FedARA: Resource-adaptive Low-rank Personalized Federated Learning via Anchor-driven Representation Alignment on Heterogeneous Edge Devices

Supplementary Material

7. Experimental Setup

7.1. Dataset and Non-IID Settings

7.1.1. Dataset

CIFAR10 [11] contains $60,000$ 32×32 color images across 10 distinct classes, with each class comprising $6,000$ images.
CIFAR100 [11] is a color image dataset with the size 32×32 for 100 classes, where each class contains 600 images.
Tiny-ImageNet [12] includes 200 classes, where each class contains 600 images and each image is of size $64 \times 64 \times 3$.

7.1.2. Non-IID Settings

Dirichlet Distribution ($Practical$). The heterogeneity of data distribution across clients is controlled by α, where a smaller α indicates a higher degree of non-IID.
Pathological Distribution ($Pathological$). For CIFAR10, each client is assigned samples from 2, 4 or 6 out of the total 10 classes, denoted as $Pathological$ (2/10), (4/10), and (6/10), respectively. For CIFAR100, each client is assigned 10, 30 or 50 out of the 100 classes, denoted as $Pathological$ (10/100), (30/100), and (50/100). For Tiny-ImageNet, each client is assigned 20 out of the 200 classes, denoted as $Pathological$ (20/200).

Regardless of the data partitioning strategy employed, the local dataset on each client is split into training and test sets with a 3:1 ratio. Visualization of the resulting data distributions for the CIFAR10 dataset under different heterogeneous configurations is provided in Fig. 7.

7.2. Model Architectures

This section details the model architectures used in our experiments.
Model-Homogeneous Setting. As described in Sec. 5.1, the specific architecture of the unified 4-layer CNN model adopted for all clients in model-homogeneous scenario together with its low-rank variant (FedARA*, $r_k = 0.5$) is presented in Tab. 5.
Model-Heterogeneous Setting. Following [36], baseline methods in the model-heterogeneous scenario employ five distinct CNNs (CNN1-CNN5) with decreasing complexity. More details about model architectures are shown in Tab. 6 (Part A). In FedARA, we construct a set of models with five distinct capacities from the base model CNN1 through low-rank decomposition by setting different rank values, which are detailed in Tab. 6 (Part B). Concretely, for CIFAR10/100, the corresponding rank ratios are $\{1.0, 0.5, 0.35, 0.25, 0.15\}$. For Tiny-ImageNet, the rank ratios are $\{1.0, 0.4, 0.35, 0.29, 0.18\}$. The resulting low-rank models correspond to baseline models from CNN1 to CNN5 in terms of parameter complexity and are randomly assigned to clients.

Table 5. Model architectures for model homogeneous setting. The decomposed CNN is a low-rank variant of the original CNN with the rank ratio $r_k = 0.5$.

Layer	CIFAR10/100		Tiny-ImageNet	
	Original	**Decomposed**	**Original**	**Decomposed**
Conv1	5×5, 32×3	5×5, 32×3	5×5, 32×3	5×5, 32×3
Maxpool1	2×2	2×2	2×2	2×2
Conv2	5×5, 64×32	5×5, 64×32	5×5, 64×32	5×5, 64×32
Maxpool2	2×2	2×2	2×2	2×2
FC1_V^T FC1_U	512×1600	256×1600 512×256	512×10816	256×10816 512×256
FC2	10/100×512	10/100×512	200×512	200×512
Size (MB)	**3.52**	**2.46**	**21.70**	**11.67**

All models adopt the final FC layer as the classifier, and the remainder of the network serves as the FE.

Settings for Other Baselines. For a fair comparison, we adapt other baseline methods accordingly:

- For **FedSPU** [24], we similarly adopt CNN1 as the base model and configure it with varying ratios of the random update parameter to simulate a model-heterogeneous environment. Specifically, the ratios on CIFAR10/100 are $\{1.0, 0.85, 0.7, 0.6, 0.45\}$, while those on Tiny-ImageNet are $\{1.0, 0.75, 0.7, 0.65, 0.45\}$.
- The mutual learning methods, including **FML** [27], **FedKD** [33], **FedMRL** [37], and **pFedAFM** [39], employ the standard 4-layer CNN in model-homogeneous scenarios. In model-heterogeneous scenarios, they utilize CNN5 to facilitate cross-client knowledge interaction.

7.3. Implementation Details

Computing Environment. All experiments are conducted on a server equipped with eight NVIDIA GeForce RTX 4090 (24GB) GPUs, dual Intel Xeon Gold 5318Y CPUs, and 251GB of DDR4 ECC RAM. Our implementation is based on PyTorch 2.0.1 with CUDA 11.7 and cuDNN 8.5.0, using Python 3.11.13.

Table 6. Details of model architectures for baselines and low-rank model families

Part A: The heterogeneous model family for baselines
Architectures of the five CNNs (CNN1 to CNN5) with decreasing complexity

Layer	CIFAR10/100					Tiny-ImageNet				
	CNN1	**CNN2**	**CNN3**	**CNN4**	**CNN5**	**CNN1**	**CNN2**	**CNN3**	**CNN4**	**CNN5**
Conv1	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3
Maxpool1	2×2	2×2	2×2	2×2	2×2	2×2	2×2	2×2	2×2	2×2
Conv2	5×5,32×16	5×5,16×16	5×5,32×16	5×5,32×16	5×5,32×16	5×5,32×16	5×5,16×16	5×5,32×16	5×5,32×16	5×5,32×16
Maxpool2	2×2	2×2	2×2	2×2	2×2	2×2	2×2	2×2	2×2	2×2
FC1	2000×800	2000×400	1000×800	800×800	500×800	2000×5408	2000×2704	1000×5408	800×5408	500×5408
FC2	500×2000	500×2000	500×1000	500×800	500×500	500×2000	500×2000	500×1000	500×800	500×500
FC3	10/100×500	10/100×500	10/100×500	10/100×500	10/100×500	200×500	200×500	200×500	200×500	200×500
Size (MB)	**10.19**	**7.09**	**5.22**	**4.23**	**2.72**	**45.50**	**24.49**	**22.58**	**18.46**	**11.71**

Part B: The low-rank model family for FedARA
Architectures with varying rank ratios (r_k), where $r_k = 1.0$ denotes the original CNN1 model without low-rank decomposition

Layer	CIFAR10/100 (Low-rank ratio r_k)					Tiny-ImageNet (Low-rank ratio r_k)				
	r_k**=1.0**	r_k**=0.5**	r_k**=0.35**	r_k**=0.25**	r_k**=0.15**	r_k**=1.0**	r_k**=0.4**	r_k**=0.35**	r_k**=0.29**	r_k**=0.18**
Conv1	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3	5×5, 16×3
Maxpool1	2×2	2×2	2×2	2×2	2×2	2×2	2×2	2×2	2×2	2×2
Conv2_V Conv2_U	5×5, 32×16	5×1, 40×16 1×5, 32×40	5×1, 28×16 1×5, 32×28	5×1, 20×16 1×5, 32×20	5×1, 12×16 1×5, 32×12	5×5, 32×16	5×5, 32×16	5×5, 32×16	5×5, 32×16	5×5, 32×16
Maxpool2	2×2	2×2	2×2	2×2	2×2	2×2	2×2	2×2	2×2	2×2
FC1_V^T FC1_U	2000×800	400×800 2000×400	280×800 2000×280	200×800 2000×200	120×800 2000×120	2000×5408	800×5408 2000×800	700×5408 2000×700	580×5408 2000×580	360×5408 2000×360
FC2_V^T FC2_U	500×2000	250×2000 500×250	175×2000 500×175	125×2000 500×125	75×2000 500×75	500×2000	200×2000 500×200	175×2000 500×175	145×2000 500×145	90×2000 500×90
FC3	10/100×500	10/100×500	10/100×500	10/100×500	10/100×500	200×500	200×500	200×500	200×500	200×500
Size (MB)	**10.19**	**6.90**	**4.92**	**3.58**	**2.25**	**45.50**	**24.94**	**21.85**	**18.19**	**11.44**

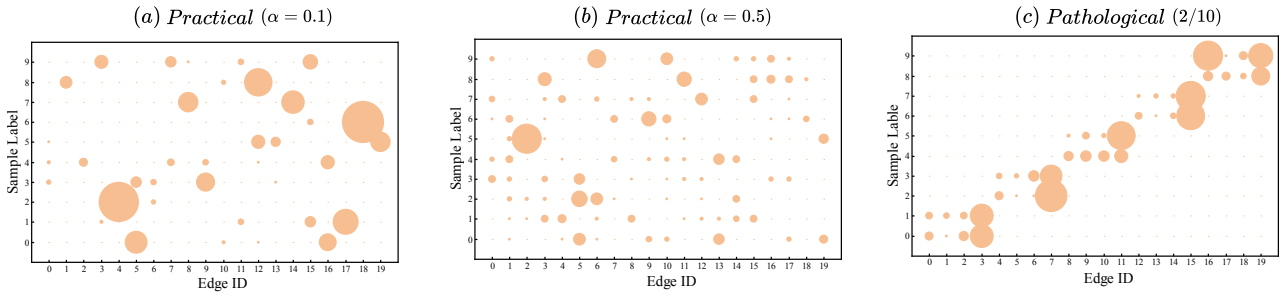

Figure 7. Visualization of CIFAR10 under different non-IID settings.

8. More Experiment

8.1. Additional Model-Homogeneous Results

Under model-homogeneous setting on the Tiny-ImageNet, we follow the same hyperparameters as in Sec. 5.1, except for anchor-constraint consistency strength λ, which is set to 20 for FedARA and 10 for FedARA*. As shown in Tab. 7, FedARA achieves the highest accuracy across all scenarios, outperforming the best baseline FedAS by 5.70% on average. Even with the lower-complexity low-rank model

Table 7. Test accuracy on Tiny-ImageNet under heterogeneous data with homogeneous models. "-" indicates method non-convergence. **Bold** indicates the best result. Underline is the second best.

Methods	*Practical* ($\alpha = 0.1$)			*Practical* ($\alpha = 0.5$)			*Pathological* (20/200)			Avg
	$P = 0.2$	$P = 0.4$	$P = 1.0$	$P = 0.2$	$P = 0.4$	$P = 1.0$	$P = 0.2$	$P = 0.4$	$P = 1.0$	
FedPer [1]	30.66	30.61	30.37	14.95	14.54	14.18	36.01	36.24	36.55	27.12
FedRep [3]	29.28	29.66	29.38	13.80	13.99	13.96	36.95	37.51	37.62	26.90
FedGen [45]	27.70	27.22	27.03	11.50	11.52	11.40	33.77	33.63	33.29	24.11
Ditto [16]	26.98	26.92	26.88	11.62	11.60	11.64	33.33	32.90	32.94	23.86
LG-FedAvg [18]	27.87	27.89	27.54	12.04	11.96	12.07	33.75	33.88	33.72	24.52
FedGH [36]	27.91	27.86	27.90	12.03	11.96	12.01	33.56	33.60	33.78	24.51
FedBABU [25]	30.11	30.53	30.71	17.12	17.05	16.97	36.19	35.84	36.34	27.87
FedProto [29]	24.79	28.08	31.11	13.54	14.43	15.87	35.84	35.86	36.48	26.22
FD [9]	28.54	28.61	28.75	12.50	12.56	12.59	34.46	34.78	35.01	25.31
FedKD [33]	33.27	33.93	–	17.19	18.06	17.61	38.12	38.44	38.70	29.41
FML [27]	–	–	–	11.19	–	–	31.47	–	–	21.33
FedALA [40]	31.59	32.04	31.71	16.95	17.19	17.13	36.35	36.63	36.99	28.50
FedMRL [37]	27.71	27.79	27.85	12.57	11.50	11.57	33.97	34.25	34.01	24.58
FedTGP [42]	30.88	31.32	31.63	15.36	15.48	15.68	36.78	37.12	37.62	27.98
FedAS [34]	34.76	34.54	34.82	18.36	18.00	17.16	39.27	39.31	40.74	30.77
pFedAFM [39]	27.46	27.68	27.73	12.08	12.10	12.08	33.99	33.97	34.13	24.58
FedARA (Ours)	**41.06** ↑6.30	**40.16** ↑5.62	**40.23** ↑5.41	**25.34** ↑6.98	**25.37** ↑7.31	**24.66** ↑7.05	**44.54** ↑5.27	**43.89** ↑4.58	**43.06** ↑2.32	**36.47** ↑5.70
FedARA* (Ours)	**39.65** ↑4.89	**39.82** ↑5.28	**39.67** ↑4.85	**24.50** ↑6.14	**24.74** ↑6.68	**24.01** ↑6.40	**42.97** ↑3.70	**43.11** ↑3.80	**42.94** ↑2.20	**35.71** ↑4.94

(FedARA*), our method still surpasses FedAS by 4.94% on average.

8.2. Comprehensive Ablation Study on the Constraint Strength

To further validate the robustness of the anchor-driven representation consistency learning mechanism beyond the results in Sec. 5.4, we extend the ablation study on the constraint strength λ to additional scenarios. This supplementary analysis covers four configurations: model-heterogeneous settings on CIFAR10 and Tiny-ImageNet, and model-homogeneous settings on CIFAR10 and CIFAR100. These experiments evaluate whether the anchor constraint λ remains consistently effective across different datasets and model architectures.

As shown in Fig. 8, under model-heterogeneous settings, the influence of λ on CIFAR10 and Tiny-ImageNet aligns with the trends observed in CIFAR100 in the main paper (see Fig. 5). Removing the anchor constraint ($\lambda = 0$) leads to significant performance degradation under both *Practical* and *Pathological* settings, further confirming that without feature alignment, local models on different clients diverge in the representation space, thereby hindering the effectiveness of knowledge aggregation. Results in Fig. 9 demonstrate that even when client models share identical architectures, the inherent statistical heterogeneity of non-IID data still causes notable representation drift, while this issue can be effectively mitigated by our proposed anchor constraint.

In summary, the anchor-driven representation consistency learning mechanism consistently alleviates feature drift caused by model-level and data-level heterogeneity, demonstrating robustness across diverse model architectures and datasets.

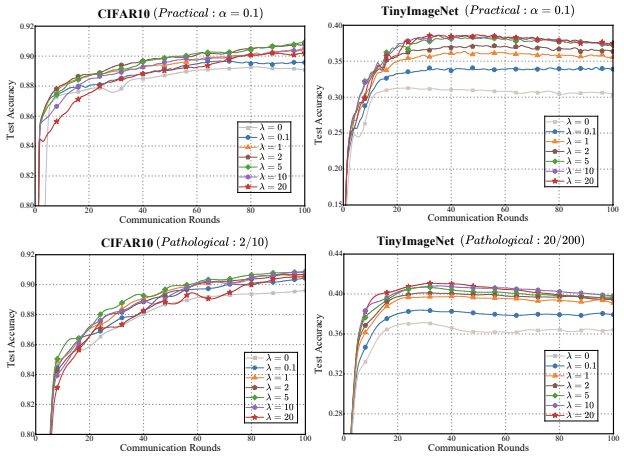

Figure 8. Ablation study on anchor constraint strength λ under model-heterogeneous settings on CIFAR10 and Tiny-ImageNet.

8.3. Additional Visualizations of Feature Representations

Feature Visualization on CIFAR10. Consistent with the findings on CIFAR100 (Fig. 6), the t-SNE visualization on CIFAR10 further validates the effect of our anchor constraint. Without the constraint, feature representations exhibit significant entanglement across classes (Fig. 10 (a)). In contrast, applying the anchor constraint organizes the feature space into well-separated and compact clusters (Fig. 10 (b)), demonstrating the scalability of our approach across different datasets.

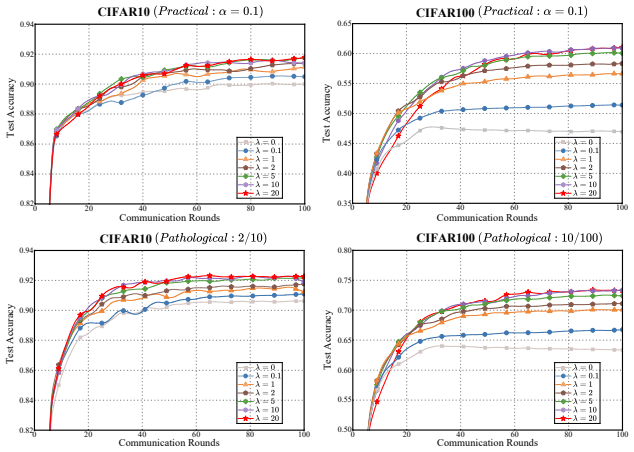

Figure 9. Ablation study on anchor constraint strength λ under model-homogeneous settings on CIFAR10 and CIFAR100.

Fine-grained Analysis of Cross-Client Feature Alignment. Fig. 11 provides a fine-grained view of how our anchor constraint achieves feature alignment at the individual class level, taking the 'deer' class from CIFAR10 as an example. Without the constraint (Fig. 11 (a)), features from the same class but different clients remain dispersed in the latent space. With the anchor constraint (Fig. 11 (b)), the features converge towards a unified representation, demonstrating effective semantic alignment across clients.

Comparative Visualization with Baseline Methods. As shown in Figs. 12 and 13, compared to FedBABU, Fedper, FedRep and FedAS, the feature distributions produced by FedARA exhibit more significant structural characteristics and stronger semantic correlations. The feature clusters under our method exhibit three key advantages: 1) tighter intra-class compactness, with samples from the same class concentrating on more focused regions, 2) clearer inter-class separation, maintaining distinct boundaries between different classes, and 3) enhanced semantic coherence, where visually similar classes remain closer in the embedding space. These observations directly validate that our anchor-driven representation consistency learning mechanism successfully mitigates feature space inconsistency across clients, leading to more discriminative and robust feature representations that explain the superior quantitative performance reported in the main paper.

8.4. Convergence Analysis under Homogeneous and Heterogeneous Model Settings

Fig. 14 illustrates the test accuracy versus communication rounds under the model-heterogeneous setting on CIFAR10. Our proposed FedARA not only achieves the highest final accuracy but also exhibits significantly faster convergence compared to all baseline methods. This accelerated convergence is attributed to the proposed anchor-driven representation consistency learning mechanism, which effectively mitigates feature space drift caused by non-IID data and model heterogeneity, thereby enabling more stable and efficient global knowledge fusion across clients with highly diverse models and data distributions.

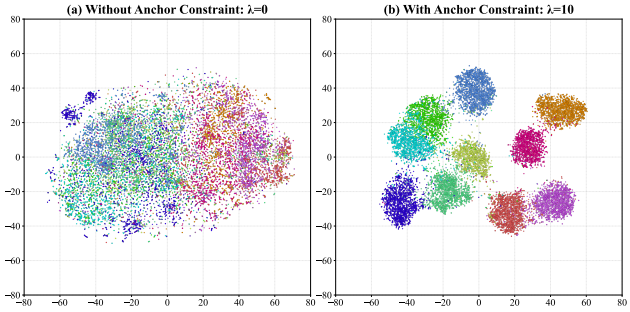

Figure 10. Disentangled clusters emerge from the anchor constraint.

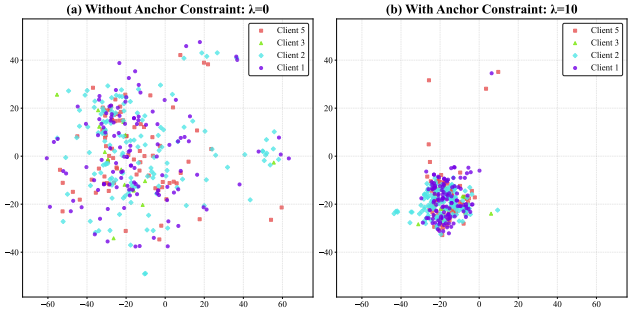

Figure 11. T-SNE visualization of 'deer' feature distributions across different clients.

Under the model-homogeneous setting (Fig. 15), FedARA consistently demonstrates faster convergence on both CIFAR10 and CIFAR100. The results indicate that even when clients share the same model architecture, the anchor alignment constraint effectively suppresses feature inconsistency induced by statistical heterogeneity, leading to more robust and efficient personalized model learning.

In conclusion, FedARA achieves superior final performance and faster convergence speed under both model-heterogeneous and model-homogeneous settings, validating the effectiveness of our framework.

8.5. Additional Results on CIFAR10 for Robustness on Non-IID Data Settings

As shown in Tab. 8, FedARA consistently achieves the highest accuracy under both $Practical$ and $Pathological$ partitioning settings, demonstrating strong robustness against varying degrees of data heterogeneity. In the $Practical$ setting, FedARA surpasses the best baseline by 1.45%, 7.21%, and 7.77% at $\alpha = 0.1, 0.5$ and 1.0, respectively. The accuracy improvement at $\alpha = 0.5$ and 1.0 is particularly significant, showing that the anchor-driven representation con-

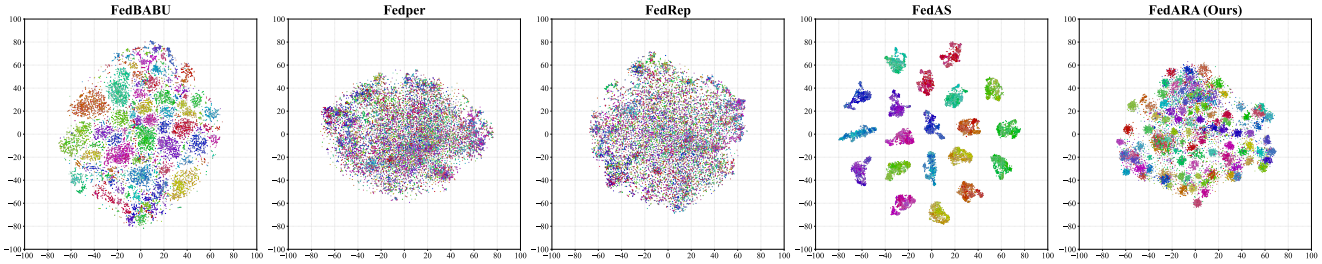

Figure 12. Visualization of feature distribution with different methods on CIFAR100.

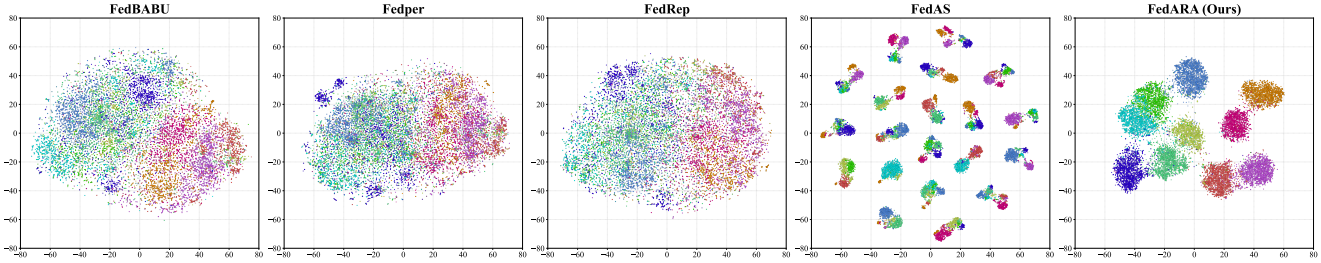

Figure 13. Visualization of feature distribution with different methods on CIFAR10.

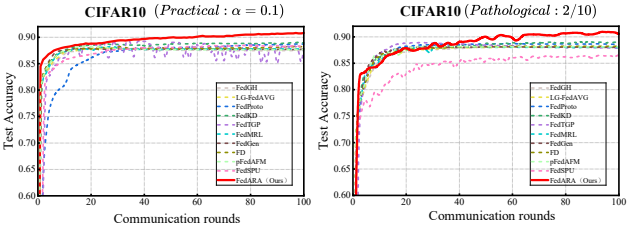

Figure 14. Accuracy vs. communication rounds under heterogeneous model settings.

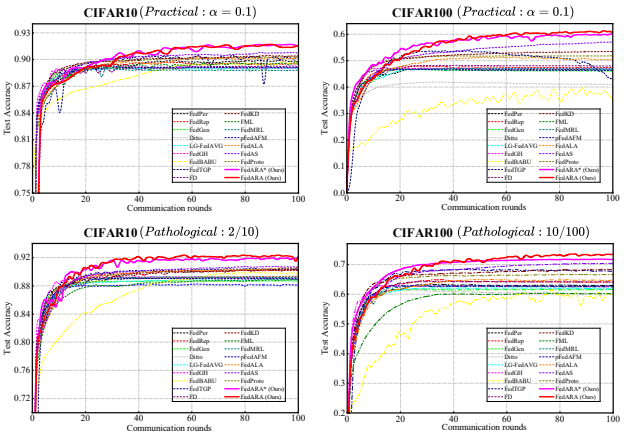

Figure 15. Accuracy vs. communication rounds under homogeneous model settings.

sistency learning mechanism can stabilize feature learning even as data distributions across clients gradually converge toward IID. Under the *Pathological* setting, which enforces disjoint label spaces across clients, FedARA also delivers superior performance, outperforming the strongest competitor by 2.06%, 3.66%, and 5.22% in the 2/10, 4/10, and 6/10 partitions, respectively.

Table 8. Robustness to Non-IID Data on CIFAR10.

Methods	*Practical*			*Pathological*		
	0.1	0.5	1.0	2/10	4/10	6/10
FedGH [36]	87.97	66.45	58.90	88.48	76.44	70.84
LG-FedAVG [18]	88.03	66.39	59.18	88.49	76.27	70.85
FML [27]	88.73	68.07	60.38	88.37	76.01	72.08
FedGen [45]	87.95	66.74	59.06	88.18	78.16	71.78
FD [9]	88.23	67.63	59.45	88.22	76.65	71.47
FedProto [29]	88.70	69.63	62.08	88.82	78.32	73.51
FedKD [33]	89.05	70.01	61.87	89.17	79.18	74.18
FedMRL [37]	88.20	66.78	58.74	88.08	76.66	71.12
FedTGP [42]	89.39	70.16	62.15	88.92	78.16	73.49
FedSPU [24]	88.49	69.24	64.23	86.58	78.35	73.31
pFedAFM [39]	88.35	66.38	59.01	88.27	76.05	70.85
FedARA (Ours)	**90.84**	**77.37**	**72.00**	**91.23**	**82.84**	**79.40**
Δ	↑1.45	↑7.21	↑7.77	↑2.06	↑3.66	↑5.22

8.6. Comprehensive Efficiency Analysis

This section provides a comprehensive analysis from computational and communication costs under both homogeneous and heterogeneous model settings.

Table 9. Computational overhead comparison under model-homogeneous setting. The target accuracy for each setting is shown in parentheses. "-" indicates that the approach fails to achieve the target accuracy.

Methods	Computational Overhead (FLOPs)							
	CIFAR10				CIFAR100			
	$Practical: \alpha = 0.1$ (90%)		$Pathological: 2/10$ (90%)		$Practical: \alpha = 0.1$ (55%)		$Pathological: 10/100$ (70%)	
	rounds	total_comp	rounds	total_comp	rounds	total_comp	rounds	total_comp
FedPer [1]	-	-	42	221.81e12	-	-	-	-
FedRep [3]	-	-	46	323.92e12	-	-	-	-
FedProto [29]	47	264.61e12	-	-	-	-	-	-
FedTGP [42]	25	140.75e12	32	180.16e12	-	-	-	-
FedKD [33]	40	424.00e12	52	551.20e12	-	-	-	-
FedALA [40]	57	348.27e12	45	274.95e12	-	-	-	-
FedAS [34]	30	200.70e12	34	227.46e12	58	389.76e12	76	510.75e12
FedARA (Ours)	27	152.01e12	16	90.08e12	29	164.14e12	33	186.78e12
FedARA*(Ours)	22	119.46e12	23	124.89e12	25	136.50e12	29	158.34e12

Table 10. Communication cost comparison under model-homogeneous setting. The target accuracy for each setting is shown in parentheses. "-" indicates that the approach fails to achieve the target accuracy.

Methods	Communication Cost (GB)							
	CIFAR10				CIFAR100			
	$Practical: \alpha = 0.1$ (90%)		$Pathological: 2/10$ (90%)		$Practical: \alpha = 0.1$ (55%)		$Pathological: 10/100$ (70%)	
	rounds	total_comm	rounds	total_comm	rounds	total_comm	rounds	total_comm
FedPer [1]	-	-	42	5.45	-	-	-	-
FedRep [3]	-	-	46	5.97	-	-	-	-
FedProto [29]	47	0.0358	-	-	-	-	-	-
FedTGP [42]	25	0.019	32	0.0244	-	-	-	-
FedKD [33]	40	1.44	52	1.59	-	-	-	-
FedALA [40]	57	7.45	45	5.89	-	-	-	-
FedAS [34]	30	3.9	34	4.41	58	7.533	76	9.87
FedARA (Ours)	27	3.5	16	2.08	29	3.76	33	4.28
FedARA* (Ours)	22	1.95	23	2.03	25	2.21	29	2.57

8.6.1. Efficiency under Model-Homogeneous Setting

Tab. 9 and Tab. 10 separately present a comparative analysis of the computational overhead and communication cost under the model-homogeneous setting, where FedARA and FedARA* are only compared with the baselines capable of achieving the pre-defined target accuracy. Several other baselines are omitted due to non-convergence for achieving the target accuracy.

Computational Cost. The results in Tab. 9 demonstrate that among the convergent methods, FedARA consistently achieves the target accuracy with the lowest computational overhead. Under the $Practical$ partition ($\alpha = 0.1$) of CIFAR10, FedARA achieves 90% accuracy in 27 rounds, requiring a total cost of 152.01e12 FLOPs. This yields a 24.3% computational saving over the FedAS (30 rounds, 200.70e12 FLOPs). Under the $Pathological$ partition, the advantage is even more pronounced, where FedARA converges in just 16 rounds, reducing the required FLOPs by over 60.4% compared to FedAS. The low-rank variant FedARA* further improves this efficiency. For instance, on CIFAR10 ($\alpha = 0.1$), it reaches the target accuracy in only 22 rounds with 119.46e12 FLOPs. This trend holds on the more complex CIFAR100 dataset, where FedARA* requires substantially fewer rounds and FLOPs than FedAS to reach the target accuracy, which achieves a 69.0% reduction in FLOPs under the $Pathological$ partition.

Communication Cost. As shown in Tab. 10, methods like FedProto and FedTGP naturally achieve minimal per-round communication costs by exchanging class-wise prototypes instead of model parameters. However, this communication efficiency comes at the expense of potential privacy risks from sharing sensitive feature-level information and a performance ceiling, hindering their practical applicability. FedKD achieves lower communication overhead than FedARA by

Table 11. Computational overhead comparison under model-heterogeneous settings. The target accuracy for each setting is shown in parentheses. "-" indicates that the approach fails to achieve the target accuracy.

Methods	Computational Overhead (FLOPs)							
	CIFAR10				CIFAR100			
	$Practical: \alpha = 0.1$ **(88%)**		$Pathological: 2/10$ **(88%)**		$Practical: \alpha = 0.1$ **(45%)**		$Pathological: 10/100$ **(60%)**	
	rounds	**total_comp**	**rounds**	**total_comp**	**rounds**	**total_comp**	**rounds**	**total_comp**
FedGH [36]	-	-	14	36.31e12	-	-	-	-
LG-FedAVG [18]	-	-	21	51.06e12	-	-	-	-
FML [27]	26	112.67e12	-	-	-	-	-	-
FedGen [45]	-	-	14	36.69e12	-	-	-	-
FD [9]	24	57.42e12	24	58.36e12	25	60.27e12	20	48.74e12
FedProto [29]	43	109.73e12	16	41.49e12	22	56.57e12	13	33.79e12
FedKD [33]	13	56.33e12	21	91.82e12	17	73.98e12	12	52.53e12
FedMRL [37]	22	95.33e12	24	104.94e12	-	-	22	96.31e12
FedTGP [42]	13	33.17e12	16	41.49e12	19	48.58e12	13	33.79e12
pFedAFM [39]	14	69.71e12	23	115.42e12	-	-	21	105.50e12
FedSPU [24]	40	89.42e12	-	-	-	-	-	-
FedARA (Ours)	10	21.89e12	15	34.19e12	9	20.68e12	13	30.11e12

transmitting approximate gradients of the model, but it sacrifices significant model accuracy (see Tabs. 1, 2 and 7). In comparison with other baselines for exchanging FEs (e.g., FedPer, FedRep, FedAS) or full model parameters (e.g., FedALA), FedARA yields significant advantages in communication efficiency. For instance, on CIFAR10 ($\alpha = 0.1$), FedARA completes training in 27 rounds with a total cost of 3.5 GB, substantially lower than FedAS (30 rounds, 3.9 GB) and FedALA (57 rounds, 7.45 GB). Additionally, on CIFAR100 ($\alpha = 0.1$), FedARA achieves significantly lower communication overhead (3.76 GB) compared to FedAS (7.533 GB), despite both transmitting FE parameters with identical architecture. This reduction is attributed to the proposed anchor-driven representation consistency learning mechanism, which accelerates convergence, enabling FedARA to reach the target accuracy in only 29 rounds, compared to 58 rounds for FedAS. The low-rank variant FedARA* further enhances communication efficiency. By transmitting a decomposed parameter set, FedARA* reduces the total communication cost by nearly half compared to the full-rank version on CIFAR10 (e.g., 1.95 GB vs. 3.5 GB under $\alpha = 0.1$), while still maintaining strong convergence guarantee that eludes many baselines on CIFAR100.

8.6.2. Efficiency under Model-Heterogeneous Setting

We comprehensively evaluate the efficiency of FedARA against SOTA methods under model-heterogeneous settings in terms of both computational and communication costs.

Computational Cost. As shown in Tab. 11, FedARA consistently achieves the target accuracy with the lowest computational cost across all settings. For instance, on CIFAR10 under the $Practical$ setting ($\alpha = 0.1$), FedARA reaches 88% accuracy in only 10 rounds, with a total computational cost of 21.89e12 FLOPs. This represents a 34.0% reduction compared to the closest competitor FedTGP (13 rounds, 33.17e12 FLOPs). Notably, FedARA also converges faster in other challenging settings. For example, on CIFAR100 ($\alpha = 0.1$), it requires only 9 rounds to reach 45% accuracy, significantly fewer than the 17-25 rounds required by other methods. Similarly, in the $Pathological$ setting, FedARA achieves the target accuracy with the lowest computational cost, fully demonstrating its robust efficiency across diverse heterogeneous scenarios.

Communication Cost. Tab. 12 details the communication overhead required for FedARA and all baselines to achieve target accuracy under model-heterogeneous scenarios. Baseline methods such as FD, FedProto and FedTGP significantly reduce communication overhead by sharing logits or features of locally visible categories, while they may raise privacy concerns. Methods like LG-FedAVG, FedGH, and FedGen achieve high communication efficiency by exchanging only classifier parameters and minimal supplementary information (e.g., generator parameters or class-wise average representation of local samples). Meanwhile, approaches such as FedKD, FedMRL, pFedAFM, and FML enable knowledge transfer by exchanging parameters of homogeneous small models or their approximate gradients. Despite requiring more training epochs to reach target accuracy, they still offer certain advantages in communication efficiency. In contrast, FedARA exhibits higher communication overhead than some baselines but boasts the lowest computational overhead (see Tab. 11) and achieves optimal classification accuracy (see Tabs. 3, 4 and 8), striking a superior balance between overall efficiency and performance.

Table 12. Communication cost comparison under model-heterogeneous settings. The target accuracy for each setting is shown in parentheses. "-" indicates that the approach fails to achieve the target accuracy.

Methods	Communication Cost (GB)							
	CIFAR10				CIFAR100			
	$Practical: \alpha = 0.1$ (88%)		$Pathological: 2/10$ (88%)		$Practical: \alpha = 0.1$ (45%)		$Pathological: 10/100$ (60%)	
	rounds	total_comm	rounds	total_comm	rounds	total_comm	rounds	total_comm
FedGH [36]	-	-	14	0.01564	-	-	-	-
LG-FedAVG [18]	-	-	21	0.0156	-	-	-	-
FML [27]	26	2.77	-	-	-	-	-	-
FedGen [45]	-	-	14	0.62	-	-	-	-
FD [9]	24	0.00035	24	0.00036	25	0.0372	20	0.0298
FedProto [29]	43	0.03203	16	0.01192	22	0.163	13	0.0966
FedKD [33]	13	0.6347	21	0.7617	17	0.7509	12	0.675
FedMRL [37]	22	2.34	24	2.56	-	-	22	2.34
FedTGP [42]	13	0.00968	16	0.01192	19	0.1416	13	0.0968
pFedAFM [39]	14	1.38	23	2.28	-	-	21	2.08
FedSPU [24]	40	8.82	-	-	-	-	-	-
FedARA (Ours)	10	2.09	15	3.13	9	1.88	13	2.71

Fig. 16 further presents the Pareto curves of communication cost, computational cost, and test accuracy for FedARA and all baseline methods under the heterogeneous model setting with $(Practical, \alpha = 0.1)$.

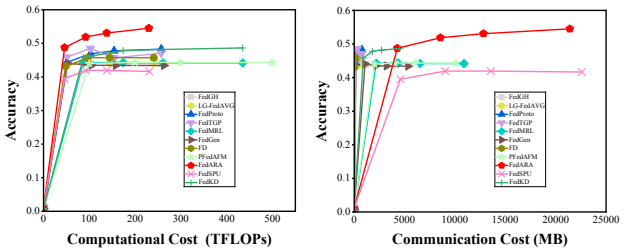

Figure 16. Pareto curves of communication and computation cost vs. test accuracy.

The aforementioned experimental results validate the dual advantages of FedARA. One is its reliable convergence where several other baselines fail, and the other is superior computational and communication efficiency among all convergent methods, stemming from its anchor-driven representation consistency learning mechanism and resource-adaptive low-rank architecture. Moreover, it not introduces additional privacy concerns.

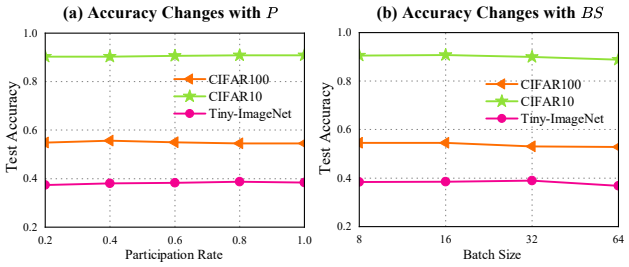

Figure 17. Accuracy changes with hyperparameters P and BS.

8.7. Other Hyperparameter Analysis

This section evaluates the robustness of FedARA to the participation rate P and the batch size BS under the scenario of model-heterogeneous setting with $Practical: \alpha = 0.1$. We set the client participation rate $P \in \{0.2, 0.4, 0.6, 0.8, 1.0\}$ and local batch size $BS \in \{8, 16, 32, 64\}$, then plotted the test accuracy under different parameter values. The corresponding results in Fig. 17 (a) and (b) show that varying hyperparameters P and BS do not induce significant performance fluctuations, confirming the high robustness of FedARA on both hyperparameters.

8.8. Privacy Analysis

For data privacy, in FedARA, each client k only exchanges the FE parameters ω_k or corresponding low-rank version ω_{r_k} with the server, while the personalized classifer θ_k and the dataset $\mathcal{D}_k$ are always stored locally within the client. This mitigates the risk of privacy leakage associated with transmitting data-related sensitive information such as class-wise average representations or logits of local samples in prototype-based or logit-based methods like FedProto, FedTGP, FedSA, FD and so forth. For model privacy, the consistency anchors required in FedARA are locally generated by client k based on the received GFE parameters and its local dataset $\mathcal{D}_k$, which does not introduce any additional privacy risks in comparison with other model-decoupled PFL such as FedAS.

Additionally, other privacy-preserving mechanisms, such as differential privacy and homomorphic encryption, can be directly integrated with FedARA for protecting the privacy of updated FE parameters during the interaction process, further enhancing the privacy of overall FedARA framework.

8.9. Impact of Low-Rank Models

Tab. 13 further evaluates the impact of heterogeneous low-rank models by replacing the CNN1-CNN5 backbones in FedProto and FedTGP with their corresponding low-rank variants (as detailed in Tab. 6). As shown, both FedProto and FedTGP suffer noticeable performance degradation when adopting low-rank heterogeneous models. This observation indicates that the performance improvements achieved by FedARA cannot be attributed to the use of low-rank models alone.

Table 13. Test Accuracy (%) on CIFAR-100 under Heterogeneous Model Settings

Method	Base Acc.	Variant	Final Acc.	Δ Acc.
FedProto	48.32	+ Low-Rank Model	44.68	-3.64
FedTGP	48.70	+ Low-Rank Model	47.64	-1.06
FedARA	**54.54**	-	-	-

8.10. Resource mapping

Tab. 14 presents a concrete mapping from the rank ratio r_k to model parameters, memory footprint, FLOPs, communication cost, and test accuracy. This mapping enables straightforward resource-aware model selection under different system constraints. Moreover, it reveals the degradation trend in performance as r_k decreases, as well as the critical collapse point under aggressive low-rank compression.

Table 14. A trade-off between resource consumption and accuracy on CIFAR-10 dataset with a homogeneous base model (CNN1).

Rank Ratio (r_k)	Param.	Memory	FLOPs	Comm.	Accuracy
$r_1 = 1.0$	2.67M	10.19MB	4.87M	9.99MB	55.23
$r_2 = 0.5$	1.81M	6.90MB	3.82M	6.71MB	55.26
$r_3 = 0.35$	1.28M	4.88MB	2.97M	4.69MB	54.38
$r_4 = 0.25$	0.93M	3.55MB	2.41M	3.36MB	54.19
$r_5 = 0.15$	0.58M	2.21MB	1.84M	2.02MB	53.19
$r_6 = 0.05$	0.23M	0.88MB	1.27M	0.69MB	50.75
$r_7 = 0.01$	0.09M	0.34MB	1.05M	0.15MB	37.34