pFedDHPO: A Differentiable Approach for Personalized Hyperparameter Optimization in Federated Learning

Jinglong Shen, *Student Member, IEEE*, Nan Cheng, *Senior Member, IEEE*, Wenchao Xu, *Member, IEEE*, Haozhao Wang, *Member, IEEE*, Wei Quan, *Senior Member, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—Hyperparameter optimization (HPO) is crucial for federated learning (FL) performance. Given the inherent data heterogeneity across clients, recent research has focused on providing personalized hyperparameters for individual clients. However, such personalized approaches introduce exponential search complexity as the number of clients increases, significantly reducing the efficiency of existing HPO methods. To address this challenge, we propose pFedDHPO, a novel personalized HPO framework that efficiently optimizes hyperparameters in a differentiable manner. Specifically, pFedDHPO formulates personalized HPO as an optimization problem targeting joint distribution parameters within the clients' search space and leverages gradient information from differentiable validation loss to substantially enhance the efficiency of the HPO process. Experimental results demonstrate that pFedDHPO achieves state-of-the-art performance compared to baseline methods, improving accuracy by up to 18.35% under extreme Non-IID data distributions. Additionally, the framework reduces communication overhead by 41.2% compared to conventional HPO methods, making it highly scalable for resource-constrained FL deployments.**

***Index Terms*—Federated learning, hyperparameter optimization, personalization, heterogeneity.**

I. Introduction

IN RECENT years, federated learning (FL) has emerged as a significant distributed machine learning framework [1]. FL allows for the utilization of data from different clients without centralizing or exchanging users' raw data, promoting model enhancement and knowledge sharing while preserving privacy. However, the implementation of FL in real-world scenarios presents significant challenges due to data heterogeneity [2], [3], [4]. For instance, in cross-hospital medical imaging analysis, participating institutions often exhibit substantially different disease prevalence patterns. A study by Zech et al. [5] demonstrated this variation when examining chest X-rays across three hospitals, where pneumonia prevalence ranged from 1.2% to 8.4% among 158,323 patients, reflecting the distinct demographic populations served. Similarly, in IoT-based smart city applications, the Chicago Array of Things project reported that air quality sensors deployed across 130 nodes showed PM2.5 concentration variations exceeding 300% between residential neighborhoods ($8.2\ \mu g/m^3$) and industrial corridors ($25.7\ \mu g/m^3$) during peak hours [6], highlighting the substantial data distribution differences across deployment zones. The data distributions among participating clients are often non-independently and identically distributed (Non-IID), and such differences in data distributions can lead to convergence instability during training and performance degradation of the global model. While many research efforts have been devoted to addressing data heterogeneity in FL, most of them focus on developing new training architectures and algorithms [7], [8] or mitigating the effect of heterogeneity by preprocessing client-side local data [9]. These schemes have the same hyperparameters for each client during the training process, making the hyperparameters not well adapted to the different distributions of each client, leading to performance loss in model training. There is still a gap in research on addressing data heterogeneity from the perspective of hyperparameter optimization (HPO).

Due to the variability in data distribution, clients may exhibit distinct optimization landscapes. As illustrated in Fig. 1, some clients' landscapes may resemble steep valleys, while others may resemble gentle hills. This requires custom hyperparameters for each client to improve federated optimization performance [10]. Additionally, several works [11], [12], [13], [14] have attempted to adaptively tune hyperparameters, such as learning rate, according to the local landscape of clients, and proved to be effective in improving the performance of the global model, especially in Non-IID scenarios. However, these works are limited to the tuning of pre-defined hyperparameter types, and it is

Received 25 November 2024; revised 4 March 2025 and 7 April 2025; accepted 7 April 2025. Date of publication 16 April 2025; date of current version 19 December 2025. This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFB2901900, in part by the Fundamental Research Funds for the Central Universities, and in part by the Innovation Fund of Xidian University under Grant YJSJ25007. The associate editor coordinating the review of this article and approving it for publication was N. Zorba. *(Corresponding author: Nan Cheng.)*

Jinglong Shen and Nan Cheng are with the School of Telecommunications Engineering, Xidian University, Xi'an 710126, Shaanxi, China (e-mail: jlshen@stu.xidian.edu.cn; dr.nan.cheng@ieee.org).

Wenchao Xu is with the Department of Computing, Hong Kong Polytechnic University, China (e-mail: wenchao.xu@polyu.edu.hk).

Haozhao Wang is with the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: hz_wang@hust.edu.cn).

Wei Quan is with the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China (e-mail: weiquan@bjtu.edu.cn).

Xuemin Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/TCCN.2025.3561292

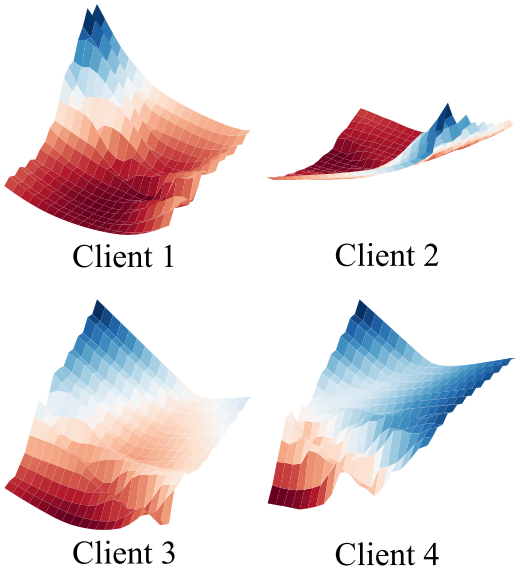

Fig. 1. Local loss landscapes of ResNet18 models trained on heterogeneous clients with Non-IID CIFAR-10 data partitions.

difficult to extend to arbitrary types. On the other hand, existing general HPO methods provide good support for various hyperparameter types. These methods typically employ heuristics to iteratively search for improved hyperparameters, optimizing computational resources by performing low-fidelity evaluations. Nevertheless, there are still challenges in applying existing general HPO methods to FL, including the search complexity and the limitations in parallel evaluation.

- *Exponential Growth in Search Complexity:* In the personalized HPO setting, different hyperparameter values need to be optimized for each client, and thus the search space is the Cartesian product of the local search space of all clients, which leads to an exponential growth of the search complexity with the number of clients.
- *Limitations in Parallel Evaluation:* In the centralized HPO scenario, the evaluation of hyperparameters can be parallelized to evaluate different hyperparameters at the same time using sufficient computational power of the computer cluster. However, in the FL scenario, the evaluation of hyperparameters can only be performed on the user's personal devices due to the constraints of privacy issues. And these devices usually have limited computational power and unreliable network connections [15], [16], so a sequential evaluation approach is needed to evaluate different hyperparameters one by one.

To address these challenges, there is an urgent need to design HPO algorithms for FL that achieve high search efficiency to better cope with the exponential growth of the search space as the number of clients increases, and to reduce the communication rounds required for HPO in the presence of unreliable network connections. Existing solutions fall into two categories: (1) client-adaptive methods that rely on manually designed hyperparameter adaptation rules, which suffer from limited applicability and dependence on heuristics; and (2) federated HPO frameworks that directly extend conventional optimization methods, which face prohibitive computational complexity in personalization scenarios.

Differentiable search methods in neural architecture search (NAS) have achieved impressive results by forming a hypernetwork of different candidate networks and introducing network architecture parameters to select between them [17]. By making the validation loss differentiable with respect to the network architecture parameters, differentiable NAS significantly improves the efficiency of the search process. Drawing inspiration from NAS while addressing FL-specific constraints, we propose **pFedDHPO**, a novel **D**ifferentiable approach for **P**ersonalized **HPO** in **FL**, which achieves up to 18.35% higher accuracy than state-of-the-art methods under Non-IID data distributions. In pFedDHPO, the hyperparameters of each client are sampled from its corresponding *distribution parameters*, and the evaluation mechanism of the hyperparameters is specially designed to ensure that the validation loss is differentiable with respect to the distribution parameters. We minimize interference with the existing FL workflow and compute the gradient of the distribution parameters to update them until they converge for all clients. pFedDHPO's credit assignment mechanism assigns importance scores to the hyperparameter decisions of individual clients based on their contributions to the validation performance, accelerating convergence by **41.2%** in communication rounds. The main contributions of this paper are as follows.

- *A Differentiable Framework for Personalized HPO in FL:* We propose pFedDHPO, the first framework that reformulates personalized HPO in FL as a differentiable optimization task over joint distribution parameters. Unlike closed-box HPO methods, our approach enables gradient-based search in an exponentially large parameter space while explicitly addressing data heterogeneity.
- *Privacy-Aware Credit Assignment Mechanism:* We design a novel differentiable validation loss mechanism coupled with client-specific importance scoring. This is the first work to integrate credit assignment with FL privacy constraints, allowing quantification of hyperparameter impact per client without exposing local data.
- *Communication-Efficient Optimization:* We introduce a budget-aware optimization framework for efficient hyperparameter search under limited federated learning communication rounds. pFedDHPO combines distributional hyperparameter sampling with validation-driven convergence detection to reduce communication overhead through: 1) seamless integration with existing FL algorithms without additional synchronization requirements, and 2) adaptive resource allocation that dynamically balances exploration-exploitation based on validation feedback, enabling early termination of underperforming configurations.

The remainder of the paper is organized as follows. Section II presents some related works of the paper. Section III shows the problem formulation for the personalized HPO in FL and presents the detailed design of the methodology. Section IV provides experimental results to evaluate the performance of the proposed method. Section V discuss

limitations and future work. Finally, Section VI concludes the paper.

II. Related Works

A. Non-IID Federated Learning

In the field of FL, numerous solutions were proposed to tackle the issue of data heterogeneity. Some works focused on reducing the differences in data distribution among clients by sharing a public dataset [18] or utilizing data samples generated by an autoencoder [9]. Others proposed innovations in model training, such as the FedProx algorithm, which enhanced the model's performance on Non-IID data by incorporating L2 regularization terms in the local updates [7]. Furthermore, at the level of learning algorithms, meta-learning [19] and multi-task learning [20] were employed to improve model adaptation to heterogeneous data, e.g., using the FedAvg algorithm for basic training and fine-tuning for personalized training. Additionally, framework-level innovations, such as client-side clustering methods [21] and distillation-based semi-supervised FL algorithms (DS-FL) [22], were designed to better adapt to heterogeneous data. Although each of these approaches addressed the Non-IID problem from a different perspective, little work was done to address data heterogeneity from the viewpoint of personalized HPO. To bridge this gap, we propose pFedDHPO, which efficiently searches for personalized hyperparameters for each client and is compatible with other FL pipelines.

B. Hyperparameter Optimization

1) General HPO Techniques: In addressing the challenges of HPO, researchers proposed a variety of strategies, each with its specific advantages and disadvantages. Grid search methods could thoroughly explore the hyperparameter space, but they were often infeasible for practical applications due to their high computational cost, which grew significantly with the number of hyperparameters [23]. As a simpler alternative, random search (RS) tended to outperform grid search in high-dimensional spaces [24]. In contrast, Bayesian optimization (BO) constructed probabilistic models to predict efficient hyperparameter combinations and provided a more intelligent search strategy that demonstrated its sample efficiency in practice [25]. To allocate computational resources more efficiently, multi-fidelity methods such as Hyperband [26] and successive halving algorithms (SHA) [27] were employed, allowing them to quickly eliminate less effective configurations. However, in a personalized HPO setting, the search space of these methods would grow exponentially with the number of clients, accompanied by the need for massive communication computational resources to evaluate different hyperparameters, which was often unacceptable in resource-sensitive FL scenarios.

2) FL-Specific HPO Methods: The growing recognition of data heterogeneity in FL underscored the critical importance of personalized hyperparameters. The authors in [12] introduced Δ-SGD, a stochastic gradient descent variant that enabled clients to adaptively adjust their step sizes based on local function smoothness. Subsequent studies [13], [14] demonstrated that client-specific hyperparameters tailored to local data distributions significantly improved fine-tuning outcomes for pre-trained models, further establishing the necessity of personalized HPO in FL. Recent advancements expanded HPO capabilities to include energy-aware joint optimization of computational and communication parameters [28] and client-driven adaptation strategies specifically designed for Non-IID scenarios [29].

While existing methods achieved adaptive hyperparameter tuning, they remained restricted to predefined hyperparameter types (e.g., learning rates or momentum) and failed to generalize to arbitrary configurations. Recent approaches addressed this limitation through meta-learning frameworks that leveraged implicit function theorem (IFT)-based optimization [14], privacy-preserving encrypted aggregation with differential privacy [30], and multi-objective Pareto optimization for security-critical applications [31].

General HPO algorithms demonstrated significant progress in federated settings. Nakka et al. [32] investigated reward-based federated HPO approaches, though subsequent analyses revealed limitations in algorithmic sensitivity. In IoT deployments, genetic algorithm-driven methods [33] optimized performance through fog computing architectures. Parallelized multi-fidelity evaluation frameworks [34] effectively balanced privacy constraints with communication and computational efficiency requirements. FLoRA [35] introduced single-shot HPO techniques that minimized communication overhead, while dynamic client-epoch adjustment strategies [36] prioritized system efficiency rather than focusing solely on model accuracy.

Despite their efficiency, these methods optimized global hyperparameters shared across all clients, neglecting client-specific requirements. To address this limitation, we propose pFedDHPO, a federated HPO framework that enable personalization while maintaining computational efficiency.

III. Methodology

The primary objective of pFedDHPO is to establish an efficient and cost-effective mechanism for adaptive HPO within the FL framework while maintaining good compatibility with classic FL pipelines. The framework of pFedDHPO is illustrated in Fig. 2. Compared with the classic FL pipeline, pFedDHPO only adds two extra parts, *Hyperparameter Sampling* and *Update Distribution Parameters*, and this design makes the pFedDHPO has good compatibility and can be easily incorporated into the classic FL pipeline. In this section, we first present the problem formulation of personalized HPO in FL. We then delve into the search space of hyperparameters after incorporating random relaxation within a differentiable environment. Subsequently, we outline the training approach for the distribution parameters of the hyperparameters. Finally, we combine insights from reinforcement learning, specifically policy gradients, to address the credit assignment problem.

A. Personalized Setting Formulation

As personalization pertains to the hyperparameters associated with local training, we focus on optimizing the hyperparameters on the client side. Consider an FL system

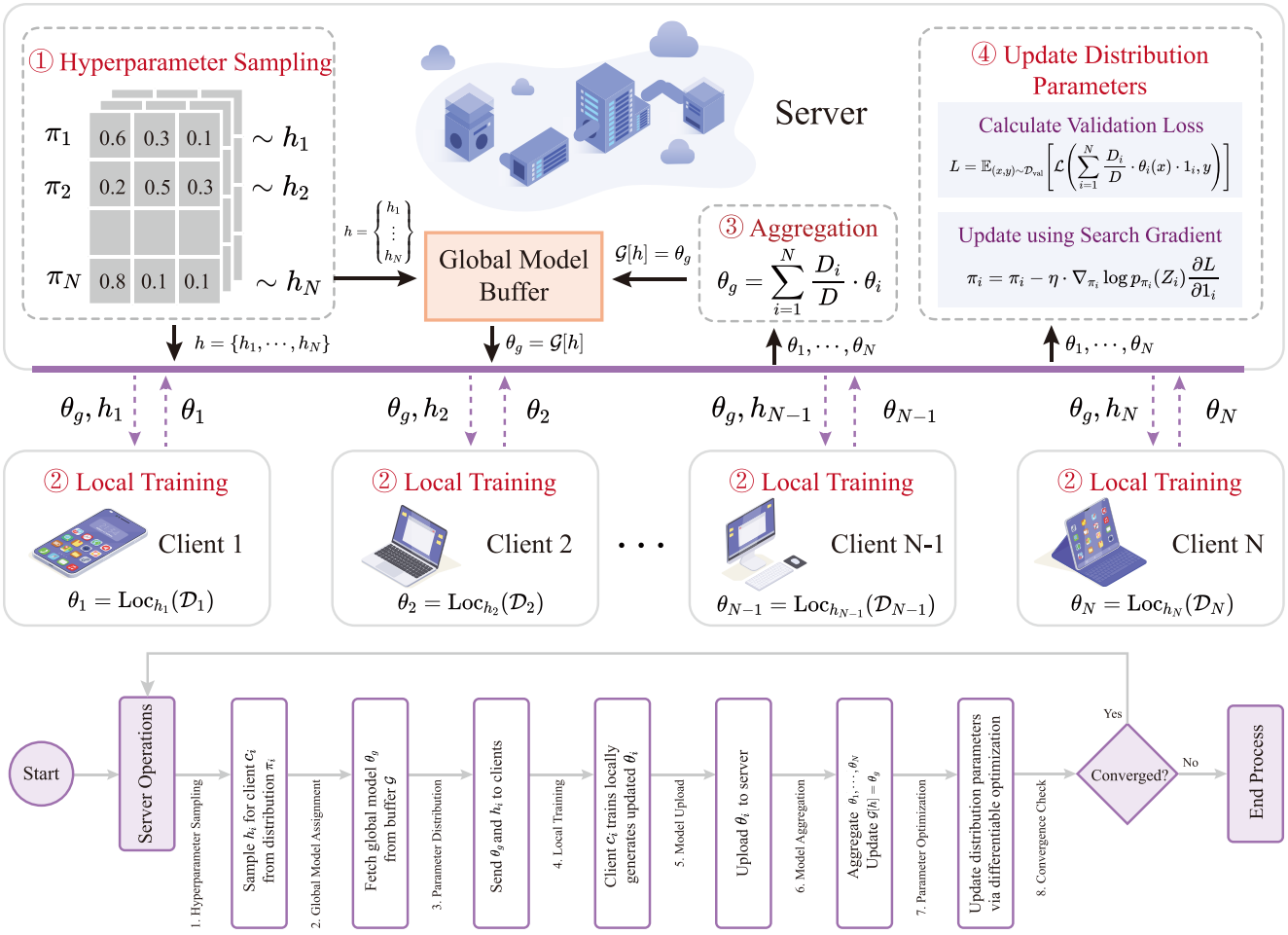

Fig. 2. Framework of pFedDHPO with four-phase workflow: (1) Hyperparameter sampling phase where the server samples $h_i \sim \pi_i$ and retrieves corresponding global model θ_g from buffer $\mathcal{G}$; (2) Client distribution phase transmitting $\{\theta_g, h_i\}$ to each client c_i; (3) Model aggregation phase updating $\mathcal{G}[h] = \theta_g$ with client models $\{\theta_i\}_{i=1}^N$; (4) Policy update phase optimizing distribution parameters via Eq. (13).

with a client set $\mathcal{C} = \{c_1, \ldots, c_N\}$ consisting of N clients, where each client possesses a training set $\mathcal{D}_i$ of size D_i, and the total size of all clients is $D = \sum_{i=1}^N D_i$. The server holds a validation set $\mathcal{D}_{\text{val}}$ to evaluate the performance of different hyperparameter groups. The hyperparameter group of the FL system is denoted by $h = \{h_1, \ldots, h_N\}$, where h_i represents the specific hyperparameters for client c_i. We assume that the hyperparameter search space is the same for each client and can be represented as $\mathcal{H} = \{h^1, \ldots, h^H\}$, consisting of H discrete hyperparameters. Therefore, the search space in the personalized setting is given by $\mathcal{H}_1 \times \cdots \times \mathcal{H}_N = \mathcal{H}^N$, which exponentially expands with the number of clients. Personalized HPO can then be formulated as solving the following bi-level optimization problem

$$\min_{h \in \mathcal{H}^N} \ \mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{val}}}[\mathcal{L}(\theta_h^*(x), y)], \tag{1}$$

$$\text{s.t.} \ \ \theta_h^* = \text{Alg}_h(\{\mathcal{D}_i | i = 1, \ldots, N\}), \tag{1a}$$

where Alg_h represents the execution of the FL system for the complete training course under the hyperparameter configuration h, returning the trained global model. $\mathcal{L}$ denotes the chosen loss function for the specific task. However, executing Alg_h requires running a complete FL training course, which is computationally expensive and introduces unacceptable communication and computational overhead. Therefore, there is a demand for multi-fidelity methods to evaluate the hyperparameters while executing only a fraction of the complete training process. The most common approach is the SHA method [27], which gradually eliminates poorly performing hyperparameters during the training process. Nevertheless, this hard elimination procedure generates noisy validation results, leading to the premature elimination of potentially high quality hyperparameters due to their less favorable early performance.

To address this issue, we transform the original bi-level optimization problem into the following surrogate problem, enabling multi-fidelity HPO

$$\min_{h \in \mathcal{H}^N} \ \mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{val}}}\left[\mathcal{L}\left(\sum_{i=1}^N \frac{D_i}{D}\theta_i(x), y\right)\right], \tag{2}$$

where $\theta_i = \text{Loc}_{h_i}(\mathcal{D}_i)$, and Loc_{h_i} represents the local training phase configured by h_i, returning the updated local model parameters θ_i trained on $\mathcal{D}_i$. Since this transformation allows us to choose different hyperparameter groups between different communication rounds, it is more flexible. We search for the best hyperparameters by optimizing (2), instead of

simply removing the hyperparameters that do not perform well in the preliminaries directly, as SHA does, and therefore avoid the premature elimination of potentially high quality hyperparameters. Having formulated the personalized HPO problem, we now address the challenge of exponential growth in the search space. This necessitates a differentiable relaxation mechanism to enable efficient learning of hyperparameter distributions, which we introduce in the following subsection.

B. Search Space and Hyperparameter Sampling

To mitigate the computational complexity of the exponentially expanding search space, we employ differentiable hyperparameter sampling. This approach enables direct optimization of hyperparameter distributions through gradient-based updates, thereby circumventing the need for exhaustive evaluation of all candidate configurations. Differentiable methods have demonstrated remarkable effectiveness in NAS [17]. However, making the HPO problem differentiable remains a challenge due to the non-trivial task of embedding sampled hyperparameters into the forward/backward propagation graph. In this paper, we introduce pFedDHPO, a differentiable method for personalized optimization of FL hyperparameters.

The search space for personalized HPO expands exponentially with the number of clients, resulting in a significantly high search complexity. Therefore, we improve the efficiency of HPO in pFedDHPO by directly searching for the optimal hyperparameter distribution for each client using an approach similar to policy gradients, referred to as *search gradients* in [37].

Since the hyperparameter decision for each client is generally tractable, we represent the optimal hyperparameter distribution for each client with a distribution $p(Z_i)$. By multiplying a one-hot random variable Z_i sampled from $p(Z_i)$ with the candidate hyperparameters $\mathcal{H}$, we obtain the optimal hyperparameter decision for client c_i as

$$h_i = Z_i^T \mathcal{H}, \tag{3}$$

where $p(Z_i)$ can be fully factorized, and its factors are parameterized by π_i. We therefore refer to Z_i as the *decision variable*. The objective function (2) can be stochastically relaxed as

$$\min_{\pi \in \Pi} \mathbb{E}_{Z \sim p_\pi(Z)} \left[\mathbb{E}_{(x,y) \sim \mathcal{D}_{val}} \left[\mathcal{L} \left(\sum_{i=1}^{N} \frac{D_i}{D} \theta_i(x), y \right) \right] \right], \tag{4}$$

where $\theta_i = \mathrm{Loc}_{Z_i^T \mathcal{H}}(\mathcal{D}_i)$, $\pi = [\pi_1, \ldots, \pi_N]$, and $Z = [Z_1, \ldots, Z_N]$. This objective function optimizes the expected performance of hyperparameters sampled from *p(Z)*. In the next subsection, we describe how pFedDHPO computes the gradient of π so as to complete the optimization of π by minimizing the validation loss L. While differentiable sampling enables gradient propagation, optimizing the hyperparameter distribution parameters necessitates incorporating the validation loss into the training loop. We now formalize the gradient computation framework for these distribution parameters.

C. Parameter Learning for Hyperparameters

The gradient of validation loss with respect to distribution parameters establishes a crucial link between hyperparameter selection and model performance. By reparameterizing discrete decision variables through the Gumbel-Softmax trick, we achieve end-to-end differentiability while maintaining the essential properties of the original search space. Computing the gradient of the validation loss L with respect to the parameters π is not straightforward because most hyperparameters are not directly involved in the forward/backward propagation process. To address this, we transform the objective function (4) from *selecting the best hyperparameters* to *selecting models trained with different hyperparameters that perform the best*. This allows us to embed Z_i into the forward/backward propagation graph during validation by multiplying Z_i with the output of the model, instead of the hyperparameters $\mathcal{H}$. This transformation is intuitive because the best-performing model often implies the best hyperparameters. Thus, we can utilize Z_i, which selects the best model, to sample the hyperparameters.

Assuming no additional communication and computational costs on the client side, each client trains a separate local model for each candidate hyperparameters. The collection of local models trained by client c_i is denoted by $\Theta_i = \{\theta^1, \ldots, \theta^H\}$, and the objective function (4) can be transformed to

$$\min_{\pi \in \Pi} \mathbb{E}_{Z \sim p_\pi(Z)} \left[\mathbb{E}_{(x,y) \sim \mathcal{D}_{val}} \left[\mathcal{L} \left(\sum_{i=1}^{N} \frac{D_i}{D} Z_i^T \Theta_i(x), y \right) \right] \right]. \tag{5}$$

To address the challenge of differentiable discrete HPO in FL, we employ the Gumbel-Softmax reparameterization. This technique combines the Gumbel-Max trick with a temperature-controlled Softmax relaxation to enable gradient-based optimization of categorical distributions. The Gumbel-Max trick generates discrete samples via $Z = \mathrm{argmax}_k(\log \pi_k + G_k)$, where $G_k \sim \mathrm{Gumble}(0, 1)$. However, the non-differentiable argmax operation is replaced with a Softmax relaxation to facilitate backpropagation [38]

$$\tilde{Z}_i^k = \frac{\exp\left((\log \pi_i^k + G_i^k)/\lambda\right)}{\sum_{l=1}^{H} \exp\left((\log \pi_i^l + G_i^l)/\lambda\right)}, \tag{6}$$

where $G_i^k = -\log(-\log(U_i^k))$ is sampled via $U_i^k \sim \mathrm{Uniform}(0, 1)$. The temperature parameter λ balances bias and gradient stability: as $\lambda \to 0$, $\tilde{Z}_i$ approaches a one-hot vector, recovering the original discrete distribution with $p(\lim_{\lambda \to 0} \tilde{Z}_i^k = 1) = \frac{\pi_i^k}{\sum_{l=1}^{N} \pi_i^l}$ [39], ensuring the unbiasedness of this relaxation.

This reparameterization enables two critical features: (1) Differentiable gradients $\dfrac{\partial L}{\partial \pi_k}$ for end-to-end optimization of hyperparameter distributions, and (2) Computational efficiency by evaluating all H candidates in a single forward-backward pass, avoiding the prohibitive cost of training H models per

client. Therefore, the partial derivative of $\tilde{Z}_i^{k'}$ with respect to π_i^k is

$$\frac{\partial \tilde{Z}_i^{k'}}{\partial \pi_i^k} = \frac{\partial \frac{\exp\left(m_i^{k'}\right)}{\sum_{l=1}^{H}\exp\left(m_i^l\right)}}{\partial m_i^k} \cdot \frac{m_i^k}{\pi_i^k} = \begin{cases} \left(1-\tilde{Z}_i^{k'}\right)\tilde{Z}_i^k \cdot \dfrac{1}{\lambda \pi_i^k}, & k'=k, \\ -\tilde{Z}_i^{k'}\tilde{Z}_i^k \cdot \dfrac{1}{\lambda \pi_i^k}, & k' \neq k, \end{cases} \tag{7}$$

where $m_i^k = (\log \pi_i^k + G_i^k)/\lambda$. Thus, it can be vectorized as

$$\frac{\partial \tilde{Z}_i}{\partial \pi_i^k} = \left(\delta(k'-k) - \tilde{Z}_i\right)\tilde{Z}_i^k \frac{1}{\lambda \pi_i^k}, \tag{8}$$

where $\delta(k'-k)$ denotes the impulse function, with $\delta(k'-k)=1$ if $k'=k$, otherwise $\delta(k'-k)=0$. And we can obtain the partial derivative of L with respect to π_i^k via chain rule as follows

$$\frac{\partial L}{\partial \pi_i^k} = \frac{\partial L}{\partial \tilde{Z}_i}\frac{\partial \tilde{Z}_i}{\partial \pi_i^k} = \frac{\partial L}{\partial \tilde{Z}_i}\left(\delta(k'-k) - \tilde{Z}_i\right)\tilde{Z}_i^k \frac{1}{\lambda \pi_i^k}, \tag{9}$$

where $\frac{\partial \tilde{Z}_i}{\partial \pi_i^k}$ is derived using the properties of the concrete distribution (Eqs. (7) and (8)). This gradient enables direct optimization of π_i^k by measuring how changes in hyperparameter probabilities affect the validation loss.

However, each client trains multiple local models based on candidate hyperparameters, incurring prohibitive computational and communication overheads. Inspired by DSNAS [40], we observe that this inefficiency primarily stems from $\tilde{Z}_i$ being a soft one-hot vector. By constraining $\tilde{Z}_i$ to be a strict one-hot vector, we can avoid training models with hyperparameters corresponding to zero values in $\tilde{Z}_i$, thereby substantially reducing computational and communication costs. We elaborate on this approach in the following subsection.

D. Credit Assignment

Building upon the differentiable framework established in previous section, we now address the credit assignment problem inherent in federated HPO. Inspired by reinforcement learning principles, we formulate hyperparameter selection as a policy optimization task where clients receive rewards proportional to their contribution to global validation performance. Credit assignment plays a crucial role in reinforcement learning, significantly impacting policy learning performance. However, as noted by [37], personalized HPO can be viewed as a task with delayed rewards, since decisions for all clients need be made before receiving any feedback. This delayed reward structure impedes convergence, which is particularly problematic in FL scenarios with limited computational and communication resources. In this subsection, we analyze how pFedDHPO approaches credit assignment for the hyperparameter decisions of each client.

As discussed in the previous subsection, the decision variable $\tilde{Z}_i$ is differentiable with respect to the distribution parameters π_i. Given $\tilde{Z}_i^k = f_{\pi_i}(G_i^k) = \frac{\exp((\log \pi_i^k + G_i^k)/\lambda)}{\sum_{l=1}^{H}\exp((\log \pi_i^l + G_i^l)/\lambda)}$, the expected gradient of the decision variable $\tilde{Z}_i$ with respect to the distribution parameter π_i can be derived as

$$\begin{aligned} &E_{\tilde{Z}_i \sim p_{\pi_i}(\tilde{Z}_i)}\left[\frac{\partial \tilde{Z}_i}{\partial \pi_i}\right] \\ &= E_{U_i \sim Uniform}\left[\frac{\partial f_{\pi_i}(-\log(-\log(U_i)))}{\partial \pi_i}\right] \\ &= \int_0^1 p(U_i)\frac{\partial f_{\pi_i}(-\log(-\log(U_i)))}{\partial \pi_i}\mathrm{d}U_i \\ &= \frac{\partial}{\partial \pi_i}\int p_{\pi_i}\left(\tilde{Z}_i\right)\tilde{Z}_i \mathrm{d}\tilde{Z}_i \\ &= \int p_{\pi_i}\left(\tilde{Z}_i\right)\frac{\partial \log p_{\pi_i}\left(\tilde{Z}_i\right)}{\partial \pi_i}\tilde{Z}_i \mathrm{d}\tilde{Z}_i \\ &= E_{\tilde{Z}_i \sim p_{\pi_i}(\tilde{Z}_i)}\left[\nabla_{\pi_i}\log p_{\pi_i}\left(\tilde{Z}_i\right)\tilde{Z}_i\right]. \end{aligned} \tag{10}$$

From this, we can obtain the search gradient of distribution parameter π_i as

$$\begin{aligned} &\mathbb{E}_{\tilde{Z}_i \sim p_{\pi_i}(\tilde{Z}_i)}\left[\frac{\partial L}{\partial \pi_i}\right] = \mathbb{E}_{\tilde{Z}_i \sim p_{\pi_i}(\tilde{Z}_i)}\left[\frac{\partial L}{\partial \tilde{Z}_i}\frac{\partial \tilde{Z}_i}{\partial \pi_i}\right] \\ &= \mathbb{E}_{\tilde{Z}_i \sim p_{\pi_i}(\tilde{Z}_i)}\left[\nabla_{\pi_i}\log p_{\pi_i}\left(\tilde{Z}_i\right)\frac{\partial L}{\partial \tilde{Z}_i}\tilde{Z}_i\right], \end{aligned} \tag{11}$$

which leverages the score function estimator. This approach avoids backpropagating through the entire training process, significantly reducing computational complexity in federated settings.

As previously discussed, the use of $\tilde{Z}_i$ as a soft one-hot vector introduces computational and communication challenges to HPO. To make pFedDHPO more feasible, we make $\lambda \to 0$ to transform $\tilde{Z}_i$ into a strict one-hot vector Z_i. This transformation leads to the following equation

$$\begin{aligned} &\lim_{\lambda \to 0}\left\{\mathbb{E}_{\tilde{Z}_i \sim p_{\pi_i}(\tilde{Z}_i)}\left[\nabla_{\pi_i}\log p_{\pi_i}\left(\tilde{Z}_i\right)\frac{\partial L}{\partial \tilde{Z}_i}\tilde{Z}_i\right]\right\} \\ &= \mathbb{E}_{Z_i \sim p_{\pi_i}(Z_i)}\left[\nabla_{\pi_i}\log p_{\pi_i}(Z_i)\frac{\partial L}{\partial 1_i}\right], \end{aligned} \tag{12}$$

where Z_i represents a strict one-hot vector, and 1_i indicates the element in Z_i whose value is 1. In this case, only the hyperparameter corresponding to the selected entry in Z_i needs to be trained, while the hyperparameters with 0 entries in Z_i no longer require training, significantly reducing computational and communication costs. The policy π_i can be updated using Monte Carlo methods, and the iterative formula is defined as follows

$$\pi_i = \pi_i - \eta \cdot \nabla_{\pi_i}\log p_{\pi_i}(Z_i)\frac{\partial L}{\partial 1_i}, \tag{13}$$

Algorithm 1 pFedDHPO Training Process

Initialize the global model buffer $\mathcal{G}$.
for round t in $\{1, \ldots, T\}$ **do**
  Sample Z_i for each client: $Z_i \sim p_{\pi_i}(Z_i)$.
  Get h_i for each client: $h_i = Z_i^T \mathcal{H}$.
  Retrieve the global model from global model buffer: $\theta_g = \mathcal{G}[h]$.
  Broadcast θ_g and h_i to the corresponding clients.
  for each client in parallel **do**
    Perform local training: $\theta_i = Loc_{h_i}(\theta_g, \mathcal{D}_i)$.
    Upload θ_i to the server.
  end for
  Aggregate local models: $\theta_g = \sum_{i=1}^{N} \frac{D_i}{D} \theta_i$.
  Update the global model buffer: $\mathcal{G}[h] = \theta_g$.
  for each (x, y) sampled from $\mathcal{D}_{\text{val}}$ **do**
    Calculate the validation loss: $L = \mathcal{L}\left(\sum_{i=1}^{N} \frac{D_i}{D} \cdot \theta_i(x) \cdot 1_i, y\right)$.
    Calculate the gradient of the dummy one: $\frac{\partial L}{\partial 1_i}$.
  end for
  Update distribution parameters using Equation (13).
end for

where η denotes the learning rate of π_i. It is evident that the search gradient is equivalent to the policy gradient of the distribution parameter π_i, and its credit is assigned as

$$R_i = -\frac{\partial L}{\partial 1_i}. \tag{14}$$

This reward allocates importance scores to different hyperparameter decisions, allowing decisions that contribute more to the validation performance to receive higher rewards. This importance allocation strategy also explains why pFedDHPO is more efficient. The complete training process is detailed in Algorithm 1. With our credit assignment mechanism in place, we integrate the differentiable HPO components into a unified FL framework, ensuring efficient resource utilization while enabling personalized hyperparameter adaptation for each client.

E. Overall Process

The pFedDHPO framework integrates hyperparameter sampling, local training, and credit-based distribution updates. By maintaining separate global model states for distinct hyperparameter configurations, we mitigate reward interference and enhance convergence stability. This section presents the complete training process of pFedDHPO. As illustrated in Fig. 2, the pFedDHPO training pipeline comprises four key components: hyperparameter sampling, local training, model aggregation, and distribution parameter updating. Notably, maintaining a single shared global model across different hyperparameter configurations would lead to reward contamination, allowing inferior hyperparameter configurations to benefit from the training progress achieved by superior configurations in previous rounds. To address this issue, we implement a global model buffer $\mathcal{G}$ that stores separate global models for each hyperparameter configuration, where each unique hyperparameter group $h = \{h_1, h_2, \ldots, h_N\}$ serves as a composite key to retrieve the corresponding global model parameters θ_g. This design isolates configuration-specific effects during aggregation and ensures bounded buffer growth through hyperparameter reuse and distribution convergence. The detailed training process is as follows:

- *Hyperparameter Sampling:* At the beginning of each communication round, the server needs to sample a new hyperparameter h_i for each client c_i from the corresponding distribution parameter π_i, to get the hyperparameter group $h = \{h_1, \ldots, h_N\}$ containing all client hyperparameters, which enables the evaluation of different hyperparameter groups. Subsequently, the server obtains the global model $\theta_g = \mathcal{G}[h]$ corresponding to the current hyperparameter group h from the global model cache $\mathcal{G}$.
- *Local Training:* The server sends the global model θ_g and hyperparameters h_i to the corresponding client c_i, and then each client starts local training based on the received global model and hyperparameters. After completing the local training, client c_i uploads the training result θ_i to the server.
- *Model Aggregation:* The server aggregates the models $\theta_1, \ldots, \theta_N$ uploaded by the client to get the updated global model θ_g and updates the global model cache $\mathcal{G}[h] = \theta_g$.
- *Distribution Parameter Update:* The server calculates the validation loss L based on the validation set $\mathcal{D}_{\text{val}}$ and the model $\theta_1, \ldots, \theta_N$ uploaded by the client

$$L = \mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{val}}}\left[\mathcal{L}\left(\sum_{i=1}^{N} \frac{D_i}{D} \cdot \theta_i(x) \cdot 1_i, y\right)\right], \tag{15}$$

where 1_i is a constant with a value of 1 used to assist in the backpropagation of the gradient. The server computes the partial differential $\frac{\partial L}{\partial 1_i}$ of the validation loss L with respect to 1_i based on (15). Subsequently, the server computes the value of $\nabla_{\pi_i} \log p_{\pi_i}(Z_i)$ and multiplies it with $\frac{\partial L}{\partial 1_i}$ to obtain the search gradient. Finally, the distribution parameter π_i for each client is updated using Eq. (13).

Our methodology introduces three key innovations that advance the state-of-the-art in FL: (1) a differentiable HPO mechanism for personalized FL via dynamic reward allocation, (2) system-agnostic robustness to data heterogeneity across arbitrary network scales, and (3) accelerated convergence through credit-based exploration-exploitation balancing. Specifically, the proposed reward allocation mechanism quantifies hyperparameter importance using gradient-based validation signals, thereby automating client-specific configuration optimization in Non-IID data settings. These contributions collectively position pFedDHPO as the first FL framework to simultaneously achieve personalized HPO automation, cross-device adaptability, and resource-efficient model optimization.

IV. Evaluations

A. Experimental Setup

To make a fair comparison, we introduce the HPO budget and define it as the total number of communication rounds used in the HPO phase. All methods are compared under the same HPO budget.

1) Baseline Algorithms: To comprehensively evaluate the performance of pFedDHPO, we selected five baseline methods representing three distinct perspectives. First, to assess pFedDHPO's performance against traditional HPO algorithms and its capacity for personalized optimization, we employed the widely-used RS [24] and BO [25] as baselines. These methods were adapted for personalized HPO by defining the search space as the Cartesian product of the clients' local search spaces. Second, to demonstrate the advantages of pFedDHPO over existing federated HPO methods and to highlight the significance of hyperparameter personalization, we included FedEx [34] and FLoRA [35] as baselines. These methods are specifically designed for FL to identify a single, global hyperparameter configuration shared by all clients. Finally, to compare pFedDHPO with rule-based hyperparameter adaptation approaches, we selected FedHPO [29] as a baseline. The implementation details are as follows:

- *RS:* The server randomly selects 30 different groups of hyperparameters and chooses different fidelities for hyperparameter evaluation based on the HPO budget. Specifically, when the HPO budget is $30, 90, \ldots, 270$ rounds, the server conducts $1, 3, \ldots, 9$ rounds of training to evaluate each group of hyperparameters, respectively.
- *BO:* The server uses BO algorithm to select 30 different groups of hyperparameters and chooses different fidelities for hyperparameter evaluation based on the HPO budget. Specifically, when the HPO budget is $30, 90, \ldots, 270$ rounds, the server conducts $1, 3, \ldots, 9$ rounds of training to evaluate each group of hyperparameters, respectively.
- *FedEx:* We utilize the SHA [27] as the wrapper for FedEx, where the server updates the distribution parameters of the hyperparameters based on the HPO budget. Once the HPO budget is exhausted, the hyperparameter associated with the maximum value in the distribution parameters is returned.
- *FLoRA:* For FLoRA, we employ BO as the local HPO optimizer. Afterwards, we aggregate the loss surfaces of individual clients using the maximum of per-party local models (MPLM) method [35], and return the hyperparameters corresponding to the minimum validation loss.
- *FedHPO:* The proposed algorithm adaptively adjusts each client's hyperparameters during local training, specifically focusing on learning rate and number of local epochs. Clients dynamically reduce their learning rate by a predefined factor when loss improvement falls below a threshold, and implement early stopping when the loss plateaus for consecutive epochs.

2) Models and Datasets: In order to conduct a comprehensive evaluation of pFedDHPO, we carefully selected three distinct types of tasks. These tasks cover scenarios with label skew and feature shift under non-iid conditions, and encompass both image and text tasks.

- *ResNet18 [41] (@CIFAR10 [42]):* The CIFAR10 dataset consists of 60,000 32x32 color images, evenly distributed among 10 classes, with each class containing 6,000 images. To simulate a *label skew* Non-IID scenario, we partitioned the CIFAR10 dataset using a Dirichlet distribution, resulting in an average of 2,000 samples per client. We then trained the ResNet18 network on this partitioned dataset to assess the performance of pFedDHPO on convolutional models for image tasks.
- *ViT-Base [43] (@Office-Home [44]):* The Office-Home dataset consists of 15,500 images from four distinct domains. These domains encompass Art, Clipart, Product, and Real-World photographs, which are categorized into 65 classes. To simulate a Non-IID scenario with *feature shift*, we assign data from a single domain to each client and control the sample size per client using a Dirichlet distribution, ensuring an average of 500 samples per client. Subsequently, we train the Vision Transformer (ViT-Base) network on this partitioned dataset to assess the performance of pFedDHPO on transformer-based models for image tasks.
- *DeBERTaV3 [45] (@SST2 [46]):* The SST-2 dataset serves as a benchmark for sentiment analysis tasks. It comprises sentences extracted from movie reviews, with each sentence labeled as either positive or negative sentiment. To simulate a Non-IID scenario with *label skew*, we partitioned the SST-2 dataset using a Dirichlet distribution, resulting in an average of 2,000 samples per client. Subsequently, we trained the DeBERTaV3 network on this partitioned dataset to assess the performance of pFedDHPO on transformer-based models for text tasks.

3) Settings: We implemented all algorithms using PyTorch and executed them on a 64-bit Ubuntu 20.04 operating system equipped with 8 A40 GPUs. To assess the compatibility of pFedDHPO with different FL pipelines, we conducted experiments on both FedAvg and FedProx. The candidate hyperparameters include learning rates [1e-1, 5e-2, 1e-2, 5e-3, 1e-3] and weight decay values [0, 1e-1, 1e-2, 1e-3, 1e-4, 1e-5]. To ensure a fair comparison, all algorithms perform HPO within the same HPO budget. After exhausting the HPO budget, we performed 50 rounds of full-fidelity training using the best performing hyperparameters to evaluate the HPO performance of the different algorithms. The reported test accuracies represent the evaluation results of the models on the test set after 50 rounds of full-fidelity training, and all results are averaged over 30 independent runs.

B. Results

1) Comparison With Baselines: We systematically evaluate the HPO capabilities and communication efficiency of pFedDHPO under constrained optimization budgets, explicitly considering practical deployment constraints. As demonstrated in Table I and Fig. 3, our framework achieves 54.94% accuracy on ResNet18 using only 30 communication rounds–a tenfold reduction in HPO budget compared to FedEx and

TABLE I
TEST ACCURACY OF MODELS WITH HYPERPARAMETERS OPTIMIZED UNDER DIFFERENT HPO BUDGETS B. THE BEST ACCURACY IS MARKED IN BOLD, AND THE SECONDARY IS MARKED IN UNDERLINE. THE SUBSCRIPT $\downarrow x.x$ IS THE PERFORMANCE DIFFERENCE FROM THE BEST RESULT, MARKED GREEN WHEN THE DIFFERENCE IS GREATER THAN 10%

		FedAvg					FedProx				
		$B=30$	$B=90$	$B=150$	$B=210$	$B=270$	$B=30$	$B=90$	$B=150$	$B=210$	$B=270$
ResNet18 (@CIFAR10)	Ours	**54.94**$_{(\downarrow 0.00)}$	**60.00**$_{(\downarrow 0.00)}$	**62.56**$_{(\downarrow 0.00)}$	**62.96**$_{(\downarrow 0.00)}$	**62.33**$_{(\downarrow 0.00)}$	**56.07**$_{(\downarrow 0.00)}$	**61.74**$_{(\downarrow 0.00)}$	**62.85**$_{(\downarrow 0.00)}$	**63.12**$_{(\downarrow 0.00)}$	**62.60**$_{(\downarrow 0.00)}$
	FedEx	40.69$_{(\downarrow 14.25)}$	42.76$_{(\downarrow 17.24)}$	43.78$_{(\downarrow 18.78)}$	44.39$_{(\downarrow 18.57)}$	44.53$_{(\downarrow 17.80)}$	41.20$_{(\downarrow 14.87)}$	42.37$_{(\downarrow 19.37)}$	43.27$_{(\downarrow 19.58)}$	44.53$_{(\downarrow 18.59)}$	44.13$_{(\downarrow 18.47)}$
	FLoRA	37.67$_{(\downarrow 17.27)}$	38.88$_{(\downarrow 21.12)}$	39.93$_{(\downarrow 22.63)}$	40.12$_{(\downarrow 22.84)}$	39.94$_{(\downarrow 22.39)}$	38.44$_{(\downarrow 17.63)}$	40.44$_{(\downarrow 21.03)}$	41.01$_{(\downarrow 21.84)}$	40.31$_{(\downarrow 22.81)}$	40.79$_{(\downarrow 21.81)}$
	RS	40.26$_{(\downarrow 14.68)}$	45.93$_{(\downarrow 14.07)}$	50.77$_{(\downarrow 11.79)}$	53.18$_{(\downarrow 9.78)}$	55.02$_{(\downarrow 7.31)}$	42.68$_{(\downarrow 13.39)}$	45.51$_{(\downarrow 16.23)}$	51.41$_{(\downarrow 11.44)}$	54.48$_{(\downarrow 8.64)}$	55.75$_{(\downarrow 6.85)}$
	BO	46.42$_{(\downarrow 8.52)}$	47.00$_{(\downarrow 13.00)}$	48.25$_{(\downarrow 14.31)}$	48.94$_{(\downarrow 14.02)}$	49.37$_{(\downarrow 12.96)}$	44.70$_{(\downarrow 11.37)}$	47.59$_{(\downarrow 14.15)}$	49.21$_{(\downarrow 13.64)}$	49.53$_{(\downarrow 13.59)}$	49.32$_{(\downarrow 13.28)}$
	FedHPO	52.15$_{(\downarrow 2.79)}$	53.30$_{(\downarrow 6.70)}$	54.35$_{(\downarrow 8.21)}$	55.63$_{(\downarrow 7.33)}$	55.97$_{(\downarrow 6.36)}$	52.68$_{(\downarrow 3.39)}$	53.90$_{(\downarrow 7.84)}$	54.65$_{(\downarrow 8.20)}$	55.87$_{(\downarrow 7.25)}$	56.12$_{(\downarrow 6.48)}$
ViT-Base (@Office-Home)	Ours	**80.33**$_{(\downarrow 0.00)}$	**80.46**$_{(\downarrow 0.00)}$	**80.63**$_{(\downarrow 0.00)}$	**81.23**$_{(\downarrow 0.00)}$	**81.86**$_{(\downarrow 0.00)}$	**80.83**$_{(\downarrow 0.00)}$	**80.29**$_{(\downarrow 0.00)}$	**80.81**$_{(\downarrow 0.00)}$	**81.67**$_{(\downarrow 0.00)}$	**81.00**$_{(\downarrow 0.00)}$
	FedEx	48.25$_{(\downarrow 32.08)}$	50.76$_{(\downarrow 29.70)}$	54.75$_{(\downarrow 25.88)}$	55.45$_{(\downarrow 25.78)}$	56.46$_{(\downarrow 25.40)}$	49.37$_{(\downarrow 31.46)}$	50.46$_{(\downarrow 29.83)}$	54.36$_{(\downarrow 26.45)}$	56.96$_{(\downarrow 24.71)}$	56.61$_{(\downarrow 24.39)}$
	FLoRA	67.05$_{(\downarrow 13.28)}$	68.14$_{(\downarrow 12.32)}$	71.09$_{(\downarrow 9.54)}$	72.71$_{(\downarrow 8.52)}$	74.40$_{(\downarrow 7.46)}$	67.55$_{(\downarrow 13.28)}$	70.91$_{(\downarrow 9.38)}$	75.04$_{(\downarrow 5.77)}$	74.56$_{(\downarrow 7.11)}$	74.49$_{(\downarrow 6.51)}$
	RS	76.78$_{(\downarrow 3.55)}$	77.77$_{(\downarrow 2.69)}$	79.19$_{(\downarrow 1.44)}$	80.21$_{(\downarrow 1.02)}$	80.94$_{(\downarrow 0.92)}$	77.27$_{(\downarrow 3.56)}$	77.40$_{(\downarrow 2.89)}$	80.63$_{(\downarrow 0.18)}$	81.20$_{(\downarrow 0.47)}$	80.59$_{(\downarrow 0.41)}$
	BO	78.83$_{(\downarrow 1.50)}$	78.29$_{(\downarrow 2.17)}$	80.08$_{(\downarrow 0.55)}$	80.80$_{(\downarrow 0.43)}$	80.85$_{(\downarrow 1.01)}$	75.85$_{(\downarrow 4.98)}$	78.17$_{(\downarrow 2.12)}$	79.67$_{(\downarrow 1.14)}$	80.72$_{(\downarrow 0.95)}$	80.78$_{(\downarrow 0.22)}$
	FedHPO	76.14$_{(\downarrow 4.19)}$	76.69$_{(\downarrow 3.77)}$	77.38$_{(\downarrow 3.25)}$	77.83$_{(\downarrow 3.40)}$	78.25$_{(\downarrow 3.61)}$	77.40$_{(\downarrow 3.43)}$	78.00$_{(\downarrow 2.29)}$	78.38$_{(\downarrow 2.43)}$	78.56$_{(\downarrow 3.11)}$	78.91$_{(\downarrow 2.09)}$
DeBERTaV3 (@SST2)	Ours	**74.51**$_{(\downarrow 0.00)}$	**83.24**$_{(\downarrow 0.00)}$	**84.66**$_{(\downarrow 0.00)}$	**84.40**$_{(\downarrow 0.00)}$	84.19$_{(\downarrow 1.28)}$	**76.83**$_{(\downarrow 0.00)}$	**83.75**$_{(\downarrow 0.00)}$	**85.16**$_{(\downarrow 0.00)}$	85.83$_{(\downarrow 0.12)}$	**85.90**$_{(\downarrow 0.00)}$
	FedEx	67.45$_{(\downarrow 7.06)}$	79.62$_{(\downarrow 3.62)}$	83.03$_{(\downarrow 1.63)}$	84.16$_{(\downarrow 0.24)}$	**85.47**$_{(\downarrow 0.00)}$	67.66$_{(\downarrow 9.17)}$	79.47$_{(\downarrow 4.28)}$	83.67$_{(\downarrow 1.49)}$	**85.95**$_{(\downarrow 0.00)}$	85.72$_{(\downarrow 0.18)}$
	FLoRA	74.02$_{(\downarrow 0.49)}$	78.76$_{(\downarrow 4.48)}$	80.81$_{(\downarrow 3.85)}$	82.71$_{(\downarrow 1.69)}$	83.08$_{(\downarrow 2.39)}$	74.52$_{(\downarrow 2.31)}$	78.30$_{(\downarrow 5.45)}$	81.23$_{(\downarrow 3.93)}$	82.92$_{(\downarrow 3.03)}$	83.67$_{(\downarrow 2.23)}$
	RS	61.50$_{(\downarrow 13.01)}$	69.67$_{(\downarrow 13.57)}$	74.15$_{(\downarrow 10.51)}$	74.76$_{(\downarrow 9.64)}$	74.27$_{(\downarrow 11.20)}$	62.99$_{(\downarrow 13.84)}$	70.68$_{(\downarrow 13.07)}$	75.69$_{(\downarrow 9.47)}$	76.65$_{(\downarrow 9.30)}$	75.57$_{(\downarrow 10.33)}$
	BO	64.88$_{(\downarrow 9.63)}$	65.98$_{(\downarrow 17.26)}$	68.84$_{(\downarrow 15.82)}$	68.90$_{(\downarrow 15.50)}$	68.47$_{(\downarrow 17.00)}$	64.76$_{(\downarrow 12.07)}$	66.79$_{(\downarrow 16.96)}$	68.90$_{(\downarrow 16.26)}$	69.25$_{(\downarrow 16.70)}$	70.15$_{(\downarrow 15.75)}$
	FedHPO	61.75$_{(\downarrow 12.76)}$	70.31$_{(\downarrow 12.93)}$	74.11$_{(\downarrow 10.55)}$	75.44$_{(\downarrow 8.96)}$	76.89$_{(\downarrow 8.58)}$	63.63$_{(\downarrow 13.20)}$	73.21$_{(\downarrow 10.54)}$	76.45$_{(\downarrow 8.71)}$	77.15$_{(\downarrow 8.80)}$	77.69$_{(\downarrow 8.21)}$

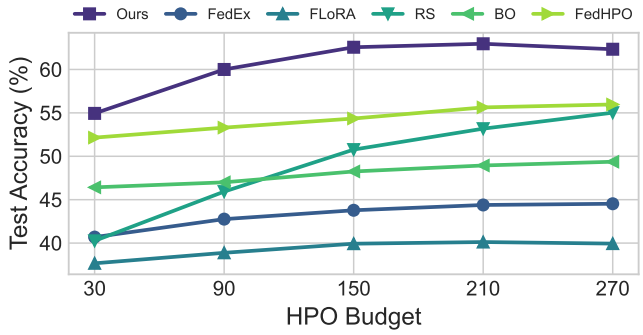

Fig. 3. Test accuracy of ResNet18 on CIFAR10 with different HPO budgets.

FLoRA while maintaining superior accuracy. This efficiency directly benefits edge computing environments with limited computational resources and communication bandwidth. Notably, our solution requires 88.9% fewer communication rounds than random search (30 versus 270 rounds) to achieve comparable performance (within 1% accuracy difference), significantly reducing operational costs in real-world FL deployments.

The observed efficiency stems from two operational innovations with significant practical implications: First, our gradient-based optimization eliminates trial-and-error exploration through direct hyperparameter updates via search gradients. This enables pFedDHPO to achieve 80.33% accuracy on ViT-Base within 30 communication rounds–a performance level that requires BO 150 rounds (5× more communication resources) to reach comparable accuracy (80.08%). Second, the credit assignment mechanism ensures robustness in challenging heterogeneous environments (Dirichlet $\alpha = 0.1$), where pFedDHPO maintains 74.51% accuracy on DeBERTaV3 within 30 rounds, while FedEx requires 66.7% more rounds to achieve similar performance. These improvements address key barriers to FL adoption by reducing communication overhead by 66.7–88.9% across diverse tasks compared to sampling-based methods.

The differentiation capability and personalized HPO strategy jointly contribute to these advantages. While personalized approaches (pFedDHPO, RS, BO) consistently outperform non-personalized baselines (FedEx, FLoRA) on ResNet18 and ViT-Base tasks, pFedDHPO achieves additional accuracy gains of 18.35-45.85% through gradient-aware optimization. These results validate the necessity of personalized HPO in Non-IID settings and demonstrate our method's dual strengths in communication efficiency and optimization effectiveness.

2) Effect of Data Heterogeneity: To systematically quantify algorithmic robustness across varying degrees of data heterogeneity, we leverage the Dirichlet distribution $\mathrm{Dir}(\alpha)$ for Non-IID client data partitioning. The probability density function (PDF) governing the class proportion vector $\mathbf{p}$ for each client follows:

$$f(\mathbf{p}|\alpha) = \frac{1}{B(\alpha)} \prod_{k=1}^{K} p_k^{\alpha-1}, \tag{16}$$

where $B(\alpha)$ denotes the multivariate beta function, and the concentration parameter α controls partition heterogeneity. Smaller values ($\alpha \to 0$) generate highly skewed Non-IID distributions, while larger values ($\alpha \geq 5.0$) approximate IID conditions. We evaluate four distinct client configurations under 30-round HPO budgets across multiple learning tasks. Complete numerical results are presented in Table II, and

TABLE II
TEST ACCURACY COMPARISON ACROSS DATASETS WITH VARYING α PARAMETERS. THE BEST ACCURACY IS MARKED IN BOLD, AND THE SECONDARY IS MARKED IN UNDERLINE. THE SUBSCRIPT $\downarrow_{x.x}$ IS THE PERFORMANCE DIFFERENCE FROM THE BEST RESULT, MARKED GREEN WHEN THE DIFFERENCE IS GREATER THAN 10%

		FedAvg					FedProx				
		$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1.0$	$\alpha = 5.0$	$\alpha = 10.0$	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1.0$	$\alpha = 5.0$	$\alpha = 10.0$
ResNet18 (@CIFAR10)	Ours	**54.94**$_{(\downarrow 0.00)}$	**65.69**$_{(\downarrow 0.00)}$	**68.12**$_{(\downarrow 0.00)}$	**69.92**$_{(\downarrow 0.00)}$	**69.96**$_{(\downarrow 0.00)}$	**56.07**$_{(\downarrow 0.00)}$	**66.83**$_{(\downarrow 0.00)}$	**68.89**$_{(\downarrow 0.00)}$	**69.83**$_{(\downarrow 0.00)}$	**69.03**$_{(\downarrow 0.00)}$
	FedEx	40.69$_{(\downarrow 14.25)}$	47.66$_{(\downarrow 18.03)}$	52.36$_{(\downarrow 15.76)}$	54.21$_{(\downarrow 15.71)}$	54.46$_{(\downarrow 15.50)}$	41.20$_{(\downarrow 14.87)}$	49.20$_{(\downarrow 17.63)}$	53.90$_{(\downarrow 14.99)}$	54.64$_{(\downarrow 15.19)}$	53.43$_{(\downarrow 15.60)}$
	FLoRA	37.67$_{(\downarrow 17.27)}$	48.61$_{(\downarrow 17.08)}$	57.07$_{(\downarrow 11.05)}$	60.86$_{(\downarrow 9.06)}$	61.16$_{(\downarrow 8.80)}$	38.44$_{(\downarrow 17.65)}$	49.13$_{(\downarrow 17.70)}$	57.04$_{(\downarrow 11.85)}$	60.39$_{(\downarrow 9.44)}$	62.33$_{(\downarrow 6.70)}$
	RS	40.26$_{(\downarrow 14.68)}$	54.89$_{(\downarrow 10.08)}$	61.91$_{(\downarrow 6.21)}$	64.63$_{(\downarrow 5.29)}$	66.21$_{(\downarrow 3.75)}$	42.68$_{(\downarrow 13.39)}$	56.75$_{(\downarrow 10.08)}$	62.81$_{(\downarrow 6.08)}$	65.28$_{(\downarrow 4.55)}$	65.85$_{(\downarrow 3.18)}$
	BO	46.42$_{(\downarrow 8.52)}$	59.32$_{(\downarrow 6.37)}$	65.18$_{(\downarrow 2.94)}$	66.88$_{(\downarrow 3.04)}$	67.13$_{(\downarrow 2.83)}$	44.70$_{(\downarrow 11.37)}$	61.59$_{(\downarrow 5.24)}$	65.35$_{(\downarrow 3.54)}$	66.05$_{(\downarrow 3.78)}$	67.44$_{(\downarrow 1.59)}$
	FedHPO	52.15$_{(\downarrow 2.79)}$	61.57$_{(\downarrow 4.12)}$	65.46$_{(\downarrow 2.66)}$	66.57$_{(\downarrow 3.35)}$	67.16$_{(\downarrow 2.80)}$	52.68$_{(\downarrow 3.39)}$	61.60$_{(\downarrow 5.23)}$	65.51$_{(\downarrow 3.38)}$	67.18$_{(\downarrow 2.65)}$	67.52$_{(\downarrow 1.51)}$
ViT-Base (@Office-Home)	Ours	**80.33**$_{(\downarrow 0.00)}$	**81.83**$_{(\downarrow 0.00)}$	**82.56**$_{(\downarrow 0.00)}$	**82.97**$_{(\downarrow 0.00)}$	**83.58**$_{(\downarrow 0.00)}$	**80.83**$_{(\downarrow 0.00)}$	**81.18**$_{(\downarrow 0.00)}$	**81.75**$_{(\downarrow 0.00)}$	82.29$_{(\downarrow 0.38)}$	82.40$_{(\downarrow 1.06)}$
	FedEx	48.25$_{(\downarrow 32.08)}$	60.84$_{(\downarrow 20.99)}$	69.02$_{(\downarrow 13.54)}$	74.51$_{(\downarrow 8.46)}$	77.19$_{(\downarrow 6.39)}$	49.37$_{(\downarrow 31.46)}$	62.01$_{(\downarrow 19.17)}$	69.11$_{(\downarrow 12.64)}$	74.14$_{(\downarrow 8.98)}$	77.85$_{(\downarrow 5.61)}$
	FLoRA	67.05$_{(\downarrow 13.28)}$	75.13$_{(\downarrow 6.70)}$	79.59$_{(\downarrow 2.97)}$	82.36$_{(\downarrow 0.61)}$	83.16$_{(\downarrow 0.42)}$	67.55$_{(\downarrow 13.26)}$	76.25$_{(\downarrow 4.93)}$	78.09$_{(\downarrow 3.66)}$	**83.12**$_{(\downarrow 0.00)}$	**83.46**$_{(\downarrow 0.00)}$
	RS	76.78$_{(\downarrow 3.55)}$	78.36$_{(\downarrow 3.47)}$	80.29$_{(\downarrow 2.27)}$	81.50$_{(\downarrow 1.47)}$	82.15$_{(\downarrow 1.43)}$	77.27$_{(\downarrow 3.56)}$	79.94$_{(\downarrow 1.24)}$	80.35$_{(\downarrow 1.40)}$	80.52$_{(\downarrow 2.60)}$	81.11$_{(\downarrow 2.35)}$
	BO	78.83$_{(\downarrow 1.50)}$	79.20$_{(\downarrow 2.63)}$	79.10$_{(\downarrow 3.46)}$	81.94$_{(\downarrow 1.03)}$	81.46$_{(\downarrow 2.12)}$	75.85$_{(\downarrow 4.98)}$	80.28$_{(\downarrow 0.90)}$	80.90$_{(\downarrow 0.85)}$	80.74$_{(\downarrow 2.38)}$	79.96$_{(\downarrow 3.50)}$
	FedHPO	76.14$_{(\downarrow 4.19)}$	78.90$_{(\downarrow 2.93)}$	81.51$_{(\downarrow 1.05)}$	82.25$_{(\downarrow 0.72)}$	83.58$_{(\downarrow 0.00)}$	77.40$_{(\downarrow 3.43)}$	78.48$_{(\downarrow 2.70)}$	81.25$_{(\downarrow 0.50)}$	82.25$_{(\downarrow 0.87)}$	82.64$_{(\downarrow 0.82)}$
DeBERTaV3 (@SST2)	Ours	**74.51**$_{(\downarrow 0.00)}$	**80.76**$_{(\downarrow 0.00)}$	**82.99**$_{(\downarrow 0.00)}$	**84.18**$_{(\downarrow 0.00)}$	**86.49**$_{(\downarrow 0.00)}$	**76.83**$_{(\downarrow 0.00)}$	**82.19**$_{(\downarrow 0.00)}$	**83.42**$_{(\downarrow 0.00)}$	**85.09**$_{(\downarrow 0.00)}$	86.07$_{(\downarrow 0.45)}$
	FedEx	67.45$_{(\downarrow 7.06)}$	71.61$_{(\downarrow 9.15)}$	78.48$_{(\downarrow 4.51)}$	85.87$_{(\downarrow 1.69)}$	85.94$_{(\downarrow 0.55)}$	67.66$_{(\downarrow 9.17)}$	75.36$_{(\downarrow 6.83)}$	80.76$_{(\downarrow 2.66)}$	84.57$_{(\downarrow 0.52)}$	**86.52**$_{(\downarrow 0.00)}$
	FLoRA	74.02$_{(\downarrow 0.49)}$	78.65$_{(\downarrow 2.11)}$	80.87$_{(\downarrow 2.12)}$	83.05$_{(\downarrow 2.82)}$	83.85$_{(\downarrow 2.64)}$	74.52$_{(\downarrow 2.13)}$	79.81$_{(\downarrow 2.38)}$	81.50$_{(\downarrow 1.92)}$	83.54$_{(\downarrow 1.55)}$	83.07$_{(\downarrow 3.45)}$
	RS	61.50$_{(\downarrow 13.01)}$	68.36$_{(\downarrow 12.40)}$	71.72$_{(\downarrow 11.27)}$	72.86$_{(\downarrow 13.01)}$	74.17$_{(\downarrow 12.32)}$	62.99$_{(\downarrow 13.84)}$	69.69$_{(\downarrow 12.50)}$	71.61$_{(\downarrow 11.81)}$	72.69$_{(\downarrow 12.40)}$	74.25$_{(\downarrow 12.27)}$
	BO	64.88$_{(\downarrow 9.63)}$	72.38$_{(\downarrow 8.38)}$	76.00$_{(\downarrow 6.99)}$	79.86$_{(\downarrow 6.01)}$	81.00$_{(\downarrow 5.49)}$	64.76$_{(\downarrow 12.07)}$	72.67$_{(\downarrow 9.52)}$	77.58$_{(\downarrow 5.84)}$	79.41$_{(\downarrow 5.68)}$	80.14$_{(\downarrow 6.38)}$
	FedHPO	61.75$_{(\downarrow 12.76)}$	72.07$_{(\downarrow 8.69)}$	78.43$_{(\downarrow 4.56)}$	81.36$_{(\downarrow 2.82)}$	82.33$_{(\downarrow 4.16)}$	63.63$_{(\downarrow 13.20)}$	74.07$_{(\downarrow 8.12)}$	79.07$_{(\downarrow 4.35)}$	81.25$_{(\downarrow 3.84)}$	82.33$_{(\downarrow 4.19)}$

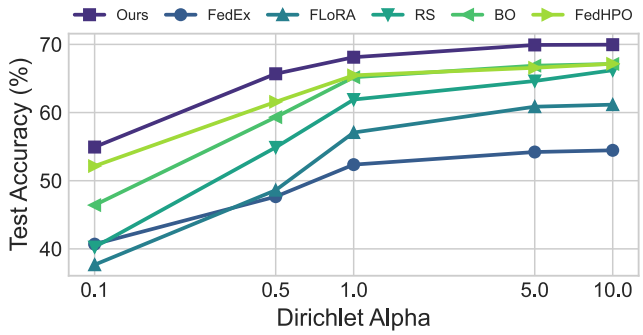

Fig. 4. Test accuracy of ResNet18 on CIFAR10 with different α parameters.

convergence patterns for ResNet18 on CIFAR10 are illustrated in Fig. 4.

Our experimental analysis reveals three principal findings:

1) *Universal α-Scaling Benefit:* All methods demonstrate progressive performance improvements with increasing α values (e.g., pFedDHPO accuracy increases from 54.94% to 69.96% on ResNet18 with CIFAR10), suggesting that reduced heterogeneity enhances model convergence rather than specifically improving HPO effectiveness.
2) *Heterogeneity-Aware Superiority:* pFedDHPO consistently outperforms baseline methods, achieving statistically significant ($p < 0.05$) absolute accuracy improvements of up to 14.25% at $\alpha = 0.1$.
3) *Robustness to Heterogeneity:* The performance gap between pFedDHPO and comparative methods widens as α decreases (Fig. 4), demonstrating superior resilience to client-side data heterogeneity during HPO execution.

These empirical results have significant implications for practical cross-device FL systems. pFedDHPO demonstrates robust performance under extreme heterogeneity ($\alpha \leq 0.5$), which is particularly valuable for deployment scenarios such as mobile edge computing and healthcare applications where natural data imbalances are common. Notably, pFedDHPO maintains 74.51% accuracy on DeBERTaV3 with $\alpha = 0.1$ (Table II), yielding absolute improvements of 7.06% and 0.49% over FedEx and FLoRA, respectively. This resilience in pathological Non-IID regimes directly addresses a key adoption barrier for production-grade FL systems. Furthermore, the method's consistent superiority across the entire α spectrum suggests fundamental improvements in client-specific hyperparameter adaptation, potentially enabling more reliable FL solutions for applications with diverse data distributions.

3) Effect of Client Scale: Scalability of HPO is critical in real-world FL deployments where the number of clients frequently exceeds laboratory-scale configurations. We systematically evaluate the scalability of pFedDHPO against baseline methods by varying the number of clients from 4 to 20 under a fixed HPO budget of 30 rounds, using Non-IID data distributions partitioned via a Dirichlet distribution ($\alpha = 0.1$). As shown in Table III and illustrated for ResNet-18 on CIFAR-10 in Fig. 5, our framework demonstrates superior scalability: pFedDHPO achieves 84.85% accuracy with 20 clients using ViT-Base, while FedEx experiences a 13.01% performance degradation under identical conditions.

TABLE III
ACCURACY EVALUATION IN FL SYSTEMS WITH DIFFERENT CLIENT SCALES. THE BEST ACCURACY IS MARKED IN BOLD, AND THE SECONDARY IS MARKED IN UNDERLINE. THE SUBSCRIPT $_{\downarrow x.x}$ IS THE PERFORMANCE DIFFERENCE FROM THE BEST RESULT, MARKED GREEN WHEN THE DIFFERENCE IS GREATER THAN 10%

		FedAvg					FedProx				
		$N=4$	$N=8$	$N=12$	$N=16$	$N=20$	$N=4$	$N=8$	$N=12$	$N=16$	$N=20$
ResNet18 (@CIFAR10)	Ours	**54.94**$_{(\downarrow 0.00)}$	**59.01**$_{(\downarrow 0.00)}$	**61.55**$_{(\downarrow 0.00)}$	**62.88**$_{(\downarrow 0.00)}$	**62.65**$_{(\downarrow 0.00)}$	**56.07**$_{(\downarrow 0.00)}$	**58.71**$_{(\downarrow 0.00)}$	**62.62**$_{(\downarrow 0.00)}$	**62.04**$_{(\downarrow 0.00)}$	**63.06**$_{(\downarrow 0.00)}$
	FedEx	40.69$_{(\downarrow 14.25)}$	42.38$_{(\downarrow 16.63)}$	43.72$_{(\downarrow 17.83)}$	42.51$_{(\downarrow 20.37)}$	38.47$_{(\downarrow 24.18)}$	41.20$_{(\downarrow 14.87)}$	43.86$_{(\downarrow 14.85)}$	43.93$_{(\downarrow 18.69)}$	42.61$_{(\downarrow 19.43)}$	41.27$_{(\downarrow 21.79)}$
	FLoRA	37.67$_{(\downarrow 17.27)}$	37.11$_{(\downarrow 21.90)}$	36.34$_{(\downarrow 25.21)}$	34.80$_{(\downarrow 28.08)}$	31.81$_{(\downarrow 30.84)}$	38.44$_{(\downarrow 17.63)}$	37.67$_{(\downarrow 21.04)}$	36.36$_{(\downarrow 26.26)}$	34.81$_{(\downarrow 27.23)}$	32.10$_{(\downarrow 30.96)}$
	RS	40.26$_{(\downarrow 14.68)}$	46.41$_{(\downarrow 12.60)}$	48.42$_{(\downarrow 13.13)}$	48.23$_{(\downarrow 14.65)}$	47.67$_{(\downarrow 14.98)}$	42.68$_{(\downarrow 13.39)}$	48.61$_{(\downarrow 10.10)}$	49.11$_{(\downarrow 13.51)}$	48.81$_{(\downarrow 13.23)}$	44.98$_{(\downarrow 18.08)}$
	BO	46.42$_{(\downarrow 8.52)}$	48.67$_{(\downarrow 10.34)}$	48.11$_{(\downarrow 13.44)}$	45.68$_{(\downarrow 17.20)}$	40.57$_{(\downarrow 22.08)}$	44.70$_{(\downarrow 11.37)}$	48.10$_{(\downarrow 10.61)}$	47.98$_{(\downarrow 14.64)}$	48.08$_{(\downarrow 13.96)}$	45.54$_{(\downarrow 17.52)}$
	FedHPO	52.15$_{(\downarrow 2.79)}$	51.82$_{(\downarrow 7.19)}$	51.57$_{(\downarrow 9.98)}$	49.83$_{(\downarrow 13.05)}$	46.44$_{(\downarrow 16.21)}$	52.68$_{(\downarrow 3.39)}$	52.33$_{(\downarrow 6.38)}$	51.03$_{(\downarrow 11.59)}$	50.07$_{(\downarrow 11.97)}$	47.68$_{(\downarrow 15.38)}$
ViT-Base (@Office-Home)	Ours	**80.33**$_{(\downarrow 0.00)}$	**82.34**$_{(\downarrow 0.00)}$	**83.11**$_{(\downarrow 0.00)}$	**83.97**$_{(\downarrow 0.00)}$	**84.85**$_{(\downarrow 0.00)}$	**80.83**$_{(\downarrow 0.00)}$	**81.89**$_{(\downarrow 0.00)}$	**83.55**$_{(\downarrow 0.00)}$	**82.96**$_{(\downarrow 0.00)}$	**84.01**$_{(\downarrow 0.00)}$
	FedEx	48.25$_{(\downarrow 32.08)}$	58.09$_{(\downarrow 24.25)}$	63.83$_{(\downarrow 19.28)}$	68.45$_{(\downarrow 15.52)}$	71.84$_{(\downarrow 13.01)}$	49.37$_{(\downarrow 31.46)}$	58.16$_{(\downarrow 23.73)}$	64.52$_{(\downarrow 19.03)}$	69.68$_{(\downarrow 13.28)}$	71.82$_{(\downarrow 12.19)}$
	FLoRA	67.05$_{(\downarrow 13.28)}$	68.54$_{(\downarrow 13.80)}$	71.99$_{(\downarrow 11.12)}$	74.09$_{(\downarrow 9.88)}$	75.72$_{(\downarrow 9.13)}$	67.55$_{(\downarrow 13.28)}$	68.59$_{(\downarrow 13.30)}$	72.42$_{(\downarrow 11.13)}$	74.65$_{(\downarrow 8.31)}$	78.19$_{(\downarrow 5.82)}$
	RS	76.78$_{(\downarrow 3.55)}$	80.51$_{(\downarrow 1.83)}$	81.70$_{(\downarrow 1.41)}$	81.90$_{(\downarrow 2.07)}$	81.18$_{(\downarrow 3.67)}$	77.27$_{(\downarrow 3.56)}$	80.73$_{(\downarrow 1.16)}$	81.91$_{(\downarrow 1.64)}$	81.89$_{(\downarrow 1.07)}$	81.48$_{(\downarrow 2.53)}$
	BO	78.83$_{(\downarrow 1.50)}$	80.02$_{(\downarrow 2.32)}$	80.26$_{(\downarrow 2.85)}$	81.33$_{(\downarrow 2.64)}$	80.35$_{(\downarrow 4.50)}$	75.85$_{(\downarrow 4.98)}$	80.13$_{(\downarrow 1.76)}$	81.37$_{(\downarrow 2.18)}$	80.56$_{(\downarrow 2.40)}$	80.27$_{(\downarrow 3.74)}$
	FedHPO	76.14$_{(\downarrow 4.19)}$	78.32$_{(\downarrow 4.02)}$	77.55$_{(\downarrow 5.56)}$	77.13$_{(\downarrow 6.84)}$	76.07$_{(\downarrow 8.78)}$	77.40$_{(\downarrow 3.43)}$	78.19$_{(\downarrow 3.70)}$	77.68$_{(\downarrow 5.87)}$	77.95$_{(\downarrow 5.01)}$	76.54$_{(\downarrow 7.47)}$
DeBERTaV3 (@SST2)	Ours	**74.51**$_{(\downarrow 0.00)}$	**76.96**$_{(\downarrow 0.00)}$	**77.81**$_{(\downarrow 0.00)}$	**76.65**$_{(\downarrow 0.00)}$	**75.39**$_{(\downarrow 0.00)}$	**76.83**$_{(\downarrow 0.00)}$	**78.60**$_{(\downarrow 0.00)}$	**78.65**$_{(\downarrow 0.00)}$	**77.46**$_{(\downarrow 0.00)}$	**76.85**$_{(\downarrow 0.00)}$
	FedEx	67.45$_{(\downarrow 7.06)}$	67.11$_{(\downarrow 9.85)}$	66.32$_{(\downarrow 11.49)}$	64.13$_{(\downarrow 12.52)}$	60.72$_{(\downarrow 14.67)}$	67.66$_{(\downarrow 9.17)}$	67.79$_{(\downarrow 10.81)}$	66.65$_{(\downarrow 12.00)}$	65.42$_{(\downarrow 12.04)}$	62.77$_{(\downarrow 14.08)}$
	FLoRA	74.02$_{(\downarrow 0.49)}$	74.01$_{(\downarrow 2.95)}$	73.34$_{(\downarrow 4.47)}$	71.72$_{(\downarrow 4.93)}$	71.19$_{(\downarrow 4.20)}$	74.52$_{(\downarrow 2.31)}$	74.23$_{(\downarrow 4.37)}$	73.90$_{(\downarrow 4.75)}$	72.07$_{(\downarrow 5.39)}$	71.93$_{(\downarrow 4.92)}$
	RS	61.50$_{(\downarrow 13.01)}$	67.23$_{(\downarrow 9.73)}$	69.62$_{(\downarrow 8.19)}$	68.24$_{(\downarrow 8.41)}$	64.80$_{(\downarrow 10.59)}$	62.99$_{(\downarrow 13.84)}$	68.46$_{(\downarrow 10.14)}$	70.25$_{(\downarrow 8.40)}$	70.11$_{(\downarrow 7.35)}$	65.81$_{(\downarrow 11.04)}$
	BO	64.88$_{(\downarrow 9.63)}$	66.23$_{(\downarrow 10.73)}$	65.89$_{(\downarrow 11.92)}$	64.94$_{(\downarrow 11.71)}$	63.53$_{(\downarrow 11.86)}$	64.76$_{(\downarrow 12.07)}$	66.62$_{(\downarrow 11.98)}$	67.36$_{(\downarrow 11.29)}$	65.58$_{(\downarrow 11.88)}$	63.78$_{(\downarrow 13.07)}$
	FedHPO	61.75$_{(\downarrow 12.76)}$	65.80$_{(\downarrow 11.16)}$	67.12$_{(\downarrow 10.69)}$	66.88$_{(\downarrow 9.77)}$	63.46$_{(\downarrow 11.93)}$	63.63$_{(\downarrow 13.20)}$	66.22$_{(\downarrow 12.38)}$	67.87$_{(\downarrow 10.78)}$	66.32$_{(\downarrow 11.14)}$	63.54$_{(\downarrow 13.31)}$

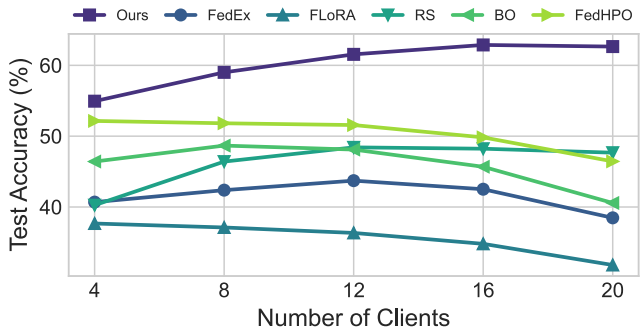

Fig. 5. Test accuracy of ResNet18 on CIFAR10 with different client scales.

This performance disparity stems from two key factors: 1) pFedDHPO leverages increased client participation through its dynamic hierarchical architecture, efficiently aggregating distributed data patterns without exponential communication overhead; 2) While baseline methods face combinatorial search complexity that scales as $\mathcal{O}(H^N)$ for N clients and H hyperparameters, our dual-level optimization approach constrains complexity to $\mathcal{O}(N)$. These computational advantages make pFedDHPO particularly well-suited for practical FL deployments in mobile or IoT environments, where client populations are large but per-device resources remain limited. Our empirical results validate the framework's effectiveness in production scenarios requiring both personalization and scalability.

4) Convergence of pFedDHPO: We conducted a case study to evaluate the convergence of pFedDHPO. Our FL system consists of three clients training ResNet18. For each client, pFedDHPO determines the optimal learning rate from a predefined set [1e-1, 5e-2, 1e-2, 5e-3, 1e-3]. We set the HPO budget for the experiment to 270 rounds, and the local dataset of each client follows a Dirichlet distribution with $\alpha = 0.1$. Fig. 7 illustrates the hyperparameters sampled by each client in each communication round. It is evident that the distribution parameters of client 1 and client 3 gradually converge to selecting a learning rate of 5e-2, while client 2 converges to selecting a learning rate of 1e-1. In the initial stage, with fewer than 100 communication rounds, the hyperparameters selected by each client change frequently. This indicates that pFedDHPO has not yet converged and is actively exploring to determine the optimal hyperparameters. However, after more than 200 rounds of communication, the variations in the hyperparameters decrease significantly. This suggests that pFedDHPO is approaching convergence, gaining a clearer assessment of the optimal hyperparameters for each client, and therefore prioritizing exploitation over exploration. Fig. 6 demonstrates the credits assigned to each client by pFedDHPO in each round of communication. As observed in the Fig. 6, the credit received by each client increases with the number of communication rounds, eventually reaching convergence after 270 rounds. Notably, client 3 receives fewer credits compared to the other clients, indicating its relatively low contribution to the global model's validation performance.

C. Statistical Significance Test

To rigorously validate the superiority of our proposed method over baseline approaches, we conducted Student's t-tests across three model architectures: ResNet18, ViT-Base, and DeBERTaV3. Table IV, Table V, and Table VI present the t-statistics, original and adjusted p-values

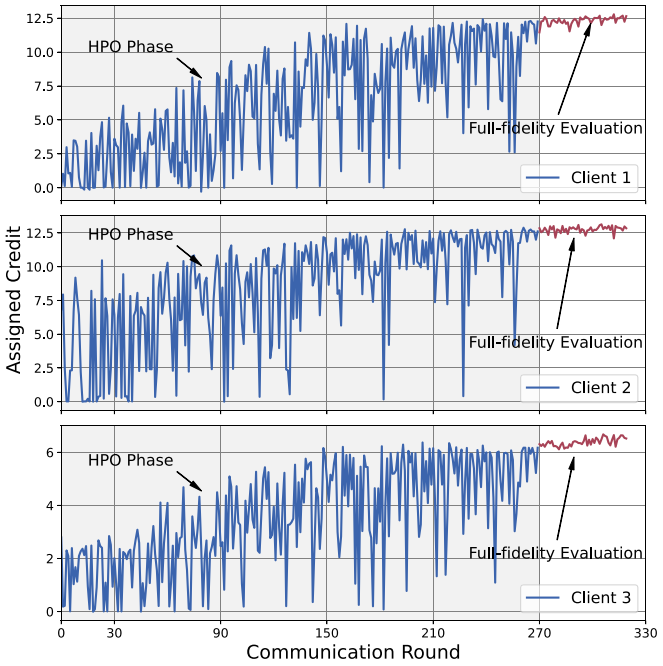

Fig. 6. Convergence behavior of client credits in pFedDHPO. Credits gradually increase and stabilize during training progression. Client 3 demonstrates significantly lower credit allocation, indicating diminished contribution to model validation performance.

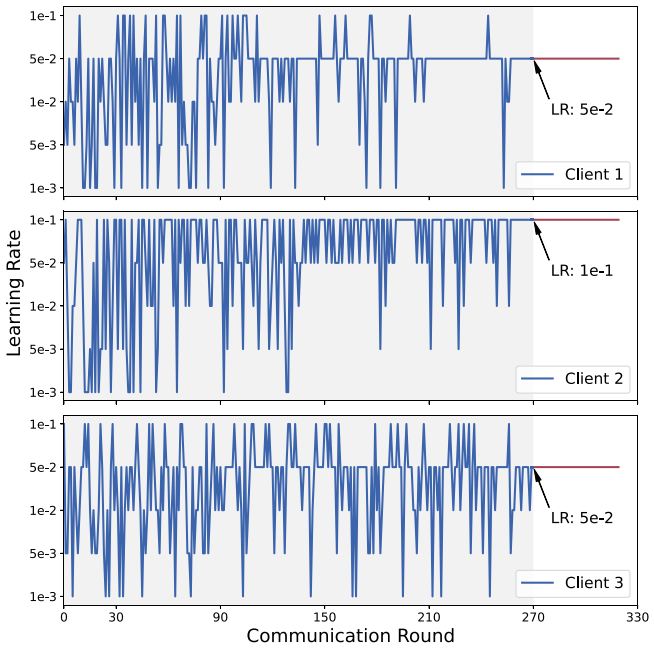

Fig. 7. Convergence of distribution parameters in pFedDHPO. Sampled hyperparameters per communication round exhibit progressively diminishing fluctuations, indicating stable convergence of the algorithm.

(Bonferroni-corrected for multiple comparisons), and significance outcomes for each baseline method.

The results consistently demonstrate statistically significant differences (adjusted $p < 0.05$) across all baseline comparisons, with exceptionally strong evidence ($p < 10^{-10}$) in most cases. Notably, FedEx and FLoRA exhibited the most pronounced performance differences in ViT-Base experiments,

TABLE IV
STUDENT'S T-TEST RESULTS FOR RESNET18 DATA

Baseline Method	Student's t-test			Significance
	T-Statistic	*Original p-value*	*Adjusted p-value*	
FedEx	13.747	3.114×10^{-14}	1.557×10^{-13}	✓
FLoRA	15.759	9.333×10^{-16}	4.666×10^{-15}	✓
RS	14.448	8.806×10^{-15}	4.403×10^{-14}	✓
BO	8.474	2.452×10^{-09}	1.226×10^{-08}	✓
FedHPO	4.124	2.856×10^{-04}	1.428×10^{-03}	✓

TABLE V
STUDENT'S T-TEST RESULTS FOR VIT-BASE DATA

Baseline Method	Student's t-test			Significance
	T-Statistic	*Original p-value*	*Adjusted p-value*	
FedEx	35.663	1.660×10^{-25}	8.300×10^{-25}	✓
FLoRA	18.875	7.835×10^{-18}	3.918×10^{-17}	✓
RS	5.622	4.511×10^{-06}	2.256×10^{-05}	✓
BO	3.356	2.223×10^{-03}	1.112×10^{-02}	✓
FedHPO	6.747	2.097×10^{-07}	1.049×10^{-06}	✓

TABLE VI
STUDENT'S T-TEST RESULTS FOR DEBERTAV3 DATA

Baseline Method	Student's t-test			Significance
	T-Statistic	*Original p-value*	*Adjusted p-value*	
FedEx	13.493	4.986×10^{-14}	2.493×10^{-13}	✓
FLoRA	13.326	6.818×10^{-14}	3.409×10^{-13}	✓
RS	20.611	7.252×10^{-19}	3.626×10^{-18}	✓
BO	15.208	2.353×10^{-15}	1.177×10^{-14}	✓
FedHPO	18.014	2.737×10^{-17}	1.369×10^{-16}	✓

as evidenced by large t-statistics (35.663 and 18.875, respectively) and extremely small p-values (8.30×10^{-25} and 3.92×10^{-17}). These findings suggest that our method achieves particularly substantial improvements when applied to vision transformer architectures, potentially because their parameter-intensive nature benefits more significantly from our proposed optimization strategy.

The DeBERTaV3 experiments demonstrated statistically significant improvements over all baselines. Although BO displayed relatively higher p-values in the ViT-Base experiments (1.11×10^{-2}), it maintained statistical significance, suggesting consistent yet more variable improvements relative to other baseline approaches.

Three key observations emerge from our analysis: (1) The progressive decrease in p-values from ResNet18 to DeBERTaV3 indicates that our method yields increasingly significant improvements as architectural complexity increases; (2) Statistical significance is maintained across all tests even after applying multiple comparison corrections, confirming the robustness of performance advantages; and (3) The magnitude of t-statistics correlates positively with model capacity, suggesting that the benefits of our optimization approach scale with architectural sophistication.

D. Practical Implications

The proposed pFedDHPO framework demonstrates significant potential for real-world deployment in industries where FL is widely adopted. Below, we discuss three representative application scenarios.

1) Healthcare: In medical institutions with strict data privacy requirements, pFedDHPO enables collaborative model training across hospitals without sharing local data. Its personalized HPO mechanism allows each institution to adaptively tune critical parameters (e.g., learning rates and regularization coefficients) based on heterogeneous data characteristics, including regional patient demographics or specialized imaging modalities (e.g., MRI protocols). This approach enhances diagnostic accuracy compared to conventional FL methods. Furthermore, the framework's credit assignment mechanism incentivizes participation by dynamically rewarding contributors based on their impact on the global model's validation performance, ensuring fair resource allocation in cross-institutional collaborations.

2) IoT Networks: pFedDHPO addresses key challenges in IoT ecosystems through its resource-efficient HPO process, achieving convergence within 270 communication rounds under severe data heterogeneity. This efficiency aligns with edge computing scenarios where devices generate Non-IID sensory data (e.g., temperature versus vibration sensors in industrial IoT). The framework's differentiable HPO backbone reduces computational overhead, enabling real-time adaptation for time-sensitive applications such as smart grid load forecasting.

3) Secure 6G Systems: The framework inherently supports secure 6G systems through adaptive security policy orchestration and distributed trust management. Its personalized hyperparameter optimization dynamically tailors context-aware configurations across heterogeneous network components, enabling edge nodes to autonomously adjust privacy preservation parameters such as differential privacy budgets based on local traffic patterns. Concurrently, user equipment can optimize lightweight encryption settings for power efficiency. The credit assignment mechanism enhances security by evaluating device reliability through validation performance, thereby implicitly identifying compromised nodes engaged in malicious activities like data poisoning. Furthermore, the differentiable optimization backbone ensures efficient resource utilization, which is critical for meeting 6G's stringent requirements in ultra-reliable low-latency scenarios, including real-time federated anomaly detection.

V. Limitations and Future Works

The proposed framework advances personalized HPO in FL, yet three key limitations warrant consideration. First, the validation-dependent reward mechanism assumes the server-maintained validation set $\mathcal{D}_{\text{val}}$ accurately represents local data distributions. In dynamic FL environments with non-stationary client data, this static validation set may fail to capture temporal distribution shifts, potentially resulting in suboptimal hyperparameter configurations. Second, our convergence analysis requires stable credit assignment across clients, which stabilizes after approximately 200 communication rounds (Fig. 6). Abrupt changes in client participation or significant data heterogeneity could disrupt this stabilization process, as demonstrated by the sensitivity to Dirichlet-α variations in Table II. Third, server-side gradient computations (e.g., $\partial L/\partial 1_i$ in Eq. (15)) introduce computational complexity of *O(N)*, which scales linearly with the number of clients *N*.

Four research directions are outlined to address these limitations:

1) Developing federated validation strategies that dynamically update $\mathcal{D}_{\text{val}}$ through privacy-preserving client feedback, eliminating dependence on static server-side validation data.
2) Designing adaptive HPO budget allocation mechanisms that optimize computational resource distribution under temporal heterogeneity, particularly for clients experiencing abrupt distribution shifts.
3) Formulating robust credit assignment algorithms with momentum-based adaptation and pseudo-label consistency metrics to enhance resilience against participation dynamics and label scarcity.
4) Extending the framework to unsupervised and semi-supervised scenarios through proxy validation metrics (e.g., clustering quality or pseudo-label consistency), enabling HPO with limited labeled data while preserving adaptation capabilities in Non-IID settings.

Additional priorities include supporting asynchronous update protocols and conducting formal theoretical analysis under non-stationary conditions. These enhancements aim to improve pFedDHPO's practicality in real-world FL deployments characterized by dynamic environments and varying levels of supervision.

VI. Conclusion

This paper introduced pFedDHPO, a novel personalized HPO framework designed specifically for FL. We formulated HPO as a differentiable optimization of distribution parameters across clients while incorporating adaptive credit assignment mechanisms. Our approach provided significant practical benefits for real-world FL deployments. The framework demonstrated computational efficiency under limited HPO budgets, making it suitable for resource-constrained environments like edge computing. pFedDHPO exhibited robust performance in the presence of substantial data heterogeneity and scaled effectively with increasing client populations, directly addressing fundamental limitations of conventional distributed learning systems. Empirical evaluations across various neural architectures confirmed consistent performance improvements in scenarios involving Non-IID data distributions and large-scale client cohorts. Although the current implementation required server-side validation data, future work will focus on eliminating this dependency to enhance applicability in privacy-sensitive domains where centralized data collection is restricted. These advancements established a foundation for practical FL systems that simultaneously addressed hardware constraints, data heterogeneity, and adaptive optimization requirements.

References

[1] Q. Li et al., "A survey on federated learning systems: Vision, hype and reality for data privacy and protection," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 4, pp. 3347–3366, Apr. 2023.

[2] J. Shen et al., "RingSFL: An adaptive split federated learning towards taming client heterogeneity," *IEEE Trans. Mobile Comput.*, vol. 23, no. 5, pp. 5462–5478, May 2023.
[3] J. Ouyang and Y. Liu, "Learning efficiency Maximization for wireless federated learning with heterogeneous data and clients," *IEEE Trans. Cogn. Commun. Netw.*, vol. 10, no. 6, pp. 2282–2295, Dec. 2024.
[4] P. Li et al., "Filling the missing: Exploring generative AI for enhanced federated learning over heterogeneous mobile edge devices," *IEEE Trans. Mobile Comput.*, vol. 23, no. 10, pp. 10001–10015, Oct. 2024.
[5] J. R. Zech, M. A. Badgeley, M. Liu, A. B. Costa, J. J. Titano, and E. K. Oermann, "Variable Generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study," *PLoS Med.*, vol. 15, no. 11, Nov. 2018, Art. no. e1002683.
[6] C. Catlett et al., "Measuring cities with software-defined sensors," *J. Soc. Comput.*, vol. 1, no. 1, pp. 14–27, 2020.
[7] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proc. Mach. Learn. Syst.*, 2020, pp. 429–450.
[8] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," in *Proc. 34th Conf. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 7611–7623.
[9] H. Wen, Y. Wu, J. Li, and H. Duan, "Communication-efficient federated data augmentation on non-IID data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2022, pp. 3377–3386.
[10] Z. Wang, W. Kuang, C. Zhang, B. Ding, and Y. Li, "FedHPO-B: A benchmark suite for federated hyperparameter optimization," 2022, *arXiv:2206.03966*.
[11] S. Mukherjee, N. Loizou, and S. U. Stich, "Locally adaptive federated learning," 2023, *arXiv:2307.06306*.
[12] J. L. Kim, M. T. Toghani, C. A. Uribe, and A. Kyrillidis, "Adaptive federated learning with auto-tuned clients," 2023, *arXiv:2306.11201*.
[13] W. Bao, T. Wei, H. Wang, and J. He, "Adaptive test-time Personalization for federated learning," in *Proc. 37th Conf. Neural Inf. Process. Syst.*, vol. 36, 2023, pp. 77882–77914.
[14] R. Lee et al., "FedL2P: Federated learning to personalize," in *Proc. 37th Conf. Neural Inf. Process. Syst.*, vol. 36, 2023, pp. 14818–14836.
[15] Y. Deng et al., "AUCTION: Automated and quality-aware client selection framework for efficient federated learning," *IEEE Trans. Parallel Distrib. Syst.*, vol. 33, no. 8, pp. 1996–2009, Aug. 2022.
[16] S. Chu et al., "Federated learning over wireless channels: Dynamic resource allocation and task scheduling," *IEEE Trans. Cogn. Commun. Netw.*, vol. 8, no. 4, pp. 1910–1924, Dec. 2022.
[17] D. Baymurzina, E. Golikov, and M. Burtsev, "A review of neural architecture search," *Neurocomputing*, vol. 474, pp. 82–93, Feb. 2022.
[18] W. Huang, M. Ye, and B. Du, "Learn from others and be yourself in heterogeneous federated learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 10143–10153.
[19] A. Fallah, A. Mokhtari, and A. Ozdaglar, "Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach," in *Proc. 34th Conf. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 3557–3568.
[20] H. Zhang, M. Tao, Y. Shi, X. Bi, and K. B. Letaief, "Federated multi-task learning with non-stationary and heterogeneous data in wireless networks," *IEEE Trans. Wireless Commun.*, vol. 23, no. 4, pp. 2653–2667, Apr. 2024.
[21] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An efficient framework for clustered federated learning," in *Proc. 34th Conf. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 19586–19597.
[22] S. Itahara, T. Nishio, Y. Koda, M. Morikura, and K. Yamamoto, "Distillation-based semi-supervised federated learning for communication-efficient collaborative training with non-IID private data," *IEEE Trans. Mobile Comput.*, vol. 22, no. 1, pp. 191–205, Jan. 2023.
[23] H. Larochelle, D. Erhan, A. Courville, J. Bergstra, and Y. Bengio, "An empirical evaluation of deep architectures on problems with many factors of variation," in *Proc. 24th Int. Conf. Mach. Learn.*, 2007, pp. 473–480.
[24] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, no. 2, pp. 281–305, 2012.
[25] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–9.
[26] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization," *J. Mach. Learn. Res.*, vol. 18, no. 185, pp. 1–52, 2018.
[27] K. Jamieson and A. Talwalkar, "Non-stochastic best arm identification and hyperparameter optimization," in *Proc. 19th Int. Conf. Artif. Intell. Stat.*, 2016, pp. 240–248.
[28] Y. Liu, X. Zhang, Z. Zeng, and S. Yu, "FedEco: Achieving energy-efficient federated learning by hyperparameter adaptive tuning," *IEEE Trans. Cogn. Commun. Netw.*, vol. 10, no. 6, pp. 2311–2324, Dec. 2024.
[29] M. Kundroo and T. Kim, "Federated learning with hyperparameter optimization," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 35, no. 9, 2023, Art. no. 101740.
[30] N. Mitic, A. Pyrgelis, and S. Sav, "How to privately tune hyperparameters in federated learning? insights from a benchmark study," 2024, *arXiv:2402.16087*.
[31] Y. Kang, Z. Ren, L. Fan, L. Yang, Y. Tong, and Q. Yang, "Hyperparameter optimization for secureboost via constrained multi-objective federated learning," 2024, *arXiv:2404.04490*.
[32] K. K. Nakka, A. Frikha, R. Mendis, X. Jiang, and X. Zhou, "Federated hyperparameter optimization through reward-based strategies: Challenges and insights," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2024, pp. 4236–4244.
[33] R. C. Patole and M. Adhikari, "Real-time prediction using fog-based federated learning and genetic hyperparameter optimisation," *IEEE Trans. Netw. Sci. Eng.*, vol. 11, no. 4, pp. 3905–3914, Jul./Aug. 2024.
[34] M. Khodak et al., "Federated hyperparameter tuning: Challenges, baselines, and connections to weight-sharing," in *Proc. 35th Conf. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 19184–19197.
[35] Y. Zhou, P. Ram, T. Salonidis, N. Baracaldo, H. Samulowitz, and H. Ludwig, "Single-shot hyper-parameter optimization for federated learning: A general algorithm & analysis," 2022, *arXiv:2202.08338*.
[36] H. Zhang et al., "Federated learning hyperparameter tuning from a system perspective," *IEEE Internet Things J.*, vol. 10, no. 16, pp. 14102–14113, Aug. 2023.
[37] S. Xie, H. Zheng, C. Liu, and L. Lin, "SNAS: Stochastic neural architecture search," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–17.
[38] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with Gumbel-softmax," in *Proc. Int. Conf. Learn. Represent.*, 2016, pp. 1–12.
[39] C. J. Maddison, A. Mnih, and Y. W. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–20.
[40] S. Hu et al., "DSNAS: Direct neural architecture search without parameter retraining," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 12084–12092.
[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[42] A. Krizhevsky, "Learning multiple layers of features from tiny images," Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada, Rep. TR-2009, 2009.
[43] B. Wu et al., "Visual transformers: Token-based image representation and processing for computer vision," 2020, *arXiv:2006.03677*.
[44] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5018–5027.
[45] P. He, J. Gao, and W. Chen, "DeBERTaV3: Improving DeBERTa using electra-style pre-training with gradient-disentangled embedding sharing," 2021, *arXiv:2111.09543*.
[46] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "GLUE: A multi-task benchmark and analysis platform for natural language understanding," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 353–355.

Jinglong Shen (Student Member, IEEE) received the B.S. degree in information engineering from Xidian University, Xi'an, China, in 2021, where he is currently pursuing the Ph.D. degree in information and communication engineering with the School of Telecommunications Engineering. His research interests include federated learning, wireless communication, and mobile edge computing. He was a recipient of the ICCC 2023 Best Demo Award. He has served as a Technical Program Committee Member for IEEE GlobeCom 2024-2025 and IEEE ICCC 2024-2025.

Nan Cheng (Senior Member, IEEE) received the B.E. and M.S. degrees from the Department of Electronics and Information Engineering, Tongji University, Shanghai, China, in 2009 and 2012, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Waterloo in 2016. He worked as a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, University of Toronto from 2017 to 2019. He is currently a Professor with the School of Electrical Engineering and Intelligentization, Dongguan University of Technology and also with the State Key Laboratory of ISN and the School of Telecommunications Engineering, Xidian University. He has published over 90 journal papers in IEEE Transactions and other top journals. His current research focuses on B5G/6G, AI-driven future networks, and space-air-ground integrated network. He serves as an Associate Editor for IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE OPEN JOURNAL OF VEHICULAR TECHNOLOGY, and *Peer-to-Peer Networking and Applications*, and serves/served as guest editors for several journals.

Wenchao Xu (Member, IEEE) received the B.E. and M.E. degrees in telecommunications engineering from Zhejiang University, Hangzhou, China, in 2008 and 2011, respectively, and the Ph.D. degree in electrical and computing engineering from the University of Waterloo, Canada, in 2018. He is a Research Assistant Professor with The Hong Kong Polytechnic University. In 2011, he joined Alcatel Lucent Shanghai Bell Company Ltd., where he was a software engineer for telecom virtualization. He has also been an Assistant Professor with the School of Computing and Information Sciences, Caritas Institute of Higher Education, Hong Kong. His research interests includes wireless communication, Internet of Things, distributed computing, and AI enabled networking.

Haozhao Wang (Member, IEEE) received the B.S. degree in computer science from the University of Electronic Science and Technology in 2016, and the Ph.D. degree from the School of Computer Science and Technology, Huazhong University of Science and Technology, where he worked as a Postdoctoral Fellow. He is currently an Assistant Professor with the School of Computer Science and Technology, Huazhong University of Science and Technology. He also worked as a Research Fellow with SLab, Nanyang Technical University. His research interests include distributed machine learning and multimodal learning.

Wei Quan (Senior Member, IEEE) received the Ph.D. degree in communication and information system from the Beijing University of Posts and Telecommunications, Beijing, China, in 2014. He is currently a Full Professor with the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing. He has coauthored more than 50 papers in prestigious international journals and conferences, including IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, *IEEE Communications Magazine*, IEEE WIRELESS COMMUNICATIONS, IEEE NETWORK, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and IEEE COMMUNICATIONS LETTERS. He is a Principle Investigator of the National Key Research and Development Program of China. His research interests focus on reliable transmission in mobile networks, vehicular networks, and industrial IoT. He is a winner of 2022 IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award. He is an Associate Editor of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, *Peer-to-Peer Networking and Applications*, *Journal of Internet Technology*, and IEEE ACCESS.

Xuemin (Sherman) Shen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is currently a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research interests include network resource management, wireless network security, the Internet of Things, 5G and beyond, and vehicular ad hoc and sensor networks. He received the R.A. Fessenden Award from IEEE, Canada, in 2019, the Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, the James Evans Avant Garde Award from the IEEE Vehicular Technology Society in 2018, the Joseph LoCicero Award in 2015 and Education Award in 2017 from the IEEE Communications Society, and the Technical Recognition Award from Wireless Communications Technical Committee in 2019 and AHSN Technical Committee in 2013. He has also received the Excellent Graduate Supervision Award from the University of Waterloo in 2006 and the Premier's Research Excellence Award from the Province of Ontario, Canada, in 2003. He served as the Editor-In-Chief of the IEEE INTERNET OF THINGS JOURNAL, IEEE NETWORK, and *IET Communications*. He served as the Technical Program Committee Chair/the Co-Chair for IEEE Globecom'16, IEEE Infocom'14, IEEE VTC'10 Fall, and IEEE Globecom'07, and the Chair for the IEEE Communications Society Technical Committee on Wireless Communications. He is the President Elect of the IEEE Communications Society. He was the vice president for technical and educational activities, the vice president for publications, a Member-At-Large on the Board of Governors, the Chair of the Distinguished Lecturer Selection Committee, and a member of IEEE Fellow Selection Committee of the ComSoc. He is a Registered Professional Engineer of Ontario, Canada, a Fellow of the Engineering Institute of Canada, the Canadian Academy of Engineering, and the Royal Society of Canada, a Foreign Member of the Chinese Academy of Engineering, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society.