Federated Learning Games for Reconfigurable Intelligent Surfaces via Causal Representations

Charbel Bou Chaaya, Sumudu Samarakoon, and Mehdi Bennis
Centre for Wireless Communications, University of Oulu, Finland
Emails: {charbel.bouchaaya, sumudu.samarakoon, mehdi.bennis}@oulu.fi

***Abstract*—In this paper, we investigate the problem of robust Reconfigurable Intelligent Surface (RIS) phase-shifts configuration over heterogeneous communication environments. The problem is formulated as a distributed learning problem over different environments in a Federated Learning (FL) setting. Equivalently, this corresponds to a game played between multiple RISs, as learning agents, in heterogeneous environments. Using Invariant Risk Minimization (IRM) and its FL equivalent, dubbed FL Games, we solve the RIS configuration problem by learning invariant causal representations across multiple environments and then predicting the phases. The solution corresponds to playing according to Best Response Dynamics (BRD) which yields the Nash Equilibrium of the FL game. The representation learner and the phase predictor are modeled by two neural networks, and their performance is validated via simulations against other benchmarks from the literature. Our results show that causality-based learning yields a predictor that is 15% more accurate in unseen Out-of-Distribution (OoD) environments.**

***Index Terms*—Reconfigurable Intelligent Surface (RIS), Federated Learning, Causal Inference, Invariant Learning.**

I. Introduction

The advent of Reconfigurable Intelligent Surfaces (RISs) will substantially boost the performance of wireless communication systems. These surfaces are manufactured by layering stacks of sheets made out of engineered materials, called meta-materials, built on a planar structure. The reflection coefficients of the meta-material elements, called meta-atoms, vary depending on their physical states. Thus, the direction of incident electromagnetic waves on RISs can be manipulated with the aid of simple integrated circuit controllers that modify meta-atoms' states. In this view, the RIS technology provides a partial control over the wireless propagation environment rendering improved spectral efficiency with a minimal power footprint [1]. RIS is considered a fundamental enabler to achieve the 6G vision of smart radio environments [2].

One of the major challenges in the RIS technology is the accurate tuning of RIS phases. To this extent, a vast majority of the existing literature on RIS-assisted communication relies on the use of Channel State Information (CSI) to train Machine Learning (ML) models that predict the optimal RIS configuration [3]–[5]. These methods seek either a centralized-controller driven approach, or a distributed multi-agent optimization technique, such as Federated Learning (FL). Other works such as [6], [7] exploit the users' locations to employ a location-based passive RIS beamforming. Either way, their main focus is to draw on the statistical correlations of the observed data, while overlooking the impacts of heterogeneous system designs (e.g., different RISs, propagation environments, users distribution, etc.). Moreover, these approaches produce a high inference accuracy within a fixed environment, from which the training and testing data are drawn. They subsequently fail to have a good Out-of-Distribution (OoD) generalization in unseen environments.

Although FL provides a learning framework where multiple agents train a collaborative model while preserving privacy, its state-of-the-art approach, such as Federated Averaging (FedAVG) [8], is known to perform poorly when the local data is non-identical across participating agents. This is due to the fact that FedAVG (and its variants) solve the distributed learning problem via Empirical Risk Minimization (ERM), that minimizes the empirical distribution of the local losses assuming that the data is identically distributed. To mitigate this issue, the authors in [9] leveraged Distributionally Robust Optimization (DRO) [10] and proposed a federated DRO for RIS configurations. Therein, the distributed learning problem is cast as a minimax problem, where the model's parameters are updated to minimize the maximal weighted combination of local losses. This ensures a good performance for the aggregated model over the worst-case combination of local distributions.

On the other hand, [11] used Invariant Risk Minimization (IRM) [12] to formulate the problem of learning optimal RIS phase-shifts. The aim of IRM is to capture causal dependencies in the data, that are invariant across different environments. In [11], the authors empirically showed that using relative distances and angles between the RIS and the transmitter/receiver as causal representations for the CSI, improves the robustness of the RIS phase predictor. However, these representations were not learned by the configuration predictor, but were predefined and fixed. Also, this solution assumes that multiple environments are known to the predictor beforehand, which is an unfeasible assumption.

The main contribution of this paper is a novel distributed IRM-based solution to the RIS configuration problem. We cast the problem of RIS phase control as a federated learning problem with multiple RISs controllers defined over heterogeneous environments, using a game-theoretic reformulation of IRM,

This work was supported by Project Infotech R2D2 and the European Commission Horizon Europe SNS JU Projects DESIRE6G (GA 101096466), ADROIT6G (GA 101095363) and VERGE (GA 101096034).

979-8-3503-1090-0/23/$31.00 © 2023 IEEE

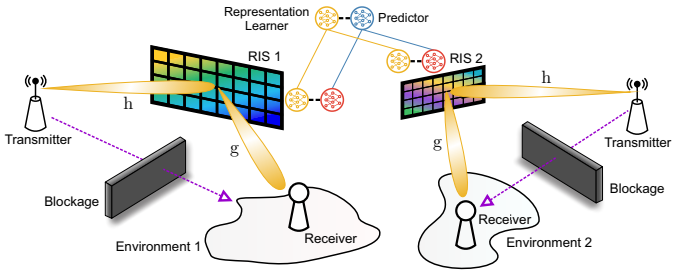


Fig. 1: System model illustrating different RIS-assisted communication scenarios. Each RIS is conceived differently from other RISs, and serves differently distributed receivers.

referred to as Federated Learning Games (FL GAMES) [13], [14]. The solution of this problem is proven to be the Nash Equilibrium of a strategic game where each RIS updates its configuration predictor by minimizing its local loss function. This game is indexed by a representation learner that is shared among the RISs to extract a causal representation from the CSI data. The representation learner and predictors are trained in a distributed and supervised learning manner. The numerical validations yield that the proposed design improves the accuracy of the predictor tested in OoD settings by 15% compared to state-of-the-art RIS designs.

The remainder of this paper is organized as follows. In Section II, we describe the system model and conventional approaches to the RIS configuration problem using FEDAVG. The FL GAMES solution that involves extracting causal representations from the data is discussed in Section III. Section IV presents the simulation results that compare the proposed algorithm with benchmarks. Concluding remarks are drawn in Section V.

II. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a set of environments $\mathcal{R}$ where each environment $r \in \mathcal{R}$ consists of a RIS-assisted downlink communication between a transmitter (Tx) and a receiver (Rx) as shown in Fig. 1. Both the Tx and the Rx are equipped with a single antenna each, and we assume that the direct link between them is blocked in which, the channel of direct link is dominated by the reflected channel. The RIS in environment r is composed by $N^r = N^r_{\mathrm{x}} N^r_{\mathrm{y}}$ reflective elements where N^r_{x} and N^r_{y} are the number of horizontal and vertical reflective elements, respectively. Additionally, the inter-element distances over horizontal and vertical axes are characterized by d^r_{x} and d^r_{y}. Each RIS element applies a phase shift on its incident signal and the reflected signals are aggregated at the Rx. Note that the location of the Tx is fixed while the location of Rx is arbitrary, which is sampled by a predefined distribution. The choices of the Rx distribution along with the parameters $(N^r_{\mathrm{x}}, N^r_{\mathrm{y}}, d^r_{\mathrm{x}}, d^r_{\mathrm{y}})$ collectively define the environment r. We assume that these environments are completely separate, i.e. each Rx receives only the signal reflected by the RIS in its corresponding environment.

A. Channel Model

For the notation simplicity, we have omitted the notion of environment during the discussion within this subsection. Let $\mathbf{g} \in \mathbb{C}^N$ be the channel between the RIS and the Rx, which is dominated by its line-of-sight (LoS) component. Hence, by denoting φ_r and ϑ_r as the azimuth and elevation angles-of-departure (AoD) from the RIS respectively, the channel is modeled as

$$\mathbf{g} = \sqrt{\alpha_r}\, \mathbf{a}_N\left(\varphi_r, \vartheta_r\right), \tag{1}$$

where α_r represents the path-loss. Additionally, we define the steering function:

$$\mathbf{a}_N\left(\varphi, \vartheta\right) = \left[e^{\frac{2\pi j}{\lambda}\Delta_1(\varphi,\vartheta)}, \cdots, e^{\frac{2\pi j}{\lambda}\Delta_N(\varphi,\vartheta)}\right]^{\mathsf{T}}, \tag{2}$$

and a set of operators for $n = 1, \ldots, N$:

$$\Delta_n\left(\varphi, \vartheta\right) = i_N(n) d_{\mathrm{x}} \cos(\vartheta)\sin(\varphi) + j_N(n) d_{\mathrm{y}} \sin(\vartheta), \tag{3}$$

$$i_N(n) = (n-1) \mod N_{\mathrm{x}}, \qquad j_N(n) = \left\lfloor \frac{n-1}{N_{\mathrm{x}}} \right\rfloor, \tag{4}$$

where λ is the wavelength, mod and $\lfloor\cdot\rfloor$ denote the modulus and floor operators. On the other hand, the channel $\mathbf{h} \in \mathbb{C}^N$ between the Tx and the RIS will have both LoS and non line-of-sight (NLoS) components. Therefore, we model $\mathbf{h}$ using Rician fading with spatial correlation, since the RIS elements are closely distanced. Accordingly, we have:

$$\mathbf{h} = \sqrt{\alpha_t}\left(\sqrt{\frac{\kappa}{1+\kappa}}\,\overline{\mathbf{h}} + \sqrt{\frac{1}{1+\kappa}}\,\widetilde{\mathbf{h}}\right), \tag{5}$$

where α_t and κ are the path-loss and the Rician coefficient respectively, $\overline{\mathbf{h}}$ is the LoS factor, and $\widetilde{\mathbf{h}}$ represents the small scale fading process in the NLoS component. Further, for the LoS link, the LoS factor is:

$$\overline{\mathbf{h}} = \mathbf{a}_N\left(\varphi_t, \vartheta_t\right), \tag{6}$$

where (φ_t, ϑ_t) are the angles-of-arrival (AoA) to the RIS. We model the NLoS link as $\widetilde{\mathbf{h}} \sim \mathcal{CN}\left(\mathbf{0}_N, \mathbf{R}\right)$, where $\mathbf{R}$ is a covariance matrix that captures the spatial correlation among the channels of the RIS elements. In the case of isotropic scattering in front of the RIS, a closed-form expression for $\mathbf{R}$ is [15, Proposition 1]:

$$[\mathbf{R}]_{m,n} = \mathrm{sinc}\left(\frac{2\left|\mathbf{u}_m - \mathbf{u}_n\right|}{\lambda}\right) \qquad m, n = 1, \ldots, N, \tag{7}$$

where $\mathbf{u}_n = \left[i_N(n)\, d_{\mathrm{x}}, j_N(n)\, d_{\mathrm{y}}\right]^{\mathsf{T}}$ represents the locations of the n^{th} element with i_N and j_N being the horizontal and vertical indices given in (4), and $\mathrm{sinc}(\cdot)$ is the normalized sampling function.

B. Downlink Rate Maximization

At every transmission slot, in each environment $r \in \mathcal{R}$, the RIS selects its phases in order to enhance the downlink rate at the Rx. Let $\boldsymbol{\theta}^r = [\theta^r_1, \ldots, \theta^r_N]^{\mathsf{T}}$ denote the phase decisions at the RIS. Thus, the received signal at the Rx is:

$$y_r = \left(\mathbf{h}_r^{\mathsf{H}} \boldsymbol{\Theta}_r\, \mathbf{g}_r\right) s_r + z_r, \tag{8}$$

where $\boldsymbol{\Theta}_r = \mathrm{diag}\left(e^{j\theta_1^r}, \ldots, e^{j\theta_N^r}\right)$ is the RIS reflection matrix, s_r is the transmitted signal satisfying the power budget constraint $\mathbb{E}\left[|s_r|^2\right] = p$, and $z_r \sim \mathcal{CN}\left(0, \sigma^2\right)$ is the additive noise with power σ^2. In this view, the objective of downlink rate maximization can be cast as follows:

$$\underset{\boldsymbol{\theta}^r \in \mathcal{C}}{\text{maximize}} \qquad \log_2\left(1 + \frac{|\mathbf{h}_r^{\mathsf{H}} \boldsymbol{\Theta}_r \, \mathbf{g}_r|^2 p}{\sigma^2}\right), \tag{9}$$

where $\mathcal{C}$ is the set of feasible RIS configurations.

In order to solve (9), a perfect knowledge of both channels $\mathbf{h}$ and $\mathbf{g}$ is assumed. Even under perfect CSI, determining the optimal set of phase shifts $\boldsymbol{\theta}^{r\star}$ requires a heuristic search due to the notion of configuration classes $\mathcal{C}$. Such solutions cannot be practically adopted since they are not scalable with the number of RIS elements. As a remedy, we resort to ML in order to devise a data-driven solution.

In this context, consider that the RIS in environment $r \in \mathcal{R}$ (later referred to as agent r) has a dataset $\mathcal{D}_r = \left\{\left(\boldsymbol{x}_j^r, c_j^r\right) \,\middle|\, j = 1, \ldots, D_r\right\}$ containing D_r samples of observed CSI $\boldsymbol{x}_j^r = \left(\mathbf{h}_j^r, \mathbf{g}_j^r\right)$ that are labeled by c_j^r corresponding to the optimal RIS phase shifts $\boldsymbol{\theta}^{r\star}$ solving (9). We then seek to construct a mapping function $f_{\boldsymbol{w}}(\cdot)$ parameterized by $\boldsymbol{w}$, that solves:

$$\underset{\boldsymbol{w}}{\text{minimize}} \qquad \frac{1}{|\mathcal{R}|}\sum_{r=1}^{|\mathcal{R}|}\frac{1}{D_r}\sum_{j=1}^{D_r}\ell\left(c_j^r, f_{\boldsymbol{w}}(\boldsymbol{x}_j^r)\right), \tag{10}$$

where $\ell(\cdot\,,\cdot)$ is the loss function in terms of phase prediction. In order to optimize the model parameter $\boldsymbol{w}$ in the ERM formulation in (10), all agents are required to share their datatsets $\mathcal{D}_r$ with a central server. Due to privacy concerns and communication constraints, (10) is recast as a FL problem by minimizing a global loss function as follows:

$$\underset{\boldsymbol{w}}{\text{minimize}} \qquad \frac{1}{|\mathcal{R}|}\sum_{r=1}^{|\mathcal{R}|}\ell_r\left(f_{\boldsymbol{w}}\right), \tag{11}$$

where $\ell_r\left(f_{\boldsymbol{w}}\right) = \frac{1}{D_r}\sum_{j=1}^{D_r}\ell\left(c_j^r, f_{\boldsymbol{w}}(\boldsymbol{x}_j^r)\right)$ is the local loss function of agent r. One of the most popular approaches in FL to solve (11) is the FEDAVG algorithm [8].

However, the formulation in (11) assumes that all agents in $\mathcal{R}$ have an equal impact on training the global model $\boldsymbol{w}$. This falls under the strong assumption of uniform and homogeneous data distribution across agents. Under data heterogeneity, drifts in the agents' local updates with respect to the aggregated model might occur, since the local optima do not necessarily coincide with the global optima. Thus, the obtained model suffers from instability in convergence, and fails to generalize to OoD samples. To obviate this issue, we resort to Invariant Risk Minimization (IRM) [12] and its FL variant dubbed FL GAMES [14].

III. FL GAMES FOR PHASE OPTIMIZATION

The key deficiency of using FEDAVG on limited datasets distributed across agents is that its trained predictor heavily relies on statistical correlations among observations. These correlations are specious since they depend on the environment from which they were sampled. Thus, overfitting to these correlations prevents FEDAVG from training a predictor that is robust in unseen environments. To overcome this issue, we turn our attention to algorithms that learn causal representations that are invariant across different agents, which improves the model's OoD generalization across many environments.

In this direction, IRM [12] jointly trains an *extraction function* $f_{\boldsymbol{\phi}}$ and a *predictor* $f_{\boldsymbol{w}}$ across training environments $\mathcal{R}$ in such a way that $f_{\boldsymbol{w}} \circ f_{\boldsymbol{\phi}}$ generalizes well in unseen environments $\mathcal{R}_{\text{all}} \supset \mathcal{R}$. The main idea is to build a parameterized feature extractor $f_{\boldsymbol{\phi}}$ that reveals the causal representations in the samples, so as to perform causal inference by optimizing $f_{\boldsymbol{w}}$. Thus, given an extraction function, the predictor $f_{\boldsymbol{w}}$ is the one that is simultaneously optimal across all training environments $\mathcal{R}$. Formally, this boils down to solving the following problem:

$$\begin{aligned} \underset{\boldsymbol{\phi},\, \boldsymbol{w}}{\text{minimize}} \quad & \frac{1}{|\mathcal{R}|}\sum_{r=1}^{|\mathcal{R}|}\ell_r\left(f_{\boldsymbol{w}} \circ f_{\boldsymbol{\phi}}\right) \\ \text{subject to} \quad & \boldsymbol{w} \in \underset{\boldsymbol{w}'}{\arg\min}\ \ell_r\left(f_{\boldsymbol{w}'} \circ f_{\boldsymbol{\phi}}\right) \qquad \forall\, r \in \mathcal{R}. \end{aligned} \tag{12}$$

Note that IRM is formulated as a single agent optimization problem, and assumes that training environments are known to the agent beforehand. Extending IRM to the distributed setting can be done using game theory (IRM GAMES [13]). Therein, different agents, each equipped with its own predictor $\boldsymbol{w}_r$, cooperate to train an ensemble model: $\boldsymbol{w}^{\text{av}} = \sum_{r\in\mathcal{R}}\frac{D_r}{\sum_{n\in\mathcal{R}} D_n}\boldsymbol{w}_r$. In contrast to (12) that designs a unique predictor across training environments, the aggregate model $\boldsymbol{w}^{\text{av}}$ satisfies:

$$\begin{aligned} \underset{\boldsymbol{\phi},\, \boldsymbol{w}^{\text{av}}}{\text{minimize}} \quad & \frac{1}{|\mathcal{R}|}\sum_{r=1}^{|\mathcal{R}|}\ell_r\left(f_{\boldsymbol{w}^{\text{av}}} \circ f_{\boldsymbol{\phi}}\right) \\ \text{subject to} \quad & \boldsymbol{w}_r \in \underset{\boldsymbol{w}_r'}{\arg\min}\ \ell_r\left(\frac{f_{\boldsymbol{w}_r'} + \sum_{n\in\mathcal{R}\setminus\{r\}} f_{\boldsymbol{w}_n}}{|\mathcal{R}|} \circ f_{\boldsymbol{\phi}}\right) \\ & \qquad\qquad \forall r \in \mathcal{R}. \end{aligned} \tag{13}$$

The set of constraints in (13) represents the Nash Equilibrium of a game where the players/environments $r \in \mathcal{R}$ select actions $\boldsymbol{w}_k$ in order to minimize their cost functions $\ell_r\left(f_{\boldsymbol{w}^{\text{av}}} \circ f_{\boldsymbol{\phi}}\right)$. Since there are no algorithms that guarantee reaching the Nash point of the aforementioned continuous non-zero sum game, Best Response Dynamics (BRD) is used due to its simplicity. It is worth mentioning that [13] also showed that the feature extraction map can be fixed to identity $f_{\boldsymbol{\phi}} = \mathrm{I}$, and the overall prediction network $f_{\boldsymbol{w}} \circ f_{\boldsymbol{\phi}}$ will be recovered by the predictor $f_{\boldsymbol{w}}$ solving (13). This version of the algorithm is called F-IRM GAMES, while the one where the extractor and predictor are learned separately is referred to as V-IRM GAMES.

IRM GAMES is a very suitable fit for FL since it allows agents distributed in different environments to train a collective IRM-based model. Hence, the authors in [14] proposed minor modifications in order to adapt it to the FL setup. First, parallelized BRD is used to mitigate the sequential dependencies and accelerate convergence. Moreover, V-IRM GAMES

requires an extra round of optimization for the extraction function parameter $\boldsymbol{\phi}$, which delays its convergence. Thus, the stochastic gradient descent (SGD) over $\boldsymbol{\phi}$ is replaced by a Gradient Descent (GD) update that takes larger steps in the direction of the global optimum. In contrast to [14], which considered highly correlated datasets, the data in our setting shows negligible oscillations in parameter updates under BRD, in which, we do not adopt buffers in training. These buffers can be used by each agent to store the historically played actions of its opponents. Then, an agent responds to a uniform distribution over these past actions. This smoothens the oscillations of BRD caused by the local correlations in the datasets. Finally, the detailed steps of the V-FL GAMES algorithm that trains both a representation learner and a predictor are presented in Algorithm 1.

IV. SIMULATION RESULTS

A. Simulation Settings

For our experiments, we consider three different environments $\mathcal{R}$. In all three environments, the Tx is located at the coordinate $(0, 35, 3)$ and the RIS comprising $N = 10 \times 10$ reflective elements is at $(10, 20, 1)$, with the coordinates given in meters within a Cartesian system. The Rx is located in an annular region around the RIS with inner and outer radii $R_{\text{min}} = 1\,\text{m}$ and $R_{\text{max}} = 5\,\text{m}$. The Rician factor κ is set to 5, and the pathloss coefficients are calculated by $\alpha_t = \frac{N d_{\text{x}} d_{\text{y}}}{4\pi d_t^2}$ and $\alpha_r = \frac{N d_{\text{x}} d_{\text{y}}}{4\pi d_r^2}$. To simplify the exhaustive search for optimal phases, we assume $\mathcal{C}$ contains two configurations classes, namely $\boldsymbol{\theta}_1 = [0, 0, \ldots, 0]^{\mathsf{T}}$ and $\boldsymbol{\theta}_2 = [0, \pi, 0, \pi, \ldots, 0, \pi]^{\mathsf{T}}$.

In environment 1, the RIS elements are distanced by $d_{\text{x}} = d_{\text{y}} = 0.5\lambda$. Therein, the Rx is uniformly placed around the RIS with a tendency to be deployed closer to the RIS following the distributions illustrated under Environment 1 in Fig. 2. In environment 2, the RIS is characterized by $d_{\text{x}} = d_{\text{y}} = 0.25\lambda$. In contrast to environment 1, here, the Rx is higher likely to be placed far from the RIS (see the distributions under Environment 2 in Fig. 2). The RIS in environment 3 has $d_{\text{x}} = d_{\text{y}} = 0.4\lambda$. Therein, the Rx's distance to the RIS is uniform but the angle distribution is concentrated in one direction as illustrated in Fig. 2 under Environment 3. Note that the data from environments 1 and 2 are used to compose the training data while environment 3 is used only for testing.

In this work, we leverage V-FL GAMES that exhibits a superior performance than F-FL GAMES (where $f_{\boldsymbol{\phi}} = \text{I}$). On the other hand, the authors in [11] showed by empirical simulations, due to the fact that RIS channels have strong LoS components, that one can select fixed causal representations based on the channels in (1) and (5). The causal representations in this case are the AoA and AoD at the RIS, (φ_t, ϑ_t) and (φ_r, ϑ_r), and the relative distances RIS-Tx d_t and RIS-Rx d_r. This benchmark variant of FL GAMES where $f_{\boldsymbol{\phi}}$ is fixed to $(\varphi_t, \vartheta_t, \varphi_r, \vartheta_r, d_t, d_r)$ is called F-FL GAMES in this paper, and is not to be confused with F-FL GAMES in [14], where $f_{\boldsymbol{\phi}} = \text{I}$.

Algorithm 1 FL GAMES for RIS

Inputs: Set of RISs: $\mathcal{R}$, Datasets: $\mathcal{D}_r$, Learning rates: $\eta_{\boldsymbol{w}}, \eta_{\boldsymbol{\phi}}$, Number of rounds: rounds, Mini-batch size: m
Outputs: $f_{\boldsymbol{\phi}}, f_{\boldsymbol{w}^{\text{av}}}$
Server executes:
Initialize $\boldsymbol{\phi}$ and $\{\boldsymbol{w}_r\}_{r\in\mathcal{R}}$
Broadcast $\boldsymbol{\phi}$ and $\{\boldsymbol{w}_r\}_{r\in\mathcal{R}}$ to all agents $r \in \mathcal{R}$
Set round $\leftarrow 1$
while round $\leq$ rounds **do**
for each agent $r \in \mathcal{R}$ **parallel do**
Compute $\nabla\boldsymbol{\phi}_r \leftarrow \nabla_{\boldsymbol{\phi}}\, \ell_r\left(f_{\boldsymbol{w}^{\text{av}}} \circ f_{\boldsymbol{\phi}}; \mathcal{D}_r\right)$
Communicate $\nabla\boldsymbol{\phi}_r$ to the server
end for
Update extractor:
$\boldsymbol{\phi} \leftarrow \boldsymbol{\phi} - \eta_{\boldsymbol{\phi}} \sum_{r\in\mathcal{R}} \frac{D_r}{\sum_{n\in\mathcal{R}} D_n} \nabla\boldsymbol{\phi}_k$
Broadcast $\boldsymbol{\phi}$ to all agents $r \in \mathcal{R}$
for each agent $r \in \mathcal{R}$ **parallel do**
Sample mini-batch $\mathcal{B}_r$ of size m from $\mathcal{D}_r$
Update predictor:
$\boldsymbol{w}_r \leftarrow \boldsymbol{w}_r - \eta_{\boldsymbol{w}} \nabla_{\boldsymbol{w}_r}\, \ell_r\left(f_{\boldsymbol{w}^{\text{av}}} \circ f_{\boldsymbol{\phi}}; \mathcal{B}_r\right)$
Communicate $\boldsymbol{w}_r$ to the server
end for
Average Predictor: $\boldsymbol{w}^{\text{av}} \leftarrow \sum_{r\in\mathcal{R}} \frac{D_r}{\sum_{n\in\mathcal{R}} D_n} \boldsymbol{w}_k$
Broadcast $\{\boldsymbol{w}_r\}_{r\in\mathcal{R}}$ to all agents $r \in \mathcal{R}$
round $\leftarrow$ round $+ 1$
end while

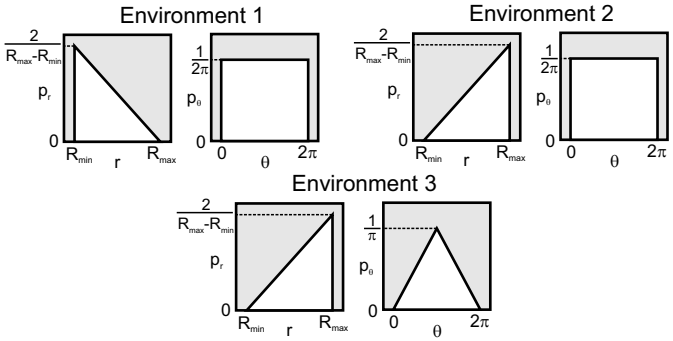


Fig. 2: Distributions of the receiver's position (distance r and angle θ from the RIS) in different environments.

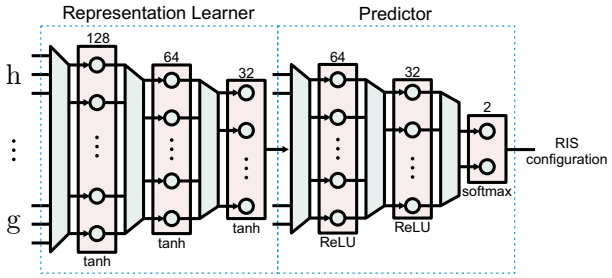
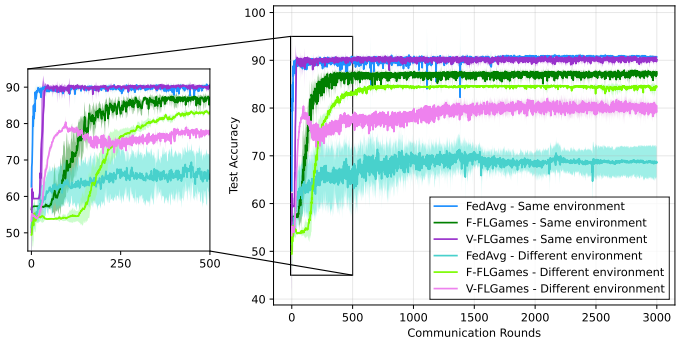
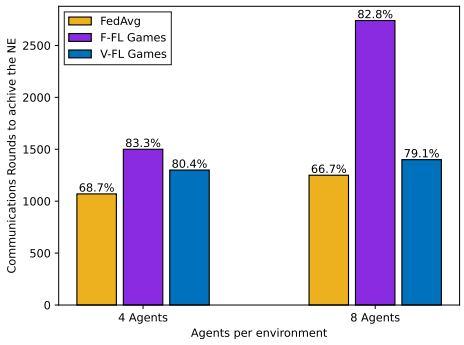


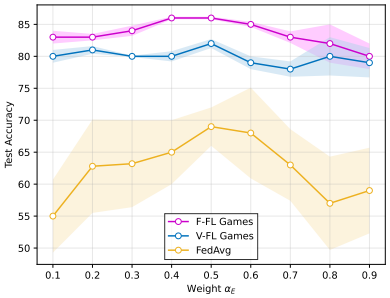
Fig. 3: Structure of the neural networks used for the representation learner and the predictor: circles represent activation functions and trapezoids correspond to the weights and biases. The activation function type and the output dimension are shown at the bottom and top of each layer.



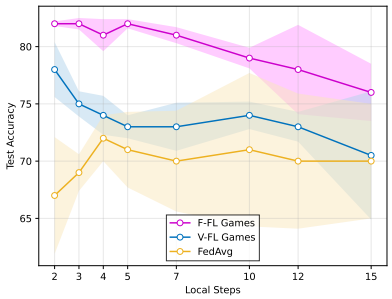
(a) Test accuracy convergence for all algorithms compared in the same or in a different environment.



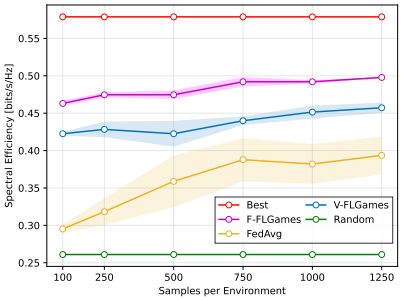
(e) Communication rounds needed for convergence with multiple agents per environment. The number on top of the bar is the corresponding test accuracy.



(b) Impact of the weight α_e on the testing accuracy.



(c) Impact of number of local iterations on the testing accuracy.



(d) Impact of the number of samples per environment on the achievable spectral efficiency.

Fig. 4: Performance comparison of different algorithms in OoD testing datasets.

For training in FEDAVG and V-FL GAMES, we collect $D_r = 1500$ CSI samples $\boldsymbol{x}_j^r = \left(\mathbf{h}_j^r, \mathbf{g}_j^r\right)$ from environments 1 and 2, that are decoupled over real and imaginary parts. This data is scaled in such a way that the normalized mean is zero and the normalized variance is one. We also record the parameters $(\varphi_t, \vartheta_t, \varphi_r, \vartheta_r, d_t, d_r)$ to train F-FL GAMES, which are scaled using a minmax scaler that normalizes the data to the interval $[-1, 1]$ by dividing by the absolute maximum. Unless stated otherwise, we use 1000 samples collected from environment 3 for testing. The design of the neural networks of the extractor and the predictor of V-FL GAMES is based on the multi-layer perceptrons architecture, and is shown in Fig. 3. Note that FEDAVG and F-FL GAMES only use the predictor part. The considered loss function is the cross-entropy. The mini-batch size used for training the predictor is $m = 32$, and the learning rates are fixed at $\eta_{\boldsymbol{\phi}} = 5 \times 10^{-4}$ and $\eta_{\boldsymbol{w}} = 2 \times 10^{-3}$. In the figures, lines correspond to the simulation results that are averaged over five runs while the their respective standard deviations are shown as shaded areas.

B. Discussion

We first plot the evolution of the testing accuracy of all algorithms in Fig. 4(a). Within the same environment (Environments 1 and 2), FEDAVG and V-FL GAMES perform similarly with an accuracy of 90%, while F-FL GAMES gives an accuracy of 87%. When testing in a different environment (Environment 3), FEDAVG's accuracy drops to 68%, highlighting its lack of robustness. In this case, F-FL GAMES and V-FL GAMES yield slightly lower accuracies of 85% and 80%, implying they do not overfit to the statistical correlations in the channels. It is also interesting to observe the convergence rate of the different methods. We notice that V-FL GAMES converges faster than F-FL GAMES, requiring around 500 communication epochs in both cases.

Fig. 4(b) demonstrates the impact of mixing data from different environments on model training. Here, we keep the total number samples to be $D_1 + D_2 = 3000$ and vary the amount of data from the two environments, in which, $\alpha_e = \frac{D_1}{D_2}$ represents the fraction of samples of environment 1 compared to environment 2. We first notice that all algorithms reach their peak performance with balanced datasets, i.e. $\alpha_e = 0.5$. When the datatsets are biased towards one of the environments, FEDAVG loses 15% of its performance. On the other hand, F-FL GAMES and V-FL GAMES maintain a steady accuracy with a slight degradation of about 3–4%.

Inspired by FEDAVG's flexibility in allowing more local computations at each agent before sharing their models, we modify our proposed algorithm to study the effect of the number of local iterations on the model accuracy. Insofar, each agent performs a few SGD updates locally prior to model sharing. Fig. 4(c) shows the impact of the number of local

steps on the testing accuracy. It can be noted that the accuracy drops when each RIS performs more local computations when using FL GAMES due to the fact that the models are overfitting to the local datasets. However, the performance of F-FL GAMES is more consistent with the local steps, losing only 2% of accuracy with seven local iterations, while V-FL GAMES loses around 8% of accuracy with 15 local updates. The reason for this behavior is that by letting each agent run over more samples from its local dataset, the testing accuracy at equilibrium decreases, since the played strategies do not account for the opponents' actions. On the other hand, the accuracy of FEDAVG slightly increases with more local iterations, but still performs poorly.

The effect of the dataset size D_r per agent on the achievable spectral efficiency is illustrated in Fig. 4(d). Note that all agents use an equal amount of samples, i.e., $\alpha_e = 0.5$ is held. For the comparison, we additionally present the optimal rate given by the best configurations (indicated by Best) and the rates given by random phase decision making (indicated by Random). All algorithms reach their best performance with the highest number of samples $D_r = 1250$ with about 21%, 14% and 32% losses compared to Best rates in V-FL GAMES, F-FL GAMES, and FEDAVG, respectively. The advantage of learning invariant causal representations with minimal amount of data is highlighted when $D_r \leq 750$. FEDAVG looses its performance rapidly. Even with 100 samples per environment, FL GAMES algorithms lose only 6% of their performance, while FEDAVG incurs more than 15% of its accuracy. FEDAVG requires around 750 samples per agent to reach its best performance, that is more than 10% less than that given by V-FL GAMES, underscoring its weakness in OoD settings. Additionally, with 1250 samples per environment, the error variance of V-FL GAMES and F-FL GAMES is 73% and 98% less than that of FEDAVG.

Finally, we vary the number of agents per environment as shown in Fig. 4(e). For this experiment, 1500 samples from each environment are shared among all agents, so more agents having less data are involved. The achieved testing accuracies of the FL GAMES algorithms are still superior than the FEDAVG benchmark. Surprisingly, doubling the number of collaborating RISs from 8 to 16 induces an 82% increase in the number of training epochs for convergence in F-FL GAMES. The same does not hold for V-FL GAMES that suffers from an 8% increase, while losing 3-4$\%$ in accuracy compared to F-FL GAMES. This implies that, with more agents owning fewer data, the training of a causal extractor and a predictor converges faster than training of only a predictor.

V. CONCLUSIONS

This paper proposes a distributed phase configuration control for RIS-assisted communication systems. The rate maximization problem is formulated using federated IRM as opposed to a heterogeneity-unaware ERM approach. Our novel robust RIS phase-shifts controller leverages the underlying causal representations of the data that are invariant over different environments. A neural network based feature extractor first uncovers the causal structure of the CSI data, then feeds it to another neural network based configuration predictor. Both neural networks are trained in a distributed supervised learning fashion, and the results are compared with the environment-unaware FEDAVG and an IRM-based predictor. The numerical results show that a phase predictor trained with the geometric properties of the environments demonstrated a better performance than a representation learner followed by a predictor. Moreover, the extractor-predictor network exhibits faster training convergence when using more RISs. The extensions for multiple users and multiple antennas at Tx and Rx are left for future works.

REFERENCES

[1] C. Huang, A. Zappone, G. C. Alexandropoulos, M. Debbah, and C. Yuen, "Reconfigurable intelligent surfaces for energy efficiency in wireless communication," *IEEE transactions on wireless communications*, vol. 18, no. 8, pp. 4157–4170, 2019.

[2] M. D. Renzo, M. Debbah, D.-T. Phan-Huy, A. Zappone, M.-S. Alouini, C. Yuen, V. Sciancalepore, G. C. Alexandropoulos, J. Hoydis *et al.*, "Smart radio environments empowered by reconfigurable ai meta-surfaces: An idea whose time has come," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, pp. 1–20, 2019.

[3] Ö. Özdoğan and E. Björnson, "Deep learning-based phase reconfiguration for intelligent reflecting surfaces," in *2020 54th Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2020, pp. 707–711.

[4] K. Stylianopoulos, N. Shlezinger, P. Del Hougne, and G. C. Alexandropoulos, "Deep-learning-assisted configuration of reconfigurable intelligent surfaces in dynamic rich-scattering environments," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 8822–8826.

[5] G. C. Alexandropoulos, K. Stylianopoulos, C. Huang, C. Yuen, M. Bennis, and M. Debbah, "Pervasive machine learning for smart radio environments enabled by reconfigurable intelligent surfaces," *Proceedings of the IEEE*, vol. 110, no. 9, pp. 1494–1525, 2022.

[6] J. Park, S. Samarakoon, H. Shiri, M. K. Abdel-Aziz, T. Nishio, A. Elgabli, and M. Bennis, "Extreme ultra-reliable and low-latency communication," *Nature Electronics*, vol. 5, no. 3, pp. 133–141, 2022.

[7] G. C. Alexandropoulos, S. Samarakoon, M. Bennis, and M. Debbah, "Phase configuration learning in wireless networks with multiple reconfigurable intelligent surfaces," in *2020 IEEE Globecom Workshops (GC Wkshps*. IEEE, 2020, pp. 1–6.

[8] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[9] C. B. Issaid, S. Samarakoon, M. Bennis, and H. V. Poor, "Federated distributionally robust optimization for phase configuration of riss," in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2021, pp. 1–6.

[10] Y. Deng, M. M. Kamani, and M. Mahdavi, "Distributionally robust federated averaging," *Advances in neural information processing systems*, vol. 33, pp. 15 111–15 122, 2020.

[11] S. Samarakoon, J. Park, and M. Bennis, "Robust reconfigurable intelligent surfaces via invariant risk and causal representations," in *2021 IEEE 22nd International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2021, pp. 301–305.

[12] M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz, "Invariant risk minimization," *arXiv preprint arXiv:1907.02893*, 2019.

[13] K. Ahuja, K. Shanmugam, K. Varshney, and A. Dhurandhar, "Invariant risk minimization games," in *International Conference on Machine Learning*. PMLR, 2020, pp. 145–155.

[14] S. Gupta, K. Ahuja, M. Havaei, N. Chatterjee, and Y. Bengio, "Fl games: A federated learning framework for distribution shifts," *arXiv preprint arXiv:2205.11101*, 2022.

[15] E. Björnson and L. Sanguinetti, "Rayleigh fading modeling and channel hardening for reconfigurable intelligent surfaces," *IEEE Wireless Communications Letters*, vol. 10, no. 4, pp. 830–834, 2020.