

# Coalition Formation for Heterogeneous Federated Learning Enabled Channel Estimation in RIS-assisted Cell-free MIMO

Nan Qi, *Senior Member, IEEE*, Haoxuan Liu, *Member, IEEE*, Theodoros A. Tsiftsis, *Senior Member, IEEE*, Alexandros-Apostolos A. Boulogeorgos, *Senior Member, IEEE*, Fuhui Zhou, *Senior Member, IEEE*, Shi Jin, *Fellow, IEEE*, Qihui Wu, *Fellow, IEEE*

arXiv:2502.05538v1 [cs.IT] 8 Feb 2025

**Abstract**—Downlink channel estimation remains a significant bottleneck in reconfigurable intelligent surface-assisted cell-free multiple-input multiple-output communication systems. Conventional approaches primarily rely on centralized deep learning methods to estimate the high-dimensional and complex cascaded channels. These methods require data aggregation from all users for centralized model training, leading to excessive communication overhead and significant data privacy concerns. Additionally, the large size of local learning models imposes heavy computational demands on end users, necessitating strong computational capabilities that most commercial devices lack. To address the aforementioned challenges, a coalition-formation-guided heterogeneous federated learning (FL) framework is proposed. This framework leverages coalition formation to guide the formation of heterogeneous FL user groups for efficient channel estimation. Specifically, by utilizing a distributed deep reinforcement learning (DRL) approach, each FL user intelligently and independently decides whether to join or leave a coalition, aiming at improving channel estimation accuracy, while reducing local model size and computational costs for end users. Moreover, to accelerate the DRL-FL convergence process and reduce computational burdens on end users, a transfer learning method is introduced. This method incorporates both received reference signal power and distance similarity metrics, by considering that nodes with similar distances to the base station and comparable received signal power have a strong likelihood of experiencing similar channel fading. Massive experiments performed that reveal that, compared with the benchmarks, the proposed framework significantly reduces the computational overhead of end users by 16%, improves data privacy, and improves channel estimation accuracy by 20%.

**Index Terms**—Coalition formation, reconfigurable intelligent surface, multiple-input multiple-output, channel estimation, het-

This article was accepted in part at the IEEE Wireless Communications and Networking Conference(WCNC), Milan, Italy, March 24–27, 2025 [1].

Nan Qi, Haoxuan Liu, Fuhui Zhou, and Qihui Wu are with the Key Laboratory of Dynamic Cognitive Systems of Electromagnetic Spectrum Space, Ministry of Industry and Information Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 210016. E-mail: nanqi.common@gmail.com, nuaalhx@nuaa.edu.cn, zhouchuhui@ieee.org, wuqihui@nuaa.edu.cn.

Theodoros A. Tsiftsis is with the Department of Informatics and Telecommunications, University of Thessaly, 35100 Lamia, Greece, E-mail: tsiftsis@uth.gr.

Alexandros -Apostolos. A. Boulogeorgos is with the Department of Electrical and Computer Engineering, University of Western Macedonia, 50100 Kozani, Greece, E-mail: aboulogeorgos@uowm.gr.

S. Jin is with the National Mobile Communications Research Laboratory and Frontiers Science Center for Mobile Information Communication and Security, Southeast University, Nanjing 210096, P.R. China, E-mail: jinshi@seu.edu.cn.

erogeneous federated learning (FL), transfer learning, distributed deep reinforcement learning.

## I. INTRODUCTION

Over the past decades, wireless technologies have been substantially evolving to meet the challenges of ultra-high capacity, ultra-low latency, and massive device access brought about by fifth-generation (5G) networks and beyond. One of the most powerful technologies is multiple-input multiple-output (MIMO) [2]–[4]. MIMO networks, comprised of multiple multi-antenna base stations (BSs) serving multiple users within a designated area, now play a pivotal role in enhancing spectral efficiency and network capacity. However, as the number of mobile communication users surges suddenly, the communication rate of individual users (especially the edge users) will be forced to decrease. To enhance system spectral efficiency and quality of service of edge users, a cell-free MIMO network architecture has recently been proposed [5]. In this scheme, all BSs are connected to a central processing unit (CPU), which operates them as a unified MIMO network without cell boundaries. This enables coherent transmission and reception, allowing the BSs to serve all users over the same time-frequency resources by applying spatial multiplexing techniques [6].

Meanwhile, reconfigurable intelligent surface/intelligent reflection surfaces (RISs / IRSs) have attracted extensive attention in both academia and industry. RISs can manipulate electromagnetic parameters like phase, amplitude, frequency, and polarization [7] to create a favorable wireless communication environment, which is considered to be a key technology in the sixth generation (6G) [8]. Researchers have investigated the fundamental constraints and system architecture, revealing that RIS-assisted cell-free MIMO can greatly increase user communication rates [9], [10].

Several studies highlight the significance of RIS in wireless networks optimize wireless network performance by simultaneously optimizing the passive beamforming and BS transmit beamforming [11]–[15]. In particular, the authors in [11] optimized beamforming by weighted minimize mean-square error (WMMSE) maximizing a wireless network user's weighted sum rate (WSR), and the complexity of the algorithm is greatly reduced by deriving closed-form solutions. Furthermore, to optimize the minimum possible rate for cell-free MIMO networks supported by numerous RISs, [12] and

TABLE I: A comparison with existing literature

	[16]–[20]	[21], [22]	[23]–[25]	[26], [27]	[28], [29]	[30]	[31]–[33]	<b>Our work</b>
Cascaded channel	✓	✓	✓	✗	✗	✓	✗	✓
Distributed training	✗	✓	✓	✓	✓	✗	✗	✓
Coalition formation game	✗	✗	✗	✗	✓	✓	✓	✓
Distributed intelligent decision-making	✗	✗	✗	✗	✗	✓	✓	✓
Heterogeneous training model	✗	✗	✗	✓	✗	✗	✗	✓

[13] have focused on the fairness problem. The authors of [14] and [15] have optimized the energy efficiency of RIS-assisted networks through network deployment as well as beamforming. Nevertheless, prior research is primarily available on the assumption that perfect channel information exists. In massive MIMO, where the electromagnetic environment is intricate and cluttered, obtaining perfect channel state information remains challenging.

### A. Related Works

Existing studies, as summarized in TABLE I, have explored channel estimation in RIS-assisted wireless systems, including least square (LS), minimum mean square error (MMSE) [16], matrix factorization [17], and compressed sensing [18] methods. Despite the great potential of these techniques, the high demands on the antenna array response (e.g., sparsity) and the high complexity of the required computations greatly restrict their practical applications. To address these issues, data-driven methodologies, notably deep learning (DL) algorithms [19], have emerged as viable alternatives for dealing with the complex non-linear correlations inherent in receiving signal samples. In particular, [20] employed pilot design and non-local attention modules in neural networks by gradually pruning less significant neurons, reducing the pilot transmission overhead, and improving channel estimation accuracy.

The performance of DL generally scales with both the model size and the volume of training data. However, large models and high-volume datasets impose heavy computational demands on a single node, often exceeding its computation capability. Distributed machine learning (DML) is a powerful solution to address this challenge by leveraging parallel processing [21]. It divides data into batches or partitions the model across multiple compute nodes for concurrent training of DL models, followed by inter-node communication to update model parameters. The research in [22] applied DML to down-link channel estimation in RIS-based networks and proposed a hierarchical neural network-based DML approach. This method improves estimation accuracy by pre-classifying user groups under different scenarios and then training scenario-specific models using the corresponding data. However, in DML method, the transmission processes involved in data distribution and model updates lead to increased communication overhead and a risk of data leakage, thereby threatening user privacy.

*To address the challenges of high communication overhead and privacy leakage, federated learning (FL) schemes, as special kinds of DML, have been recently proposed. In FL, rather than transmitting the entire dataset, only model updates, such as the gradients of model parameters, are exchanged.* There have been some important works on how to utilize FL for RIS channel estimation [23], [24]. Elbir *et. al.* applied

FL to channel estimation for RIS-assisted MIMO systems [23]. In comparison with the centralized learning method, the ChannelNet learning method in [23] significantly lowers the amount of communication overhead by utilizing the FL method. However, the performance, while approaching that of MMSE, remains suboptimal due to the aggregation of parameter updates from all participating users. Compared to [23], [24] achieves high-precision channel estimation while reducing the number of model parameters by integrating a deep residual network into the FL framework, thereby further decreasing the communication overhead. In addition, [25] employs hierarchical residual networks and federated learning to classify user distribution scenarios, addressing the decline in channel estimation accuracy caused by uneven user location distribution. Moreover, the large size of the local learning models places a significant computational burden on end-user devices, requiring powerful computing capabilities that exceed the resources available on many existing devices. *To adapt to the different computing and storage capabilities of the end-user, the network model needs to be flexibly resizable, and heterogeneous federated learning (HFL) is ideally suited to address the requirement.* The authors of [26] articulated customized model aggregation strategies to cope with different types of heterogeneity and increase the feasibility. Data-free knowledge distillation techniques were introduced to HFL, enabling the efficient transfer of model knowledge without requiring access to raw data. This approach is particularly beneficial in resource-constrained environments [27].

In multi-user RIS-MIMO networks, channel attenuation varies significantly among geographically dispersed users. This variability necessitates partitioning users into multiple federated learning (FL) groups, with each group aggregating its own channel model. However, *most existing FL applications in channel estimation do not address the optimization of user grouping; instead, they use predefined FL groups. This approach may deteriorate channel estimation accuracy in the aggregated models.* A coalition formation game provides a framework for multiple agents (players) to cooperate and share resources (e.g., local model parameters of neural networks) to form coalitions for mutual benefits or shared goals (e.g., improved channel estimation accuracy). This framework supports user group partitioning, allowing players to form coalitions with others to improve their outcomes and determining how the overall benefits, such as rewards or utility, are distributed among members. *Therefore, the problem of federated learning user grouping can be readily converted into a coalition formation game.* However, few studies have addressed the FL user grouping problem by using coalition formation games. In [28] and [29], coalition formation was employed to quantify user data contribution [28], and prevent user selfishness

[29]. However, all the above works focus on a single FL aggregated model. Specifically, the work in [28] was the first to incorporate coalition games into personalized federated learning, utilizing Shapley values to assess the contribution of other users' models to an individual model, subsequently selecting parameters that enhance performance and mitigate degradation from malicious users. Literature The authors of [29] introduced a coalition member trust mechanism to prevent selfish devices from harming their coalition, mitigating the problems of non-independent and identically distributed (I.I.D) data and communication bottlenecks. Moreover, the authors of [30] investigated the trade-off between channel estimation accuracy and communication overhead in centralized computation using clustering algorithms and proved the algorithm's stable convergence. This reveals the feasibility of combining coalition formation games with federated learning for channel estimation.

To address the FL user grouping problem through coalition formation games, two subproblems need to be tackled: 1) Establishing stable coalitions is essential: This involves designing an overall payoff function, a payoff distribution policy, and a coalition formation order that align to improve channel estimation accuracy, while ensuring the existence of a stable coalition partition. 2) The identification of a stable coalition partition. In particular, for subproblem 2), the coalition formation of the FL user is mathematically intractable under an unknown channel model, especially when coupled with the iterative framework of federated learning. This makes it infeasible to formulate the overall coalition payoff and utility distribution among coalition members, both of which are essential for guiding coalition formation. Conventional approaches, such as mixed-integer programming and heuristic algorithms (e.g., ant colony optimization, genetic algorithms), are no longer applicable. In order to tackle those problems, a data-driven deep reinforcement learning (DRL) approach can be considered to address the coalition formation problem. The authors of [31] introduced the DRL-based coalition formation solution for peer-to-peer energy trading problem, aiming to minimize the cost of energy trading. In addition, coalition formation games are also applied in multi-agent task allocation to model task preferences of different nodes, enhancing task completion efficiency through collaboration among multiple coalitions [32], [33].

### B. Motivations and Contributions

Motivated by the above observations and analyses, this paper proposes a coalition-formation-guided heterogeneous FL approach for channel estimation in RIS-assisted cell-free MIMO systems. Specifically, the FL user grouping problem is modeled as a coalition formation problem, and the existence of a stable coalition formation solution is theoretically analyzed. *To the best of our knowledge, this is the first work that proposes a coalition-formation-guided FL method for channel estimation in RIS-MIMO, which can be readily applied to MIMO networks.* Moreover, achieving stable coalition formation solutions is particularly challenging. To address this challenge, we will propose a distributed DRL method that enables each end user to independently decide whether to leave

or join a coalition namely, a FL user group, thereby alleviating the burden of a central decision node. To reduce the size of local models for individual users that have very limited computational resources, such as smartwatches or tablets, a heterogeneous FL (HFL)-based approach will be proposed. Lastly, to accelerate the convergence of the DRL-federated learning process, further reducing computational burdens on end users, we design a transfer learning method.

The major contributions of this work are summarized as follows

- This is the first work to utilize coalition formation to enhance FL user grouping and significantly improve channel estimation accuracy. A coalition altruistic order is proposed, and it is proven that the formulated game has at least one stable coalition partition solution that (at least locally) minimizes the overall channel estimation mean square error (MSE).
- DRL framework, namely, Qmix-based reinforcement learning, is proposed to seek for the stable coalition formation solution. The distributed framework facilitates autonomy and collaboration among the users in the coalition, where each user can independently make decisions on joining or leaving one coalition based on local information, while taking into account the interests of the whole coalition. It significantly reduces the information exchange overhead, compared with the centralized DQN approach, but only brings slight channel estimation accuracy loss (for example, approximately 2.5% loss at a 5dB signal-to-noise ratio).
- A HFL-based approach is proposed to reduce the size of local neural network models for individual users with very limited computational resources, leading to a substantial decrease in computational and power costs during the training process compared to homogeneous FL. While heterogeneous FL may result in a slight reduction in accuracy, it significantly lowers computational costs (measured in Flops) by 16% and complexity. More importantly, under the HFL framework, only a part of model parameters are transmitted, reducing the quantity of data that models transmit, thus improving data privacy compared with that of homogeneous FL.
- To accelerate the convergence of HFL, we propose a transfer learning approach, referred to as “FL+RSRP+Dis”, which incorporates both received signal reference power (RSRP) and distance similarity metrics to enhance the initialization of local model parameters.

### C. Organizations

The RIS-assisted wireless communication system is first introduced in the paper, followed by the problem formulation. Then, a deep learning method for downlink channel estimation is subsequently presented. Section II discusses the system model along with the foundational models. In Section III, a coalition game-enabled federated learning method is proposed to address the channel estimation problem. Section IV presents numerical results and analyzes convergence and complexity. Concluding remarks are given in Section V.

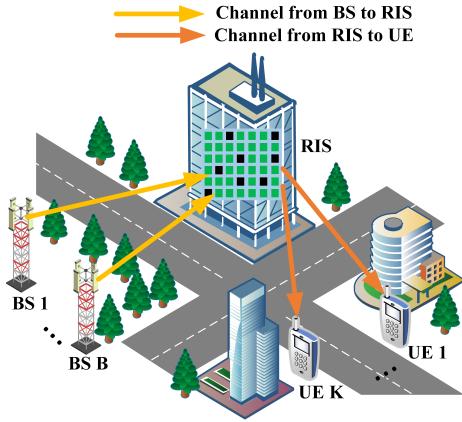


Fig. 1: RIS-assisted cell-free MIMO network.

## II. SYSTEM MODEL AND PROBLEM FORMULATION

An RIS-assisted cell-free MIMO wireless downlink communication system is considered, as illustrated in Figure 1. This system comprises  $B$  BSs and  $K$  user equipment (UEs). Each BS is equipped with  $N_t$  antennas, while each UE is equipped with a single antenna. A single RIS comprising  $N$  elements is deployed between BSs and UEs to enhance the signal quality. All BSs jointly serve each UE, with the beamforming vector from the BSs to UE  $k$  denoted by  $\mathbf{F}_k = [\mathbf{f}_{1,k}, \dots, \mathbf{f}_{b,k}, \dots, \mathbf{f}_{B,k}]^T \in \mathbb{C}^{(N_t \times B) \times 1}$ , where  $\mathbf{f}_{b,k} \in \mathbb{C}^{N_t \times 1}$  represents the beamforming vector from BS  $b$  to UE  $k$ . The phase shift matrix of RIS can be expressed as  $\Phi = \text{diag}(e^{j\theta_1}, e^{j\theta_2}, \dots, e^{j\theta_N}) \in \mathbb{C}^{N \times N}$ , where  $\theta_n$  is the phase-shift coefficient of the  $n$ -th RIS element.

### A. Channel Modeling

The channel matrix between BS  $b$  to RIS and the channel matrix between RIS to UE  $k$  are denoted by  $\mathbf{G}_b \in \mathbb{C}^{N \times N_t}$ ,  $\mathbf{v}_k^H \in \mathbb{C}^{1 \times N}$ , respectively. The channel matrices can be represented by the widely used Saleh-Valenzuela channel model. Specifically,  $\mathbf{G}_b$  is given by

$$\mathbf{G}_b = \sqrt{\frac{N^t N}{L_G}} \sum_{l_1=1}^{L_G} \alpha_{l_1}^G \mathbf{a} \left( \vartheta_{l_1}^{G_r}, \psi_{l_1}^{G_r} \right) \mathbf{u}_b \left( \vartheta_{l_1}^{G_t}, \psi_{l_1}^{G_t} \right)^T, \quad (1)$$

where  $L_G$  represents the number of paths between the BS and the RIS,  $\alpha_{l_1}^G$ ,  $\vartheta_{l_1}^{G_r}$  ( $\psi_{l_1}^{G_r}$ ), and  $\vartheta_{l_1}^{G_t}$  ( $\psi_{l_1}^{G_t}$ ) represent the complex gain, the azimuth (elevation) angle at the RIS, and the azimuth (elevation) angle at BS  $b$  for the  $l_1$ -th path.  $\mathbf{u}_b(\vartheta, \psi) \in \mathbb{C}^{N_t \times 1}$  and  $\mathbf{a}(\vartheta, \psi) \in \mathbb{C}^{N \times 1}$  represent the normalized array steering vector associated to BS  $b$  and the RIS, respectively. For a typical  $N_1 \times N_2$  ( $N = N_1 \times N_2$ ) uniform planar array,  $\mathbf{a}(\vartheta, \psi)$  can be represented by [34]

$$\mathbf{a}(\vartheta, \psi) = \frac{1}{\sqrt{N}} \left[ e^{-j2\pi d \cos(\psi)\mathbf{n}_1/\lambda} \right] \otimes \left[ e^{-j2\pi d \sin(\psi) \cos(\theta)\mathbf{n}_2/\lambda} \right], \quad (2)$$

where  $\mathbf{n}_1 = [0, 1, \dots, N_1 - 1]^T$  and  $\mathbf{n}_2 = [0, 1, \dots, N_2 - 1]^T$ ,  $\lambda$  is the carrier wavelength, and  $d$  is the antenna spacing which is typically set to  $d = \lambda/2$ .

Similarly, the channel matrix between RIS to UE  $k$ , denoted by  $\mathbf{v}_k^H$  can be represented by

$$\mathbf{v}_k^H = \sqrt{\frac{N}{L_k}} \sum_{l_2=1}^{L_k} \alpha_{l_2}^k \mathbf{a} \left( \vartheta_{l_2}^k, \psi_{l_2}^k \right)^T, \quad (3)$$

where  $L_k$  represents the number of paths between the RIS and the  $k$ -th user;  $\alpha_{l_2}^k$ ,  $\vartheta_{l_2}^{G_r}$ , and  $\psi_{l_2}^{G_r}$  represent the complex gain, the azimuth angle, and the elevation angle at the RIS for the  $l_2$ -th path.

It is assumed that the channels between BSs and UEs are completely obstructed. It is worth noting that the model can be easily extended to scenarios involving direct BS-UE links. Given the influence of cascaded channel gain quality on the received signal quality from the BS, this study focuses on the estimation of cascaded channel gains.

Consequently, the signal received by UE  $k$  from BSs is

$$y_k = \sum_{b=1}^B \left\{ \mathbf{v}_k^H \mathbf{G}_b \mathbf{f}_{b,k} \right\} s_k + n_k = \phi \mathbf{h}_k \mathbf{F}_k s_k + n_k, \quad (4)$$

where  $s_k$  represents the pilot signal,  $\phi = \text{diag}(\Phi) \in \mathbb{C}^{1 \times N}$ ,  $\mathbf{h}_k = \text{diag}(\mathbf{v}_k^H) \mathbf{G} \in \mathbb{C}^{N \times (N_t \times B)}$ ,  $\mathbf{G} = [\mathbf{G}_1, \dots, \mathbf{G}_b, \dots, \mathbf{G}_B] \in \mathbb{C}^{N \times (N_t \times B)}$  represents the cascaded channel between BSs and UE  $k$ . Note that during the channel estimation process, both the phase shift matrix  $\Phi$  of RIS and the beamforming vector  $\mathbf{F}_k$  at BSs are predetermined and known to both communication parties like pilot signals. Thus, the task of downlink channel estimation is to estimate the channel matrix  $\mathbf{h}_k$  given  $\Phi$  and  $\mathbf{F}_k$ .

To address the computational challenge, a federated learning-based approach is introduced for downlink cascaded channel estimation. The following two subsections introduce the deep neural network (DNN) model utilized at the UE. Building upon this, a transfer-learning based model is presented.

### B. Basic DNNs Model for Local Channel Estimation

At the UE, a DNN is deployed for downlink cascaded channel estimation, establishing a non-linear mapping between received pilots and cascaded channel parameters. This complex mapping is mathematically represented by

$$\hat{\mathbf{h}}_{b,k} = f_\omega(y_{b,k}), \quad (5)$$

where  $f_\omega(\cdot)$  stands for the non-linear mapping function governed by the weight parameters  $\omega$ ,  $\hat{\mathbf{h}}_{b,k}$  is the estimated channel matrix from the  $b$ -th BS to  $k$ -th UE. To effectively train the DNN, the UE accumulates a sufficient volume of training data beforehand. The training dataset is defined by  $\{\mathbf{y}_{b,k}, \mathbf{h}_{b,k}\}_{D_{b,k}}$ ,  $\forall b, k$ , where  $\mathbf{h}_{b,k}$  is the target channel matrix from the  $b$ -th BS to  $k$ -th UE, and  $D_{b,k}$  represents the size of training dataset for the  $k$ -th UE served by the  $b$ -th BS. The loss function can be represented by

$$\mathcal{L}(\omega) = \frac{1}{D_{b,k}} \sum_{D_{b,k}} \left\| \hat{\mathbf{h}}_{b,k} - \mathbf{h}_{b,k} \right\|_2^2, \quad (6)$$

The goal of training the DNN is to minimize the loss function by optimizing the weight parameters  $\omega$ , i.e.,

$$\min_{\omega} \mathcal{L}(\omega) = \frac{1}{D_{b,k}} \sum_{D_{b,k}} \left\| f_\omega(y_{b,k}) - \mathbf{h}_{b,k} \right\|_2^2. \quad (7)$$

Upon defining the objective function, the DNN undergoes iterative optimization using the given training dataset. In each iteration  $t$ , the weights  $\omega$  are updated using gradient descent, i.e.,

$$\omega_{t+1} = \omega_t - \eta_t \mathbf{g}(\omega_t), \quad (8)$$

where  $\omega_t$  and  $\omega_{t+1}$  are the weights in the  $t$ -th and  $(t+1)$ -th iterations for DNN, respectively. Additionally,  $\mathbf{g}(\omega_t)$  is the gradient vector for  $\omega_t$ , and  $\eta_t$  is the learning rate. After training the DNN, the UE can estimate the channel directly based on the trained DNN.

However, the requirement of sending user data to a centralized node, such as a BS, for unified training incurs substantial overhead and raises significant privacy concerns. To address this issue, we utilize a distributed learning paradigm known as FL, which enables decentralized training.

### C. FL-based Channel Estimation Network Training

Compared with DNN-based methods, the proposed FL-based method reduces communication overhead and enhances privacy, as FL only requires the exchange of model parameters. This leads to a significant reduction in communication costs. The FL integrates DNN and can be fundamentally divided into several procedures:

- *FL user group selection*: The central node (CN) identifies a subset of UEs, denoted by  $\widehat{M} \subset \{1, 2, \dots, M\}$  to participate in the FL process. We refer to the UEs in  $\widehat{M}$  as the active UEs.
- *Model broadcast*: The CN broadcasts the current  $t$ -th global model  $\omega_t^g$  to the active UEs.
- *Local gradient calculation*: Each active UE calculates its local gradient by utilizing its dataset. The gradient is achieved as  $\mathbf{g}(\omega_{m,t}^l) \triangleq \nabla \mathcal{L}(\omega_t^g, D_m)$ , where  $\nabla \mathcal{L}(\omega_t^g, D_m)$  is the gradient of  $\mathcal{L}$  at  $\omega_t = \omega_t^g$ .
- *Model aggregation*: Active UEs transmit their computed local gradients  $\mathbf{g}(\omega_{m,t}^l)$  to the CN via wireless channels. Subsequently, the CN aims to compute a weighted aggregate of these local gradients  $\left\{ \mathbf{g}(\omega_{m,t}^l) : m \in \widehat{M}_t \right\}$ , which is used to update the global model. Adopting the prevalent FL configuration from [35], the weight of each local gradient vector is normalized. The global gradient vector can be thus represented as

$$\mathbf{r}_t \triangleq \sum_{m \in \widehat{M}_t} \left\{ \frac{\alpha_m}{\sum_{i \in \widehat{M}_t} \alpha_i} \mathbf{g}(\omega_{m,t}^l) \right\}, \quad (9)$$

where  $\alpha_m$  represents the corresponding weight attributed to UE  $m$ 's contribution in the update of the global model's parameters, such as distance weight. Therefore, (8) can be written as

$$\omega_{t+1} = \omega_t - \eta_t \mathbf{r}_t. \quad (10)$$

### III. COALITION FORMATION FOR FL ENABLED CHANNEL ESTIMATION

In this section, we design a coalition formation-guided FL framework for channel estimation by integrating the previously introduced DNN and FL approaches. First, each local user trains the DNN using the data that they have collected. Next, to reduce communication overhead and improve channel estimation accuracy, we introduce a coalition formation-guided FL (CFFL) architecture, where a DRL method is employed to intelligently guide the coalition formation. Finally, to enhance the efficiency of the framework, we further integrate transfer learning into the CFFL method.

#### A. DNN for the Downlink Channel Estimation (DCE)

To augment the volume of data available to individual UEs and enhance estimation accuracy, we present a strategy wherein geographically proximate neighboring UEs, within a certain radius from the active UE, contribute a portion of their data to train the DNN network of the active UE. This approach not only ensures an adequate supply of training samples but also minimizes communication overheads compared with conventional distributed ML paradigms. Therefore, a higher learning efficiency and lower transmission overhead can be achieved.

A deep neural network (DNN) architecture is adopted, comprising three convolutional layers followed by a linear layer. Each convolutional layer includes a series of operations: a convolutional operation, a batch normalization operation, and an activation function, respectively.

Considering that both received pilot signal  $y_{b,k}$  and the cascaded channel  $\mathbf{h}_{b,k}$  are complex, the DNN inputs and outputs are transformed to real-valued representations. Specifically,  $y_{b,k}$  is reshaped into a  $Q_1 \times Q_2$  2-dimensional matrix, with its real and imaginary components serving as dual input feature maps for the first convolutional layer. The DNN output is structured as a  $2N \times M$ -element vector, where the first  $N \times M$  entries correspond to the real part of  $\mathbf{h}_{b,k}$ , and the remaining  $N \times M$  entries represent its imaginary counterpart.

Without loss of generality, we assume that UE  $k$  is surrounded by a total of  $Z$  neighboring UEs. The aim of training the DNN is to minimize the sum of the loss functions by optimizing the weights  $\omega$ , i.e.,

$$\begin{aligned} \min_{\omega} \mathcal{L}(\omega) &= \frac{1}{D_k} \sum_{D_k} \| \mathbf{f}_{\omega}(y_k) - \mathbf{h}_k \|_2^2 \\ &+ \sum_{z=1}^Z \left\{ \frac{1}{D_z} \sum_{D_z} \| \mathbf{f}_{\omega}(y_z) - \mathbf{h}_z \|_2^2 \right\}, \end{aligned} \quad (11)$$

where  $D_z \geq a D_k$ ,  $a$  is an empirical value, generally set to 0.1 [36].

#### B. Transfer Learning-based Channel Estimation

The FL process involves the iterative training of local models followed by the aggregation of a global model, which can result in prolonged training times. To accelerate model convergence, while maintaining high accuracy in channel estimation, two types of transfer learning strategies are documented. The fundamental concept is that the DNN network weights of a central node are not trained from scratch; instead, they are initialized based on the trained DNN network weights of neighboring central nodes. This initialization incorporates a weighted aggregation of the weights from neighboring nodes, considering both distance proximity and the similarity of the received reference signal power (RSRP) [37] to these nodes. This approach facilitates the determination of the central node's initial DNN network parameters; thereby, accelerating subsequent DNN parameter iterations. The details of the two types of transfer learning strategies are presented in the following sections.

The first strategy is distance (Dis)-similar transfer learning. Given that users close to one another are likely to exhibit

similar environmental conditions and channel characteristics, transferring federated channel estimation models across proximate spatial locations not only accelerates training but also maintains a certain level of accuracy. The weight under the first transfer strategy is represented as

$$\alpha_m' = \begin{cases} 1 - \frac{L_m}{L}, L_m \leq L, m = 1, 2, \dots, K \\ 0, \text{ otherwise} \end{cases} \quad (12)$$

where  $L_m$  represents the distance between UE  $m$  and the CN aggregating the global model, and  $L$  is the predefined reference distance for normalization. Finally, replacing  $\alpha_m'$  with  $\alpha_m$  in (9) results in the “FL+Dis” strategy.

The second strategy of FL is based not only on Dis-similarity, but also integrates power similarity (hereinafter referred to as “RSRP+Dis”). This is motivated by the insights drawn from BS association rules prevalent in cellular networks. When the received signal power of two neighboring users closely matches the RSRP [37], it suggests a strong likelihood that the two signals of neighboring users have experienced similar channel fading; hence, sharing comparable channel fadings.

Consequently, the weight under the second transfer learning strategy can be formalized as

$$\alpha_m'' = \begin{cases} \left\{ 1 - \frac{|P_m - P_{center}|}{2\Delta} \right\} \alpha_m', P_{center} - \Delta < P_m < P_{center} + \Delta \\ 0, \text{ otherwise} \end{cases}, \quad (13)$$

where  $P_m$  represents the power of the channel sample at UE  $m$ , while  $P_{center}$  represents the predetermined central reference power level at the center user,  $\Delta$  represents the predefined range of comparable power levels [36]. Finally, replacing  $\alpha_m''$  with  $\alpha_m$  in (9) results in the “FL+RSRP+Dis” strategy.

### C. HFL Enabled Channel Estimation

In the above application of the FL algorithm, the CNs and users’ devices share the same neural network structure as well as data properties, and are therefore referred to as homogeneous FL; whereas in reality, the local user generally has very limited computational and power capacity and is not capable of handling neural network training with high computational complexity and power consumption. To address that, a HFL-based approach is proposed that reduces the size of local neural network models for individual users, leading to a substantial decrease in computational costs and power consumption during the training process compared to homogeneous FL. The algorithm for HFL is described in Figure 3.

The whole channel estimation process is executed iteratively, with each round including 6 steps:

*Step 1- Global model broadcast:* A central/edge server or node leverages its superior computing capabilities to provide a powerful global model. A global model  $W_g$  is initialized at the central server, which is larger than the users’ private neural network models. Within the heterogeneous FL framework, the global model  $W_g$  is decomposed into the shared part  $W_g^1$  and distillation part  $W_g^2$ , as shown in Figure 4.  $W_g$  is distilled into a small neural network  $W_f$  that is broadcast to and used by local users. Note that  $W_f$  is composed of  $W_g^1$  and  $W_g^2$  as

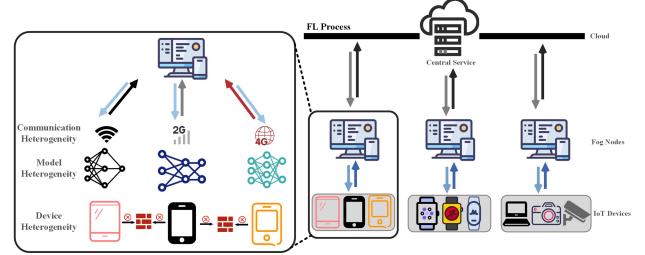


Fig. 2: The concept of heterogeneous federated learning networks.

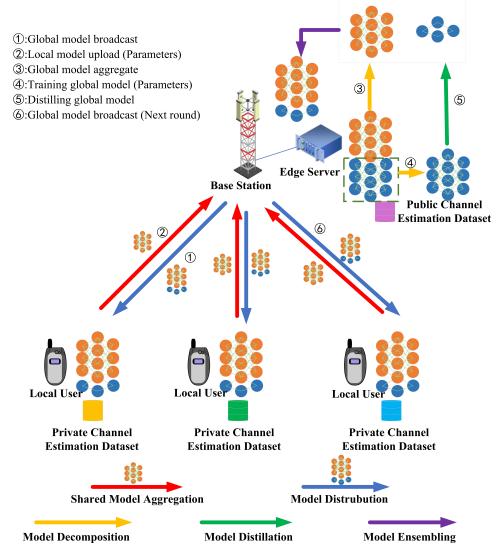


Fig. 3: Aggregation process of heterogeneous federated learning model.

depicted in Figure 4. The structure of  $W_g^1$  is identical to that of  $W_f^1$ , but with different parameters.

*Step 2- Local model upload:* After receiving  $W_f$ , each local users train their individual local models to minimize MSE (i.e., update  $W_f$  independently<sup>1</sup>) by utilizing their respective data set collected by themselves. Finally, the local user  $l$ ’s private local model is obtained, denoted by  $W_l$ , which can be decomposed into the shared part  $W_l^1$  and distillation part  $W_l^2$ .

*Step 3- Shared part aggregating of the global model:* After user  $l$  completes training its local model, the user proceeds to transmit the parameters of the shared part  $W_l^1$  through the dedicated channel. The central server first aggregates the parameters of the shared part based on the average criterion and generates the shared part parameters of the global model, that is

$$W_g^1 = (\sum_{l=1}^L \beta_l W_l^1)/L, \quad (14)$$

where  $L$  is the number of users,  $\beta_l$  is the weight of  $l$ -th model parameter in the global model aggregation.

*Step 4- Distillation part training of the global model:* The central server proceeds to train the distillation part  $W_g^2$  through part of the data set. Note that to reduce the user’s downlink data privacy, we use the data collected by the edge

<sup>1</sup>Note that though we assume all users share the same neural network size, it can be easily extended to situations where users have different model size.

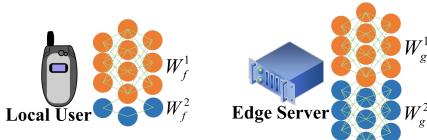


Fig. 4: Composition of models.

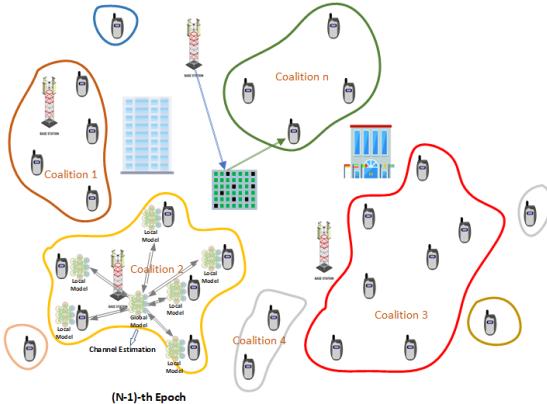


Fig. 5: Coalition formation diagram for channel estimation in RIS-assisted cell-free MIMO networks.

server. Specifically, we consider the reciprocity of uplink and downlink channels [38]. During the uplink transmission phase, the data uploaded by users to BS are stored as training data samples for this phase. By estimating the downlink channel and leveraging channel reciprocity, we obtain the uplink channel information.

After Steps 3 and 4, a global model is generated. Different from traditional homogenous FL approach, we first distill the global model before broadcasting it to the local users.

*Step 5- Global model distilling:* Specifically, local user model  $W_f$  is obtained by distilling  $W_g$ . The network structure of  $W_f$  is identical to that of the user's model.  $W_f^1$  shares the same model parameters as  $W_g^1$ , while the parameters of the distillation part  $W_f^2$  is initialized independently. In the training process, the data set is utilized, and only the distillation part, i.e.,  $W_f^2$  is iteratively updated till convergence is achieved or the maximum epoch is reached. Then, model  $W_f$ , which has the same network structure as the user's local model, is finally obtained.

*Step 6- Global model broadcast:* Next round of HFL starts. The central server broadcasts  $W_f$  to the users for the subsequent round of channel estimation training. Note that after training, the server uses the global model  $W_g$ , while the local users utilize the distilled models  $W_f$  for the HFL process.

**Remark 1.** Under the HFL framework, the size of local neural network models for individual users is reduced, leading to a substantial decrease in computational and power costs during the training process, compared to homogeneous FL. While HFL results in a slight reduction in accuracy, it significantly lowers computational costs (measured in flops) and complexity. Additionally, under the homogeneous FL framework, only a portion of the model parameters is transmitted during the training process. This approach reduces the quantity of data exchanged; thereby, enhancing privacy protection for

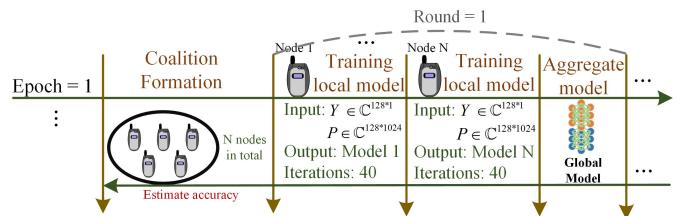


Fig. 6: CFFL for channel estimation in RIS-assisted cell-free MIMO networks.

individual users.

#### D. Coalition Formation-enabled FL Method

To enhance channel estimation accuracy in FL frameworks, whether homogeneous or heterogeneous, optimizing the user grouping strategy is an effective approach. This optimization facilitates the construction of a high-quality aggregation neural network; thus, improving channel estimation accuracy. In this context, a coalition is a FL user group. The FL user grouping problem is converted into a coalition formation problem. Accordingly, a CFFL method is documented to improve the accuracy of the HFL approach. The details are described as follows. First, the initial coalition combinations are determined using a reinforcement learning algorithm. Each user employs its local dataset to perform DNN-based channel estimation, generating a locally trained model. The model parameters from local users are then transmitted to the central node of each coalition via the uplink channels. Subsequently, the central node performs federated learning model aggregation and distributes the aggregated and distilled model to each user within the coalition. The global utility of the current coalition combination is evaluated by averaging the normalized mean square error (NMSE) across all users within the coalition. This NMSE value serves as a reward for the reinforcement learning algorithm, which updates the coalition combinations iteratively. This process is repeated until convergence. In what follows, we present two reinforcement learning approaches of coalition formation, namely i) DQN-enabled coalition formation, and ii) Qmix-enabled coalition formation.

##### 1) DQN-enabled Coalition Formation

In this section, DQN is utilized to solve the FL coalition formation problem. The flow diagram, which is presented in Figure 6 illustrates the initial round of the first epoch in CFFL. Following this, a total of  $T$  epochs and each epoches consists of  $E$  rounds are executed. Then we introduce the state, action, and reward of the DQN in the CFFL framework as follows.

- *State :*  $S_t = \{C_t, D_t\}$  is defined as the state information of the coalition combination at  $t$ -th epoch, in which  $C_t = \{c_{1,t}, \dots, c_{K,t}\} \in \mathbb{R}^{1 \times K}$  (with  $c_{k,t} = 0, 1, \dots, J$ ) is the current coalition choice of  $K$  users,  $c_{k,t} = 0$  represents the  $k$ -th user who does not join any coalition and trains on its own,  $J$  denotes the maximum number of coalitions.  $D_t \in \mathbb{R}^{1 \times J}$  is the set of number of users in  $J$  coalitions.
- *Action :*  $A_t = \{C_t\}$  is the coalition choice of all users.
- *Reward :*  $R_t = 1 - \bar{e}$ , in which  $\bar{e}$  represents the average channel estimation accuracy of all users after one epoch of CFFL.

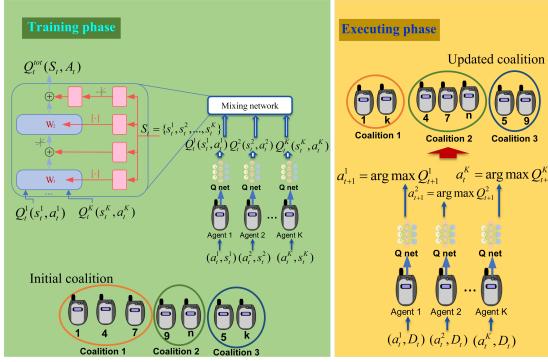


Fig. 7: Qmix-enabled coalitions formation.

### 2) Qmix-enabled CFFL

DQN is a centralized algorithm that constructs a large joint action-value mapping function from global observations. However, it may lead to a rapid increase in complexity as the number of users grows, while failing to satisfy privacy-preserving and communication-efficient requirements. Alternatively, distributed frameworks can be applied, wherein each agent has its local observation and makes decisions independently. Qmix is a distributed reinforcement learning algorithm, where each agent has its individual value function. These individual value functions are then combined into a global value function for the entire team through a centralized mixing network. Correspondingly, the distributed framework facilitates autonomy and collaboration among the users in the coalition, where each user can make decisions based on local information, while also taking into account the interests of the whole coalition with the help of the mixing network.

In Qmix architecture, each user can only achieve the observation of which coalition it joins and has no information on the decisions of other users. This design eliminates the need for information exchange, which consequently reduce communication costs associated with tracking other users' coalition choices. In the training process, the coalition is initialized. Based on their current action and the observed state  $S$ , including  $\{C, D\}$ , each user calculates the Q-value, as shown in Figure 7. Subsequently, the Q-values obtained by each user along with all the information about the observed environment will be uniformly input into the mixing network. The network is trained to generate the global Q-value as well as the Q-value weights for each user. Each user's Q network is iteratively updated to maximize the global Q value until convergence or the end of training.

During the execution phase, users don't need to go through the mixing network for centralized execution; they can execute actions through their own Q network by using their current action and the information concerning the number of users in the coalition.

### 3) FL User Coalition Formation Game

The traditional coalition formation game divides the set of players into several disjoint sets through the cooperation between users, and the players of the game pursue the maximization of their own interests. In this section, the participating UEs in channel estimation are game players, forming distinct coalitions, where members within each coalition engage in

model aggregation through federated learning. According to the previous analysis, we configure the utility of the entire network as the aggregate sum of utilities across all coalitions. The coalition game model can be represented as

$$\mathcal{G} = \{\mathcal{K}, U(S), S\}, \quad (15)$$

where  $\mathcal{K}$  is the set of UEs participating in the game for channel estimation,  $S = \{S_1, \dots, S_J\}$  represents the coalition structure formed among these UEs.  $c_{k,t} = j$  means that, at  $t$ -th epoch, UE  $k$  belongs to coalition  $S_j$ , and  $U(S_j)$  is the utility of coalition  $j$ .

The utility all coalitions is formulated as

$$U = \sum_{S_j \in S} U(S_j), \quad (16)$$

where  $U(S_j) = A - \sum_{p=1}^P e_{j,p}$  is the utility of coalition  $j$ ,  $A$  is a positive constant introduced to ensure non-negativity of utilities, and  $e_{j,p}$  represents the normalized mean squared error between the estimated channel for UE  $p$  within coalition  $j$  and the true channel. The utility assigned to the  $p$ -th user in the  $j$ -th coalition, denoted by  $u(a_{j,p})$ , is given by

$$u(a_{j,p}) = \frac{|a_{j,p}|}{\sum_{q \in S_j} |a_{j,q}|} U(S_j), \quad (17)$$

where  $|a_{j,p}|$  represents the magnitude of data volume contributed by UE  $a_{j,p}$ .

### 4) Analysis of Nash Equilibrium (NE)

The design of a preference order that ensures a stable coalition structure while maximizing the global utility is crucial for guiding coalition formation. The preference relationship of altruistic criteria between coalition  $S_1$  and coalition  $S_2$  is formally defined in **Definition 1**.

**Definition 1.** For any UE  $k \in \mathcal{K}$ , and two coalitions  $S_1 \subseteq S$  and  $S_2 \subseteq S$ , the altruistic criterion is defined as

$$\begin{aligned} S_1 \succ_j S_2 \Leftrightarrow & u_j(S_1) + \sum_{k \in S_1 \setminus \{j\}} u_k(S_1) + \sum_{k \in S_2 \setminus \{j\}} u_k(S_2 \setminus \{j\}) \\ & > u_j(S_2) + \sum_{k \in S_1 \setminus \{j\}} u_k(S_1 \setminus \{j\}) + \sum_{k \in S_2 \setminus \{j\}} u_k(S_2). \end{aligned} \quad (18)$$

It means that when UE  $k$  joins a new coalition, the utilities of both the newly formed coalition, and the original coalition to which UE  $k$  previously belonged, are improved.

**Theorem 1.** There is at least one stable coalition structure under the altruistic criterion, and the optimal deconstruction of the utility maximization problem of the whole network is a stable coalition structure.

*Proof:* Refer to Appendix A. ■

## IV. NUMERICAL RESULTS AND PERFORMANCE ANALYSES

In the simulation, we consider a scenario comprising 4 BSs, 1 RIS, and 10 UEs. Each BS is equipped with  $N_t = 16 = 4 \times 4$  antennas, while the RIS consists of  $N = 64 = 8 \times 8$  elements. All channel data utilized in this analysis were extracted from the DeepMIMO dataset [39]. The two-dimensional geographical distribution of the 10 UEs is shown in Figure 8. The white box represents the distributed buildings along the streets. The



Fig. 8: Sparse distribution scenario: two-dimensional geographical distribution of 10 users.

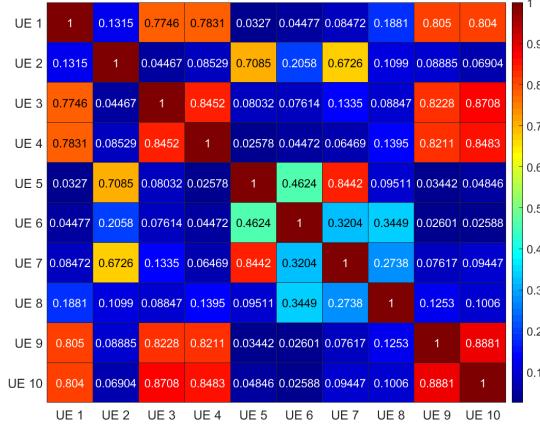


Fig. 9: Channel correlation coefficient with 10 sparse users.

main street (the horizontal one) is 600 m long and 40 m wide, the cross street (the vertical one) is 440 m long and 40 m wide. The two streets have buildings on both sides, the height of every building is marked in the figure.

Along the main street, all the buildings have bases of the same dimensions, 30 m  $\times$  60 m and the buildings have bases of 60 m  $\times$  60 m in the second street. In the HFL framework, the model of the center node has 5 convolutional layers as well as 1 fully connected layer, while the model of the edge nodes has only 3 convolutional layers and 1 fully connected layer. For the sake of fairness, the pilot signal length used for MMSE is set to 1024, whereas the proposed algorithm employs a pilot signal length of 128. Some of the parameters of this section are given in Table II.

TABLE II: Parameter Settings

Parameters	Value	Parameters	Value
Number of BSs $B$	4	Number of UEs $K$	10
Number of RISs $R$	1	Number of BS antennas $N_t$	$4 \times 4 = 16$
Number of RIS elements $N$	$8 \times 8 = 64$	Number of samples per user	4000
Batch size for training	64	Training test ratio	9
Learning rate	$1e^{-3}$	Epoch	6000
Maximum number of coalitions	3	System Bandwidth	100MHz

#### A. Impact of user channel correlation on the resulting formed FL user coalition

To verify the rationality of the resulting coalition formation, we investigate the relationship between the formed coalition and user channel correlation. Without loss of generality, we



Fig. 10: Dense distribution scenario: two-dimensional geographical distribution of 10 users.

consider 10 UEs under two scenarios: sparse and dense distributions, as illustrated in Figure 8 and Figure 10, respectively. Without loss of generality, the maximum number of coalitions is set to 3.

For the sparse scenario, the user channel correlation coefficients are shown in Figure 9. Of note the channel correlations among users vary, motivating the grouping of users with strong channel correlation to improve the efficiency of federated learning. Applying the proposed algorithm, the resulting user actions (i.e., coalition choices) are  $A_t = [1, 0, 3, 3, 2, 0, 2, 0, 1, 1]$ , where 0 indicates that the user does not join the coalition, and other numbers indicate which coalition the users joins. As can be seen in Figure 8, though being geographically neighboring nodes, 2 users (e.g., the 5th and 6th users) are not necessarily placed in the same coalition. This is reasonable, as indicated in Figure 9, where the channel correlation between the 5th and 6th users is weak. In contrast, users 1, 9, and 10 are grouped in the same coalition due to their high channel correlation, as illustrated in Figure 9.

A similar conclusion can be achieved for the dense distribution scenario. As shown in Figure 10, the actions of all users are  $A_t = [3, 3, 0, 2, 0, 2, 1, 0, 1, 2]$ . For instance, the channel correlation between User 2 and User 5 is weak, and they do not choose the same coalition. In addition, User 2 prefers to join the same coalition as User 1 rather than User 10, as the channel correlation between User 2 and User 1 is stronger. This can be observed from the first row of Figure 11. The above analysis highlights the significant impact of channel correlation on coalition formation and algorithm performance, thereby demonstrating the validity of the proposed algorithm.

#### B. Channel estimation accuracy comparison

The performance comparisons between the proposed scheme and several benchmark algorithms are illustrated in Figure 12. Note that to show the performance gains brought by coalition formation, except “DQN+FL” and “Qmix+HFL”, all other algorithms in Figure 12 assume that all users form a single large FL group rather than being divided into multiple coalitions. The following conclusions can be achieved. 1) Impact of SNR: It is shown that when the SNR increases, all algorithms achieve higher estimation accuracy. 2) NMSE

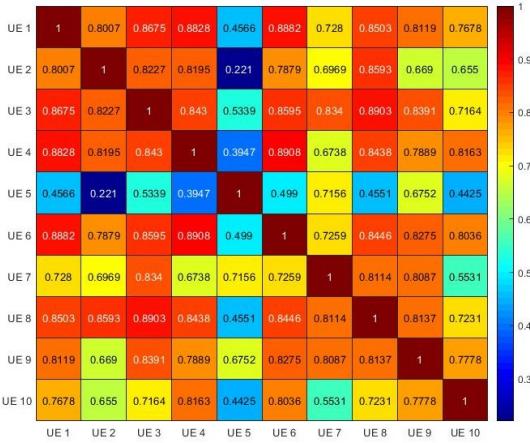


Fig. 11: Channel correlation coefficient with 10 intensive users.

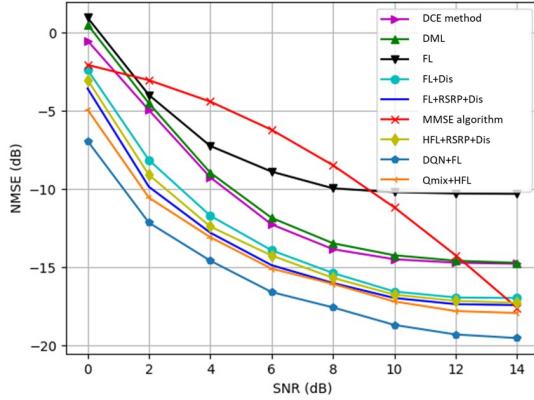


Fig. 12: NMSE versus SNR under different algorithms.

gains from transfer learning: Compared with the “DML”, conventional “FL”, “MMSE algorithm”, and “DCE method”, the “FL+Dis” and “FL+RSRP+Dis” improve channel estimation accuracy by 33% ~ 50%. Therefore, we conclude the integration of transfer learning significantly enhances NMSE performance. This is because the transferred model has already captured a portion of the underlying channel features, and subsequent training further refines the model, enhancing estimation accuracy. Besides, it can be seen that “FL+RSRP+Dis” outperforms “FL+Dis” in channel estimation accuracy. This is because, compared to strategies that consider only distance similarity, “FL+RSRP+Dis” integrates both RSRP and distance similarity enables the usage of more accurate similar channel estimation neural network parameters for transfer, as illustrated in (13) and (9). 3) HFL-based Algorithm: The HFL-based algorithm (“HFL+RSRP+Dis”) outperforms other benchmark methods by at least 5% but slightly underperforms compared to homogeneous federated learning (“FL+RSRP+Dis”) by 3%. This is due to the slight reduction in feature extraction capability in the distilled small model compared to the original large model. However, model distillation significantly reduces computational and hardware requirements for users (detailed numerical analyses are presented in Section IV-G). 4) Impact of reinforcement learning: The introduction of reinforcement learning algorithms improves federated learning performance. While “Qmix+HFL”,

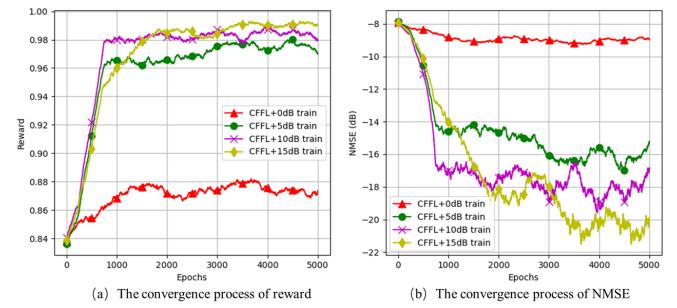


Fig. 13: The performance of DQN-based algorithm at different SNR vs epochs.

a distributed decision-making algorithm, shows slight performance degradation compared to the “DQN+FL”, its NMSE performance ranks just below that of “DQN+FL”. Additionally, in the SNR range of 0 to 2 dB, Qmix+HFL achieves approximately a 30% improvement in NMSE compared to “HFL+RSRP+Dis”.

### C. Impact of training SNR on performance

Figure 13(a) depicts the convergence process of the DQN-enabled federated learning, where the training SNR (i.e., the SNR of the sample data) is set to 0 dB, 5 dB, 10 dB, and 15 dB, respectively. It is seen that rewards improve as SNR increases, which is reasonable. Figure 13(b) depicts how NMSE varies with epochs during DQN training. The performance of CFFL is much better compared to “FL+RSRP+Dis” in Figure 12. For example, when SNR= 5 dB, the NMSE achieved by the proposed algorithm is around -16 dB. Notably, compared with “FL+RSRP+Dis” and “FL+Dis” (all users form one large FL group, instead of being divided into different groups), our proposed CFFL brings NMSE gains of 18% and 28%, respectively. This is because with the reinforcement learning method, the coalition can more reasonably divide the users into different FL groups. In other words, users experiencing similar channel fading are grouped into a coalition; thus, enhancing the effectiveness of federated learning. Consequently, this results in a more accurate global model for each FL group and improves channel estimation accuracy.

### D. Performance comparison of HFL and FL, DQN and Qmix

Figure 14 compares the channel estimation performance resulting from the formation of federated learning user coalitions guided by DQN and Qmix, as well as by homogeneous and heterogeneous federated learning methods. We set SNR= 5 dB. The performance of federated learning based on Qmix, i.e., “Qmix+FL”, is about 97.5% of that achieved with DQN, indicating a slight decrease in reward. This is because Qmix, as a distributed algorithm, only utilizes its local observation information and its own Q-network in the neural network executing phase, leading to a performance degradation compared to the centralized DQN model. Meanwhile, the performance of HFL is about 96.8% of that of homogeneous federated learning algorithms. However, as aforementioned, a slight decline in channel estimation performance is compensated by significant reduction in information exchange, local user model complexity, and an improvement in user privacy security.

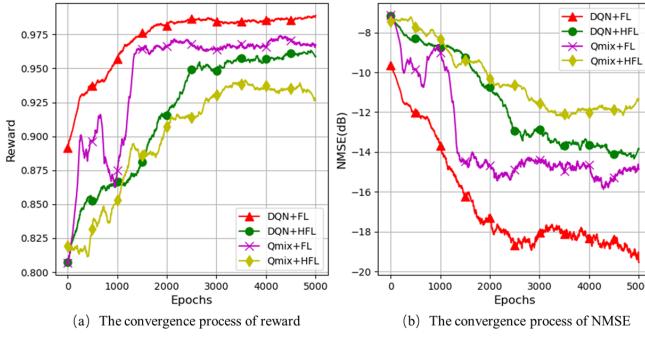


Fig. 14: Reward vs Epochs under different algorithms.

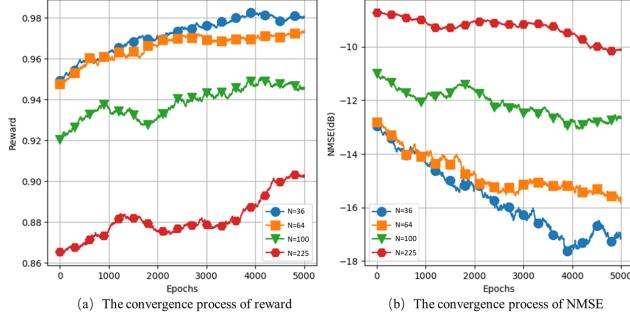


Fig. 15: The performance of DQN-based algorithm at different sizes of RIS vs epochs.

#### E. Impact of RIS size on performance

Figure 15 illustrates the performance of the DQN-based algorithm as the RIS size varies. The figure shows that the accuracy of channel estimation decreases with an increasing number of RIS elements. For example, at epoch 4000, the NMSE decreases by 11.4%, 28%, and 45.1% for RIS numbers of 64, 100, and 225, respectively, compared to a element number of 36. This decline occurs because changes in RIS size do not affect the signal dimensionality at the users, while the cascaded channel matrix grows larger in dimension. Given that the dimensionality of the received signal, the amount of sample data, and the network structure remain the same, inferring cascaded channels of larger dimensions (i.e., network outputs) becomes more challenging, leading to a degradation in performance.

#### F. Impact of neural network size on the channel estimation accuracy

Figure 16 shows how the number of neural network layer affects the channel estimation accuracy. Take the heterogeneous Qmix algorithm as an example. The results indicate that as the number of layers in the heterogeneous network decreases, channel estimation accuracy also declines. For instance, at epoch 5000, the NMSE decreases by 14.1% for convolutional layer numbers of 4 (i.e., “Qmix+HFL+4conv”) and by 25% for a layer number of 3 (i.e., “Qmix+HFL+3conv”), compared to a convolutional layer number of 5 (i.e., “Qmix+FL+5conv”). It is because of the reduced inference capability due to the decrease of the neural network layer. Therefore, it is important to select an appropriate neural network layer by balancing estimation accuracy with the hardware capabilities of the deployed devices.

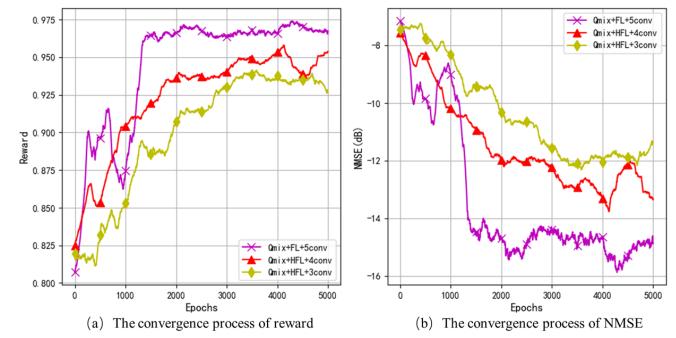


Fig. 16: The performance of Qmix-based algorithm at different network layer vs epochs.

TABLE III: Model complexity comparison between HFL and homogeneous FL. The number of users is 10. The model of the GPU is RTX 1660S.

Network	FL	HFL	Improvement
Total parameter	8,428,416 (5 convolutional layers and 1 fully connected layer)	8,409,856 (3 convolutional layers and 1 fully connected layer)	0.22%
Flops	0.84882272G (1 big local model)	0.695730176G (1 distilled local model)	18.04%
Total Flops	9.337044992G (1 big global model and 10 big local models)	7.806124033G (1 big global model and 10 small local models)	16.40%
Time for 1 Epoch (s)	1.05349256	0.893589525	15.18%

#### G. Model complexity comparison between FL and HFL

Model complexity is compared between HFL and homogeneous federated learning. The time taken to complete each episode is averaged over 10 episodes. In HFL, users generate a small (distilled) local model, while the central server utilizes a larger global model. In contrast, homogeneous FL employs large neural network models for both users and the central server, resulting in significant differences in model complexity. As shown in Table III, HFL reduces computational overhead by approximately 16%.

#### H. Communication overhead comparison

The comparison of communication interactions among the four algorithms of CFFL is listed in Table IV. Specifically,  $K$  denotes the number of users,  $W_{FL}$  denotes the number of model parameters to be transmitted by a single user in homogeneous FL,  $W_{HFL}$  denotes the number of model parameters to be transmitted by a single user in HFL. Consequently, the amount of interactions for DQN-based FL in a round is  $2KW_{FL}$ . Let  $E$  denote the number of rounds to be performed in each epoch,  $O$  represent the amount of environmental information observed by the user,  $A$  indicate the amount of action information transmitted by the user, and  $P$  denote the policy information provided by the executive network. Then, the total amount of communication interactions for DQN-FL used in an epoch is given by  $E(2KW_{FL}) + K[(O + A) + P]$ . Let  $T_1, T_2, T_3, T_4$  represent the number of converged epochs for “DQN+FL”, “DQN+HFL”, “Qmix+FL”, “Qmix+HFL”, respectively. Then the total amount of communication interactions for DQN-FL is given by  $T_1 \cdot \{E(2KW_{FL}) + K[(O + A) + P]\}$ . The total communication interactions for the other algorithms can be calculated in a similar manner. It can be observed from Section III-C that the amount of communication interaction is primarily determined by the number of model parameters,

TABLE IV: Comparison of communication interactions

	<b>One Round of FL</b>	<b>One Epoch of CFFL</b>	<b>The total amount of interaction data required until convergence</b>
DQN + FL	$2KW_{FL}$	$E(2KW_{FL}) + K[(O + A) + P]$	$T_1 \{E(2KW_{FL}) + K[(O + A) + P]\}$
DQN + HFL	$2KW_{HFL}$	$E(2KW_{HFL}) + K[(O + A) + P]$	$T_2 \{E(2KW_{HFL}) + K[(O + A) + P]\}$
Qmix + FL	$2KW_{FL}$	$E(2KW_{FL}) + K[(O + (C_{max} + 1)) + (C_{max} + 1)]$	$T_3 \{E(2KW_{FL}) + K[(O + (C_{max} + 1)) + (C_{max} + 1)]\}$
Qmix + HFL	$2KW_{HFL}$	$E(2KW_{HFL}) + K[(O + (C_{max} + 1)) + (C_{max} + 1)]$	$T_4 \{E(2KW_{HFL}) + K[(O + (C_{max} + 1)) + (C_{max} + 1)]\}$

specifically  $W_{HFL}$  and  $W_{FL}$ . The difference in the number of parameters between FL and HFL in one round is given by  $2KW_{FL} - 2KW_{HFL}$ , where  $K = 10$ ,  $W_{FL} = 8.428M$ ,  $W_{HFL} = 8.409M$  under the scenario settings. For the difference in communication interactions between DQN and Qmix, it mainly depends on the dimensions of the action size that needs to be determined and the dimension of the vector executing the policy, specifically  $A = (C_{max} + 1)K = 40$ ,  $P = 40$ , where  $C_{max} = 3$  is the maximum number of coalitions. This difference becomes larger as the number of users increases. Thus, the difference in the amount of algorithmic communication interactions becomes significantly large after  $E$  rounds and  $T$  epochs of iterations.

## V. CONCLUSION

A coalition formation-guided HFL channel estimation method was proposed. In particular, we utilized Qmix to implement distributed reinforcement learning for coalition formation, which enhancing the grouping of FL users. Additionally, Qmix facilitates autonomy and cooperation among users in the coalition, where each user can independently make decisions based on local information, while considering the interests of the entire coalition. This approach significantly reduces the overhead associated with information exchange compared to the centralized DQN method. Furthermore, the HFL-based approach was proposed to reduce the size of local neural network models for individual users. Numerical results showed that although HFL slightly reduces channel estimation accuracy, it significantly lowers computational cost and complexity, and reduces training duration by approximately 15%. More importantly, HFL transmits only a subset of model parameters, reducing the amount of data transmitted and better protecting local user privacy compared to homogeneous FL. Simulations revealed that, proposed method reduces the end-user computational overhead by 16%, improves data privacy, and enhances channel estimation accuracy by 20%. Of note in our context, the HFL (without coalition formation) and the DQN-enabled homogeneous FL serve as the lower and upper performance bounds, respectively. The Qmix-enabled algorithm outperforms the lower performance bound baseline by 25% while maintaining an accuracy loss of less than 4% compared to the upper-performance bound.

## APPENDIX A

Considering the altruistic cooperation among UEs, the utility function of UE  $j$  is defined as

$$U = u(a_{m,j}) + \sum_{i \in S_m \setminus \{j\}} u(a_{m,i}) + \sum_{i \in S_n} u(a_{n,i}), \quad (19)$$

where  $S_m$  represents the coalition that UE  $j$  leaves,  $S_n$  represents the new coalition that UE  $j$  chooses to join,  $S_m \setminus \{j\}$

represents the coalition  $S_m$  after removing UE  $j$ . The potential function is defined as

$$\varphi = \sum_{j \in S_k, S_k \in S} u(a_{k,j}). \quad (20)$$

If the UE unilaterally changes its coalition choice from  $S_m$  to  $S_n$ , then the UE utility function changes by

$$\begin{aligned} U' - U &= \left\{ u'(a_{n,j}) + \sum_{i \in S'_n \setminus \{j\}} u'(a_{n,i}) + \sum_{i \in S'_m} u'(a_{m,i}) \right\} \\ &\quad - \left\{ u(a_{m,j}) + \sum_{i \in S_m \setminus \{j\}} u(a_{m,i}) + \sum_{i \in S_n} u(a_{n,i}) \right\} \\ &= \underbrace{\{u'(a_{n,j}) - u(a_{m,j})\}}_{\textcircled{1}} + \underbrace{\left\{ \sum_{i \in S'_n \setminus \{j\}} u'(a_{n,i}) - \sum_{i \in S_n} u(a_{n,i}) \right\}}_{\textcircled{2}} \\ &\quad + \underbrace{\left\{ \sum_{i \in S'_m} u'(a_{m,i}) - \sum_{i \in S_m \setminus \{j\}} u(a_{m,i}) \right\}}_{\textcircled{3}} \end{aligned} \quad (21)$$

where  $S'_m$  represents the coalition that UE  $j$  leaves,  $S'_n$  represents the new coalition that UE  $j$  chooses to join.

On the other hand, since the UE unilaterally changes its coalition selection, the change in the potential function can be calculated as

$$\begin{aligned} \varphi' - \varphi &= \left\{ \sum_{j \in S_k, S_k \in S'} u'(a_{k,j}) \right\} - \left\{ \sum_{j \in S_k, S_k \in S} u(a_{k,j}) \right\} \\ &= \underbrace{\{u'(a_{n,j}) - u(a_{m,j})\}}_{\textcircled{4}} + \underbrace{\left\{ \sum_{i \in S'_n \setminus \{j\}} u'(a_{n,i}) - \sum_{i \in S_n} u(a_{n,i}) \right\}}_{\textcircled{5}} \\ &\quad + \underbrace{\left\{ \sum_{i \in S'_m} u'(a_{m,i}) - \sum_{i \in S_m \setminus \{j\}} u(a_{m,i}) \right\}}_{\textcircled{6}} \\ &\quad + \underbrace{\left\{ \sum_{i \in S_k, S_k \in S' \setminus \{S'_n, S'_m\}} u'(a_{k,i}) - \sum_{i \in S_k, S_k \in S \setminus \{S_n, S_m\}} u(a_{k,i}) \right\}}_{\textcircled{7}} \end{aligned} \quad (22)$$

Since UE  $j$  unilaterally changes its coalition choice, it will only affect the coalition from which UE  $j$  is departing and to which it is joining, namely  $S_m, S_n, S'_m, S'_n$ . Therefore,  $\textcircled{7}$  in (22) is 0. In addition,  $\textcircled{1} = \textcircled{4}$ ,  $\textcircled{2} = \textcircled{5}$ ,  $\textcircled{3} = \textcircled{6}$ . Hence, we have

$$\varphi' - \varphi = U' - U. \quad (23)$$

(23) indicates that when UE  $j$  ( $\forall j$ ) independently changes its coalition selection, the variation in the utility function is equal to that of the potential function. This implies that the coalition formation game is an exact potential game, which indicates that there is at least one pure strategy Nash equilibrium [40]. Consequently, no UE can improve its utility by unilaterally changing its coalition. Since the potential function

and the objective function are positively and linearly related, maximizing the potential function is equivalent to maximizing the utility of the coalition formation of the whole network. This concludes the proof of **Theorem 1**.

## REFERENCES

- [1] H. Liu, N. Qi, X. Li, A. A. A. Boulogeorgos, T. A. Tsiftsis, M. Xiao, and J. Röning, "IRS channel estimation in cell-free MIMO network: A coalition formation guided federated learning approach," *IEEE Wireless Communications and Networking Conference (WCNC)*, 2025.
- [2] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74–80, 2014.
- [3] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 186–195, 2014.
- [4] E. Björnson, J. Hoydis, and L. Sanguinetti, "Massive MIMO networks: Spectral, energy, and hardware efficiency," *Found. Trends Signal Process.*, vol. 11, no. 3, pp. 154–655, 2017.
- [5] X. Zhang, H. Qi, X. Zhang, and L. Han, "Spectral efficiency improvement and power control optimization of massive mimo networks," *IEEE Access*, vol. 9, pp. 11 523–11 532, 2021.
- [6] H. Yang, Z. Xiong, J. Zhao, D. Niyato, L. Xiao, and Q. Wu, "Deep reinforcement learning-based intelligent reflecting surface for secure wireless communications," *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 375–388, 2021.
- [7] E. Björnson, "A programmable wireless world with reconfigurable intelligent surfaces," July 2020. [Online]. Available: <https://www.youtube.com/watch?v=9cBn5pi9Ms>
- [8] X. Jindan, Y. Chau, H. Chongwen, H. Naveed Ul, G. C. Alexandropoulos, M. D. Renzo, and M. Debbah, "Reconfiguring wireless environments via intelligent surfaces for 6G: reflection, modulation, and security," *SCIENCE CHINA Information Sciences*, vol. 66, no. 3, pp. 130 304–2023.
- [9] K. Zhi, C. Pan, H. Ren, K. Wang, M. Elkashlan, M. D. Renzo, R. Schober, H. V. Poor, J. Wang, and L. Hanzo, "Two-timescale design for reconfigurable intelligent surface-aided massive MIMO systems with imperfect CSI," *IEEE Transactions on Information Theory*, vol. 69, no. 5, pp. 3001–3033, 2023.
- [10] T. Zhou, K. Xu, X. Xia, W. Xie, and J. Xu, "Achievable rate optimization for aerial intelligent reflecting surface-aided cell-free massive MIMO system," *IEEE Access*, vol. 9, pp. 3828–3837, 2021.
- [11] K. Wang, N. Qi, X. Guan, Q. Shi, M. Xiao, S. Jin, and K.-K. Wong, "Transmit/passive beamforming design for multi-IRS assisted cell-free MIMO networks," *IEEE Systems Journal*, vol. 17, no. 4, pp. 6282–6291, 2023.
- [12] T. Noh and J. Choi, "Cell-free MIMO systems powered by intelligent reflecting surfaces," *IEEE Communications Letters*, vol. 26, no. 5, pp. 1076–1080, 2022.
- [13] K. Wang, N. Qi, M. Xiao, S. Hu, and B. Sihlbom, "Fairness-aware location and hybrid beamforming optimization in IRSs-aided cell-free MIMO system," in *2023 IEEE/CIC International Conference on Communications in China (ICCC)*, 2023, pp. 1–6.
- [14] H. Liu, N. Qi, K. Wang, T. A. Tsiftsis, W. Wang, and Y. Liu, "Network deployment with energy efficiency optimization in IRS-assisted cell-free MIMO system," *Physical Communication*, vol. 63, p. 102287, 2024.
- [15] K. Wang, N. Qi, H. Liu, A.-A. A. Boulogeorgos, T. A. Tsiftsis, M. Xiao, and K.-K. Wong, "Reconfigurable intelligent surfaces aided energy efficiency maximization in cell-free networks," *IEEE Wireless Communications Letters*, vol. 13, no. 6, pp. 1596–1600, 2024.
- [16] D. Mishra and H. Johansson, "Channel estimation and low-complexity beamforming design for passive intelligent surface assisted MISO wireless energy transfer," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 4659–4663.
- [17] Z.-Q. He and X. Yuan, "Cascaded channel estimation for large intelligent metasurface assisted massive MIMO," *IEEE Wireless Communications Letters*, vol. 9, no. 2, pp. 210–214, 2020.
- [18] K. Ardash, S. Gherekhloo, A. L. F. de Almeida, and M. Haardt, "Trice: A channel estimation framework for RIS-aided millimeter-wave MIMO systems," *IEEE Signal Processing Letters*, vol. 28, pp. 513–517, 2021.
- [19] H. Feng, Y. Xu, and Y. Zhao, "Deep learning-based joint channel estimation and csi feedback for RIS-assisted communications," *IEEE Communications Letters*, vol. 28, no. 8, pp. 1860–1864, 2024.
- [20] M. B. Mashhadi and D. Gündüz, "Pruning the pilots: Deep learning-based pilot design and channel estimation for MIMO-OFDM systems," *IEEE Transactions on Wireless Communications*, vol. 20, no. 10, pp. 6315–6328, 2021.
- [21] S. Hu, X. Chen, W. Ni, E. Hossain, and X. Wang, "Distributed machine learning for wireless communication networks: Techniques, architectures, and applications," *IEEE Communications Surveys Tutorials*, vol. 23, no. 3, pp. 1458–1493, 2021.
- [22] L. Dai and X. Wei, "Distributed machine learning based downlink channel estimation for RIS assisted wireless communications," *IEEE Transactions on Communications*, vol. 70, no. 7, pp. 4900–4909, 2022.
- [23] A. M. Elbir and S. Coleri, "Federated learning for channel estimation in conventional and RIS-assisted massive MIMO," *IEEE Transactions on Wireless Communications*, vol. 21, no. 6, pp. 4255–4268, 2022.
- [24] W. Shen, Z. Qin, and A. Nallanathan, "Federated learning enabled channel estimation for RIS-aided multi-user wireless systems," in *2022 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2022, pp. 812–817.
- [25] B. Qiu, X. Chang, X. Li, H. Xiao, and Z. Zhang, "Federated learning-based channel estimation for RIS-aided communication systems," *IEEE Wireless Communications Letters*, vol. 13, no. 8, pp. 2130–2134, 2024.
- [26] F. Yu, W. Zhang, Z. Qin, Z. Xu, D. Wang, C. Liu, Z. Tian, and X. Chen, "Heterogeneous federated learning," 2022. [Online]. Available: <https://arxiv.org/abs/2008.06767>
- [27] Z. Zhu, J. Hong, and J. Zhou, "Data-free knowledge distillation for heterogeneous federated learning," vol. 139, pp. 12 878–12 889, 18–24 Jul 2021. [Online]. Available: <https://proceedings.mlr.press/v139/zhu21b.html>
- [28] L. Wu, S. Guo, Y.-X. Ding, Y. Zhan, and J. Zhang, "A coalition formation game approach for personalized federated learning," *ArXiv*, vol. abs/2202.02502, 2022. [Online]. Available: <https://api.semanticscholar.org/CorpusID:246634014>
- [29] S. Arisdakessian, O. A. Wahab, A. Mourad, and H. Otrok, "Coalitional federated learning: Improving communication and training on non-iid data with selfish clients," *IEEE Transactions on Services Computing*, vol. 16, no. 4, pp. 2462–2476, 2023.
- [30] Z. Wang, F. Yang, S. Yan, S. Memon, Z. Zhao, and C. Hu, "Joint design of coalition formation and semi-blind channel estimation in fog radio access networks," *China Communications*, vol. 16, no. 11, pp. 1–15, 2019.
- [31] M. Sadeghi and M. Erol-Kantarci, "Deep reinforcement learning based coalition formation for energy trading in smart grid," pp. 200–205, 2021.
- [32] W. Dai, A. Bidwai, and G. Sartoretti, "Dynamic coalition formation and routing for multirobot task allocation via reinforcement learning," pp. 16 567–16 573, 2024.
- [33] K. Xiong, T. Zhang, G. Cui, S. Wang, and L. Kong, "Coalition game of radar network for multitarget tracking via model-based multiagent reinforcement learning," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 59, no. 3, pp. 2123–2140, 2023.
- [34] C. Hu, L. Dai, T. Mir, Z. Gao, and J. Fang, "Super-resolution channel estimation for mmwave massive MIMO with hybrid precoding," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8954–8958, 2018.
- [35] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *International Conference on Artificial Intelligence and Statistics*, 2016.
- [36] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," 2018. [Online]. Available: <https://api.semanticscholar.org/CorpusID:51929263>
- [37] N. Gupta1, "5G Cell Selection Criteria," [https://www.telecomhall.net/t\\_5g-cell-selection-criteria/23860](https://www.telecomhall.net/t_5g-cell-selection-criteria/23860), Jun. 2023, (accessed Feb. 2024).
- [38] X. Hu, R. Zhang, and C. Zhong, "Semi-passive elements assisted channel estimation for intelligent reflecting surface-aided communications," *IEEE Transactions on Wireless Communications*, vol. 21, no. 2, pp. 1132–1142, 2022.
- [39] A. Alkhateeb, "Deepmimo: A generic deep learning dataset for millimeter wave and massive mimo applications," in *Proc. of Information Theory and Applications Workshop (ITA)*, San Diego, CA, 02 2019, pp. 1–8.
- [40] D. Monderer and L. S. Shapley, "Potential games," *Games and Economic Behavior*, vol. 14, no. 1, pp. 124–143, 1996.