furthermore, a meta-learning-based gated recurrent unit (GRU) network is proposed to reduce the number of signaling transmission rounds. For the sum-rate maximization problem, simulation results show that the proposed two-stage neural network can perform closely to the fully cooperative global CSI-based benchmark with significantly reduced communication overhead. Moreover, using the first stage alone can already outperform the existing local CSI-based benchmark.**

I. Introduction

Cloud radio access network (C-RAN) [1], [2], also known as cell-free multiple-input multiple-output (MIMO) [3] network, is envisioned as a key building block for future wireless networks. In an uplink C-RAN system, the user signals received by distributed remote radio heads (RRHs) are jointly estimated and decoded in the central processor (CP), thereby effectively addressing the issue of inter-cell interference [4]. However, due to the limited fronthaul capacity, it is necessary for each RRH to compress the received high-dimensional signal vectors before forwarding them to the CP. This is typically done by applying a dimension reducing compression matrix followed by a uniform quantizer [5]. In this paper, we focus on the design of dimension reducing compression matrices at RRHs since the uniform quantization step can be readily incorporated into the proposed scheme afterwards. This is a challenging problem because the optimal compression matrices need to be designed jointly at the CP using global channel state information (CSI), leading to significant communication overhead. To address this issue, this paper shows that by carefully designing a gated recurrent unit (GRU)-based meta-learning framework, we are able to find near-optimal compression matrices with very small communication overhead.

Assuming that the global CSI is available at the CP, the centralized design of the compression matrices is already a highly nontrivial problem. In [6], a heuristic channel selection-based matched filtering scheme is proposed to maximize the joint mutual information between all compressed signals and original user signals. Further, block coordinate descent (BCD)-based algorithms are developed in [7] and [8] to optimize the estimation mean squared error (MSE) and joint mutual information, respectively. In both of these works, the closed-form optimal compression matrix at one RRH are derived assuming others are fixed. However, in all these works, the transmission of full CSI from the RRHs to the CP and the designed compression matrices from the CP to the RRHs can lead to significant communication overhead.

To reduce the communication overhead for CSI transmission, local CSI-based methods are proposed in [5], [9]–[11], where compression matrices are designed at each RRH individually using the available local CSI. Specifically, [9]–[11] propose to design each compression matrix using the largest eigenvectors of the covariance matrix of the received signal at each RRH. However, this eigenvalue decomposition (EVD)-based approach can be highly sub-optimal since it minimizes the local reconstruction error at each RRH without taking into account the optimization of the end-to-end system objective. To address this problem, [5] proposes a data-driven approach to optimize the end-to-end MSE based on local CSI. However, although this deep learning approach outperforms the EVD-based method, there is still a gap to the global CSI-based benchmark.

This paper aims to design near-optimal compression matrices with as small communication overhead as possible. Toward this end, we propose a novel two-stage deep learning framework, where in the first stage, compression matrices are derived from the local CSI at the RRHs using fully connected deep neural networks (DNNs), and in the second stage, the designed compression matrices are further refined iteratively using the gradient of the system objective with respect to beamforming matrices. To reduce the signaling dimension for gradient transmission in each iteration, we propose an efficient signaling strategy, which allows the gradient at each RRH to be calculated from a low-dimensional signal from the CP. Further, to reduce the overall refinement rounds, we propose a meta-learning-based GRU network that can be trained to design an efficient refinement update based on historical and current gradients. Simulation results show that the trained neural network with only the first stage can already outperform the existing local CSI-based benchmark without introducing additional communication costs. Moreover, when a few iterations are allowed in the second stage, the proposed network can achieve almost the same performance as the global CSI-based benchmark but with significant reduction in communication overhead.

II. System Model and Problem Formulation

A. System Model

Consider a C-RAN system where N single antenna users communicate with the CP in the uplink through B spatially distributed RRHs, each equipped with M antennas. As shown in Fig. 1, the b-th RRH receives signals from all users as:

$$\boldsymbol{y}_b = \mathbf{H}_b \boldsymbol{x} + \boldsymbol{z}_b \ , \tag{1}$$

where $\mathbf{H}_b = [\boldsymbol{h}_{b1} \cdots \boldsymbol{h}_{bN}] \in \mathbb{C}^{M \times N}$ is the channel matrix between RRH b and all users, $\boldsymbol{x} \sim \mathcal{CN}(\mathbf{0}, P_x \mathbf{I})$ is the transmitted signal from all users, and $\boldsymbol{z}_b \sim \mathcal{CN}\left(\mathbf{0}, \sigma_z^2 \mathbf{I}\right)$ is the additive white Gaussian noise. We consider a quasi-static block-fading channel model, where $\mathbf{H}_b$ remains constant in each coherence block. We assume that perfect local CSI $\mathbf{H}_b$ is available at RRH b for each coherence block.

To reduce the fronthaul traffic load, each RRH compresses the received user signal vector by reducing its dimension before forwarding it to the CP. The received signal vector after compression for RRH b is given by:

$$\boldsymbol{v}_b = \mathbf{W}_b \mathbf{H}_b \boldsymbol{x} + \mathbf{W}_b \boldsymbol{z}_b \ , \tag{2}$$

where $\mathbf{W}_b \in \mathbb{C}^{K \times M}$ is a full rank dimension reducing matrix at RRH b. We denote the effective channel matrix after compression for RRH b as:

$$\mathbf{F}_b = \mathbf{W}_b \mathbf{H}_b \ , \tag{3}$$

where the i-th column $\boldsymbol{f}_{bi} \in \mathbb{C}^K$ is the effective channel vector from user i to RRH b.

At the CP, user signals are recovered based on the compressed signal vectors from the RRHs. Because estimating each user's signal using compressed vectors from all the RRHs is impractical for a large network, a clustering scheme is needed to limit the computational complexity in the CP. We adopt a user-centric clustering scheme [12], where each user is at the center of its serving RRH cluster, and clusters for different users may overlap. Specifically, each user is associated with its strongest RRHs. We use Θ_n to denote the serving cluster of RRHs for user n, and use $|\Theta_n|$ to denote the cluster size. Then, the collective signal model for the RRH cluster Θ_n can be written as:

$$\bar{\boldsymbol{v}}_n = \bar{\mathbf{F}}_n \boldsymbol{x} + \bar{\mathbf{W}}_n \bar{\boldsymbol{z}}_n \ , \tag{4}$$

where $\bar{\boldsymbol{v}}_n = \left[\cdots \boldsymbol{v}_b^{\mathsf{H}} \cdots\right]_{b \in \Theta_n}^{\mathsf{H}} \in \mathbb{C}^{|\Theta_n| K}$ is the collective received signal vector, $\bar{\mathbf{F}}_n = \left[\cdots \mathbf{F}_b^{\mathsf{H}} \cdots\right]_{b \in \Theta_n}^{\mathsf{H}} \in \mathbb{C}^{|\Theta_n| K \times N}$ is the collective effective channel matrix after compression, $\bar{\mathbf{W}}_n \in \mathbb{C}^{|\Theta_n| K \times |\Theta_n| M}$ is the collective block-diagonal compression matrix whose block-diagonal entries are given by $\mathbf{W}_b$, for $b \in \Theta_n$, and $\bar{\boldsymbol{z}}_n = \left[\cdots \boldsymbol{z}_b^{\mathsf{H}} \cdots\right]_{b \in \Theta_n}^{\mathsf{H}} \in \mathbb{C}^{|\Theta_n| M}$ is the collective noise vector. Further, we use $\boldsymbol{h}_i^{(n)} = \left[\cdots \boldsymbol{h}_{bi}^{\mathsf{H}} \cdots\right]_{b \in \Theta_n}^{\mathsf{H}} \in \mathbb{C}^{|\Theta_n| M}$ and $\boldsymbol{f}_i^{(n)} = \bar{\mathbf{W}}_n \boldsymbol{h}_i^{(n)} \in \mathbb{C}^{|\Theta_n| K}$ to denote the original and effective channel vector from user i to the RRHs in Θ_n, respectively.

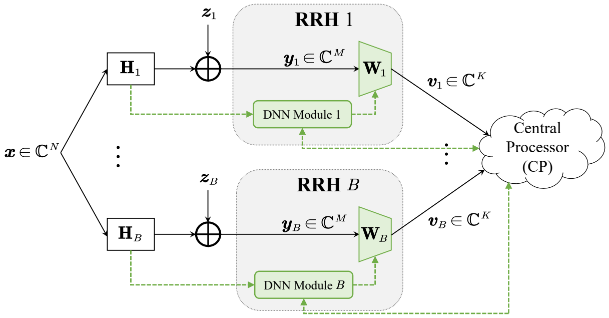

$\boldsymbol{z}_1$
RRH 1
$\mathbf{H}_1$
$\boldsymbol{y}_1 \in \mathbb{C}^M$
$\mathbf{W}_1$
$\boldsymbol{v}_1 \in \mathbb{C}^K$
DNN Module 1
$\boldsymbol{x} \in \mathbb{C}^N$
⋮
Central Processor (CP)
$\boldsymbol{z}_B$
RRH B
$\mathbf{H}_B$
$\boldsymbol{y}_B \in \mathbb{C}^M$
$\mathbf{W}_B$
$\boldsymbol{v}_B \in \mathbb{C}^K$
DNN Module B

Fig. 1. Uplink C-RAN system model. The compression matrices $\mathbf{W}_b$ at RRH b is designed using the proposed two-stage DNN.

The achievable rate of user n can be written as:

$$R_n = \log\left(1 + \frac{P_x \left|\boldsymbol{c}_n^{\mathsf{H}} \boldsymbol{f}_n^{(n)}\right|^2}{P_x \sum_{i \neq n} \left|\boldsymbol{c}_n^{\mathsf{H}} \boldsymbol{f}_i^{(n)}\right|^2 + \sigma_z^2 \left\|\bar{\mathbf{W}}_n^{\mathsf{H}} \boldsymbol{c}_n\right\|^2}\right) \ , \tag{5}$$

where $\boldsymbol{c}_n$ is the linear receive beamformer for user n in the CP. Without loss of generality, we constrain $\mathbf{W}_b$ to be a semi-orthogonal matrix, i.e., $\mathbf{W}_b \mathbf{W}_b^{\mathsf{H}} = \mathbf{I}$. This constraint can be satisfied by taking QR decomposition $\tilde{\mathbf{W}}_b^{\mathsf{H}} = \mathbf{QR}$, where $\mathbf{Q}$ is a $M \times K$ matrix with orthonormal columns and $\mathbf{R}$ is a $K \times K$ upper triangular matrix. We can then set the normalized $\mathbf{W}_b$ as $\mathbf{Q}^{\mathsf{H}}$. This operation does not change the achievable rate R_n since the triangular matrix $\mathbf{R}$ is invertable.

With the semi-orthogonal $\mathbf{W}_b$ as chosen above, the achievable rate in (5) can now be rewritten as:

$$R_n = \log\left(1 + \frac{P_x \left|\boldsymbol{c}_n^{\mathsf{H}} \boldsymbol{f}_n^{(n)}\right|^2}{P_x \sum_{i \neq n} \left|\boldsymbol{c}_n^{\mathsf{H}} \boldsymbol{f}_i^{(n)}\right|^2 + \sigma_z^2 \left\|\boldsymbol{c}_n\right\|^2}\right) . \tag{6}$$

We use the linear minimum mean squared error (LMMSE) estimator to design the receive beamformer $\boldsymbol{c}_n$ as [13]:

$$\boldsymbol{c}_n = \left(\bar{\mathbf{F}}_n \bar{\mathbf{F}}_n^{\mathsf{H}} + \sigma_z^2 / P_x \mathbf{I}\right)^{-1} \boldsymbol{f}_n^{(n)} \ . \tag{7}$$

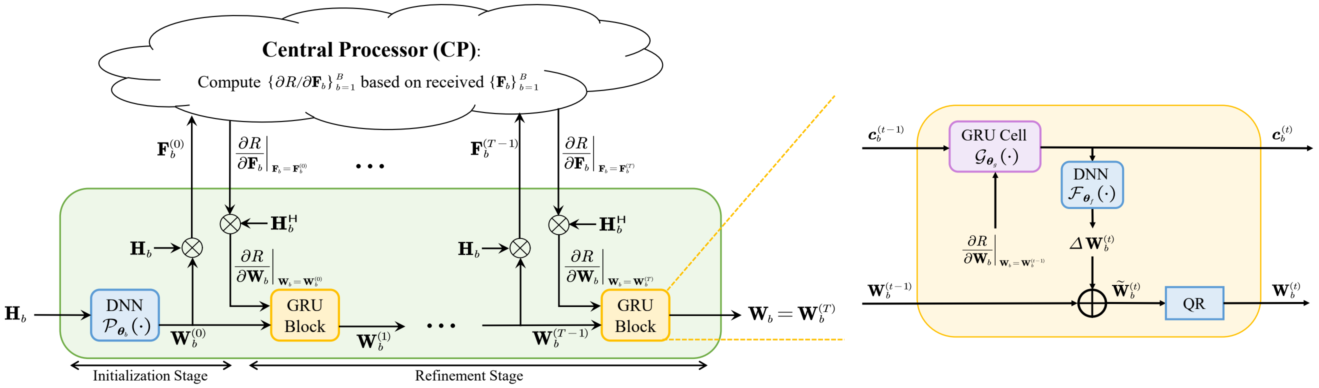
B. Problem Formulation

With the above system model in place, the sum rate maximization problem can now be formulated as:

$$\underset{\{\mathcal{F}_b(\cdot)\}_{b=1}^B}{\text{maximize}} \quad \mathbb{E}\left[\sum_{n=1}^N R_n\right] \tag{8a}$$

$$\text{subject to} \quad \mathbf{W}_b = \mathcal{F}_b\left(\{\mathbf{H}_b\}_{b=1}^B\right), \quad b = 1, \ldots, B \ , \tag{8b}$$

$$\mathbf{W}_b \mathbf{W}_b^{\mathsf{H}} = \mathbf{I}, \quad b = 1, \ldots, B \ , \tag{8c}$$


Central Processor (CP):
Compute $\{\partial R/\partial \mathbf{F}_b\}_{b=1}^{B}$ based on received $\{\mathbf{F}_b\}_{b=1}^{B}$
$\mathbf{F}_b^{(0)}$
$\left.\frac{\partial R}{\partial \mathbf{F}_b}\right|_{\mathbf{F}_b=\mathbf{F}_b^{(0)}}$
$\mathbf{F}_b^{(T-1)}$
$\left.\frac{\partial R}{\partial \mathbf{F}_b}\right|_{\mathbf{F}_b=\mathbf{F}_b^{(T)}}$
$\mathbf{H}_b^{\mathsf{H}}$
$\mathbf{H}_b$
$\left.\frac{\partial R}{\partial \mathbf{W}_b}\right|_{\mathbf{W}_b=\mathbf{W}_b^{(0)}}$
$\left.\frac{\partial R}{\partial \mathbf{W}_b}\right|_{\mathbf{W}_b=\mathbf{W}_b^{(T)}}$
DNN
$\mathcal{P}_{\boldsymbol{\theta}_b}(\cdot)$
GRU Block
$\mathbf{W}_b^{(0)}$
$\mathbf{W}_b^{(1)}$
$\mathbf{W}_b^{(T-1)}$
$\mathbf{W}_b = \mathbf{W}_b^{(T)}$
Initialization Stage
Refinement Stage
$\boldsymbol{c}_b^{(t-1)}$
GRU Cell
$\mathcal{G}_{\boldsymbol{\theta}_g}(\cdot)$
$\boldsymbol{c}_b^{(t)}$
$\mathcal{F}_{\boldsymbol{\theta}_f}(\cdot)$
$\left.\frac{\partial R}{\partial \mathbf{W}_b}\right|_{\mathbf{W}_b=\mathbf{W}_b^{(t-1)}}$
$\Delta \mathbf{W}_b^{(t)}$
$\mathbf{W}_b^{(t-1)}$
$\widetilde{\mathbf{W}}_b^{(t)}$
QR
$\mathbf{W}_b^{(t)}$

(a) The uplink and downlink low-dimensional signaling scheme.

(b) The GRU block for updating the compression matrix at RRH b in the t-th iteration.

Fig. 2. A block diagram of the proposed two-stage meta-learning algorithm for designing compression matrices.

where the expectation in the objective function is over the distribution of channel matrices $\{\mathbf{H}_b\}_{b=1}^{B}$.

Solving the problem (8) is challenging in terms of both communication cost and computational complexity. Specifically, the optimal compression matrices $\{\mathbf{W}_b\}_{b=1}^{B}$ need to be designed jointly at the CP with the global CSI $\{\mathbf{H}_b\}_{b=1}^{B}$, but this requires significant communication overhead for transmitting the local CSI from the RRHs to the CP and transmitting the designed compression matrices from the CP back to the RRHs through the fronthaul. Moreover, given the global CSI, this problem is also computationally difficult to solve due to its non-convexity and the complex user-centric clustering setting.

To reduce the communication overhead, most of the existing works [9]–[11] solve the problem (8) using the EVD method, which optimizes the compression matrix individually at each RRH based on the local CSI:

$$\begin{aligned} \underset{\mathbf{W}_b}{\text{minimize}} \quad & \mathbb{E}\left[\left\|\boldsymbol{y}_b - \mathbf{W}_b^{\mathsf{H}}\mathbf{W}_b\boldsymbol{y}_b\right\|_2^2\right] && \text{(9a)} \\ \text{subject to} \quad & \mathbf{W}_b\mathbf{W}_b^{\mathsf{H}} = \mathbf{I} \; . && \text{(9b)} \end{aligned}$$

However, this EVD-based approach is highly sub-optimal since independent local metrics do not correspond to the end-to-end system objective. In this paper, we propose a novel data-driven approach to solve this problem. By carefully designing the signaling scheme and the neural network structure, we can achieve near-optimal performance with significantly reduced communication overhead.

III. Two-Stage Design of Dimension Reducing Matrices

In this section, we propose a two-stage deep learning framework including an initialization stage and a refinement stage to solve the problem (8), as shown in Fig. 2. Specifically, the compression matrix at each RRH is first derived from the local CSI using a DNN in the first stage, and then refined iteratively using the gradient of the objective in the second stage. To reduce the communication overhead in the refinement stage, a low-dimensional signaling scheme is proposed for each iteration, and a meta-learning-based GRU network is designed to reduce the number of iteration rounds.

A. Stage One: Initialization Using Local CSI

In the first stage, each RRH uses a DNN to design its compression matrix based on its local CSI. Mathematically, RRH b maps its local CSI $\mathbf{H}_b$ to the compression matrix $\mathbf{W}_b^{(0)}$ according to:

$$\mathbf{W}_b^{(0)} = \mathcal{P}_{\boldsymbol{\theta}_b}\left(\mathbf{H}_b\right) \; , \tag{10}$$

where $\mathcal{P}_{\boldsymbol{\theta}_b}(\cdot)$ denotes the DNN parameterized by $\boldsymbol{\theta}_b$. As shown in Fig. 2(a), the B DNNs in stage one are concatenated with the network modules in stage two and the overall network is trained in an end-to-end manner using unsupervised learning.

As a special case, stage one can be implemented alone as a local CSI-based method. Specifically, at the beginning of each channel coherence block, RRH b first designs the compression matrix $\mathbf{W}_b$ using the DNN $\mathcal{P}_{\boldsymbol{\theta}_b}$ based on the local CSI $\mathbf{H}_b$, and then transmits the effective channel $\mathbf{F}_b$ to the CP to design the receive beamformer in (7). This local CSI-based deep learning method can outperform the heuristic EVD method in [9] since the DNNs learn to optimize the end-to-end loss function instead of the intermediary MSE loss as in (9). As such, even though the global CSI is inaccessible to each RRH, DNNs can implicitly learn to make use of the statistical distribution of the global CSI through training.

B. Stage Two: Iterative Refinement Using Global Signaling

After the compression matrices are initialized at each RRH using the local CSI, a refinement process can be implemented to further improve the performance by exchanging some low-dimensional global information between the RRHs and the CP iteratively. To ensure the efficiency of the iterative algorithm, we propose a low-dimensional signaling transmission scheme for each iteration, and a GRU-based meta-learning framework to accelerate the convergence speed.

TABLE I
COMMUNICATION OVERHEAD PER RRH

Methods	Uplink Transmission	Downlink Transmission	Total Communication Overheads
Local CSI (EVD/DNN)	$\mathbf{F}_b$	–	KN
Two-stage DNN	$\mathbf{F}_b^{(t)}$ $t = 0, \dots, T$	$\frac{\partial R}{\partial \mathbf{F}_b}\Big\vert_{\mathbf{F}_b=\mathbf{F}_b^{(t)}}$ $t = 0, \dots, T-1$	$(2T+1)KN$
Global CSI	$\mathbf{H}_b$	$\mathbf{W}_b$	$MN + KM$

1) Low Dimensional Signaling Scheme: A straightforward signaling scheme is based on gradient descent (GD). Specifically, RRH b sends the compression matrix $\mathbf{W}_b$ to the CP; the CP calculates the gradient $\partial R/\partial \mathbf{W}_b$ and sends it back to RRH b to update the compression matrix $\mathbf{W}_b$ using the GD algorithm. However, computing $\partial R/\partial \mathbf{W}_b$ requires global CSI matrices $\{\mathbf{H}_b\}_{b=1}^{B}$, which are not available in the CP. In this paper, we make a key observation that since the sum rate is only a function of effective CSI matrices according to (6) and (7), we have:

$$\frac{\partial R}{\partial \mathbf{W}_b} = \frac{\partial R}{\partial \mathbf{F}_b}\mathbf{H}_b^{\mathsf{H}} \ , \tag{11}$$

which implies that the gradient with respect to $\mathbf{W}_b$ can be recovered at RRH b based on the gradient with respect to $\mathbf{F}_b$ plus the local CSI $\mathbf{H}_b$. Therefore, we propose a signaling scheme as shown in Fig. 2(a). Specifically, in the t-th iteration, RRH b sends the CP its effective CSI $\mathbf{F}_b^{(t-1)}$. After collecting the effective CSI matrices from all the RRHs, the CP computes the gradient $\partial R/\partial \mathbf{F}_b$ evaluated at $\mathbf{F}_b = \mathbf{F}_b^{(t-1)}$ and transmits it back to RRH b. Then, RRH b recovers the gradient $\partial R/\partial \mathbf{W}_b$ evaluated at $\mathbf{W}_b = \mathbf{W}_b^{(t-1)}$ according to (11) and performs the GD update given by:

$$\tilde{\mathbf{W}}_b^{(t)} = \mathbf{W}_b^{(t-1)} + \alpha_t \frac{\partial R}{\partial \mathbf{W}_b}\bigg|_{\mathbf{W}_b=\mathbf{W}_b^{(t-1)}} \ , \tag{12}$$

where α_t is the step size. At the end of each iteration, we take the QR decomposition of $\left(\tilde{\mathbf{W}}_b^{(t)}\right)^{\mathsf{H}} = \mathbf{QR}$ and set the $\mathbf{Q}^{\mathsf{H}}$ matrix as the orthogonalized $\mathbf{W}_b^{(t)}$ to satisfy the constraint (8c). The final compression matrices are decided after T iterations, i.e., $\mathbf{W}_b = \mathbf{W}_b^{(T)}$.

We quantify the amount of communication overhead as the number of entries in the signaling matrices. For each iteration of the proposed scheme, both the uplink and downlink signaling are $K \times N$ dimensional, consuming a total of $(2T+1)KN$ overhead per RRH for T refinement iterations and one final uplink transmission. Comparatively, the global CSI-based approach requires the transmission of the $M \times N$ dimensional full CSI in the uplink, and the $K \times M$ dimensional compression matrix in the downlink for each RRH. Since the compression dimension K is much smaller than the number of antennas M in a C-RAN system, the amount of overhead can be significantly reduced by the proposed low-dimensional iterative signaling scheme, provided that the number of iterations required is small. The communication overhead of different methods is listed in Table I.

Algorithm 1 Proposed two-stage meta-learning algorithm

```
# Stage 1: Initialization using local CSI
Initialize $\mathbf{W}_b^{(0)}$ via (10), $b = 1, \dots B$
# Stage 2: Iterative refinement using global signaling
Initialize GRU hidden state vector $\boldsymbol{c}_b^{(0)}$
for $t = 1 : T$ do
    RRH $b$ sends effective CSI $\mathbf{F}_b^{(t-1)}$ to CP, $b = 1, \dots B$
    CP computes gradient $\partial R/\partial \mathbf{F}_b|_{\mathbf{F}_b=\mathbf{F}_b^{(t-1)}}$
      and sends it back to RRH $b$, $b = 1, \dots B$
    for each RRH $b = 1, \dots B$ do
        Compute gradient $\partial R/\partial \mathbf{W}_b|_{\mathbf{W}_b=\mathbf{W}_b^{(t-1)}}$ via (11)
        Update hidden state vector $\boldsymbol{c}_b^{(t)}$ via (13)
        Compute update term $\Delta\mathbf{W}_b^{(t)}$ via (14)
        Update $\mathbf{W}_b^{(t)}$ via (15) and QR decomposition.
    end for
end for
Set $\mathbf{W}_b = \mathbf{W}_b^{(T)}$
```

2) Reducing Signaling Rounds via Meta-Learning : To reduce the number of communication rounds T, we need an algorithm that can converge quickly. However, GD in general has a slow convergence rate since it only uses the gradient information in each time-step, and the optimal step size for each iteration is difficult to choose, especially since the RRHs do not have access to the full CSI. To accelerate the convergence speed, we propose a meta-learning-based GRU network to learn the update step based on the current and historical gradient information [14], [15]. Specifically, in the t-th refinement iteration of RRH b, the GRU cell takes the previous hidden state vector $\boldsymbol{c}_b^{(t-1)}$ and the gradient $\partial R/\partial \mathbf{W}_b$ evaluated at the previous compression matrix $\mathbf{W}_b^{(t-1)}$ as inputs, and outputs the new hidden state vector $\boldsymbol{c}_b^{(t)}$ according to:

$$\boldsymbol{c}_b^{(t)} = \mathcal{G}_{\boldsymbol{\theta}_g}\left(\boldsymbol{c}_b^{(t-1)}, \frac{\partial R}{\partial \mathbf{W}_b}\bigg|_{\mathbf{W}_b=\mathbf{W}_b^{(t-1)}}\right) \ , \tag{13}$$

where $\mathcal{G}_{\boldsymbol{\theta}_g}(\cdot)$ denotes the GRU hidden state update function parameterized by $\boldsymbol{\theta}_g$. Using another DNN $\mathcal{F}_{\boldsymbol{\theta}_f}(\cdot)$ parameterized by $\boldsymbol{\theta}_f$, the update term $\varDelta\mathbf{W}_b^{(t)}$ is mapped from the hidden state vector $\boldsymbol{c}_b^{(t)}$ according to:

$$\Delta\mathbf{W}_b^{(t)} = \mathcal{F}_{\boldsymbol{\theta}_f}\left(\boldsymbol{c}_b^{(t)}\right) \ . \tag{14}$$

Finally, the compression matrix for RRH b at iteration t is updated as:

$$\tilde{\mathbf{W}}_b^{(t)} = \mathbf{W}_b^{(t-1)} + \varDelta\mathbf{W}_b^{(t)} \ , \tag{15}$$

followed by the QR decomposition step. The above GRU block is concatenated for T iterations, as shown in Fig. 2(a). The overall two-stage network is trained in an end-to-end manner using the loss function $-\sum_{t=0}^{T} R\left(\mathbf{W}_b^{(t-1)}\right)$ to optimize the network parameters $\left\{\{\boldsymbol{\theta}_b\}_{b=1}^{B}, \boldsymbol{\theta}_g, \boldsymbol{\theta}_f\right\}$. The overall algorithm is summarized in **Algorithm 1**.

We remark that the proposed meta-learning-based approach can converge much faster than GD because the GRU can learn an efficient update rule for this specific problem from training data rather than using manually designed general update rules.

IV. Simulation Results

A. System Setup

In this section, we evaluate the performances of the proposed scheme with 19-cell wrap-around cellular network simulation topology. We consider a dense urban network, where the distance between two neighbouring RRHs is 150m and the height of an RRH is 30m. During each scheduling time slot, 2 users are uniformly generated in each cell, and the transmit power of each user is 23dBm. The system bandwidth is 20MHz and background noise level is -169dBm/Hz. A signal-to-interference-plus-noise ratio (SINR) gap of 6dB is considered to account for the coding and modulation scheme used in practice. We assume that the carrier frequency is 2.9GHz and the channel follows Rayleigh fading with the distance dependent path loss $41.74 + 29\log 10\,(d)$ [16]. Simulations are performed in the real field for simplicity.

We perform simulations for two scenarios: the number of antennas $M = 8$ and $M = 32$, respectively. In both cases, we set the compression dimension $K = 2$. To make sure that the C-RAN system is operating in the correct regime, the cluster size $|\Theta_n|$ should satisfy $K\,|\Theta_n| > M$ so that the number of effective antennas in a cluster is greater than the number of antennas M at a single RRH. In our simulation, we set $|\Theta_n| = 7$ for $M = 8$, and $|\Theta_n| = 17$ for $M = 32$.

We compare the performance of proposed data-driven scheme with the following benchmarks:

- **Single-cell processing.** This corresponds to the setting of traditional cellular MIMO network without cooperation, where each base station employs the LMMSE beamformer [13] for users in its own cell using the local CSI.
- **EVD using local CSI [9].** The rows of the compression matrix for RRH b are derived by taking the eigenvectors corresponding to the K largest eigenvalues of the received signal's covariance matrix $\mathbf{\Sigma}_{\boldsymbol{y}_b\boldsymbol{y}_b} = P_x\mathbf{H}_b\mathbf{H}_b^{\mathsf{H}} + \sigma_z^2\mathbf{I}$. The effective channel $\mathbf{F}_b = \mathbf{W}_b\mathbf{H}_b$ is forwarded to the CP for designing receive beamformers.
- **GD using global CSI.** To reap full cooperation gain, each RRH forwards its local CSI to the CP, which jointly optimizes the compression matrices for all RRHs using the global CSI with GD. The designed compression matrices are then transmitted back to each RRH.
- **DNN using local CSI (first stage) + GD.** The second stage of the proposed network is replaced with generic GD, which has the same signaling scheme as in Sec. III-B but uses the update rule in (12). The constant step size is tuned manually so that GD can achieve the best performance within the number of iterations allowed.

B. Neural Network Implementation Details

In the first stage of the proposed framework, a common 3-layer fully connected DNN with hidden layers of width $[2048, 512]$ and tanh nonlinearity is used at all RRHs. Since the distance between RRH b and the i-th scheduled user d_{bi} may change drastically in different scheduling timeslots, the 2-norm of the input feature $\boldsymbol{h}_{bi}$ varies significantly during the neural network training stage, which causes severe training difficulties. Therefore, we sort the columns $\{\boldsymbol{h}_{bi}\}_{i=1}^{N}$ according to the distance d_{bi} before $\mathbf{H}_b$ is flattened and fed into the input layer. We add a normalization layer after the output layer where QR decomposition is adopted to guarantee $\mathbf{W}_b\mathbf{W}_b^{\mathsf{H}} = \mathbf{I}$. In the second stage of the proposed framework, the GRU and DNN in each iteration block are reused for every iteration and every RRH to save memory. The hidden unit of GRU has size of $2KM$, and the DNN has one hidden layer of size $2KM$. We implement the deep learning models in PyTorch [17] and train them using the Adam optimizer [18].

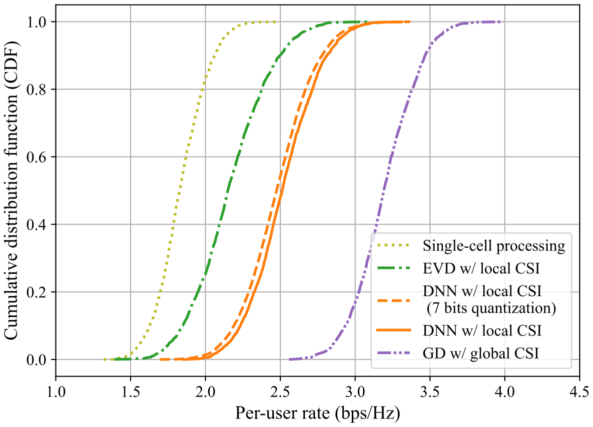


Fig. 3. CDF curve of per-user rate ($M = 8$).

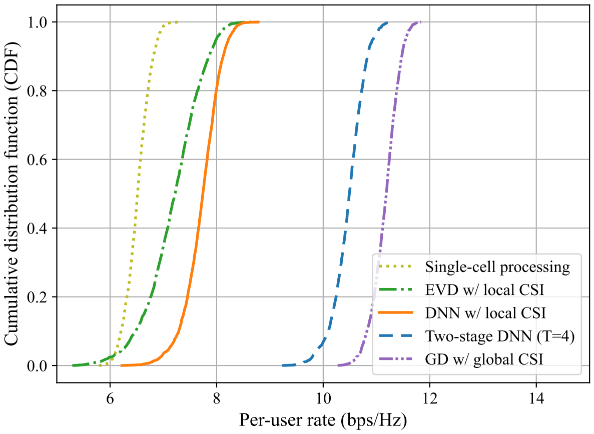


Fig. 4. CDF curve of per-user rate ($M = 32$).

C. Simulation Results

We first illustrate the benefit of using the local CSI-based DNN method. That is, the compression matrices are designed using only the first stage of the proposed two-stage DNN. As shown in Fig. 3 and Fig. 4, the local CSI-based DNN methods can achieve better performance compared to the EVD method. Note that this gain is obtained without introducing any additional communication cost; it comes from the use of

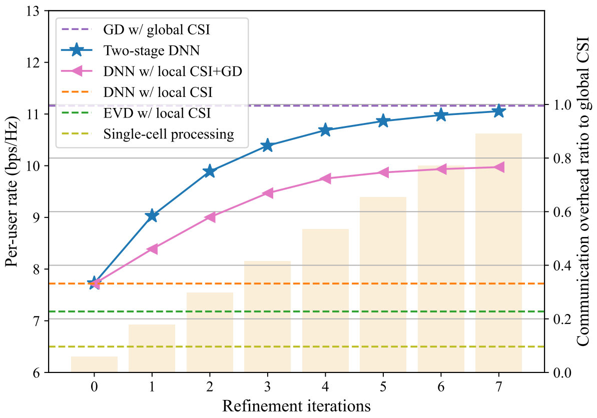


Fig. 5. Left vertical axis: Convergence curve for the per-user rate. Right vertical axis: Bar graph indicating the communication overhead ratio of the proposed iterative refinement scheme to the global CSI-based method. ($M = 32$).

an end-to-end loss function instead of a heuristic local loss function and the implicit utilization of the distribution of the global CSI learned by the DNN through training.

We remark that uniform quantization can be readily incorporated into the proposed framework. That is, after applying dimension reducing matrices, we uniformly quantize each dimension with 7 bits and model the quantization error as additive Gaussian noise [9]. As shown in Fig. 3, the performance degradation caused by quantization is negligible.

Even though the local CSI-based DNN can significantly outperform EVD, there is still a large gap from the global CSI-based GD method. For a system equipped with small number of antennas, e.g., $M = 8$, the dimension of the global CSI may not be very large; thus, it is feasible to send all the CSI matrices from the RRHs to the CP and use the GD method to design the compression matrices based on the global CSI. However, for large-scale antenna systems, e.g., $M = 32$, the significant overhead requirement makes the global CSI-based approach impractical. In this case, the proposed iterative refinement scheme can provide an efficient trade-off between performance and communication overhead. Specifically, as shown in Fig. 5, the gap between the local CSI-based DNN and the global CSI-based GD method can be closed by 85% with only 4 refinement iterations, which corresponds to only 50% of overhead required by the global CSI-based approach. Moreover, it can be observed that the GRU-based meta-learning method converges much faster than GD due to the use of the optimized update rule for this specific problem learned from training data. This significantly reduces communication rounds for CSI sharing and gradient transmission.

V. Conclusion

This paper investigates the problem of designing compression matrices in an uplink C-RAN system. A two-stage deep learning framework is proposed to optimize the end-to-end sum rate objective, where the compression matrices are derived from the local CSI using fully connected neural networks in the first stage and further refined iteratively using the downlink signaling from the CP in the second stage. To reduce the communication overhead, a low-dimensional signaling scheme is proposed to reduce the overhead for each iteration, and a novel GRU-based meta-learning framework is proposed to accelerate convergence speed of the refinement process. Simulation results show that the performance of the proposed neural network can quickly converge to that of the global CSI-based benchmark with significantly smaller overhead, and that using the first stage alone can already outperform the local CSI-based EVD benchmark.

References

[1] A. Checko *et al.*, "Cloud RAN for mobile networks—a technology overview," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 405–426, Firstquarter 2014.

[2] O. Simeone, A. Maeder, M. Peng, O. Sahin, and W. Yu, "Cloud radio access network: Virtualizing wireless access for dense heterogeneous systems," *J. Commn. Net.*, vol. 18, no. 2, pp. 135–149, Apr. 2016.

[3] E. Björnson and L. Sanguinetti, "Scalable cell-free massive MIMO systems," *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 4247–4261, July 2020.

[4] W. Yu *et al.*, "Cooperative beamforming and resource optimization in C-RAN," in *Cloud Radio Access Networks Principles, Technologies, and Applications*. Cambridge Univ. Press, 2017, pp. 54–81.

[5] F. Sohrabi, T. Jiang, and W. Yu, "Learning progressive distributed compression strategies from local channel state information," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 3, pp. 573–584, Apr. 2022.

[6] F. Wiffen, M. Z. Bocus, A. Doufexi, and W. H. Chin, "MF-based dimension reduction signal compression for fronthaul-constrained distributed MIMO C-RAN," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Virtual Conference, May 2020, pp. 1–8.

[7] I. D. Schizas, G. B. Giannakis, and Z.-Q. Luo, "Distributed estimation using reduced-dimensionality sensor observations," *IEEE Trans. Signal Process.*, vol. 55, no. 8, pp. 4284–4299, Aug. 2007.

[8] F. Wiffen, W. H. Chin, and A. Doufexi, "Distributed dimension reduction for distributed massive MIMO C-RAN with finite fronthaul capacity," in *IEEE Asilomar Conf. Signals, Syst., Comput.* Pacific Grove, CA, USA: IEEE, Nov. 2021, pp. 1228–1236.

[9] L. Liu, W. Yu, and O. Simeone, *Fronthaul-Aware Design for Cloud Radio Access Networks*. Cambridge Univ. Press, 2017, p. 48–75.

[10] L. Liu and R. Zhang, "Optimized uplink transmission in multi-antenna C-RAN with spatial compression and forward," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5083–5095, Oct. 2015.

[11] F. Wiffen, M. Z. Bocus, A. Doufexi, and A. Nix, "Distributed MIMO uplink capacity under transform coding fronthaul compression," in *IEEE Int. Conf. Commun. (ICC)*, Shanghai, China, May 2019, pp. 1–6.

[12] C. Zhu and W. Yu, "Stochastic modeling and analysis of user-centric network MIMO systems," *IEEE Trans. Commun.*, vol. 66, no. 12, pp. 6176–6189, Dec. 2018.

[13] M. Joham, W. Utschick, and J. A. Nossek, "Linear transmit processing in MIMO communications systems," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2700–2712, Aug. 2005.

[14] M. Andrychowicz *et al.*, "Learning to learn by gradient descent by gradient descent," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 29, Dec. 2016.

[15] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, Sept. 2014.

[16] S. Sun *et al.*, "Investigation of prediction accuracy, sensitivity, and parameter stability of large-scale propagation path loss models for 5G wireless communications," *IEEE Trans. Veh. Technol.*, vol. 65, no. 5, pp. 2843–2860, 2016.

[17] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 32, Dec. 2019.

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations (ICLR)*, Banff, AB, Canada, Dec. 2014.