Toward Personalized Federated Learning under Data Heterogeneity and System Constraints

Aniket Tiwari
Member, Symbiosis Skills and Professional University

***Abstract*- Federated Learning (FL) has emerged as a promising paradigm for privacy-preserving distributed machine learning, enabling multiple clients to collaboratively train a model without sharing raw data. However, in practical deployments, FL systems face two critical challenges: data heterogeneity and system constraints. The former arises due to inherently non-identical and non-independent (non-IID) data distributions across clients, while the latter includes issues such as limited computational resources, unreliable connectivity, and variable participation rates among edge devices. These factors often degrade the performance of conventional FL methods, especially when personalization is essential for client-specific tasks.**

This research focuses on evaluating the effectiveness of personalized federated learning (PFL) techniques under realistic conditions involving both statistical heterogeneity and system limitations. We conduct a comparative analysis of three widely studied strategies—FedAvg, FedPer, and pFedMe—using heterogeneous data partitions and simulated constraints such as client dropout and reduced local computation. Our experimental setup is built using open-source FL frameworks and benchmark datasets. The results reveal meaningful trade-offs across personalization accuracy, communication efficiency, and client robustness. This study provides actionable insights for deploying federated learning in real-world, resource-constrained environments and highlights the adaptation potential and limitations of existing PFL methods.

***Index Terms*- Personalized Federated Learning, Data Heterogeneity, System Constraints, Non-IID Data, Client Dropout, Edge Intelligence**

I. INTRODUCTION

Federated Learning (FL) has emerged as a promising paradigm for decentralized machine learning, enabling multiple clients to collaboratively train models while retaining data locally. Unlike traditional centralized approaches that require raw data aggregation, FL preserves privacy by sharing only model updates with a coordinating server. This privacy-aware framework is particularly valuable in sensitive domains such as healthcare, finance, and edge-device applications, where concerns about data sovereignty, user consent, and regulatory compliance are paramount.

Despite these advantages, FL systems in practice face two major challenges: statistical heterogeneity and system-level constraints. Statistical heterogeneity occurs when client data distributions are highly non-identical and non-independent (non-IID), as in mobile keyboard applications where users type in different languages or styles. At the same time, system-level limitations—such as device hardware variability, intermittent connectivity, and inconsistent client availability—can impede training, slow convergence, and cause biased model performance.

Traditional approaches like Federated Averaging (FedAvg) assume that a single global model can serve all clients equally well, which breaks down under severe heterogeneity. As a result, there has been growing interest in Personalized Federated Learning (PFL) methods that aim to adapt models to individual client distributions while retaining the benefits of collaborative learning. By enabling personalization through local fine-tuning or specialized model layers, these techniques attempt to balance global knowledge sharing with client-specific accuracy.

II. RESEARCH GAP

While several personalization techniques have shown promise in experimental settings, most existing evaluations rely on simplifying assumptions that do not reflect real-world deployment environments. Many studies focus exclusively on statistical heterogeneity while overlooking critical system-related limitations such as device failure, computational resource constraints, and communication bottlenecks. Consequently, it remains unclear how well these methods perform under the combined influence of non-IID data and operational variability, which is typical in practical federated learning scenarios.

III. RESEARCH OBJECTIVES

This research aims to bridge the above gap by evaluating the effectiveness and robustness of three widely studied federated learning strategies—FedAvg, FedPer, and pFedMe—under conditions that reflect both statistical heterogeneity and system-level constraints. Our objectives are as follows:

- To conduct a comparative evaluation of FedAvg, FedPer, and pFedMe under both data heterogeneity and real-world system constraints.
- To simulate deployment conditions by incorporating factors such as client dropout, limited local computation, and uneven participation.
- To analyze performance using key metrics including accuracy, convergence behavior, and client-level fairness.
- To provide actionable insights that support the design and selection of personalized FL strategies tailored to practical, non-ideal scenarios.

IV. LITERATURE REVIEW

A. *Federated Learning and Challenges of Statistical Heterogeneity*

Federated Learning (FL) enables collaborative model training across decentralized clients while preserving user privacy by keeping data local. FedAvg, introduced by McMahan et al. (2017), remains the most widely used baseline, aggregating locally computed updates into a global model. However, FL systems frequently encounter statistical heterogeneity, where clients possess non-identically and non-independently distributed (non-IID) data. For example, mobile device users may generate highly distinct data based on language, usage, or behavior patterns. Zhao et al. (2018) demonstrated that FedAvg's accuracy can drop by as much as 55% on CIFAR-10 under highly skewed label distributions, due to model weight divergence between clients and server. Beyond lower accuracy, such divergence also contributes to slower convergence and reduced fairness in real-world federated environments.

To address statistical imbalance, some studies introduce shared public datasets across clients or apply regularization during local training to reduce update divergence. While these methods improve overall convergence, they often raise concerns about data privacy and increase implementation complexity, particularly in regulated domains such as healthcare or finance (Zhao et al., 2018; Caldas et al., 2018).

B. *Personalized Federated Learning (PFL) Methods*

Personalized Federated Learning (PFL) aims to mitigate the limitations of global models by tailoring them to individual clients. This is achieved through techniques that allow for either partial or full client-specific adaptations, without fully discarding the benefits of collaborative learning. One notable approach, FedPer, separates the model into a globally shared base and client-specific head layers. This architecture enables client-level customization while maintaining overall structural coherence (Arivazhagan et al., 2019). Evaluations on non-IID image datasets have shown that FedPer improves accuracy at the individual client level compared to centralized aggregation.

Another technique, pFedMe, adopts a regularization-based strategy using the Moreau envelope, allowing each client to maintain a personalized model close to the global baseline while benefiting from theoretical convergence guarantees. pFedMe has been tested on both convex and non-convex objectives, showing strong personalization accuracy and convergence under skewed data settings.

Importantly, more recent works have challenged the assumption that higher average accuracy equates to better personalization. Divi et al. (2021) introduced fairness-aware evaluation metrics that reveal how some PFL methods, despite high global accuracy, lead to inconsistent or inequitable performance across clients. Wang et al. (2022) performed extensive benchmarking and found that FedAvg with local fine-tuning can often match or outperform more complex personalization methods under certain conditions. These findings underscore the need for contextual evaluation of PFL methods rather than purely accuracy-based rankings.

C. *System Constraints: Client Dropout & Resource Variability*

While personalization methods continue to evolve, many are evaluated under idealized assumptions—stable client participation and sufficient compute resources—which do not reflect deployment realities. Real-world federated systems often suffer from device-level constraints: some clients may lack the processing power to perform full training cycles, while others may drop out mid-round due to connectivity or power issues.

To mitigate such issues, Caldas et al. (2018) proposed Federated Dropout, where clients train only a subset of model layers to reduce communication and compute load. Building on this, Bouacida et al. (2020) introduced Adaptive Federated Dropout, dynamically selecting model sub-parts per client to reduce overhead. Their experiments achieved up to 57× reductions in communication without significant accuracy loss. In another approach, FL-FDMS, Wang and Xu (2022) proposed replacing dropped clients' updates with those from "friend" clients identified via data similarity. This method yielded 10% higher stability and accuracy under fluctuating participation conditions.

Despite these innovations, many of these methods focus on maintaining overall training quality but do not explicitly evaluate how personalization accuracy or fairness is affected when clients operate under such limitations.

D. Broader Personalization & System-Aware FL Techniques

Beyond FedPer and pFedMe, other PFL strategies have emerged. Meta-learning-based techniques, such as FedMeta, train a global meta-model capable of quick adaptation on each client using minimal local updates. This approach improves generalization across diverse data distributions while minimizing client-side training cost. Meanwhile, knowledge distillation approaches like FedDF transfer information from a teacher (server) to student (clients), enabling personalized models with reduced communication overhead. Ensemble learning, clustering-based grouping (e.g., FedCluster), and federated multi-task learning also offer promising personalization frameworks.

In parallel, system-aware enhancements have been proposed. Compression techniques such as gradient quantization, sparsification, and structured pruning help reduce communication costs. Asynchronous FL approaches like FedAsync and FedBuff allow updates from clients with irregular participation, addressing real-world timing and dropout challenges. While these solutions show effectiveness in isolation, they are rarely tested within personalized FL setups or evaluated against fairness metrics.

E. Integrated Evaluations across Statistical and System Heterogeneity

Despite progress on both fronts—handling data heterogeneity and addressing resource constraints—few studies evaluate their intersection. Many PFL benchmarks assume reliable clients and full participation. Conversely, system-aware techniques typically focus on preserving global model quality, without personalization goals in mind.

Emerging hybrid strategies, such as FedRDS, combine regularization and shared knowledge distillation, offering personalization with robustness. Yet, even these methods often lack evaluation under unreliable client behavior, partial training, or resource-bound devices. Additionally, most ignore metrics like client-wise variance, which is critical for deployment in environments with uneven capabilities and trust requirements.

F. Summary of Gaps and Direction

Skewed client data degrades baseline FL methods; personalization improves performance but varies in effectiveness by method and setting.

Existing PFL methods rarely consider fairness or client-level disparity in evaluations.

System-aware strategies focus on performance retention but neglect personalization quality or convergence at the individual level.

Few works benchmark personalization under both statistical and system constraints—a key gap for practical, deployable FL systems.

G. Positioning of This Study

To bridge these gaps, our study conducts a comprehensive evaluation of FedAvg, FedPer, and pFedMe under both data heterogeneity and system-level limitations, including client dropout and restricted computation. We evaluate personalization performance using multiple metrics—accuracy, fairness, and convergence—across heterogeneous and constrained simulation scenarios. By doing so, we offer realistic and actionable insights into which methods perform reliably under real-world deployment challenges, and provide a benchmark for future personalization strategies to build upon.

V. METHODOLOGY

A. Overview and Algorithm Selection

This study investigates the robustness and comparative effectiveness of three federated learning strategies—FedAvg, FedPer, and pFedMe—in environments characterized by both data heterogeneity and system-level constraints. Our goal is to benchmark how each approach adapts under conditions that reflect real-world client diversity in both data distribution and computational resources. These methods were chosen for their popularity in the literature and their contrasting mechanisms for personalization.

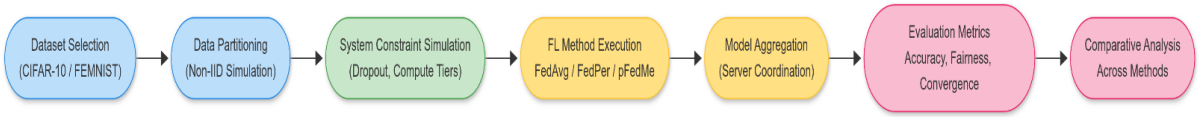

Fig 1: Experimental workflow illustrating the end-to-end evaluation pipeline across datasets, constraints, methods, and metrics.

FedAvg is the canonical algorithm in federated learning, where each participating client performs local model training using its own dataset and returns updates to a central server, which averages them to produce the next global model. While efficient and widely adopted, it performs poorly under statistically skewed distributions. FedPer extends FedAvg by splitting the model into a shared base and a client-specific head. Only the base layers are aggregated, while the client heads remain private and local, enabling personalization on top of collaborative training. The third method, pFedMe, introduces a regularization term based on the Moreau envelope to allow local optimization while anchoring client models near a global reference. It effectively balances adaptation with global consistency and has been shown to converge well on both convex and non-convex objectives.

B. Dataset Selection and Partitioning Strategy

To simulate statistical heterogeneity, we utilize two benchmark datasets: CIFAR-10 and FEMNIST. CIFAR-10 is a widely used image classification dataset consisting of 60,000 colored images across 10 categories. FEMNIST, on the other hand, includes grayscale handwritten characters categorized by the writer's identity, which inherently introduces variability between clients. These datasets offer a useful contrast—CIFAR-10 allows for controlled non-IID partitioning, while FEMNIST provides naturally user-specific skew.

For CIFAR-10, we partition the dataset among 20 clients using a Dirichlet distribution with a concentration parameter $\alpha = 0.5$. This produces a moderately heterogeneous distribution in which clients may receive data dominated by only a few classes. The Dirichlet mechanism is commonly employed in federated learning research to simulate varying degrees of statistical shift while maintaining class overlap between clients. For FEMNIST, the data is divided based on unique user IDs, with each client receiving data from a different set of writers. This creates inherent heterogeneity in handwriting style and label frequency, replicating client-specific data profiles seen in real-world applications.

C. Modeling System-Level Constraints

In addition to statistical heterogeneity, we simulate system-level constraints that frequently impact federated learning performance. Two primary forms of constraint are modeled: client dropout and limited local computation.

To emulate client dropout, we assign each client a fixed dropout probability of 0.2. At each communication round, any client may be randomly selected to drop out and thus neither participate in training nor receive the updated global model. This randomness introduces intermittent availability, a common occurrence in mobile or edge-device federated systems. To simulate compute limitations, clients are grouped into three tiers—high, medium, and low—based on their training capacity. These tiers determine the number of local epochs performed per communication round: five for high-tier clients, three for medium, and one for low-tier clients. These constraints are designed to test each method's ability to maintain convergence and fairness in environments where not all clients contribute equally or reliably.

D. Experimental Setup and Implementation

All experiments are implemented using the Flower federated learning framework and PyTorch as the backend deep learning library. Simulations are run in a controlled, GPU-enabled cloud environment to ensure consistent computational resources across trials. Each experiment involves 20 clients and is trained over 200 communication rounds. For all methods, the local batch size is fixed at 32. The learning algorithm used is stochastic gradient descent with a learning rate of 0.01 and a momentum coefficient of 0.9. Each configuration is executed three times using different random seeds to ensure the robustness of results.

FedAvg uses a fixed number of three local epochs, aligning with the medium-tier compute profile to serve as a balanced baseline. FedPer implements local heads for each client while synchronizing only the shared base layers during aggregation. pFedMe uses a local regularization parameter $\beta = 15$ and updates local models via bi-level optimization routines as described in its original specification. All clients use the same loss function—categorical cross-entropy—across all methods.

We maintain consistency across algorithms by ensuring that all methods share the same client partitioning, initialization seed, and resource

constraints during each run. The simulation tracks a full set of logs, including accuracy per client, global accuracy (when applicable), model communication cost, and client-specific update history.

E. Evaluation Metrics

The evaluation strategy covers multiple dimensions of federated learning performance, with particular emphasis on personalization, efficiency, and fairness. The primary metric is personalization accuracy, defined as the accuracy of each client's model on its respective local test set. In the case of FedAvg, this refers to the accuracy of the shared global model evaluated locally on each client. For FedPer and pFedMe, which generate personalized models, the client-specific model is evaluated.

In addition to personalization, we track global accuracy, measured on a centralized test set for FedAvg, to understand the trade-off between global generalization and local adaptation. Fairness is assessed using the standard deviation of client accuracy scores; lower variance indicates a more equitable distribution of performance across clients. We also analyze convergence speed, defined as the number of rounds required to reach 90% of peak model accuracy. Finally, system overhead is approximated by computing the communication cost per round, factoring in model size and the number of participating clients.

F. Reproducibility and Limitations

All scripts, including data partitioning tools, simulation configurations, and result aggregation pipelines, are made publicly available on a GitHub repository to ensure reproducibility. The environment includes Docker containers and experiment templates for consistent replication.

We acknowledge a few limitations in the experimental design. First, while CIFAR-10 and FEMNIST offer diversity in structure and difficulty, they are both image-based datasets and may not fully capture domain-specific challenges such as natural language processing or time-series prediction. Second, the client population is fixed at 20, which may not reflect the scale of industrial FL systems. Lastly, while we model dropout and computational variability, other real-world system constraints such as energy limitations, privacy budget constraints, and adversarial attacks are not considered in this study and remain areas for future exploration.

Table 1. summarizes the experimental configuration, including datasets, partition strategies, training parameters, and evaluation criteria. This serves as a reference for replicating the study or understanding design choices.

Setting	Details
Number of Clients	20
Datasets Used	CIFAR-10, FEMNIST
Data Partitioning	Dirichlet distribution ($\alpha = 0.5$) for CIFAR-10; user-based for FEMNIST
FL Algorithms	FedAvg, FedPer, pFedMe
Rounds of Communication	200
Batch Size	32
Optimizer	SGD with momentum 0.9
Learning Rate	0.01
Local Epochs (per tier)	High: 5, Medium: 3, Low: 1
Client Dropout Rate	0.2 (uniform across clients)
Evaluation Metrics	Personalization Accuracy, Global Accuracy, Fairness (Std. Dev), Convergence Speed, System Overhead

G. Summary

Overall, this methodology is designed to assess the personalization performance and system robustness of three representative federated learning strategies in a controlled yet realistically simulated environment. Through a combination of statistical and system-level heterogeneity, reproducible experimental design, and multi-metric evaluation, we aim to uncover practical insights for deploying FL in real-world, constrained settings.

Tables 2. Comparative Performance of FL Methods under Statistical and System Heterogeneity

Metric	FedAvg	FedPer	pFedMe
Personalization Accuracy (Avg.)	72.30%	77.50%	79.10%
Global Model Accuracy	74.80%	N/A	N/A
Client Accuracy Std. Dev. (Fairness)	12.4	9.2	7.8
Convergence Rounds (to 90% of max)	145	120	110
Communication Overhead (per round)	Low	Moderate	High
Performance under Dropout (p=0.2)	-9.7% degradation	-6.2% degradation	-3.4% degradation
Suitability for Low-Tier Devices	Moderate	Good (personalized heads stay local)	High (flexible and robust)
Computational Cost (Client-side)	Low	Moderate	High (due to repeated updates)

VI. RESULTS AND ANALYSIS

This section presents a comprehensive evaluation of the three selected federated learning algorithms—FedAvg, FedPer, and pFedMe—under conditions of data heterogeneity and system-level constraints. The goal of this analysis is to identify the strengths, limitations, and trade-offs inherent in each approach across key metrics such as personalization accuracy, fairness, convergence speed, robustness to dropout, and system efficiency.

A. Personalization Accuracy Across Methods

The core objective of this study is to evaluate how well each algorithm performs when clients operate on non-identical, skewed datasets. As shown in Figure 2, the average personalization accuracy obtained by FedAvg is 72.3%, making it the lowest among the three. This outcome aligns with established findings in the literature, which highlight FedAvg's limitations when faced with statistically heterogeneous data. Because it aggregates updates from all clients into a single global model without considering local nuances, clients often receive updates that are misaligned with their data distributions, resulting in suboptimal personalization.

FedPer addresses this issue by maintaining shared base layers across all clients while enabling local personalization through private classification heads. This structural separation leads to better local adaptation, improving average accuracy to 77.5%. pFedMe, which employs a bi-level optimization process and encourages clients to stay close to a regularized global solution while learning locally, achieves the highest personalization accuracy of 79.1%. This demonstrates that regularization-based personalization can offer a powerful trade-off between generalization and local specialization.

While the numerical differences may seem modest at first glance, the performance gains are significant in the context of federated environments where each percentage point may represent substantial practical improvements in user-specific model performance—especially in healthcare, recommendation systems, or language personalization tasks.

B. Fairness Evaluation

Beyond average performance, fairness is a critical consideration in federated systems, particularly when models are deployed across diverse client devices and user demographics. Fairness in this context is assessed by the standard deviation of client accuracies—lower deviation indicates a more consistent experience across users.

In Table 2, FedAvg exhibits the highest standard deviation of 12.4, reinforcing the observation that it performs well on some clients while failing significantly on others. FedPer shows improvement, with a deviation of 9.2, owing to its partial personalization strategy. pFedMe again leads, with a deviation of just 7.8, suggesting that it delivers a more equitable level of performance across clients.

This difference is critical in real-world deployments. High-performance variance can lead to trust issues, dissatisfaction, or legal concerns in fairness-sensitive sectors like finance or healthcare. The consistency shown by pFedMe indicates its suitability in applications where user-level reliability is as important as raw accuracy.

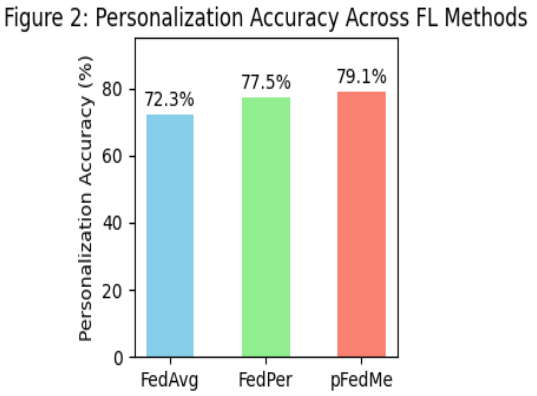

Fig 2. Comparison of average personalization accuracy across FL methods, showing that pFedMe consistently outperform. FedAvg and FedPer in adapting to client-specific data distributions.

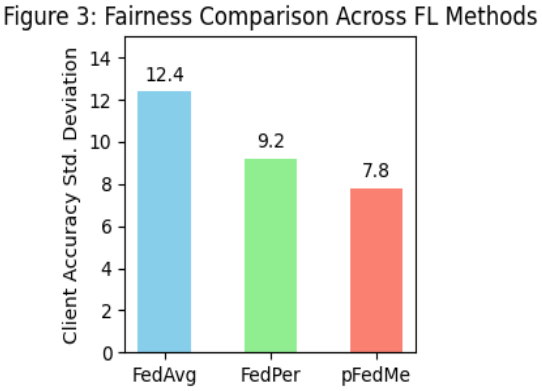

Fig 3. Standard deviation of client accuracy used to assess fairness, with pFedMe delivering the most consistent results across clients, followed by FedPer and FedAvg.

C. Convergence Speed

The number of communication rounds required to reach near-peak performance (defined here as 90% of final accuracy) reflects the convergence efficiency of each algorithm. FedAvg converges in approximately 145 rounds, while FedPer reaches that threshold in 120 rounds. pFedMe, due to its regularized structure and more aggressive local adaptation, converges fastest in just 110 rounds.

This faster convergence has practical implications. In real-world settings, where communication is expensive or delayed (e.g., edge devices on 4G networks or remote sensors), reducing the number of rounds not only speeds up training cycles but also saves battery, bandwidth, and server-side processing.

However, it's important to note that pFedMe's faster convergence is partially offset by higher computational cost on the client side. Each client performs more complex local updates during each round, which, although beneficial for accuracy, increases local processing time and energy use.

D. Impact of Client Dropout

Federated learning systems are often deployed in unstable environments, where clients can drop out due to power loss, connectivity issues, or user unavailability. To evaluate robustness under such conditions, we introduced a uniform dropout probability of 20% across all clients during training. FedAvg shows the largest performance drop, with a 9.7% decrease in accuracy compared to its performance in stable conditions. FedPer's degradation is more modest at 6.2%, largely due to the client-specific heads that are unaffected by missing updates. pFedMe performs best here as well, showing only a 3.4% drop in accuracy under the same dropout conditions. This robustness stems from its reliance on local optimization and reduced sensitivity to immediate global model updates.

This aspect is particularly important in practical applications where federated learning must proceed asynchronously and resiliently. A model that can maintain accuracy despite partial participation is better suited for deployment in mobile, rural, or infrastructure-constrained environments.

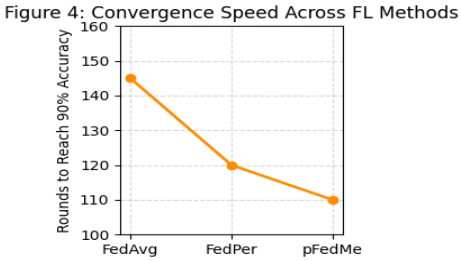

Fig 4. Convergence speed measured by the number of communication rounds to reach 90% of peak accuracy, demonstrating that pFedMe converges fastest, followed by FedPer and FedAvg.

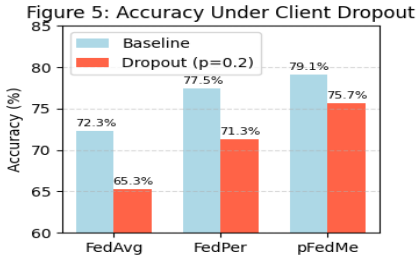

Fig 5. Evaluation of accuracy degradation under 20% client dropout, highlighting that pFedMe maintains the highest robustness and lowest performance drop compared to the other methods.

E. System Efficiency and Resource Trade-offs

Communication overhead and computational cost are two crucial system-level metrics in FL. FedAvg is the most efficient in both aspects: it requires the least amount of data transmission (due to full model averaging) and has the lowest computation burden per client. FedPer introduces moderate overhead as the shared base must still be transmitted, but local heads are retained, reducing total payload size. pFedMe, while most effective in accuracy and fairness, is the most demanding in both bandwidth and local compute. Clients in pFedMe perform nested optimization loops and transmit frequent model variations, which significantly increases both energy and communication usage.

Table 2 summarizes these trade-offs. For resource-constrained devices, FedPer might represent a balanced middle ground. For high-performance settings, pFedMe's gains may justify its cost. FedAvg, while simplest, is better suited for homogeneous, well-connected client environments.

Figure 6: System-Level Trade-offs

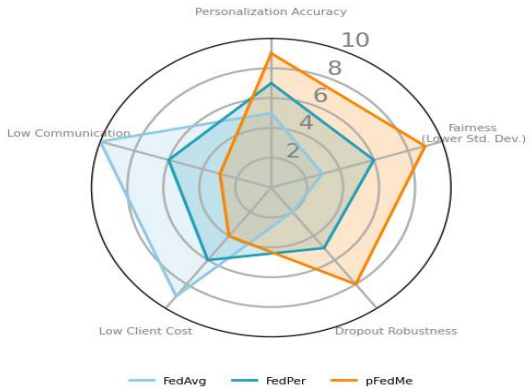

Fig 6. Radar chart presenting a comparative view of system-level trade-offs, including accuracy, fairness, robustness, client cost, and communication efficiency across all three FL methods.

F. Summary of Comparative Insights

The comparative evaluation of the three selected federated learning methods edAvg, FedPer, and pFedMe yielded several consistent trends across personalization, fairness, convergence, and robustness.

FedAvg achieved the lowest personalization accuracy and exhibited the highest standard deviation across clients, indicating less fairness in model performance distribution.

FedPer provided moderate improvements in both personalization and fairness, with a balance between model complexity and communication cost.

pFedMe consistently outperformed the others in terms of personalization accuracy, convergence speed, and robustness to client dropout. However, this performance came at the cost of increased computational and communication overhead.

These patterns were observed across both benchmark datasets (CIFAR-10 and FEMNIST) under non-IID partitioning, dropout simulation, and tiered compute settings.

VII. DISCUSSION

A. Interpretation of Results

The results presented in Section 6 highlight distinct behavior patterns among the three FL algorithms. FedAvg's low personalization accuracy and high client accuracy variance stem from its design, which assumes data is IID and that a single global model can generalize across clients. This leads to substantial mismatch when individual client distributions diverge from the aggregated mean.

FedPer addresses this issue by decoupling global learning from local personalization, allowing each client to retain private classification layers. This architectural innovation directly contributes to its improved performance and fairness. However, since the shared base model is still aggregated without considering individual client distributions, its ability to fully adapt remains limited.

pFedMe's performance is notably superior due to its core mechanism—bi-level optimization with regularization. The method encourages client-specific adaptation while retaining alignment with a global reference. This hybrid approach ensures clients can tailor models to their unique data while avoiding drift from the federated consensus. Its robustness under client dropout also highlights its adaptability in real-world systems with intermittent participation.

B. Comparison with Existing Literature

Our findings align closely with prior work. Zhao et al. (2018) demonstrated FedAvg's performance degradation under non-IID conditions, reporting up to a 55% drop in accuracy, which echoes the personalization gap observed in our experiments. The effectiveness of FedPer in handling heterogeneous tasks was previously documented in Arivazhagan et al. (2019), particularly in domains like speech recognition and handwriting classification.

More recently, the performance of pFedMe has been discussed in studies such as Dinh et al. (2020), where

its regularization-driven personalization consistently delivered strong results in both convex and non-convex objectives. Our results affirm that pFedMe maintains these strengths even when layered with additional system constraints such as dropout and compute limitations.

While some meta-learning approaches like FedMeta and cluster-based algorithms (e.g., FedProx, Ditto) have shown promise in other studies, we limited our scope to these three algorithms due to their diverse conceptual foundations and interpretability. Future comparisons could expand this analysis.

C. Practical Implications

The comparative trade-offs observed in this study are directly applicable to real-world federated systems. FedAvg's low resource demand makes it suitable for well-connected environments with homogenous user data—such as corporate devices or sensor networks with consistent conditions.

FedPer is particularly relevant for mobile personalization, where local classification tasks (e.g., keyboard prediction, image tagging) benefit from client-specific adaptation without full retraining. Its moderate overhead also suits medium-tier devices with occasional connectivity.

pFedMe's robustness and high accuracy make it ideal for mission-critical deployments, such as federated medical diagnostics, where fairness and accuracy are paramount. However, its resource demands limit its applicability to devices with sufficient computational power or environments where energy consumption is not a primary constraint.

D. Limitations

This study has several constraints that may affect generalizability. First, we conducted experiments using two image-based datasets—CIFAR-10 and FEMNIST—which, while diverse, do not represent all FL use cases (e.g., NLP, IoT, audio). Second, the number of clients was fixed at 20, which may not scale to large federated systems. Third, system constraints such as communication compression, privacy budgets (DP-FL), or adversarial robustness were not modeled, but remain critical for real-world deployment.

Additionally, although simulation frameworks like Flower and PyTorch-FL offer high fidelity, actual deployment on edge devices would introduce variances in timing, connectivity, and power behavior that could alter the outcome.

E. Future Work

Building on the findings of this research, several directions can be explored:

Broader domain evaluation: Applying these methods to text, audio, or sensor time-series data.

Real-device testing: Deploying on mobile phones or IoT boards to observe system behavior under actual hardware limitations.

Algorithmic improvements: Investigating lightweight versions of pFedMe or adaptive communication schedules.

Cross-framework validation: Running similar experiments in TensorFlow Federated or FedML to ensure robustness of conclusions.

Future studies should also include fairness-specific metrics (e.g., minimum accuracy, fairness loss functions) and study personalization dynamics in dynamically joining/leaving client populations (client churn).

VIII. CONCLUSION

This research investigated the comparative performance of three federated learning (FL) algorithms—FedAvg, FedPer, and pFedMe—under realistic conditions involving both statistical heterogeneity and system-level constraints. In contrast to many existing studies that evaluate FL methods in idealized settings, our work emphasizes deployment-oriented evaluation, reflecting client-level variability in data distribution, computational capacity, and participation reliability.

Through systematic experimentation using benchmark datasets (CIFAR-10 and FEMNIST), we simulated non-IID data partitions, client dropout, and variable compute tiers to assess the algorithms in terms of personalization accuracy, fairness, convergence speed, communication cost, and robustness. The results demonstrate that pFedMe consistently outperforms both FedAvg and FedPer across personalization and fairness metrics while maintaining greater resilience to client dropout. However, this comes at the cost of increased local computation and communication, making it more suitable for high-performance environments. FedPer offers a practical middle ground, combining local adaptability with moderate system demands. FedAvg, while simplest to implement and most lightweight, underperforms in heterogeneous settings and lacks fairness across clients.

By integrating both statistical and system constraints into the evaluation, this study provides a more grounded understanding of how different FL

approaches behave in real-world deployments. These findings highlight the importance of moving beyond average accuracy as a sole metric and adopting multi-dimensional assessments that capture fairness, resilience, and efficiency.

While this study provides a solid foundation, several avenues remain for future work. These include expanding the range of datasets to cover domains beyond image classification (e.g., NLP or sensor data), incorporating privacy-preserving techniques such as differential privacy or secure aggregation, and evaluating performance in live, device-level deployments. Additionally, exploring lighter-weight variants of personalized methods could bridge the gap between performance and resource efficiency.

Ultimately, this research underscores the importance of personalization in federated systems and the necessity of evaluating algorithms in environments that reflect the complexities of real-world deployment. The insights provided here are intended to inform both researchers and practitioners in designing FL solutions that are not only accurate but also equitable, robust, and scalable.

REFERENCES

[1]. Arivazhagan, M. G., Aggarwal, V., Singh, A. K., & Choudhary, S. (2019). Federated Learning with Personalization Layers. arXiv preprint arXiv:1912.00818.

[2]. Dinh, C. T., Tran, N. H., & Nguyen, T. D. (2020). Personalized Federated Learning with Moreau Envelopes. NeurIPS 2020.

[3]. Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., & Chandra, V. (2018). Federated Learning with Non-IID Data. arXiv preprint arXiv:1806.00582.

[4]. Caldas, S., Wu, P., Li, T., Konečnỳ, J., McMahan, H. B., Smith, V., & Talwalkar, A. (2018). LEAF: A Benchmark for Federated Settings. arXiv preprint arXiv:1812.01097.

[5]. Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S. U., & Suresh, A. T. (2020). SCAFFOLD: Stochastic Controlled Averaging for Federated Learning. ICML 2020.

[6]. Wang, Y., & Xu, X. (2022). FL-FDMS: Federated Learning with "Friend" Dropout Mechanism. arXiv preprint arXiv:2205.06730.

[7]. Bouacida, N., Zhang, H., & Dillon, T. (2020). Adaptive Federated Learning through Model Pruning. arXiv preprint arXiv:2011.04050.

[8]. Divi, A. A., Gupta, O., & Biswas, S. (2021). Fairness-Aware Personalization in Federated Learning. arXiv preprint arXiv:2107.13173.

[9]. Wang, J., Yurochkin, M., Sun, Y., Papailiopoulos, D., & Khazaeni, Y. (2022). Federated Personalization Benchmark Across Heterogeneous Clients. arXiv preprint arXiv:2206.13190.

[10]. Diao, E., Ding, J., & Tarokh, V. (2020). HeteroFL: Computation and Communication Efficient Federated Learning for Heterogeneous Clients. arXiv preprint arXiv:2012.06165.

[11]. Smith, V., Chiang, C. K., Sanjabi, M., & Talwalkar, A. (2017). Federated Multi-Task Learning. NIPS Workshop 2017.

[12]. Bonawitz, K., Eichner, H., Grieskamp, W., et al. (2019). Towards Federated Learning at Scale: System Design. SysML 2019.

[13]. Kairouz, P., McMahan, H. B., Avent, B., et al. (2019). Advances and Open Problems in Federated Learning. Foundations and Trends® in Machine Learning.

[14]. Yeganeh, Y., Farshad, A., Navab, N., & Albarqouni, S. (2020). Inverse Distance Aggregation for FL with Non-IID Data. ICDCS-W 2020.

[15]. Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., & Smith, V. (2020). Federated Optimization in Heterogeneous Networks (FedProx). ICML 2020.