Appendix

We provide details omitted in the main paper.

- Appendix A: additions to the description of FedBPG.
- Appendix B: detailed experiment setup
- Appendix C: additional experimental setup and evaluations
- Appendix D: the advantages of ED in FedBPG

1 Additions to the description of FedBPG

1.1 Training of the Generator

To generate public data for assisting in ensemble distillation, the server maintains a conditional generator G that produces pseudo data to capture the data distribution of clients, as follows:

$$\widetilde{x} = G(z, y; \theta) \tag{1}$$

here, θ is the parameter of G, $z \sim \mathcal{N}(0, 1)$ is standard Gaussian noise, and y is the class label of $\widetilde{x}$ sampled from a predefined distribution $p(y)$, depending on the proportion of data labeled with y for each client.

To ensure the quality of the public data used for knowledge transfer, three requirements are considered. Firstly, authenticity is crucial, meaning that the public data should be perceived as correct by the shared models of all clients. Since different clients possess varying knowledge about different types of data, each piece of public data is weighted to cater more to the shared models that excel in recognizing the corresponding data type. This is modeled by the following function:

$$\mathcal{L}_{aut} = \sum_{k=1}^{N} \frac{|\mathcal{D}_k^y|}{|\mathcal{D}^y|} \ell(\widetilde{x}, y; \omega_k). \tag{2}$$

here, $|\mathcal{D}_k^y|$ represents the amount of data labeled y in client k, and $|\mathcal{D}^y|$ denotes the total amount of data labeled y across all clients. The proportion of data labeled y for each client is used to measure the extent to which a client excels in knowledge about data type y.

Next, diversity in the generated public data is considered. To avoid the generator producing similar data for each label, a diversity regularization term is introduced:

$$\mathcal{L}_{div} = e^{\frac{1}{Q*Q} \sum_{i,j \in \{1,\ldots,Q\}} (-||\widetilde{x}_i - \widetilde{x}_j||_2 * ||z_i - z_j||_2)} \tag{3}$$

where $\widetilde{x}_i$ is generated using z_i.By considering diversity loss, we obtain diverse public data that is distributed across the data space.

Finally, effectiveness is addressed. Even with authenticity and diversity, if the shared models of clients and the global model produce identical outputs for public data, the effectiveness of knowledge transfer is compromised. To enhance knowledge transfer effectiveness, the difference in outputs between the global model and each client's shared model is emphasized:

$$\mathcal{L}_{eff} = \sum_{k=1}^{N} \frac{|\mathcal{D}_k^y|}{|\mathcal{D}^y|} |\omega_k(\widetilde{x}) - \overline{\omega}(\widetilde{x})| \tag{4}$$

similar to the pursuit of authenticity, weights are assigned to each shared model according to their knowledge level for different data types. This pursuit of effectiveness enhances the subsequent adjustment of the global model.

In summary, the training of the generator involves a comprehensive approach considering three crucial factors. The overall loss function, denoted as $\mathcal{L}_{gen}$, encompasses three components, as $\mathcal{L}_{gen} = \mathcal{L}_{aut} + \lambda_{div}\mathcal{L}_{div} + \lambda_{eff}\mathcal{L}_{eff}$, λ_{div} and λ_{eff} control the weight of diversity and validity of the generated public data. So, the generator can be updated in a way that can be expressed as:

$$\theta = \theta - \eta \nabla_\theta \mathcal{L}_{gen} \tag{5}$$

By setting out the overall loss function through the requirements of authenticity, diversity and effectiveness of the public data, and then using it to update the parameters of the generator, we can get a generator that produces data that meets our requirements.

1.2 Fine-tuning of the Global Model

For the fine-tuning of the global model, we provide an additional pseudo-code (Algorithm 2) to help with the generator process. We have nearly given detailed training of the generator before this, so when fine-tuning the global model, we first use the generator to generate pseudo-data. Then, if there is any pseudo-data retained from the previous round then we first use these for training,if not then we directly use the generated pseudo-data. We first use these data to feed the shared model of each client, we can get an output, and use this output to calculate the empirical loss, this loss is used to measure

Algorithm 1 Detailed expression of Tuning the Global Model

Input: client shared model, public data generated by the generator and retained from the previous round, feasibility threshold ζ.
Parameter: shared model ω, generator θ, global model $\bar{\omega}$,
Output: global model $\bar{\omega}$

```
initialize aggregated model $\bar{\omega}$ and each element $\alpha_k$=1.
for iteration $t$ = 1, ..., $E$ do
    calculate the loss of distillation $\mathcal{L}_{glo}$ by Eq.11.
    calculate $\delta_i$ for the public data generated and retained
    from the previous round by Eq. 12.
    if $\delta_i > 0$ then
        Select data that meets the conditions to be retained
        in the next round of training.
    else if $\xi < \zeta$ then
        update the loss function $\mathcal{L}_{glo}$ to Eq.13.
    end if
    fine_tuning global model $\bar{\omega}$ by Eq.14.
end for
return $\bar{\omega}$.
```

the mastery of the current data of each client's model, if the model's loss is smaller, this represents the model's mastery of the current data is just right. We reflect this relationship by transforming this loss with a negative exponential function. Finally, we normalize the transformed loss, which gives us the weights. These weights are then multiplied by the output of the corresponding client's shared model pair to get a consensus. And we then use these consensus to refer to the global model. If the loss obtained by these consensus will be smaller than that of the global model, it is considered that at this point our global model has more knowledge about the current pseudo-data, and then we directly use the KL scatter to measure the model's results. If not, the empirical loss is added to guide the global model update in cases where we can ensure the quality of the pseudo-data.

2 Detailed Experiment Setup

Here, we present detailed data distributions for two heterogeneous setups, as depicted in Figure 1 and Figure 2. In these figures, the horizontal coordinate represents the client's ID, the vertical coordinate represents the data class, and the color in each image reflects the number of data instances corresponding to the client in that class. We have a total of twenty clients, with 10 classes each for FMNIST and CIFAR 10, 100 classes for CIFAR 100, and 200 for Tiny-ImageNet.

Upon observation, in the practical heterogeneous setting, the data distribution appears more random without a discernible trend. Some clients possess fewer types of data simultaneously. For example, in the CIFAR 10 dataset, Client ID 17 has only 1 type of data, while others, like Client ID 1, have up to 7 types of data.

Contrastingly, in the Pathological heterogeneous setting, the data distribution for each client follows a more regular pattern. Firstly, each client has an equal number of useful data types. Secondly, there are several clients with similar data distributions, providing utility for methods that consider client similarity, such as FedFomo. This regularity can be advantageous for certain approaches.

Settings	CIFAR 10		CIFAR 100	
Method	acc =80%	acc =85%	acc =40%	acc =45%
Per-FedAvg	19.4±1.85	98.0±8.67	185.6±64.57	N/A
APFL	1.8±0.75	7.6±0.80	13.2±0.40	24.0±1.10
FedFomo	2.0±0.0	9.8±0.40	20.6±0.80	N/A
FedRep	3.0±0.63	12.4±0.80	19.0±0.63	30.8±1.17
FedRoD	4.2±0.40	15.6±1.02	24.8±1.72	39.0±3.22
Ditto	5.0±0.00	16.6±0.80	28.2±0.98	49.4±1.50
FedAMP	1.8±0.40	7.4±0.49	13.8±0.40	24.6±1.36
FedPHP	329.8±3.25	494.4±11.77	329.8±3.25	494.4±11.78
APPLE	7.0±0.00	30.2±1.47	7.0±0.00	30.2±1.47
FedProto	8.2±1.60	41.6±3.32	29.2±0.75	45.6±2.58
FedALA	3.0±0.00	14.2±1.17	21.6±1.36	34.6±0.80
FedBPG	6.2±0.40	11.4±0.49	11.2±0.40	15.2±0.40

Table 1: Evaluation of different FL methods on CIFAR10 and CIFAR100 (in the practical heterogeneous setting), in terms of the number of communication rounds to reach target test accuracy (acc).

Furthermore, in both settings, the amount of data owned by each client significantly decreases compared to FMNIST and CIFAR 10, particularly with CIFAR 100 and Tiny-ImageNet. These datasets, being more challenging to categorize, result in a rapid drop in classification accuracy for these two data sets.

3 Additional Experimental Setup and Evaluations

3.1 Communication Effciency

In the main text, we presented a table outlining communication efficiency in the Pathological heterogeneous setting, while Table 1 provides the corresponding information for the Practical heterogeneous setting. It is important to note that our analysis excludes consideration of the generalization method and FedBABU.

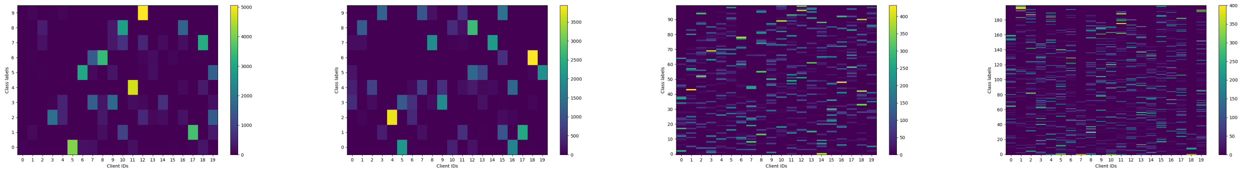
Notably, in the CIFAR 100 setting, we observed a downgrade in accuracy targets from 50% and 55% to 40% and 45%, respectively. This adjustment was necessary due to differences in the characteristics of the client's data distribution between the two settings. In the Practical heterogeneous setting, achieving the same level of accuracy proved to be more challenging.

Furthermore, for CIFAR 10, we adopted a unity criterion, revealing that results manifest more rapidly in the Pathological heterogeneous setting. This reaffirms the notion that the Pathological heterogeneous setting is more amenable to personalized approaches and reinforces the simplicity of this setting for such methodologies.

In our approach, we not only secured a dominant position in the CIFAR 100 dataset but also demonstrated rapid convergence in the CIFAR 10 dataset. This further attests to the efficiency of our method, showcasing accelerated convergence while maintaining consistent uploading overhead per round akin to FedAvg. This reduction in communication overhead stands as a testament to the effectiveness of our approach.

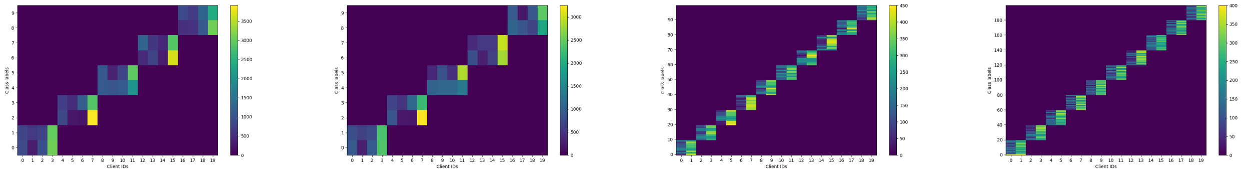
3.2 Ablation Studies

Table 2 presents the outcomes of the ablation experiment conducted in the pathological heterogeneous setting, where ED



(a) FMNIST (b) CIFAR 10 (c) CIFAR 100 (d) Tiny-IamgeNet

Figure 1: Classes allocated to each client n the practical heterogeneous setting



(a) FMNIST (b) CIFAR 10 (c) CIFAR 100 (d) Tiny-IamgeNet

Figure 2: Classes allocated to each client n the pathological heterogeneous setting

Methods	FMNIST	CIFAR 10	CIFAR 100	Tiny-ImageNet
w/o ED	99.58±0.01	91.56±0.26	67.45±0.44	43.24±0.13
w/o AF	99.44±0.01	91.11±0.15	65.93±0.38	39.12±0.63
w/o DP	99.58±0.01	91.76±0.22	67.28±0.22	43.06±0.01
FedBPG	99.58±0.02	91.89±0.20	67.45±0.13	43.09±0.25

Table 2: Results of ablation studies in the pathological heterogeneous setting , where ED is ensemble distillation, AF is adaptive fusion and DP denotes deep personalization.

denotes ensemble distillation, AF represents adaptive fusion, and DP stands for deep personalization. Notably, our approach demonstrated superiority in both CIFAR 10 and CIFAR 100 datasets.

As discussed in the main text, our methodology involves leveraging generator-generated pseudo-data for fine-tuning the global model. However, when applied to the Tiny-ImageNet dataset, certain challenges arise. Primarily, the original Tiny-ImageNet images contain a substantial amount of information, intensifying the difficulty for the generator during the simulation process. Furthermore, relative to other datasets, each client possesses a reduced quantity of data for each specific category, diminishing the modeling capacity of individual clients. This decrease in client modeling ability subsequently impacts the training of the generator. Consequently, our method does not exhibit a significant advantage on Tiny-ImageNet, especially when compared to scenarios where ensemble distillation is not employed. Nevertheless, improved results can be attained by enhancing the model complexity of the generator. It's important to note that, due to equipment limitations, this paper does not delve into effectively modeling more intricate data.

Method	CIFAR 10	CIFAR 100
APFL	+0.07	+0.71
FedFomo	-0.14	-1.11
FedRep	-0.08	-0.48
FedRoD	-0.16	-0.26
FedBABU	-0.09	-0.36
Ditto	-0.42	+4.25
FedAMP	+0.03	-0.61
FedPHP	-0.79	+1.85
APPLE	-0.12	-2.02
FedProto	-0.31	-0.13
FedALA	-0.13	-2.05

Table 3: Deep personalization in combination with other method

3.3 Integration of Deep Personalization

Table 3 provides the outcomes of integrating the Deep Personalization (DP) component of our method with other personalized methodologies. Interestingly, these results highlight that DP does not consistently improve the performance of these methods, particularly those tailored for personalization. This phenomenon arises because many of today's personalized techniques often exhibit a tendency to compromise the generalization ability of federated learning, posing a risk of overfitting. Consequently, additional personalization considerations may not always be beneficial for such methods.

Contrastingly, our personalization methods are uniquely designed to account for generalization, making deep personalization effective in our approach. In the context of generalization methods, which typically do not incorporate personalization considerations, the introduction of personalization represents a significant qualitative advancement.

It is noteworthy that within the same federated learning

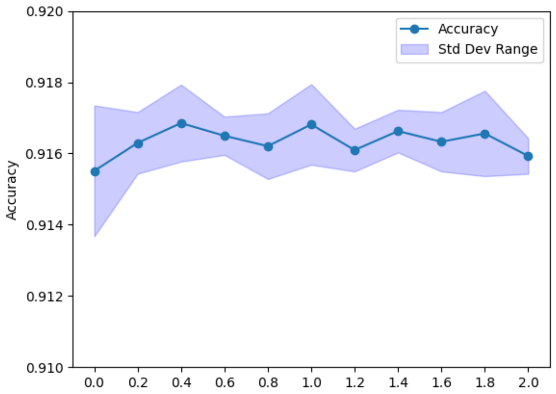


Figure 3: Performance of FedBPG using different initializations of the parameter ζ.

framework, each client has the autonomy to decide whether to opt for deep personalization or not. Importantly, this choice does not impact the client's participation in the federated learning process in any way. This flexibility ensures that clients can decide whether they need a DP approach based on their specific needs and requirements.

3.4 Effect of Hyperparameters

In Figure 3, we showcase the performance of FedBPG with varying initializations of the parameter ζ. When ζ is set to zero, it indicates the exclusion of our pseudo-data retention strategy,it's less effective at this point. As ζ increases, it reflects our confidence in the quality of pseudo-data when utilizing the empirical loss directly. Smaller ζ values imply a higher quality requirement for the data. Notably, a substantial increase in ζ leads to a decline in effectiveness, as using classification loss to update the model on poor-quality data can mislead the overall model.

To assess the impact of hyperparameter selection, we choose values for λ_{div} and λ_{eff} from the range [0.0, 0.5, 1.0, 1.5, 2.0]. Figure 4 presents the test accuracy through a box plot. Notably, results on both extremes are less favorable compared to those in the middle. This pattern emerges because the magnitudes of the λ_{div} and λ_{eff} parameters signify choices that consider the diversity and validity of pseudo-data. If these parameters are set too small, these properties are neglected, leading to a degradation in performance. Conversely, excessively large parameter values tend to bias pseudo-data towards specific features, hindering a balanced consideration of both diversity and validity.

3.5 The process of change in α

Figures 5 and 6 illustrate the evolution of α, with Figure 5 focusing on the client with ID 1 and Figure 6 on the client with ID 3. Each sub-figure demonstrates different initialization scenarios, and within the same figure, various lines portray the evolution of α for different layers.

A noteworthy observation is the distinctive trend in α changes among different clients, highlighting our method's capability to adjust the balance between generalization and personalization based on each client's unique data characteristics. Additionally, the diverse trajectories of α corresponding to different layers for the same client further validate the efficacy of our hierarchical approach.

Commonalities also surface in the α values across different client initializations. Specifically, for α values corresponding to bias, a gradual and slow decline is observed, reaching minimal values around the 600th round. In contrast, α values associated with weights, particularly for the first fully connected layer, exhibit distinct patterns. When initialized to 1, they initially remain constant and may even trend upward at lower initial values, maintaining a higher value. However, after a certain period, they rapidly decline, reaching a minimum value around the 400th round.

The observed trend in the adaptive layer-wise fusion, where α values consistently decrease over training epochs, particularly for biases, signifies a dynamic shift in the model's learning strategy. Small α values indicate an increased reliance on global model information, highlighting a strengthened focus on generalization capabilities. This trend is evident in the gradual decline of α values, reflecting a progressive reduction in dependence on personalized biases and a growing emphasis on capturing broader patterns in the data.

Regarding weights, the initial non-decreasing or even increasing trend in α values suggests an early preference for global model weight information. However, as training progresses, these α values decrease, signaling a diminishing reliance on global weights and a more pronounced focus on learning personalized weights specific to each client. Notably, the rapid decline in α values for the first fully connected layer's weights underscores a swift transition towards personalized weight adaptation, emphasizing the client-specific characteristics captured in this layer.

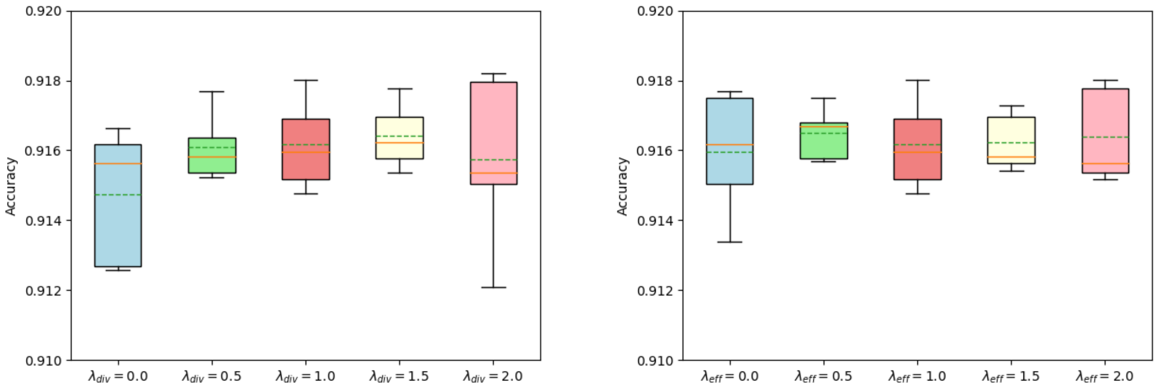
In summary, the observed dynamics in α values reveal an intricate balance between global and personalized learning. The decreasing trend in α values overall indicates a gradual shift towards prioritizing global model information for biases and a more personalized approach in capturing individualized features through weights, aligning with the model's evolving strategy over training epochs.

4 The Advantages of ED in FedBPG

The operations introduced on the server side in our approach can also be considered as an independent generalizability method, which we will analyze separately in the following section, denoted as FedBPG_G.

4.1 Performance in Heterogeneous Scenarios

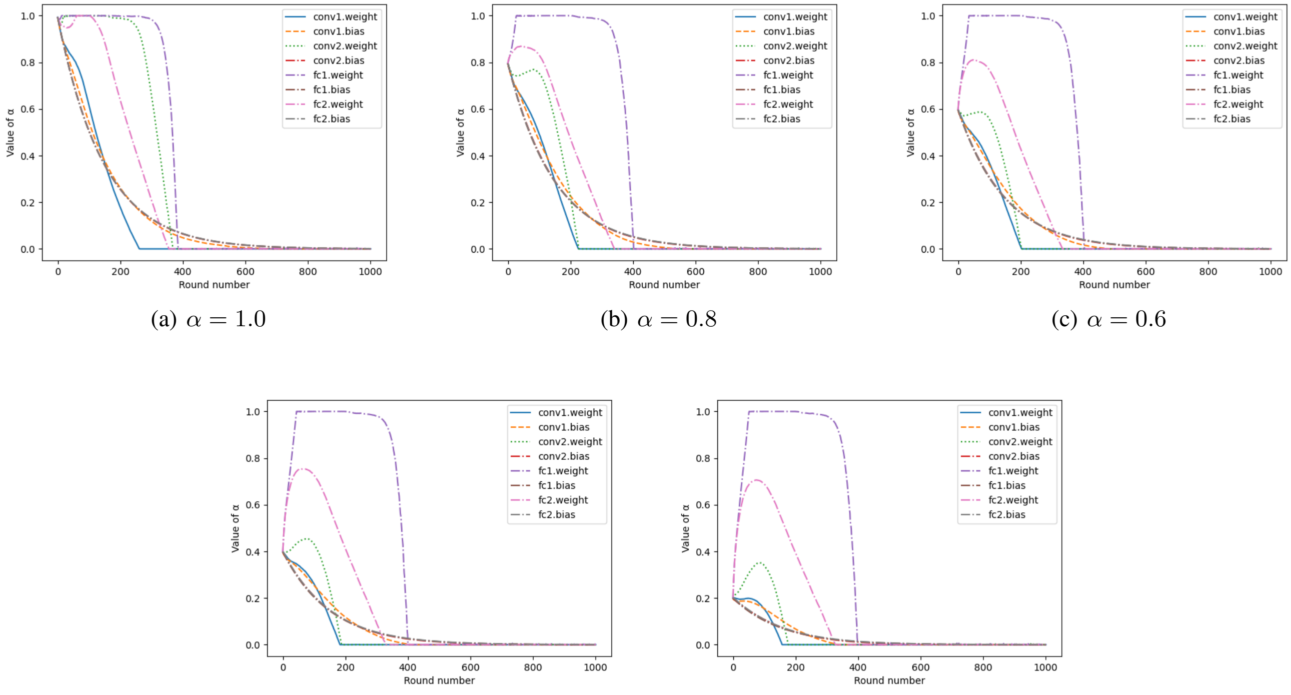
Similar to the main text, we present the results of FedBPG_G in the two heterogeneous cases compared to other generalization methods in Table 4. It is evident that the results in the Pathological heterogeneous setting are inferior to those in the Practical heterogeneous setting, a phenomenon consistent with our observations in the main text. Moreover, our method demonstrates superior results for both CIFAR 10 and CIFAR 100. However, the results for Tiny-ImageNet are less favorable due to the increased complexity of the Tiny-ImageNet



(a) Box plot w.r.t. λ_{div}

(b) Box plot w.r.t. λ_{eff}

Figure 4: Performance of FedBPG using different hyperparameters (a)λ_{div}, (b)λ_{eff} on CIFAR 10.



(a) $\alpha = 1.0$

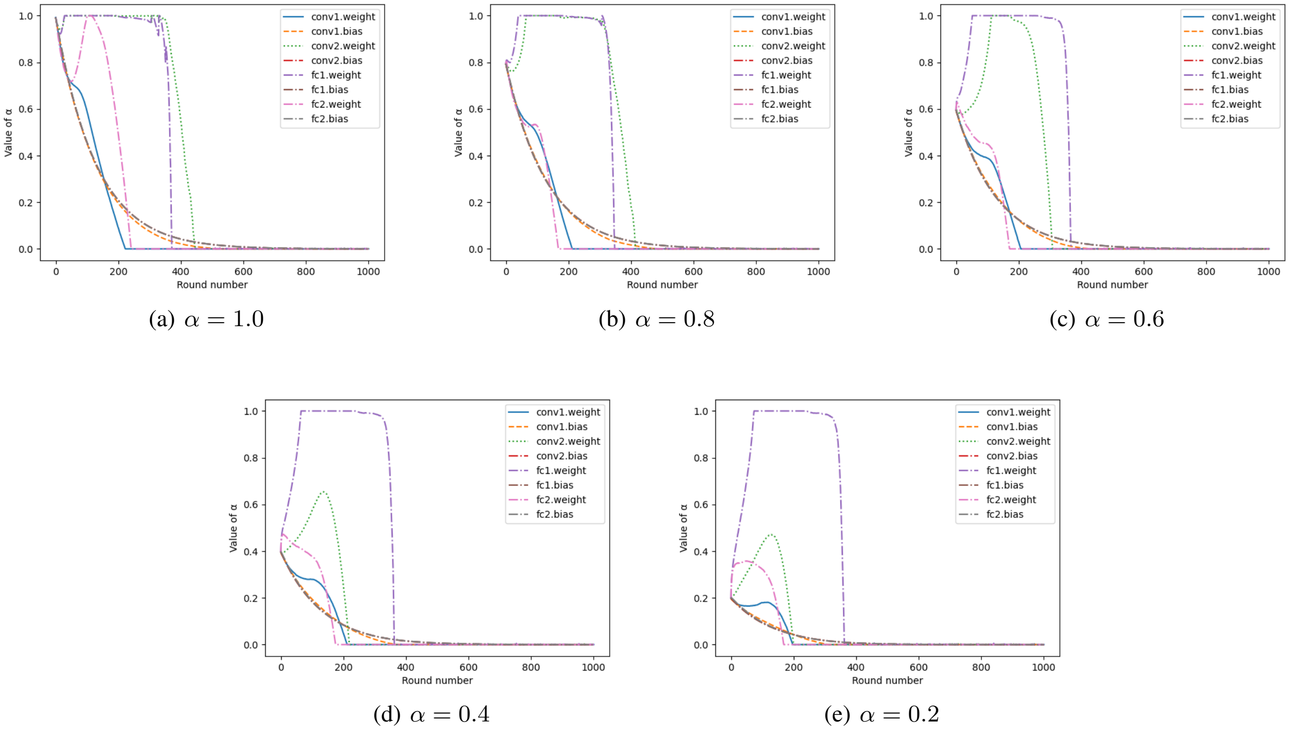
(b) $\alpha = 0.8$

(c) $\alpha = 0.6$

(d) $\alpha = 0.4$

(e) $\alpha = 0.2$

Figure 5: The variation in α for Client ID 1



(a) $\alpha = 1.0$

(b) $\alpha = 0.8$

(c) $\alpha = 0.6$

(d) $\alpha = 0.4$

(e) $\alpha = 0.2$

Figure 6: The variation in α for Client ID 3

Settings	Pathological heterogeneous setting				Practical heterogeneous setting			
Method	FMNIST	CIFAR 10	CIFAR 100	Tiny-ImageNet	FMNIST	CIFAR 10	CIFAR 100	Tiny-ImageNet
FedAvg	84.64±0.08	61.09±0.61	28.49±0.18	13.11±0.26	87.85±0.10	62.50±0.33	32.51±0.33	17.77±0.18
SCAFFOLD	79.29±0.81	57.23±0.63	27.23±0.34	13.15±0.12	87.16±0.28	61.96±0.77	**34.54±0.33**	**19.80±0.21**
FedProx	84.56±0.22	61.11±0.37	28.60±0.30	**13.16±0.14**	87.81±0.12	62.40±0.26	32.45±0.33	17.77±0.11
MOON	84.67±0.19	61.13±0.49	28.55±0.25	13.13±0.31	87.76±0.14	62.21±0.30	32.49±0.23	17.79±0.14
FedGen	**86.16±0.25**	61.58±0.69	27.39±0.38	11.06±0.39	**88.00±0.16**	62.75±0.37	32.15±0.36	15.78±0.08
FedFTG	84.51±0.08	61.20±0.58	28.78±0.28	13.07±0.40	87.60±0.18	62.20±0.52	32.64±0.16	17.62±0.41
FedBPG_G	84.61±0.12	**61.64±0.16**	**28.90±0.07**	13.11±0.26	87.82±0.13	**62.82±0.38**	32.68±0.11	17.62±0.10

Table 4: The test accuracy (%) in the pathological heterogeneous setting and practical heterogeneous setting for our generalisation method. **Bold** /underline fonts highlight the best/second best baseline.

Settings	CIFAR 10				CIFAR 100			
Method	acc=50%		acc=55%		acc=20%		acc=25%	
	DIR	PAT	DIR	PAT	DIR	PAT	DIR	PAT
FedAvg	156.0±4.77	77.0±6.29	237.4±6.02	156.2±15.61	46.0±1.67	49.2±2.32	70.0±3.46	105.8±6.91
SCAFFOLD	217.0±16.84	344.2±39.44	304.4±16.57	549.6±94.09	55.2±2.04	174.8±12.62	90.2±3.71	443.4±84.21
FedProx	153.6±1.96	76.2±6.21	240.2±6.91	153.8±14.34	45.6±1.85	49.2±1.94	70.0±2.97	105.4±5.24
MOON	154.8±3.92	76.6±4.84	238.2±8.57	160.6±16.21	45.8±2.23	49.6±2.06	70.2±3.31	104.2±4.87
FedGen	124.4±5.82	76.2±7.19	183.6±10.38	156.6±25.44	43.0±1.10	50.6±1.85	63.0±1.41	342.2±39.69
FedFTG	154.0±5.10	85.8±4.12	238.0±14.38	165.8±15.69	44.8±1.47	51.4±3.14	69.4±1.85	110.2±11.44
FedBPG_G	150.0±3.63	79.2±3.06	229.8±7.10	146.4±6.97	45.8±1.83	51.8±2.86	69.6±2.24	114.6±14.71

Table 5: Evaluation of different FL methods on CIFAR10 and CIFAR100, in terms of the number of communication rounds to reach target test accuracy (acc)

Settings	Pathological heterogeneous setting				Practical heterogeneous setting			
Method	FMNIST	CIFAR 10	CIFAR 100	Tiny-ImageNet	FMNIST	CIFAR 10	CIFAR 100	Tiny-ImageNet
FedAvg+	99.58±0.01	91.56±0.26	67.45±0.44	43.24±0.13	97.96±0.04	91.56±0.11	56.71±0.33	41.43±0.46
FedProx+	99.58±0.01	91.59±0.26	67.40±0.41	**43.39±0.09**	97.95±0.04	91.61±0.09	56.75±0.29	41.42±0.32
FedDyn+	**99.63±0.02**	91.35±0.08	67.36±0.33	40.39±0.08	97.74±0.04	91.33±0.09	54.19±0.35	40.21±0.56
FedBPG	99.58±0.02	**91.89±0.20**	**67.45±0.13**	43.09±0.25	**97.96±0.02**	**91.80±0.12**	**56.93±0.10**	**41.68±0.08**

Table 6: The test accuracy (%) in the pathological heterogeneous setting and practical heterogeneous setting for applying different generalisations. **Bold** /underline fonts highlight the best/second best baseline.

data. The quality of pseudo-data generated by our method using the generator is compromised in this scenario, a challenge similarly faced by FedFTG and FedGen.

It is noteworthy that FedGen outperforms other methods, although it incurs higher computational costs by dispatching generators to each client for on-site pseudo-data generation. In contrast, our approach utilizes the generator directly on the server side for pseudo-data generation. Relatively speaking, the computational resources on the server side are more abundant than those on the client side, so our approach is more acceptable.

4.2 Communication Effciency

Table 5 evaluates various FL methods based on the number of communication rounds required to attain the target test accuracy (50% and 55% for CIFAR 10, 20% and 25% for CIFAR 100, respectively) in both the pathological (PAT) and practical (DIR) heterogeneous settings.

While we mentioned earlier that the ultimate results in the practical setting would be better than those in the pathological setting, it is noteworthy that the convergence in the pathological setting is faster. This accelerated convergence is attributed to the relatively simpler data distribution in the pathological setting, despite it being less suitable for generalization methods. In contrast, our approach demonstrates a quicker attainment of the target accuracy. Specifically, FedGen achieves a faster convergence speed than ours, but it incurs higher communication costs per round compared to our method. As indicated in Table 4, SCAFFOLD achieves commendable final results; however, Table 5 reveals that its convergence is currently the slowest. Consequently, our method exhibits favorable outcomes, as evidenced by both Table 4 and Table 5.

4.3 Other alternatives to generalization

In our pursuit of enhancing the generalization ability of the global model, we explore alternative methods such as FedAvg, FedProx, and FedDyn. The outcomes, presented in Table 6, underscore the superiority of our approach across various scenarios. In the majority of cases, our method demonstrates outstanding performance, reaffirming the efficacy of our proposed generalization approach.