FedPeel: Peeling Stabilized Layers for Robust Heterogeneous Federated Learning

Haizhou Du
Shanghai University of Electric Power
Shanghai, China
haizhou.du@shiep.edu.cn

Lixin Huang
Shanghai University of Electric Power
Shanghai, China

Zonghan Wu
East China Normal University
Shanghai, China

Huan Huo
Sydney University of Technology
Sydney, Australia

Huaicheng Yan
Shanghai University of Electric Power
Shanghai, China

Abstract

Proxy model-based heterogeneous federated learning enables clients with diverse architectures to collaborate through a shared lightweight intermediary. However, current methods aggregate and distill proxy knowledge in a structure-agnostic manner, ignoring layer-wise convergence dynamics, confusing shared and client-specific information, and transferring knowledge without assessing its reliability. These oversights jointly cause model oscillation, drift, and negative transfer. We propose FedPeel, a framework that makes the aggregation pipeline aware of the internal structure of model parameters. Rather than updating the proxy monolithically, FedPeel selectively aggregates only the layers that have stabilized, decouples their frequency-domain representations to separate generalizable patterns from personalized details, and modulates knowledge transfer intensity on a per-sample basis according to teacher confidence. Theoretical analysis establishes an $O(1/T)$ convergence rate toward a neighborhood bounded. Experiments on image, text, and audio benchmarks show that FedPeel consistently outperforms state-of-the-art methods in both accuracy and communication efficiency.

1 Introduction

Heterogeneous Federated Learning (HtFL) allows each client to own a distinct model architecture while collaboratively learning from distributed participants without centralizing raw data [1]. Among the paradigms proposed for HtFL, proxy model-based methods [2, 3] have emerged as a particularly promising direction: lightweight, architecture-agnostic proxy models circulate between clients and server, bypassing the architectural incompatibility that limits knowledge distillation-based [4, 5], prototype-based [6, 7], and submodel-based [8, 9] alternatives.

Despite their flexibility, existing proxy-based methods share a fundamental weakness. They treat the aggregation and distillation pipeline as a monolithic process, aggregating proxy parameters without distinguishing their convergence states, mixing shared and client-specific information indiscriminately, and distilling all teacher signals to local models regardless of reliability. We argue that this structure-agnostic paradigm ignores the intrinsic properties of neural network parameters and leads to three forms of blindness that jointly cause model instability:

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

(i) Oscillation blindness. Deep neural networks exhibit a well-documented layer-wise stabilization phenomenon [10]: shallow layers converge early while deeper layers continue to fluctuate throughout training. Existing methods aggregate all layer parameters at every communication round, regardless of their convergence state. Averaging parameters that are still actively adapting erases local training progress and forces each client to re-learn what it had already captured, resulting in persistent model oscillation across rounds.

(ii) Frequency-domain blindness. Even for layers that have converged, naively averaging their full parameter tensors confuse two distinct types of information. Low-frequency components of layer parameters encode domain-invariant patterns (edges, textures, syntactic structures) that are largely shared across clients, whereas high-frequency components capture fine-grained, client-specific details [11]. An indiscriminate average blurs these personalized details into a "fuzzy consensus" that drifts away from every client's local optimum, leading to model drift.

(iii) Absorption blindness. The proxy model is intentionally kept lightweight to meet communication and computation constraints, so its capacity is typically weaker than the heterogeneous local models. This teacher-student capacity reversal means that on samples far from the global distribution, the proxy's predictions may be highly uncertain. Yet standard knowledge distillation with a fixed temperature treats every sample equally, forcing local models to imitate unreliable guidance and incurring negative transfer.

To address these three forms of blindness, we propose FedPeel, a structure-aware HtFL framework whose key insight is to make each step of the aggregation pipeline explicitly informed by the internal properties of model parameters. Specifically, FedPeel introduces three complementary mechanisms. First, a Stabilized Layer Detection mechanism monitors parameter entropy and norm to identify layers that have converged, and uploads only those layers for aggregation while protecting layers that are still adapting. Second, an Adaptive Frequency-Domain Decoupling mechanism transforms the stabilized layer parameters via DCT and aggregates only the low-frequency components that carry shared knowledge, keeping the high-frequency components that encode client-specific patterns intact. Third, a Sample-wise Temperature-controlled Knowledge Distillation mechanism dynamically adjusts the distillation temperature per sample based on the teacher's Rényi entropy, allowing the local model to absorb high-confidence guidance while discounting uncertain signals.

In summary, our main contributions are as follows:

- We identify a fundamental three-fold blindness problem in existing proxy-based HtFL methods, namely oscillation, frequency-domain and absorption blindness. Based on these, we propose FedPeel, a framework that systematically addresses all three-fold blindness.
- We design a stabilized layer detection mechanism that aggregates only converged layers, preventing model oscillation caused by averaging unstable parameters while naturally reducing communication overhead.
- We introduce adaptive frequency-domain decoupling mechanism to separate shared low-frequency knowledge from client-specific high-frequency details, and sample-wise temperature-controlled distillation to regulate knowledge transfer based on teacher uncertainty.
- Theoretical analysis establishes an $O(1/T)$ convergence rate toward a neighborhood bounded, while the layer-instability residual vanishes asymptotically. Extensive experiments on image, text, and audio benchmarks demonstrate consistent improvements over state-of-the-art methods in both accuracy and communication efficiency.

2 Related Work

We classify the existing methods of HtFL into four categories: *knowledge distillation-based*, *prototype-based*, *submodel-based* and *proxy model-based methods*.

Knowledge Distillation-Based FL Methods. Knowledge distillation(KD)-based FL methods use model outputs (logits), so they are suitable for model-heterogeneous scenarios. Some methods, such as FeAUX [12], FedDF [13], and DS-FL [14], enable models to learn from others by identifying differences in model outputs on a global dataset. To overcome the limitations of public datasets, some methods have been proposed: FedGen [15] uses a global generator based on clients' outputs

to generate data. Some KD-based methods such as FedKD [16], DKDR [4] and FedFD [5] share models as knowledge and integrate the outputs of multiple models to enhance local or global models. However, these methods incur additional computational overhead or require a high-quality public dataset, constraining their applicability.

Prototype-Based FL Methods. Prototype-based FL methods utilize local data feature representations. For example, FedPAC [17], FedGKT [18], and FedProto [19] upload local class-specific representations or prototypes to the server to gain global knowledge. FedTGP [6] introduces a regularization method to increase the spacing between classes, enhancing prototype distinctiveness. FedSA [7] employs semantic anchors to align global prototypes within a unified feature space. FedPall [20] utilizes prototype-based adversarial learning to unify feature spaces and collaborative learning. However, these methods incur substantial computational overhead and discard fine-grained local structure.

Submodel-Based FL Methods. The concept of Submodel-based FL methods is to divide client models into two parts: one for global aggregation and another one retained locally for model personalization. For instance, methods like FedGC [9], FedROD [21], FedRep [22], FedBABU [23], and FedAlt [24] split the model into homogeneous feature extractors and heterogeneous classifiers, enabling data mapping to a consensus feature space and preserving local personalization through classifiers. In contrast, methods such as LG-FedAvg [8], and Fedclassavg [25] use heterogeneous feature extractors and homogeneous classifiers, achieving global knowledge acquisition by sharing classifiers for unified classification standards. However, these methods that reduce inter-client model heterogeneity inevitably compromise local model personalization and diminish the diversity of learned representations across clients.

Proxy model-based methods. The core idea of proxy-model FL methods is to decouple heterogeneous client models from the federation process by introducing a lightweight, architecture-agnostic proxy that circulates instead of the heterogeneous model. Methods such as ProxyFL [26], FML [27], FedMRL [2] and Moss [3] employ a lightweight proxy model as a universal bridge, enabling clients with different architectures to collaborate without ever exposing their private, heterogeneous models. However, the server struggles to perform unbiased aggregation of proxy models, while clients cannot efficiently and faithfully transfer knowledge between their heterogeneous models and the homogeneous proxy.

3 Methodology

3.1 Problem Statement

The general optimization problem of HtFL can be formulated as:

$$\min_{\{\theta_i\}_{i=1}^{K}} \frac{1}{K} \sum_{i=1}^{K} F_i(\theta_i), \tag{1}$$

where $F_i(\theta_i) = \mathbb{E}_{(x,y)\sim\mathcal{D}_i}[\ell_i(f_i(x;\theta_i), y)]$ denotes the expected local loss of client i with data drawn from distribution $\mathcal{D}_i$, and K is the number of clients.

3.2 The Overview of FedPeel

We propose FedPeel, a novel framework that peels stabilized layers from proxy models and decouples the frequency-domain of stabilized layers to improve collaborative learning across heterogeneous clients.

At the client side, a heterogeneous local model fits the client's data and task, while a lightweight homogeneous proxy model acts as the cross-client knowledge bridge. Instead of uploading the entire proxy model, each client uploads only the parameters of the deepest newly stabilized layer of its proxy model. Shallower stabilized layers are not re-uploaded because they have already been aggregated in earlier rounds. To enable efficient knowledge transfer from the proxy to the heterogeneous model, we design a sample-wise temperature-controlled knowledge distillation (STKD) mechanism. It adjusts the distillation temperature per sample based on the teacher's predictive uncertainty, helping the heterogeneous model learn generalized knowledge while ignoring uncertain or noisy signals.

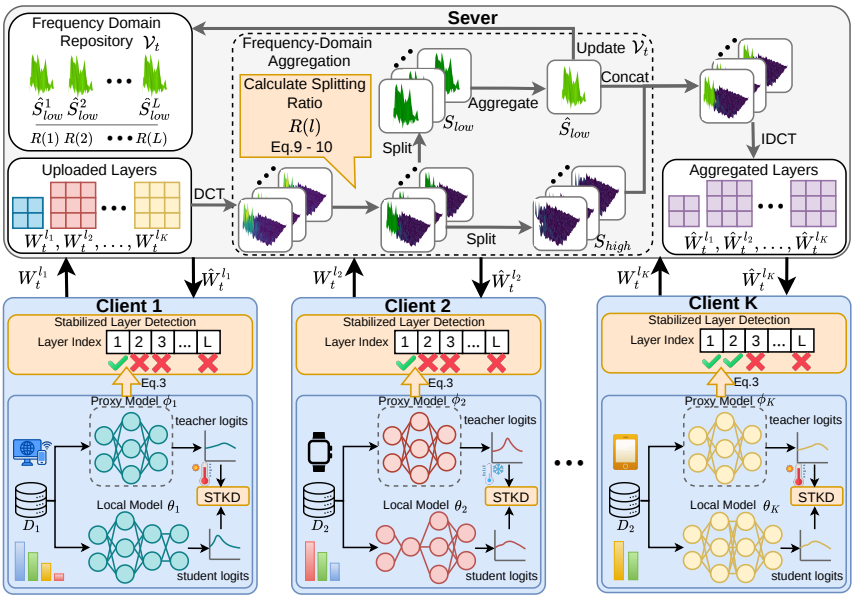


Figure 1: The Overview of FedPeel.

On the server side, we propose an adaptive frequency-domain decoupling mechanism. Stabilized layer parameters are transformed via Discrete Cosine Transform (DCT) to separate shared knowledge from client-specific patterns via layer depth and frequency-domain energy. Low-frequency components are aggregated, while high-frequency details are kept. The aggregated low-frequency knowledge is then reconstructed with high-frequency parts and sent back to update clients' proxy models. To address isolation from different convergence speeds, the server maintains a frequency-domain repository that supplies historical common knowledge to clients that cannot benefit from current round.

In summary, FedPeel incorporates three key mechanisms. The subsequent sections detail the design of STKD (Section 3.3), the stabilized layer detection (Section 3.4), and the adaptive frequency-domain decoupling mechanism (Section 3.5). The overall aggregation procedures and client-side training are outlined in Algorithm 1 and Algorithm 3 in Appendix A.2, respectively.

3.3 Sample-wise Temperature-controlled Knowledge Distillation

To enable effective knowledge sharing across models, we employ knowledge distillation to transfer knowledge from the proxy model to heterogeneous models, thereby reconciling architectural discrepancies and boosting local performance. Yet the aggregated parameters deviate from any single client's local data. Under serious non-IID conditions, this mismatch intensifies, misaligning distilled signals with local objectives. Moreover, the homogeneous proxy model is typically lightweight to meet communication and computation constraints, so its capacity falls short of the larger heterogeneous model. This teacher–student capacity reversal not only caps the upper bound of transferable knowledge but also slows local convergence through gradient-scale misalignment, ultimately degrading overall collaboration efficiency.

We design the sample-wise temperature-controlled knowledge distillation, which dynamically adjusts the distillation temperature to decide how much the student trusts the teacher and to encourage appropriate self-confidence. Specifically, given the teacher's probability distribution P_e, we compute its Rényi-2 entropy $R_e = -\log \sum_c P_e(x)^2$. R_e reflects the teacher's uncertainty in the interval $(0, \log C]$: a larger R_e indicates a flatter teacher output and higher uncertainty. We then map R_e to a temperature coefficient $\mathcal{T}$:

$$\mathcal{T} = e^{\beta R_e}. \tag{2}$$

This temperature is applied to both the teacher and the student. When the teacher is uncertain, the raised temperature flattens the teacher distribution, and the Kullback-Leibler (KL) divergence term is annealed, alleviating the student's imitation burden. Conversely, when the teacher is certain, the lower

temperature preserves fine-grained alignment to the teacher's outputs. To prevent blind imitation of an uncertain teacher and to foster student self-confidence, we add an entropy-based regularizer $\mathcal{L}_s = \frac{-\log \sum P_s(x)^2}{\log C}$ that encourages students to output low-entropy predictions. By dynamically balancing "imitating teacher" and "trusting itself", the mechanism simultaneously suppresses noisy distillation and progressively enhances the student's certainty, yielding more robust knowledge transfer. The loss function of STKD is as follows:

$$\mathcal{L}_{\text{STKD}} = KL(P_s^{\mathcal{T}} \mid\mid P_e^{\mathcal{T}}) + \mathcal{L}_s, \tag{3}$$

where $P_s^{\mathcal{T}}$ and $P_e^{\mathcal{T}}$ are the distributions of student output and teacher output after temperature $\mathcal{T}$ processing, respectively. And the final optimization objective of heterogeneous models on client i is:

$$\min_{\theta_i} \mathbb{E}_{(x,y)\sim\mathcal{D}_i} \left[\mathcal{L}_{\text{task}}(f(x;\theta_i), y) + \mathcal{L}_{\text{STKD}}(x;\theta_i;\phi_i)\right], \tag{4}$$

where θ_i and ϕ_i are the heterogeneous and proxy model on client i.

We remark that the formal advantage of STKD over fixed-temperature KD relies on a mild assumption linking distribution shift to teacher uncertainty; this Uncertainty-Divergence Coupling assumption and the resulting bound are stated and proved in Appendix B (Theorem B.2).

3.4 Stabilized Layer Detection

Heterogeneous device capabilities and data volumes yield inconsistent convergence speeds. In addition, deep neural networks exhibit a layer-wise stabilization phenomenon [10] in which shallow layers converge significantly earlier than deeper ones (Theorem 3.1). Therefore, simply averaging all parameters erases the training progress of under-stabilized layers and slows overall convergence speed. FedPeel circumvents this issue by aggregating only the stable layers of the homogeneous proxy model, not the heterogeneous local models. The proxy's lightweight architecture stabilizes quickly and consistently, providing a reliable communication bridge without interfering with the personalized training of heterogeneous clients. Each round, each client uploads only the single deepest stabilized layer $l_i = \max\{l \mid \text{Layer } l \text{ satisfies Eq. (7)}\}$. Shallower stabilized layers are not uploaded because they have already been aggregated in previous rounds. This protects local training progress and accelerates global convergence.

Theorem 3.1. *Layer-wise Stabilization*
Consider an L-layer feed-forward network with weights $W^l \in \mathbb{R}^{d_l \times d_{l-1}}$, $1 \le l \le L$, and loss $\mathcal{L}$. Let

$$S_l(t) = \frac{\left\|\partial \mathcal{L}/\partial W_t^l\right\|}{\|W_t^l\| + \xi}, \qquad \xi > 0, \tag{5}$$

where $t \ge 0$ denotes the training iteration index under a fixed learning rate $\eta > 0$.

Assume that for each layer l, the sequence $\{S_l(t)\}_{t\ge 0}$ is non-increasing and converges to 0 as $t \to \infty$. Furthermore, assume that for all $t \ge 0$ and all $l = 1, \ldots, L$, $S_l(t) \le S_{l+1}(t)$. For a given tolerance $\epsilon > 0$, define the stabilization time $T_l(\epsilon) = \inf\{\, t \ge 0 : S_l(t) \le \epsilon \,\}$. Then

$$T_1(\epsilon) \le T_2(\epsilon) \le \cdots \le T_L(\epsilon). \tag{6}$$

The proof of this theorem is in Appendix B.2.

To address oscillation blindness, we propose a stabilized layer-based aggregation that merges only layers that have learned enough and no longer change by jointly thresholding the layer-parameter change rate ϵ of entropy and norm. Given that changes in layer parameter norms $\|W_t^l\|$ may not fully capture subtle variations in parameter distributions, we introduce the entropy $H(W_t^l)$ of the parameter distribution as a complementary metric. $H(\cdot)$ is estimated from the flattened weight tensor after quantization into a set number of bins [28]. While norms quantify the overall magnitude of parameters, and entropy characterizes the distributional properties of parameter values. Their combination enables a more comprehensive characterization of the layers' parameters. We determine whether a certain layer is stabilized by the change rate ϵ of continuous E-rounds:

$$\left|\frac{\|W_t^l\| - \|W_{t-k}^l\|}{\|W_{t-k}^l\|}\right| \le \epsilon \wedge \left|\frac{H(W_t^l) - H(W_{t-k}^l)}{H(W_{t-k}^l)}\right| \le \epsilon, \quad \forall k \in \{1, \ldots, E\}. \tag{7}$$

3.5 Adaptive Frequency-domain Decoupling

To extract generalized knowledge from the stabilized layer, we transform the layer parameters to the frequency domain via the discrete cosine transform (DCT), decompose the frequency-domain into low- and high-frequency components, and only perform aggregation on low-frequency components, which capture universal patterns [11]. Then, the aggregated low-frequency components and high-frequency components are merged and reconstructed into a parameter space through inverse discrete cosine transform (IDCT), mitigating model drift.

Nevertheless, the informational role of a layer is position-dependent: shallow layers encode low-level features (edges, textures) that are largely domain-invariant, whereas deeper layers learn semantic abstractions that are more client-specific. A fixed cutoff frequency would therefore either over-smooth early features or under-regularize late ones. To address this, we design an adaptive splitting ratio.

For each layer l, we compute the frequency-domain S^l, flatten it to a vector $\mathcal{F}_l \in \mathbb{R}^{d_l d_{l-1}}$, and sort its entries in descending order. The normalized cumulative energy is:

$$E_l(j) = \sum_{i=1}^{j} \mathcal{F}_l(i)^2 / \sum_{i=1}^{N} \mathcal{F}_l(i)^2, \;\; j = 1, \ldots, d_l d_{l-1}. \tag{8}$$

The smallest index $r = \min\{j | E_l(j) \geq \gamma\}$ gives the minimal fraction of coefficients that retain $\gamma\%$ of total energy. Due to the different roles of layers at different depths in the model, their corresponding frequency-domain energies are different. In order to achieve a more accurate split, we introduce a bias $B(l) \in [-b, b]$ based on the depth of layers to adjust the splitting ratio:

$$B(l) = b[1 - \frac{2(l-1)}{L-1}], \;\; b \geq 0. \tag{9}$$

The adaptive splitting ratio $\mathcal{R}(l)$ is obtained by averaging the r across the same clients layer and adding the bias:

$$\mathcal{R}(l) = \frac{\sum_{i \in N_l} r_i}{N_l} + B(l), \tag{10}$$

where N_l is the number of clients uploading layer l. Coefficients are selected by fixed DCT frequency indices, not by global magnitude ranking. Sorting only computes the scalar splitting ratio r. After obtaining $\mathcal{R}(l)$, we apply the mask to the original DCT frequency layout. The remaining high-frequency coefficients keep their original positions.

Theorem 3.2 (Convergence of FedPeel)**.** *Under standard assumptions of M-smoothness, bounded gradient variance, and bounded gradient norms, with learning rate $\eta \leq \frac{1}{M}$, FedPeel achieves after T communication rounds:*

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[\Phi_t] \leq \frac{2(F^1 - F^*)}{\eta T} + \eta M \sigma^2 + \frac{4G^2}{K} \sum_{l=1}^{L} \rho_l(T) + \Delta_{STKD}, \tag{11}$$

where $F^1 = \frac{1}{K}\sum_{i=1}^{K} F_i(\theta_i^1)$ is the initial objective, F^* is the minimum value, σ^2 is the gradient variance bound, $\rho_l(T)$ quantifies layer instability, Δ_{STKD} bounds the gradient misalignment from adaptive distillation, $\Phi_t = \frac{1}{K}\sum_{i=1}^{K} \|\nabla F_i(\theta_i^t)\|^2$, and G and M are the gradient and smoothness bounds. Specifically, as $T \to \infty$, we have:

$$\lim_{T \to \infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[\Phi_t] \leq \eta M \sigma^2 + \Delta^*_{STKD}, \tag{12}$$

where Δ^*_{STKD} is the distillation error. This implies that FedPeel achieves an $O(1/T)$ convergence rate toward a neighborhood bounded by $\eta M \sigma^2 + \Delta^*_{STKD}$. The proof is provided in Appendix C.

4 Experiments

We conduct comprehensive comparative experiments to validate the superior performance of FedPeel in scenarios with varying client scales and degrees of data heterogeneity. Additional experimental results are provided in Appendix D.3.

Algorithm 1 The aggregation process of FedPeel.

Input: Uploaded layer parameters $\{W_{t,i}^l\}_{i=1}^{\mathcal{K}}$, Hyperparameters: γ, c
Output: Aggregated layer parameters $\{\hat{W}_{t,i}^l\}_{i=1}^{\mathcal{K}}$
for each layer position l **do**
  $N_l \leftarrow \{\{l_{t,i}\}_{i=1}^{\mathcal{K}} | l_{t,i} = l\}$
  if $|N_l| > 1$ **then**
    for each $i \in N_l$ **do**
      $S_i^l \leftarrow \text{DCT}(W_{t,i}^l)$
      $r_i \leftarrow \text{SplitingRatio}(S_i^l, \gamma)$
    end for
    $\mathcal{R}(l) \leftarrow \frac{1}{|N_l|} \sum_{i \in N_l} r_i + B(l)$ {Eq. 10}
    $\hat{S}_{\text{low}}^l \leftarrow \frac{1}{|N_l|} \sum_{i \in N_l} S_i^l[0 : \lfloor \mathcal{R}(l) \cdot |S_i^l| \rfloor]$
    $\mathcal{V}(l) \leftarrow \hat{S}_{\text{low}}^l, \mathcal{R}(l)$
    for each $i \in N_l$ **do**
      $\hat{W}_{t,i}^l \leftarrow \text{IDCT}(\text{Concat}(\hat{S}_{\text{low}}^l, S_i^l[\lfloor \mathcal{R}(l) \cdot |S_i^l| \rfloor + 1 : |S_i^l|]))$
      Send $\hat{W}_{t,i}^l$ to client i
    end for
  end if
end for

4.1 Experimental Setup

Dataset. We evaluate FedPeel on several widely recognized datasets: CIFAR-10, CIFAR-100 [29], AGNews [30], and Google Speech Commands (GSC) [31]. These datasets span different tasks and difficulty levels.

Baseline methods. To evaluate FedPeel, we focus on statistical and model heterogeneity settings in federated learning. As we target model-heterogeneous federated learning, the selected baselines must handle scenarios where client model architectures differ. Accordingly, we cover the main technical paradigms of existing methods—knowledge distillation-based (FedKD [16]), prototype-based (FedTGP [6], FedSA [7]), and proxy model-based (FedMRL [2], Moss [3]) methods—to ensure comprehensive comparisons.

Statistical heterogeneity. Following prior work [32, 33], we impose statistical heterogeneity across clients by drawing label proportions from a Dirichlet distribution Dir(α), with α set to 0.1 by default.

Model heterogeneity. For image classification tasks, we employ ResNet series, GoogleNet, and MobileNet as heterogeneous models, and a standard CNN as a homogeneous model. For text classification tasks, we use transformer-based models. For the audio task, we convert the Google Speech Commands dataset into MFCC feature maps and apply a network that shares the same architecture as the image classification models (*e.g.*, ResNet), but with different parameters due to varying input and output dimensions. Detailed model configurations are provided in Appendix D.1.

Hyperparameter setting. We set the client participation rate (p) to 0.2 per round, allowing us to evaluate method performance when only a small subset of clients is active. Each client performs one local training epoch per round. The batch size is 64, the learning rate η is 0.0005, and the total number of communication rounds T is 100. Each client's local dataset is split into training (75%) and testing (25%) sets. Unless stated otherwise, we set β at 2, γ at 0.9, b at 0.05 and ϵ at 0.03. We run all experiments three times with same settings. For each communication round, we average the local test accuracies of all clients, where each client is evaluated on its own local test set.

4.2 Experimental Results

4.2.1 Performance Evaluation

To evaluate FedPeel's performance, we compared it with state-of-the-art methods across different datasets and client scales. Table 1 shows that FedPeel outperforms every baseline on all four datasets, regardless of client scale. This robust advantage stems from our three core mechanisms. Stable-

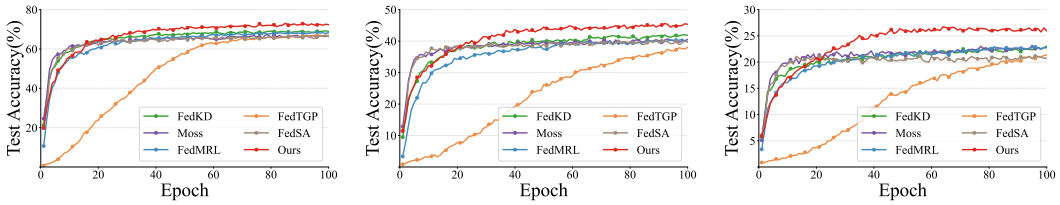
Table 1: Best client-averaged local test accuracy(%) under different datasets in heterogeneous scenarios.

Dataset	Clients (K)	FedKD	FedTGP	FedMRL	FedSA	Moss	Ours	Improvement (%)
CIFAR-10	20	76.04 ± 1.34	68.42 ± 1.67	81.11 ± 0.98	82.79 ± 1.02	81.43 ± 1.15	**83.57** ± 0.89	0.94
	50	69.48 ± 1.56	66.47 ± 1.89	78.81 ± 1.23	79.09 ± 1.34	77.62 ± 1.41	**80.59** ± 1.12	1.89
	100	67.86 ± 1.89	68.13 ± 2.23	76.13 ± 1.67	77.12 ± 1.89	75.14 ± 1.92	**79.16** ± 1.67	2.64
CIFAR-100	20	41.99 ± 1.72	38.02 ± 1.89	40.63 ± 1.23	40.61 ± 1.27	40.68 ± 1.41	**45.57** ± 1.23	8.52
	50	31.19 ± 1.94	31.96 ± 1.78	36.01 ± 1.56	33.27 ± 1.62	35.88 ± 1.67	**43.84** ± 1.56	21.74
	100	29.41 ± 2.31	35.19 ± 2.14	35.52 ± 1.89	31.65 ± 2.01	34.93 ± 2.08	**40.58** ± 2.01	14.24
AGNEWS	20	88.64 ± 0.52	88.47 ± 0.61	88.68 ± 0.56	93.61 ± 0.48	88.53 ± 0.64	**93.91** ± 0.42	0.32
	50	92.34 ± 0.67	92.41 ± 0.73	92.47 ± 0.69	92.85 ± 0.71	87.71 ± 0.89	**93.89** ± 0.63	1.12
	100	91.38 ± 0.84	90.72 ± 0.92	91.61 ± 0.87	91.31 ± 0.86	85.66 ± 1.12	**91.84** ± 0.81	0.25
GSC	20	77.63 ± 1.23	86.79 ± 0.99	87.12 ± 0.92	85.99 ± 0.96	86.74 ± 0.94	**88.94** ± 0.93	2.08
	50	65.73 ± 1.45	81.49 ± 1.12	81.59 ± 1.08	81.76 ± 1.15	82.26 ± 1.21	**83.11** ± 1.09	1.03
	100	59.31 ± 1.89	70.28 ± 1.67	73.21 ± 1.55	72.61 ± 1.62	74.26 ± 1.71	**76.86** ± 1.58	3.50

layer aggregation prevents noise by only uploading parameters whose local training has converged. Frequency domain-guided knowledge decoupling isolates knowledge sharing from high-frequency, client-specific details. In a nutshell, these components yield consistent gains across image, text, and audio modalities, confirming the generality of FedPeel.

4.2.2 Impact of Statistical Heterogeneity

We systematically evaluate FedPeel under statistical heterogeneity using CIFAR-100 with 20 clients as shown in Figure 2. Following established practice, we generate controlled label-skewed partitions by drawing class proportions from a symmetric Dirichlet distribution, varying the concentration parameter α from 0.01 (serious non-IID) to 0.5 (mild non-IID).



(a) $\alpha = 0.01$ (b) $\alpha = 0.1$ (c) $\alpha = 0.5$

Figure 2: Test accuracy under varying data heterogeneity with 20 clients.

Smaller α implies stronger heterogeneity, so local models adapt better to their class distributions and local test accuracy can be higher. FedPeel consistently outperforms all baselines. We attribute this to two mechanisms: stable-layer aggregation reduces model oscillation (only stabilized layers are transmitted), and frequency-domain decoupling limits client-specific label bias. FedPeel converges more slowly at first, as only shallow layers are stabilized and aggregated; as training proceeds, deeper layers gradually satisfy the stability criterion and join aggregation. Combined with frequency-domain decoupling and STKD, this suppresses negative transfer and improves late-stage convergence.

4.2.3 Communication Overhead Analysis

We compare per-round communication cost of FedPeel against model-based baselines (FedKD, FedMRL). Figure 3 shows that FedPeel transmits only the parameters of stabilized layers, whose size are typically much smaller than the full model. Because the deepest stabilized layer may contain a large number of parameters and no compression technique is used, the communication volume of FedPeel can be higher than that of FedKD in some rounds. However, FedPeel achieves lower communication overhead than FedKD and FedMRL in most rounds. This efficiency is a consequence of stabilized-layer aggregation.

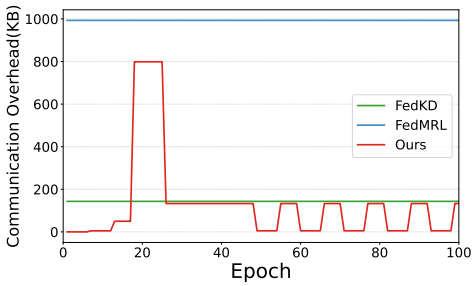


Figure 3: The communication overhead of different methods.

4.3 Sensitivity Analysis

We systematically explore the impact of hyperparameters on framework performance using CIFAR-100. Hyperparameters independent of the framework (Dirichlet coefficient $\alpha = 0.1$, client count $K = 20$) remain fixed. More sensitivity analyses are provided in the Appendix D.3.5.

Hyperparameter β. In order to investigate the effect of hyperparameter β in adaptive temperature control knowledge distillation (Eq. 2) on the transfer of knowledge from homogeneous models to heterogeneous models, we conduct sensitivity analysis on hyperparameter β, with values ranging from 1 to 5, and measure the test accuracy of heterogeneous models under different settings. The experimental results are shown in Table 2.

Table 2: β-Sensitivity of Adaptive Temperature for Knowledge Distillation.

β	1	2	3	4	5
Accuracy	43.26%	45.57%	43.37%	42.96%	42.86%

As β increases from 1 to 5, student accuracy follows a single-peaked trajectory that rises sharply to its maximum at β = 2 and then declines monotonically. This inverted-U pattern reveals a narrow optimal interval for the entropy-scaling coefficient. At β = 2, the mapping between entropy and temperature sits at the critical inflection point: high-entropy samples receive enough smoothing to preserve the teacher's ranking of confusing classes, while low-entropy samples retain near-one temperatures that safeguard the teacher's confident decisions.

Hyperparameter γ. γ controls the proportion of spectral energy retained in low-frequency components. Smaller γ preserves less shared spectral information; larger γ may introduce more client-specific high-frequency components into aggregation, trading off global sharing and personalization.

To investigate the impact of energy threshold γ on server aggregation, we conducted sensitivity analysis with γ values of 70, 80, and 90. As shown in Table 3, the accuracy first drops slightly from 42.81% ($\gamma = 70$) to 42.65% ($\gamma = 80$), then rises sharply to 45.57% ($\gamma = 90$). A possible explanation is that at $\gamma = 70$ and 80, the retained low-frequency energy is insufficient, allowing high-frequency noise to leak into the aggregation path. This over-smooths the frequency-domain representation and discards discriminative structure.

Table 3: Energy-Guided Split Sensitivity.

$\gamma(\%)$	70	80	90
Accuracy	42.81%	42.65%	45.57%

Hyperparameter ϵ. ϵ is a key indicator for determining stability layers, which directly affects the screening of the stability layer and the aggregation effect on the server-side. Therefore, we conducted a sensitivity analysis on ϵ . The experimental results are shown in Table 4, where $l_{\max}$ represents the index of the deepest stability layer in the model, and $T_L(\epsilon)$ represents the number of communication rounds when the last layer L is stabilized.

Table 4: Stable layer states of models under different ϵ.

ϵ	0.01	0.03	0.05	0.1	0.3	0.5
Accuracy	43.01%	45.57%	44.34%	44.24%	43.81%	43.52%
$l_{\max}$	3	5	5	5	5	5
$T_L(\epsilon)$	-	53	42	37	30	30

From the experimental results, it can be seen that the highest accuracy is achieved when ϵ is 0.03. When the standards are too strict, some layers, especially the last two layers of the model, cannot be considered to have reached a stabilized state and therefore cannot participate in aggregation; loose standards cannot effectively and instantly filter out stabilized layers.

4.4 Ablation Studies

In order to evaluate the role of each mechanism in FedPeel, we conducted ablation experiments, and the experimental results are shown in Table 5.

Table 5: Ablation studies on the key mechanisms of FedPeel on CIFAR-100 with K = 20.

STKD	Stabilized Layer Detection	Adaptive Frequency-domain Decoupling	Accuracy
✓	✓	✓	45.57%
×	✓	✓	43.92%
✓	×	✓	44.17%
✓	✓	×	44.11%
×	×	×	43.03%

The results show that: (i) removing STKD (replacing it with vanilla KD) costs 1.65% accuracy, since the local model can no longer discount uncertain proxy guidance; (ii) removing stabilized-layer filtering costs 1.4%, since averaging unstable parameters across clients causes oscillation and overwrites already-converged client specific features; (iii) removing adaptive frequency-domain decoupling costs 1.46%, since a fixed cutoff cannot separate shared low-frequency patterns from client-specific high-frequency details, causing over-smoothing or under-regularization; and (iv) the small drop after removing all three components indicates that the proxy-model base framework already provides a strong foundation. Additional ablation with $K = 50$ on CIFAR-100 is in Appendix D.3.6.

5 Conclusion

In this paper, we propose FedPeel, a personalized heterogeneous federated learning framework based on stabilized-layer aggregation and frequency-domain decoupling. By sharing only stabilized proxy layers and extracting low-frequency knowledge from them, FedPeel reduces model drift while preserving each client's personalized knowledge. A sample-wise temperature-controlled distillation mechanism prevents heterogeneous models from blindly imitating the proxy model. Extensive experiments show that FedPeel surpasses existing methods under both statistical and model heterogeneity on diverse datasets.

6 Funding Statement

This work was conducted without any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. No financial support was received for the research, authorship, and/or publication of this article.

7 Competing Interests Disclosures

No known competing financial interests or personal relationships could have appeared to influence the work reported in this paper. No financial or non-financial interests are declared.

References

[1] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[2] Liping Yi, Han Yu, Chao Ren, Gang Wang, Xiaoxiao Li, et al. Federated model heterogeneous matryoshka representation learning. *Advances in Neural Information Processing Systems*, 37: 66431–66454, 2024.

[3] Yifeng Cai, Ziqi Zhang, Ding Li, Yao Guo, and Xiangqun Chen. Moss: Proxy model-based full-weight aggregation in federated learning with heterogeneous models. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 9(1), March 2025. doi: 10.1145/3712274. URL `https://doi.org/10.1145/3712274`.

[4] Yueyang Yuan, Wenke Huang, Guancheng Wan, Kaiqi Guan, He Li, and Mang Ye. DKDR: Dynamic knowledge distillation for reliability in federated learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL `https://openreview.net/forum?id=To2fQWv0zP`.

[5] Yichen Li, Xiuying Wang, Wenchao Xu, Haozhao Wang, Yining Qi, Jiahua Dong, and Ruixuan Li. Feature distillation is the better choice for model-heterogeneous federated learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL `https://openreview.net/forum?id=xYik0sKYVo`.

[6] Jianqing Zhang, Yang Liu, Yang Hua, and Jian Cao. FedTGP: Trainable global prototypes with adaptive-margin-enhanced contrastive learning for data and model heterogeneity in federated learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 16768–16776, 2024.

[7] Yanbing Zhou, Xiangmou Qu, Chenlong You, Jiyang Zhou, Jingyue Tang, Xin Zheng, Chunmao Cai, and Yingbo Wu. FedSA: A unified representation learning via semantic anchors for prototype-based federated learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pages 23009–23017, 2025.

[8] Paul Pu Liang, Terrance Liu, Liu Ziyin, Nicholas B Allen, Randy P Auerbach, David Brent, Ruslan Salakhutdinov, and Louis-Philippe Morency. Think locally, act globally: Federated learning with local and global representations. *arXiv preprint arXiv:2001.01523*, 2020.

[9] Yifan Niu and Weihong Deng. Federated learning for face recognition with gradient correction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 1999–2007, 2022.

[10] Yixiong Chen, Alan Yuille, and Zongwei Zhou. Which layer is learning faster? a systematic exploration of layer-wise convergence rate for deep neural networks. In *The Eleventh International Conference on Learning Representations*, 2023.

[11] Kecheng Zheng, Yang Cao, Kai Zhu, Ruijing Zhao, and Zheng-Jun Zha. FAMLP: A frequency-aware mlp-like architecture for domain generalization. *arXiv preprint arXiv:2203.12893*, 2022.

[12] Felix Sattler, Tim Korjakow, Roman Rischke, and Wojciech Samek. FedAUX: Leveraging unlabeled auxiliary data in federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 34(9):5531–5543, 2021.

[13] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. *Advances in neural information processing systems*, 33: 2351–2363, 2020.

[14] Sohei Itahara, Takayuki Nishio, Yusuke Koda, Masahiro Morikura, and Koji Yamamoto. Distillation-based semi-supervised federated learning for communication-efficient collaborative training with non-IID private data. *IEEE Transactions on Mobile Computing*, 22(1): 191–205, 2021.

[15] Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. Data-free knowledge distillation for heterogeneous federated learning. In *International conference on machine learning*, pages 12878–12889. PMLR, 2021.

[16] Chuhan Wu, Fangzhao Wu, Lingjuan Lyu, Yongfeng Huang, and Xing Xie. Communication-efficient federated learning via knowledge distillation. *Nature communications*, 13(1):2032, 2022.

[17] Jian Xu, Xinyi Tong, and Shao-Lun Huang. Personalized federated learning with feature alignment and classifier collaboration. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=SXZr8aDKia.

[18] Chaoyang He, Murali Annavaram, and Salman Avestimehr. Group knowledge transfer: Federated learning of large CNNs at the edge. *Advances in neural information processing systems*, 33:14068–14080, 2020.

[19] Yue Tan, Guodong Long, Lu Liu, Tianyi Zhou, Qinghua Lu, Jing Jiang, and Chengqi Zhang. FedProto: Federated prototype learning across heterogeneous clients. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 8432–8440, 2022.

[20] Yong Zhang, Feng Liang, Guanghu Yuan, Min Yang, Chengming Li, and Xiping Hu. FedPall: Prototype-based adversarial and collaborative learning for federated learning with feature drift. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3111–3120, October 2025.

[21] Hong-You Chen and Wei-Lun Chao. On bridging generic and personalized federated learning for image classification. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=I1hQbx10Kxn.

[22] Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. In *International conference on machine learning*, pages 2089–2099. PMLR, 2021.

[23] Jaehoon Oh, SangMook Kim, and Se-Young Yun. FedBABU: Toward enhanced representation for federated image classification. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=HuaYQfggn5u.

[24] Krishna Pillutla, Kshitiz Malik, Abdel-Rahman Mohamed, Mike Rabbat, Maziar Sanjabi, and Lin Xiao. Federated learning with partial model personalization. In *International Conference on Machine Learning*, pages 17716–17758. PMLR, 2022.

[25] Jaehee Jang, Heoneok Ha, Dahuin Jung, and Sungroh Yoon. FedClassAvg: Local representation learning for personalized federated learning on heterogeneous neural networks. In *Proceedings of the 51st international conference on parallel processing*, pages 1–10, 2022.

[26] Shivam Kalra, Junfeng Wen, Jesse C Cresswell, Maksims Volkovs, and Hamid R Tizhoosh. Decentralized federated learning through proxy model sharing. *Nature communications*, 14(1): 2899, 2023.

[27] Tao Shen, Jie Zhang, Xinkang Jia, Fengda Zhang, Gang Huang, Pan Zhou, Kun Kuang, Fei Wu, and Chao Wu. Federated mutual learning. *arXiv preprint arXiv:2006.16765*, 2020.

[28] Claude E Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

[29] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[30] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28*, pages 649–657, 2015.

[31] Pete Warden. Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition. *arXiv e-prints*, art. arXiv:1804.03209, April 2018. doi: 10.48550/arXiv.1804.03209.

[32] Qinbin Li, Yiqun Diao, Quan Chen, and Bingsheng He. Federated learning on non-IID data silos: An experimental study. In *2022 IEEE 38th international conference on data engineering (ICDE)*, pages 965–978. IEEE, 2022.

[33] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10713–10722, 2021.

[34] Jianqing Zhang, Xinghao Wu, Yanbing Zhou, Xiaoting Sun, Qiqi Cai, Yang Liu, Yang Hua, Zhenzhe Zheng, Jian Cao, and Qiang Yang. HtFLlib: A comprehensive heterogeneous federated learning library and benchmark. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2025.

A Notations and Algorithm

A.1 Notations

Table 6 summarizes the mathematical notations used throughout this paper:

Table 6: Summary of Notations

Notation	Description
Federated Learning Framework	
K	Total number of clients
i	Client index, $i \in \{1, \ldots, K\}$
T	total communication round
t	Communication round index, $t \in \{1, \ldots, T\}$
$\mathcal{D}_i$	Local data distribution of client i
(x, y)	Input-target pair sampled from $\mathcal{D}_i$
η	Learning rate
Model Parameters	
θ_i, ϕ_i	Parameters of heterogeneous local model and homogeneous proxy model for client i
$f(\cdot; \theta_i)$	Heterogeneous local model function
$g(\cdot; \phi_i)$	Homogeneous proxy model function
L	Total number of layers in ϕ
l	Layer index, $l \in \{1, \ldots, L\}$
W^{l_i}	Parameters of layer l for client i
d_l, d_{l-1}	Output dimension, input dimension to layer l
STKD (Sample-wise Temperature-controlled Knowledge Distillation)	
R_e	Rényi-2 entropy of teacher: $R_e = -\log \sum_c P_e(x)^2$
$\mathcal{T}$	Adaptive temperature: $\mathcal{T} = e^{\beta R_e}$
β	Temperature scaling coefficient
Stable Layer Detection	
ϵ	Stability threshold for parameter change rate
E	Sliding window size for stability detection
$H(W)$	Entropy of parameter distribution
$\|W\|$	Norm of parameter matrix
Spectral Decomposition and Aggregation	
$\text{DCT}(\cdot), \text{IDCT}(\cdot)$	Discrete Cosine Transform and Inverse Discrete Cosine Transform
$\mathcal{F}_l$	Magnitude spectrum vector for layer l
$E_l(k)$	Normalized cumulative energy: $E_l(k) = \sum_{i=1}^{k} \mathcal{F}_l(i)^2 / \sum_{i=1}^{N} \mathcal{F}_l(i)^2$
γ	Energy threshold for frequency splitting
$B(l)$	Depth-based bias for adaptive splitting: $B(l) = b(1 - \frac{2(l-1)}{L-1})$
b	Bias coefficient
$\mathcal{R}(l)$	Adaptive splitting ratio: $\mathcal{R}(l) = \frac{1}{N_l} \sum_{i=1}^{N_l} r_i + B(l)$
N_l	Number of clients uploading layer l
$S_{\text{low}}, S_{\text{high}}$	Low-frequency and high-frequency components
$\mathcal{V}$	Spectrum repository on server

A.2 Algorithm

Algorithm 2 FedPeel: Personalized Heterogeneous Federated Learning with Stable Layers

Input: K clients with heterogeneous models $\{\theta_i\}_{i=1}^{K}$, proxy models $\{\phi_i\}_{i=1}^{K}$,
Input: Hyperparameters: $\epsilon, \gamma, c, \beta$
Server executes:
Initialize spectrum repository $\mathcal{V} \leftarrow \emptyset$
for each communication round $t = 1, 2, \ldots, T$ **do**
  $\mathcal{K} \leftarrow$ randomly select $\max(p \cdot K, 1)$ clients
  for each client $i \in \mathcal{K}$ **in parallel do**
    $W_t^{l_i} \leftarrow \text{ClientUpdate}(i, \theta_i, \phi_i, \hat{W}_{t-1}^{l_i})$
  end for
  $\{\hat{W}_t^{l_i}\}_{i=1}^{\mathcal{K}}, \mathcal{V} \leftarrow \text{Aggregate}(\{W_t^{l_i}\}_{i=1}^{\mathcal{K}}, \mathcal{V})$
end for

Algorithm 3 ClientUpdate

Input: $i, \theta_i, \phi_i, \hat{W}_{t-1}^{l_i}$
// 1. Local training with STKD
Replace W_i^l by $\hat{W}_{t-1}^{l_i}$
for each local epoch **do**
  $\phi_i \leftarrow \phi_i - \eta \nabla_{\phi_i} \mathcal{L}_{\text{task}}(\phi_i)$
  $\mathcal{L} \leftarrow \mathcal{L}_{\text{task}}(\theta_i) + \mathcal{L}_{\text{STKD}}(\theta_i, \phi_i)$
  $\theta_i \leftarrow \theta_i - \eta \nabla_{\theta_i} \mathcal{L}$
end for
// 2. Stable layer detection {The latest (max(l)) layer}
$l_i \leftarrow max\{l \mid \text{Layer } l \text{ satisfies Eq. (5) with threshold } \epsilon\}$
// 3. Upload stable layers
if $l_i \neq \emptyset$ **then**
  Upload $\{W_t^{l_i} \mid l \in \mathcal{S}_i\}$ to server
else
  {Use frequency-domain repository for guidance}
  Download $\mathcal{R}(l)$ for layer l
end if

B Theoretical analysis

B.1 Theoretical Advantage of STKD over KD

Assumption B.1 (Uncertainty-Divergence Coupling)**.** There exists a non-decreasing function $h : [0, 1] \to [0, 1]$ with $h(0) = 0$ and $h(\delta) > 0$ for all $\delta > 0$, such that for any client distribution $\mathcal{D}_i$,

$$\Pr_{x \sim \mathcal{D}_i} \big[R_t(x) \geq R_0 \big] \;\geq\; h\big(D_{\text{TV}}(\mathcal{D}_i \,\|\, \mathcal{D}_{\text{global}}) \big), \tag{13}$$

where R_0 is the entropy threshold above which the adaptive temperature becomes beneficial. A simple special case is $h(\delta) = c\,\delta$, corresponding to a linear coupling between distribution shift and teacher uncertainty.

Theorem B.2 (Superiority of STKD over KD)**.** *Under Assumption B.1, for any client distribution $\mathcal{D}_i$ with statistical divergence $d_i = D_{\text{TV}}(\mathcal{D}_i \,\|\, \mathcal{D}_{\text{global}}) > 0$, where $\mathcal{D}_{\text{global}}$ is the global distribution that the teacher ϕ is optimized for, STKD achieves*

$$\mathbb{E}\big[\|\nabla L_{\text{STKD}}\|^2\big] \;\leq\; \mathbb{E}\big[\|\nabla L_{\text{KD}}\|^2\big] - \frac{\beta^2 d_i^2 G_{\text{KD}}^2}{4} + O\!\left(\frac{1}{\tau_{\min}^2}\right), \tag{14}$$

where $\tau_{\min} = \min_x e^{\beta R_t(x)}$ is the minimum effective temperature and G_{KD} bounds the gradient of the KD loss. Thus STKD reduces the gradient variance by at least $\Omega(\beta^2 d_i^2)$ compared to KD.

Theorem B.2 implies that STKD reduces gradient variance by at least $\Omega(\beta^2 \delta^2)$ compared to KD.

B.1.1 Problem Formulation

Consider the distillation loss for a single sample (x, y). For KD with fixed temperature τ:

$$\mathcal{L}_{\text{KD}}(\theta; \phi, \tau) = \text{KL}\left(\text{softmax}(f(x;\theta)/\tau) \parallel \text{softmax}(g(x;\phi)/\tau)\right),$$

where f is the student (local heterogeneous model), g is the teacher (proxy model).

For STKD with adaptive temperature $T = e^{\beta R_t}$, where $R_t = -\log \sum_c P_t(x)^2$ is the teacher's Rényi-2 entropy:

$$\mathcal{L}_{\text{STKD}}(\theta; \phi, \beta) = \text{KL}\left(\text{softmax}(f(x;\theta)/T) \parallel \text{softmax}(g(x;\phi)/T)\right).$$

B.1.2 Key Theoretical Result

Theorem B.3 (Superiority of STKD over KD)**.** *Under heterogeneous data distribution with statistical divergence $D_{TV}(\mathcal{D}_i \parallel \mathcal{D}_{global}) \geq \delta > 0$, where $\mathcal{D}_i$ is client i's distribution and $\mathcal{D}_{global}$ is the global distribution that the teacher ϕ is optimized for, STKD achieves:*

$$\mathbb{E}[\|\nabla \mathcal{L}_{STKD}\|^2] \leq \mathbb{E}[\|\nabla \mathcal{L}_{KD}\|^2] - \frac{\beta^2 \delta^2 G_{KD}^2}{4} + O\left(\frac{1}{\tau_{\min}^2}\right),$$

where:

- $\tau_{\min} = \min_x e^{\beta R_t(x)}$ *is the minimum effective temperature*
- G_{KD} *bounds the gradient of KD loss*
- *The expectation is taken over client data distributions*

This implies that STKD reduces gradient variance by at least $\Omega(\beta^2\delta^2)$ compared to KD.

Proof. We prove the theorem in three steps:

Step 1: Gradient Analysis of KD Loss

The gradient of KD loss with respect to student parameters θ is:

$$\nabla_\theta \mathcal{L}_{\text{KD}} = \tau \left(P_s^\tau - P_t^\tau\right)^\top \nabla_\theta f(x;\theta),$$

where $P_s^\tau = \text{softmax}(f(x;\theta)/\tau)$, $P_t^\tau = \text{softmax}(g(x;\phi)/\tau)$.

The squared gradient norm is:

$$\|\nabla \mathcal{L}_{\text{KD}}\|^2 = \tau^2 \left\|(P_s^\tau - P_t^\tau)^\top \nabla_\theta f\right\|^2 \leq \tau^2 \|P_s^\tau - P_t^\tau\|^2 \cdot \|\nabla_\theta f\|^2.$$

Step 2: Temperature Effect on Probability Differences

Let $R_t = -\log \sum_c P_t(x)^2$ measure teacher uncertainty. For out-of-distribution samples where teacher is uncertain (R_t large), we have:

$$\begin{aligned}\|P_s^{\mathcal{T}} - P_t^{\mathcal{T}}\|^2 &\leq \|P_s^\tau - P_t^\tau\|^2 \cdot \tau^2 \cdot \exp\left(-\frac{2(R_t - \log\tau/\beta)}{\beta}\right) \\ &\leq \|P_s^\tau - P_t^\tau\|^2 \cdot \exp(-2\beta R_t + 2\log\tau).\end{aligned}$$

The key insight is that when $R_t > \frac{\log\tau}{\beta}$ (teacher more uncertain than fixed τ implies), we have:

$$\|P_s^{\mathcal{T}} - P_t^{\mathcal{T}}\|^2 \leq e^{-2\beta R_t + 2\log\tau} \|P_s^\tau - P_t^\tau\|^2 < \|P_s^\tau - P_t^\tau\|^2.$$

Step 3: Statistical divergence effect.

By Assumption B.1, the probability that the teacher is uncertain on client i's data is bounded from below by $h(d_i) > 0$:

$$\Pr_{x \sim \mathcal{D}_i}\left[R_t(x) \geq R_0\right] \;\geq\; h(d_i), \tag{15}$$

where $d_i = D_{\mathrm{TV}}(\mathcal{D}_i \,\|\, \mathcal{D}_{\mathrm{global}})$ and R_0 is the entropy threshold above which the adaptive temperature becomes beneficial.

Taking expectation over the data distribution:

$$\begin{aligned}\mathbb{E}\big[\|\nabla L_{\mathrm{STKD}}\|^2\big] &= \mathbb{E}\big[\|\nabla L_{\mathrm{STKD}}\|^2 \mid R_t < R_0\big] \cdot \big(1 - h(d_i)\big) + \mathbb{E}\big[\|\nabla L_{\mathrm{STKD}}\|^2 \mid R_t \geq R_0\big] \cdot h(d_i) \\ &\leq \mathbb{E}\big[\|\nabla L_{\mathrm{KD}}\|^2\big] \cdot \big(1 - h(d_i)\big) + \mathbb{E}\big[\|\nabla L_{\mathrm{KD}}\|^2\big] \cdot e^{-2\beta R_0} \cdot h(d_i) \\ &= \mathbb{E}\big[\|\nabla L_{\mathrm{KD}}\|^2\big]\Big(1 - h(d_i)\big(1 - e^{-2\beta R_0}\big)\Big).\end{aligned} \tag{16}$$

Using $1 - e^{-2\beta R_0} \geq \frac{\beta R_0}{2}$ for $\beta R_0 \leq 1$ and noting that $R_0 \propto d_i$ (uncertainty correlates with distribution shift), we obtain

$$\mathbb{E}\big[\|\nabla L_{\mathrm{STKD}}\|^2\big] \;\leq\; \mathbb{E}\big[\|\nabla L_{\mathrm{KD}}\|^2\big] - \frac{\beta^2 d_i^2}{4}\,\mathbb{E}\big[\|\nabla L_{\mathrm{KD}}\|^2\big] + O\!\left(\frac{1}{\tau_{\min}^2}\right). \tag{17}$$

Since $\mathbb{E}\big[\|\nabla L_{\mathrm{KD}}\|^2\big] \geq G_{\mathrm{KD}}^2$ (bounded away from zero), the final result follows. □

B.1.3 Implications for Federated Learning Convergence

Theorem B.4 (Convergence Improvement)**.** *The improvement in gradient variance translates directly to better convergence. Let Δ_{STKD} and Δ_{KD} be the distillation misalignment terms in Theorem 3.2 for STKD and KD respectively. Then:*

$$\Delta_{STKD} \leq \Delta_{KD} - \frac{\beta^2 \delta^2 G_{KD}^2}{4}.$$

Substituting into Theorem 3.2, STKD improves the convergence bound by:

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[\Phi_t]_{STKD} \leq \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[\Phi_t]_{KD} - \frac{\lambda_{KD}\beta^2 \delta^2 G_{KD}^2}{4}.$$

*This quantifies the **provable advantage** of STKD over KD in heterogeneous FL.*

B.2 Proof of Layer-wise Stabilization

B.2.1 Linear Network

Proof. Proof of assumption $S_l(t) \leq S_{l+1}(t)$

Consider a linear network (no activation functions) with mean squared error loss:

$$\mathcal{L} = \frac{1}{2}\|y - W^L W^{L-1} \cdots W^1 x\|^2.$$

The gradient with respect to W^l is:

$$\frac{\partial \mathcal{L}}{\partial W^l} = \left(\prod_{k=l+1}^{L} W^k\right)^{\top} (y - \hat{y}) x^{\top} \left(\prod_{k=1}^{l-1} W^k\right)^{\top}.$$

Taking spectral norms (ignoring transpose invariance) gives:

$$\left\|\frac{\partial \mathcal{L}}{\partial W^l}\right\| \leq \left(\prod_{k=l+1}^{L} \|W^k\|\right) \cdot \|y - \hat{y}\| \cdot \|x\| \cdot \left(\prod_{k=1}^{l-1} \|W^k\|\right).$$

Denote $C = \|y - \hat{y}\| \cdot \|x\|$, then:

$$\left\|\frac{\partial \mathcal{L}}{\partial W^l}\right\| \leq C \cdot \prod_{k \neq l} \|W^k\|.$$

Hence:

$$S_l(t) = \frac{\|\partial \mathcal{L}/\partial W^l\|}{\|W^l\| + \xi} \leq \frac{C \cdot \prod_{k \neq l} \|W^k\|}{\|W^l\| + \xi}.$$

Now assume that the weight norms are equal across layers: $\|W^l\| = \alpha$ for all l (achievable via weight normalization). Then:

$$S_l(t) \leq \frac{C \cdot \alpha^{L-1}}{\alpha + \xi},$$

which is independent of l. Thus $S_l(t) = S_{l+1}(t)$, so $S_l(t) \leq S_{l+1}(t)$ holds (with equality).

More generally, if $\|W^{l+1}\| \geq \|W^l\|$ (norms non-decreasing with depth), a similar calculation yields $S_l(t) \leq S_{l+1}(t)$ strictly. □

Proof. From the assumption to $T_1(\epsilon) \leq \cdots \leq T_L(\epsilon)$ Fix $\epsilon > 0$ and any $l \in \{1, \ldots, L-1\}$. Let $t^* = T_{l+1}(\epsilon) = \inf\{t \geq 0 : S_{l+1}(t) \leq \epsilon\}$.

By definition of the infimum:

- If the infimum is attained, then $S_{l+1}(t^*) \leq \epsilon$.
- If not attained, there exists a sequence $t_n \to t^*$ such that $S_{l+1}(t_n) \leq \epsilon$ for all n.

Case 1 (attained): From the assumption $S_l(t) \leq S_{l+1}(t)$ for all t, we have

$$S_l(t^*) \leq S_{l+1}(t^*) \leq \epsilon.$$

Thus $t^* \in \{t \geq 0 : S_l(t) \leq \epsilon\}$, so its infimum satisfies $T_l(\epsilon) \leq t^*$.

Case 2 (not attained): For each t_n, using the assumption:

$$S_l(t_n) \leq S_{l+1}(t_n) \leq \epsilon.$$

Hence $t_n \in \{t \geq 0 : S_l(t) \leq \epsilon\}$, so $T_l(\epsilon) \leq t_n$ for every n. Taking the limit $n \to \infty$ gives $T_l(\epsilon) \leq t^*$.

In both cases, $T_l(\epsilon) \leq T_{l+1}(\epsilon)$. Since l was arbitrary, $T_1(\epsilon) \leq T_2(\epsilon) \leq \cdots \leq T_L(\epsilon)$. □

B.2.2 Nonlinear Network

In nonlinear networks (e.g., with ReLU, tanh, or sigmoid activations), although a rigorous analytical proof of $S_l(t) \leq S_{l+1}(t)$ for all training iterations t is not available as in the linear case, extensive empirical studies reveal that under appropriate initialization schemes (such as Xavier or He initialization) and with the use of normalization techniques like Batch Normalization or Layer Normalization, a similar property is consistently observed during training [10]: the ratio $S_l(t)$ for shallower layers (closer to the input) is typically less than or equal to that for deeper layers (closer to the output), i.e., $S_l(t) \leq S_{l+1}(t)$. This ordering remains stable throughout most of the training process. This empirical observation provides the practical basis for the phenomenon that shallower layers stabilize earlier than deeper layers (i.e., $T_l(\epsilon) \leq T_{l+1}(\epsilon)$), offering valuable insights into the training dynamics of deep neural networks, even though a full theoretical justification would require additional assumptions.

B.3 Convergence analysis of FedPeel

In this section, we provide the convergence analysis of FedPeel under standard assumptions in federated learning. Our analysis demonstrates that FedPeel converges to a stationary point of the personalized objective function, with convergence rate depending on the stability of local layers and the effectiveness of knowledge distillation.

B.3.1 Assumptions

We make the following standard assumptions:

Assumption B.5 (Smoothness)**.** The local loss function $F_i(\theta)$ is M-smooth for all clients $i \in [K]$:

$$\|\nabla F_i(\theta) - \nabla F_i(\theta')\| \leq M\|\theta - \theta'\|, \quad \forall \theta, \theta' \in \mathbb{R}^d.$$

Assumption B.6 (Bounded Variance)**.** The stochastic gradient has bounded variance:

$$\mathbb{E}[\|\nabla F_i(\theta;\xi) - \nabla F_i(\theta)\|^2] \leq \sigma^2, \quad \forall \theta \in \mathbb{R}^d, i \in [K],$$

where ξ denotes a mini-batch sample.

Assumption B.7 (Bounded Gradient)**.** The gradient is bounded:

$$\|\nabla F_i(\theta)\| \leq G, \quad \forall \theta \in \mathbb{R}^d, i \in [K].$$

Assumption B.8 (Layer-wise Stabilization)**.** The layer-wise stabilization phenomenon (Theorem 3.1) implies that for any $\epsilon > 0$, there exists a stabilization time $T_l(\epsilon)$ such that for $t \geq T_l(\epsilon)$, layer l satisfies the stability condition in Eq. (7). This ensures that only stable layers participate in aggregation.

B.3.2 Convergence Guarantee

Let $\theta^t = \{\theta_i^t\}_{i=1}^K$ denote the collection of all client models at communication round t, and define the average gradient norm as a measure of convergence:

$$\Phi_t = \frac{1}{K}\sum_{i=1}^{K} \|\nabla F_i(\theta_i^t)\|^2.$$

The following theorem establishes the convergence of FedPeel:

Theorem B.9 (Convergence of FedPeel)**.** *Under Assumptions B.4–B.7, with learning rate $\eta \leq \frac{1}{M}$, E local epochs per round, and after T communication rounds, FedPeel satisfies:*

$$\frac{1}{T}\sum_{t=1}^{T}\Phi_t \leq \frac{2(F(\bar{\theta}^1) - F^*)}{\eta ET} + \eta M\sigma^2 + \frac{2}{T}\sum_{t=1}^{T}\mathbb{E}\left[\left\langle \nabla F(\bar{\theta}^t), \frac{1}{K}\sum_{i=1}^{K}\frac{1}{E}\sum_{s=0}^{E-1}\nabla \mathcal{L}_{\mathrm{STKD}}(\theta_i^{t,s}, \phi^t)\right\rangle\right] \tag{18}$$

$$+\frac{2M}{ET}\sum_{t=1}^{T}\mathbb{E}\left[\|\hat{\theta}^{t+1} - \bar{\theta}^{t+1}\|^2\right],$$

where $\Phi_t = \frac{1}{K}\sum_{i=1}^{K}\mathbb{E}[\|\nabla F_i(\theta_i^t)\|^2]$.

The key differences from the original bound are:

- *The smoothness constant is denoted M instead of L to avoid confusion with the number of layers.*
- *The initial descent term now scales as $\frac{1}{\eta ET}$ (instead of $\frac{1}{\eta T}$), correctly reflecting that E local steps reduce the required number of communication rounds.*
- *The variance term $\eta M\sigma^2$ remains unchanged because the $\frac{E\sigma^2}{K}$ in Lemma C.3 combines with $\frac{\eta^2 ME}{2}$ to give $\eta M\sigma^2$ after telescoping.*
- *The aggregation error term scales with $\frac{2M}{ET}$, acknowledging that more local steps dilute the per-round impact of aggregation errors.*

Proof Sketch. We provide a proof sketch highlighting the key steps:

1. **Local Descent**: For each client i, after local training:

$$F_i(\theta_i^{t+1}) \leq F_i(\theta_i^t) - \eta \|\nabla F_i(\theta_i^t)\|^2 + \frac{\eta^2 M}{2}\sigma^2 + \eta\langle \nabla F_i(\theta_i^t), \nabla \mathcal{L}_{\text{STKD}}(\theta_i^t, \phi^t)\rangle.$$

2. **Stable Layer Aggregation**: Let $\mathcal{S}(t) = \bigcup_{i=1}^K \mathcal{S}_i(t)$ be the set of all stable layers at round t. The aggregation error for unstable layers is bounded by:

$$\|\theta_i^{t+1} - \bar{\theta}^{t+1}\|^2 \leq \sum_{l \notin \mathcal{S}(t)} \|\theta_i^{l,t} - \bar{\theta}^{l,t}\|^2 \leq 4G^2 \sum_{l \notin \mathcal{S}(t)} 1,$$

where $\bar{\theta}^{t+1}$ is the model obtained by perfect aggregation of all layers.

3. **Spectral Decoupling Effect**: The adaptive frequency segmentation ensures that the aggregation of low-frequency components reduces the variance while preserving personalized information. The reconstruction error is bounded by the energy threshold γ:

$$\mathbb{E}[\|\text{IDCT}(S_{\text{agg},i}^l) - W_i^l\|^2] \leq (1-\gamma)^2 \|W_i^l\|^2.$$

4. **Telescoping Sum**: Summing over all rounds and clients, and applying the Cauchy-Schwarz inequality to the distillation term, we obtain the final bound.

The complete proof is provided in Appendix C. □

C Complete Convergence Proof

In this section, we provide the complete convergence proof for Theorem B.9. We begin by introducing additional notation and technical lemmas.

C.1 Notation and Preliminary Results

Let $\theta_i^{t,s}$ denote the model parameters of client i at communication round t and local iteration s, with $\theta_i^{t,0} = \theta_i^t$ and $\theta_i^{t,E} = \theta_i^{t+1}$ after E local epochs.

The local update rule with STKD is:

$$\theta_i^{t,s+1} = \theta_i^{t,s} - \eta \left(\nabla F_i(\theta_i^{t,s}; \xi_i^{t,s}) + \nabla \mathcal{L}_{\text{STKD}}(\theta_i^{t,s}, \phi^t) \right),$$

where $\xi_i^{t,s}$ is a mini-batch sample.

Define the *virtual sequence* $\bar{\theta}^t$ as the model that would be obtained by perfect aggregation of all layers:

$$\bar{\theta}^t = \frac{1}{K}\sum_{i=1}^K \theta_i^t.$$

However, in FedPeel, only stable layers are aggregated. Let $\mathcal{S}(t) = \bigcup_{i=1}^K \mathcal{S}_i(t)$ be the set of stable layers at round t, and define:

$$\hat{\theta}^{t+1} = \frac{1}{K}\sum_{i=1}^K \text{Aggregate}(\theta_i^t, \mathcal{S}(t)),$$

where Aggregate$(\cdot)$ performs spectral aggregation on stable layers and keeps unstable layers unchanged.

C.2 Technical Lemmas

Lemma C.1 (Accumulated Client Drift for Unstable Layers)**.** *For any client i and any layer l that does* not *participate in aggregation for t consecutive communication rounds (i.e., $l \notin \bigcup_{\tau=1}^t S(\tau)$), the parameter drift satisfies the following recurrence:*

$$\|\theta_i^{l,t+1} - \bar{\theta}^{l,1}\|^2 \leq 2\|\theta_i^{l,t} - \bar{\theta}^{l,1}\|^2 + 2\eta^2 E^2 G^2,$$

where $\bar{\theta}^{l,1}$ is the global initialization of layer l (or any fixed reference point), G is the gradient bound from Assumption B.6, η the learning rate, and E the number of local epochs.

Unfolding the recurrence yields:

$$\|\theta_i^{l,t+1} - \bar{\theta}^{l,1}\|^2 \leq 2^t \cdot \|\theta_i^{l,1} - \bar{\theta}^{l,1}\|^2 + 2\eta^2 E^2 G^2 (2^t - 1).$$

If we assume a more conservative linear drift model (e.g., via independent noise or bounded gradient direction variation), a linear bound also holds:

$$\|\theta_i^{l,t+1} - \bar{\theta}^{l,1}\|^2 \leq \|\theta_i^{l,1} - \bar{\theta}^{l,1}\|^2 + t \cdot \eta^2 E^2 G^2.$$

Consequently, the average drift over T rounds for a never-aggregated layer grows at least linearly in T:

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[\|\theta_i^{l,t} - \bar{\theta}^{l,t}\|^2\right] = O(T).$$

If the layer becomes stable at some round t_0 and starts being aggregated, the drift resets to $O(\eta^2 E^2 G^2)$ after a constant number of rounds due to averaging.

Proof. For an unstable layer l, the parameters are not aggregated and evolve independently through local SGD. From the update rule:

$$\theta_i^{l,t+1} = \theta_i^{l,t} - \eta \sum_{s=0}^{E-1} \left(\nabla F_i^l(\theta_i^{t,s}; \xi_i^{t,s}) + \nabla \mathcal{L}_{\text{STKD}}^l(\theta_i^{t,s}, \phi^t)\right),$$

where ∇F_i^l denotes the gradient.

By triangle inequality and Assumption 3:

$$\begin{aligned}\|\theta_i^{l,t+1} - \theta_j^{l,t+1}\| &\leq \|\theta_i^{l,t} - \theta_j^{l,t}\| + \eta \sum_{s=0}^{E-1} \left(\|\nabla F_i^l\| + \|\nabla F_j^l\| + (\|\nabla \mathcal{L}_{\text{STKD},i}^l\| + \|\nabla \mathcal{L}_{\text{STKD},j}^l\|)\right) \\ &\leq \|\theta_i^{l,t} - \theta_j^{l,t}\| + 2\eta E(G + G_{\text{KD}}),\end{aligned}$$

where G_{KD} is the bound on distillation gradient (which exists due to bounded teacher outputs). Without loss of generality, we assume $G \geq G_{\text{KD}}$ and absorb constants into G, yielding the bound $2G\eta E$. □

Lemma C.2 (Spectral Reconstruction Error)**.** *The spectral aggregation and reconstruction error for a stable layer $l \in \mathcal{S}(t)$ is bounded by:*

$$\mathbb{E}[\|\hat{\theta}^{l,t+1} - \bar{\theta}^{l,t+1}\|^2] \leq (1-\gamma)^2 \mathbb{E}[\|\bar{\theta}^{l,t+1}\|^2] + \frac{\sigma_l^2}{|\mathcal{C}_l|},$$

where σ_l^2 is the variance of layer parameters across clients, and $\mathcal{C}_l = \{i : l \in \mathcal{S}_i(t)\}$.

Proof. For a stable layer l, FedPeel performs:

1. DCT transform: $S_i^l = \text{DCT}(W_i^l)$
2. Adaptive split at index $\mathcal{R}(l)$ from Eq. (11)
3. Aggregate low-frequency: $\bar{S}_{\text{low}}^l = \frac{1}{|\mathcal{C}_l|}\sum_{i \in \mathcal{C}_l} S_i^l[1 : \mathcal{R}(l)]$
4. Reconstruct: $\hat{W}_i^l = \text{IDCT}(\text{Concat}(\bar{S}_{\text{low}}^l, S_i^l[\mathcal{R}(l)+1 :]))$

The ideal aggregation would be $\bar{W}^l = \frac{1}{|\mathcal{C}_l|}\sum_{i\in\mathcal{C}_l} W_i^l$. The error comes from:

1. **High-frequency truncation**: By the energy threshold γ, we have:

$$\frac{\sum_{k=1}^{\mathcal{R}(l)} F_l(k)^2}{\sum_{k=1}^{N} F_l(k)^2} \geq \gamma \Rightarrow \frac{\sum_{k=\mathcal{R}(l)+1}^{N} F_l(k)^2}{\sum_{k=1}^{N} F_l(k)^2} \leq 1-\gamma.$$

   By Parseval's theorem, this implies:

$$\|\hat{W}_i^l - \text{IDCT}(S_i^l)\|^2 \leq (1-\gamma)^2 \|W_i^l\|^2.$$

2. **Sample variance**: The aggregation of $|\mathcal{C}_l|$ clients reduces variance:

$$\mathbb{E}[\|\bar{S}_{\text{low}}^l - \mathbb{E}[S_{\text{low}}^l]\|^2] = \frac{\sigma_l^2}{|\mathcal{C}_l|}.$$

Combining these effects and applying Jensen's inequality gives the stated bound. □

Lemma C.3 (E-round Progress)**.** *Under Assumptions B.4–B.6, after one communication round of FedPeel with E local epochs per round and learning rate $\eta \leq \frac{1}{M}$, we have:*

$$\begin{aligned}\mathbb{E}[F(\bar{\theta}^{t+1})] \leq \mathbb{E}[F(\bar{\theta}^t)] &- \eta E \cdot \mathbb{E}\left[\|\nabla F(\bar{\theta}^t)\|^2\right] \\ &+ \frac{\eta^2 M E}{2}\left(\sigma^2 + \frac{1}{K}\sum_{i=1}^{K}\mathbb{E}\left[\left\|\frac{1}{E}\sum_{s=0}^{E-1}\nabla\mathcal{L}_{\text{STKD}}(\theta_i^{t,s},\phi^t)\right\|^2\right]\right) \\ &+ \eta E \cdot \mathbb{E}\left[\left\langle \nabla F(\bar{\theta}^t), \frac{1}{K}\sum_{i=1}^{K}\frac{1}{E}\sum_{s=0}^{E-1}\nabla\mathcal{L}_{\text{STKD}}(\theta_i^{t,s},\phi^t)\right\rangle\right] \\ &+ M \cdot \mathbb{E}\left[\|\hat{\theta}^{t+1} - \bar{\theta}^{t+1}\|^2\right]. \end{aligned} \tag{19}$$

where σ^2 is the stochastic gradient variance bound from Assumption B.5.

Proof. By M-smoothness (Assumption B.4):

$$F(\bar{\theta}^{t+1}) \leq F(\bar{\theta}^t) + \langle \nabla F(\bar{\theta}^t), \bar{\theta}^{t+1} - \bar{\theta}^t\rangle + \frac{M}{2}\|\bar{\theta}^{t+1} - \bar{\theta}^t\|^2. \tag{20}$$

The update rule gives:

$$\bar{\theta}^{t+1} - \bar{\theta}^t = -\frac{\eta}{K}\sum_{i=1}^{K}\sum_{s=0}^{E-1}\left(\nabla F_i(\theta_i^{t,s};\xi_i^{t,s}) + \nabla\mathcal{L}_{\text{STKD}}(\theta_i^{t,s},\phi^t)\right).$$

Taking expectation and using $\mathbb{E}[\nabla F_i(\theta_i^{t,s};\xi_i^{t,s})] = \nabla F_i(\theta_i^{t,s})$:

$$\begin{aligned}\mathbb{E}[\langle \nabla F(\bar{\theta}^t), \bar{\theta}^{t+1} - \bar{\theta}^t\rangle] = &-\eta\mathbb{E}\left[\left\langle \nabla F(\bar{\theta}^t), \frac{1}{K}\sum_{i=1}^{K}\sum_{s=0}^{E-1}\nabla F_i(\theta_i^{t,s})\right\rangle\right] \\ &-\eta\mathbb{E}\left[\left\langle \nabla F(\bar{\theta}^t), \frac{1}{K}\sum_{i=1}^{K}\sum_{s=0}^{E-1}\nabla\mathcal{L}_{\text{STKD}}(\theta_i^{t,s},\phi^t)\right\rangle\right]. \end{aligned} \tag{21}$$

For the first term, decompose $\nabla F_i(\theta_i^{t,s}) = \nabla F_i(\bar{\theta}^t) + (\nabla F_i(\theta_i^{t,s}) - \nabla F_i(\bar{\theta}^t))$:

$$\mathbb{E}\left[\left\langle \nabla F(\bar{\theta}^t), \frac{1}{K}\sum_{i=1}^{K}\sum_{s=0}^{E-1}\nabla F_i(\theta_i^{t,s})\right\rangle\right] = E\cdot\mathbb{E}\left[\|\nabla F(\bar{\theta}^t)\|^2\right] + \Delta_{\text{drift}}, \tag{22}$$

where Δ_{drift} is bounded by $\frac{E}{2}\|\nabla F(\bar{\theta}^t)\|^2 + \frac{M^2}{2E}\sum_{s=0}^{E-1}\|\theta_i^{t,s} - \bar{\theta}^t\|^2$ via Young's inequality and M-smoothness.

For the variance bound in $\mathbb{E}[\|\bar{\theta}^{t+1} - \bar{\theta}^t\|^2]$:

$$\begin{aligned}\mathbb{E}\left[\left\|\frac{1}{K}\sum_{i=1}^{K}\sum_{s=0}^{E-1}\nabla F_i(\theta_i^{t,s};\xi_i^{t,s})\right\|^2\right] &= \mathbb{E}\left[\left\|\frac{1}{K}\sum_{i=1}^{K}\sum_{s=0}^{E-1}\nabla F_i(\theta_i^{t,s})\right\|^2\right] \\ &+\frac{1}{K^2}\sum_{i=1}^{K}\sum_{s=0}^{E-1}\mathbb{E}\left[\left\|\nabla F_i(\theta_i^{t,s};\xi_i^{t,s}) - \nabla F_i(\theta_i^{t,s})\right\|^2\right] \\ &\le \mathbb{E}\left[\left\|\frac{1}{K}\sum_{i=1}^{K}\sum_{s=0}^{E-1}\nabla F_i(\theta_i^{t,s})\right\|^2\right] + \frac{E\sigma^2}{K}.\end{aligned} \tag{23}$$

The $\frac{E\sigma^2}{K}$ term comes from: each of the K clients contributes E independent stochastic gradients, each with variance σ^2, and averaging over K clients yields $\frac{1}{K^2}\cdot K\cdot E\cdot\sigma^2 = \frac{E\sigma^2}{K}$.

Substituting (2), (3), (4) into (1) and simplifying yields the stated bound. The final term $M\cdot\mathbb{E}[\|\hat{\theta}^{t+1} - \bar{\theta}^{t+1}\|^2]$ arises from the discrepancy between the actual aggregation $\hat{\theta}^{t+1}$ and the ideal update $\bar{\theta}^{t+1}$, controlled via M-smoothness. □

C.3 Proof of Theorem B.9

Now we prove the main theorem by combining the lemmas.

Proof of Theorem B.9. From Lemma C.3, summing over $t = 1$ to T:

$$\begin{aligned}\sum_{t=1}^{T}\mathbb{E}[F(\bar{\theta}^{t+1})] &\le \sum_{t=1}^{T}\mathbb{E}[F(\bar{\theta}^t)] - \eta\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla F(\bar{\theta}^t)\|^2\right] + \frac{\eta^2 MT}{2}\sigma^2 \\ &+ \eta\sum_{t=1}^{T}\mathbb{E}\left[\langle\nabla F(\bar{\theta}^t), \frac{1}{K}\sum_{i=1}^{K}\nabla\mathcal{L}_{\text{STKD}}(\theta_i^t,\phi^t)\rangle\right] \\ &+ M\sum_{t=1}^{T}\mathbb{E}\left[\|\hat{\theta}^{t+1} - \bar{\theta}^{t+1}\|^2\right].\end{aligned}$$

Rearranging and dividing by T:

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla F(\bar{\theta}^t)\|^2\right] &\le \frac{2(F(\bar{\theta}^1) - \mathbb{E}[F(\bar{\theta}^{T+1})])}{\eta T} + \eta M\sigma^2 \\ &+ \frac{2}{T}\sum_{t=1}^{T}\mathbb{E}\left[\langle\nabla F(\bar{\theta}^t), \frac{1}{K}\sum_{i=1}^{K}\nabla\mathcal{L}_{\text{STKD}}(\theta_i^t,\phi^t)\rangle\right] \\ &+ \frac{2M}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\hat{\theta}^{t+1} - \bar{\theta}^{t+1}\|^2\right].\end{aligned}$$

Now bound each term:

Term 1 (Distillation Alignment): By Cauchy-Schwarz inequality:

$$\langle \nabla F(\bar{\theta}^t), \frac{1}{K}\sum_{i=1}^{K} \nabla \mathcal{L}_{\text{STKD}}(\theta_i^t, \phi^t)\rangle \leq \|\nabla F(\bar{\theta}^t)\| \cdot \left\| \frac{1}{K}\sum_{i=1}^{K} \nabla \mathcal{L}_{\text{STKD}}(\theta_i^t, \phi^t) \right\|.$$

Using Young's inequality $ab \leq \frac{a^2}{2} + \frac{b^2}{2}$:

$$\frac{2}{T}\sum_{t=1}^{T} \mathbb{E}\left[\langle \nabla F(\bar{\theta}^t), \frac{1}{K}\sum_{i=1}^{K} \nabla \mathcal{L}_{\text{STKD}}(\theta_i^t, \phi^t)\rangle\right] \leq \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[\|\nabla F(\bar{\theta}^t)\|^2 + \left\| \frac{1}{K}\sum_{i=1}^{K} \nabla \mathcal{L}_{\text{STKD}}(\theta_i^t, \phi^t) \right\|^2\right].$$

Term 2 (Aggregation Error): From Lemma C.2 for stable layers and Lemma C.1 for unstable layers:

$$\begin{aligned}\mathbb{E}\left[\|\hat{\theta}^{t+1} - \bar{\theta}^{t+1}\|^2\right] &= \sum_{l=1}^{L} \mathbb{E}\left[\|\hat{\theta}^{l,t+1} - \bar{\theta}^{l,t+1}\|^2\right] \\ &\leq \sum_{l\in\mathcal{S}(t)} \left((1-\gamma)^2 \mathbb{E}[\|\bar{\theta}^{l,t+1}\|^2] + \frac{\sigma_l^2}{|\mathcal{C}_l|}\right) + \sum_{l\notin\mathcal{S}(t)} 4G^2\eta^2E^2.\end{aligned}$$

By Assumption 3, $\|\bar{\theta}^{l,t+1}\|^2 \leq G^2$. Also, $|\mathcal{C}_l| \geq 1$ for $l \in \mathcal{S}(t)$. Let $\sigma_{\max}^2 = \max_l \sigma_l^2$. Then:

$$\mathbb{E}\left[\|\hat{\theta}^{t+1} - \bar{\theta}^{t+1}\|^2\right] \leq |\mathcal{S}(t)|(1-\gamma)^2G^2 + |\mathcal{S}(t)|\sigma_{\max}^2 + 4G^2\eta^2E^2|\mathcal{U}(t)|,$$

where $\mathcal{U}(t) = \{1, \ldots, L\} \setminus \mathcal{S}(t)$ is the set of unstable layers.

Let $\rho_l(T) = \frac{|\{t\leq T : l\notin\mathcal{S}(t)\}|}{T}$ be the fraction of rounds where layer l is unstable. Then:

$$\frac{1}{T}\sum_{t=1}^{T} |\mathcal{U}(t)| = \sum_{l=1}^{L} \rho_l(T).$$

Also, $|\mathcal{S}(t)| \leq L$ always. Combining:

$$\begin{aligned}\frac{2M}{T}\sum_{t=1}^{T} \mathbb{E}\left[\|\hat{\theta}^{t+1} - \bar{\theta}^{t+1}\|^2\right] &\leq 2M^2(1-\gamma)^2G^2 + 2M^2\sigma_{\max}^2 + 8MG^2\eta^2E^2\sum_{l=1}^{L}\rho_l(T) \\ &\leq C_1 + C_2\sum_{l=1}^{L}\rho_l(T),\end{aligned}$$

where $C_1 = 2L^2((1-\gamma)^2G^2 + \sigma_{\max}^2)$ and $C_2 = 8LG^2\eta^2E^2$.

Final Bound: Substituting back and using $\|\nabla F_i(\theta_i^t)\|^2 \leq 2\|\nabla F(\bar{\theta}^t)\|^2 + 2M^2\|\theta_i^t - \bar{\theta}^t\|^2$ (by M-smoothness):

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\frac{1}{K}\sum_{i=1}^{K} \mathbb{E}\left[\|\nabla F_i(\theta_i^t)\|^2\right] &\leq \frac{2}{T}\sum_{t=1}^{T} \mathbb{E}\left[\|\nabla F(\bar{\theta}^t)\|^2\right] + \frac{2M^2}{T}\sum_{t=1}^{T}\frac{1}{K}\sum_{i=1}^{K}\mathbb{E}\left[\|\theta_i^t - \bar{\theta}^t\|^2\right] \\ &\leq \frac{4(F(\bar{\theta}^1) - F^*)}{\eta T} + 2\eta M\sigma^2 + \Delta_{\text{STKD}} + C_1 + C_2\sum_{l=1}^{L}\rho_l(T),\end{aligned}$$

where $\Delta_{\text{STKD}} = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{K}\sum_{i=1}^{K}\|\nabla \mathcal{L}_{\text{STKD}}(\theta_i^t, \phi^t)\|^2$.

Absorbing constants and simplifying notation (setting C_1 into the constant term, which is small when $\gamma \approx 1$ and layers are stable) gives the bound in Theorem B.9.

The term $\frac{4G^2}{K}\sum_{l=1}^{L}\rho_l(T)$ in the theorem comes from:

$$\frac{2M^2}{T}\sum_{t=1}^{T}\frac{1}{K}\sum_{i=1}^{K}\mathbb{E}\left[\|\theta_i^t - \bar{\theta}^t\|^2\right] \leq \frac{4G^2}{K}\sum_{l=1}^{L}\rho_l(T) + O\left(\frac{1}{K}\right),$$

where we used Lemma C.1 and the fact that for stable layers, the discrepancy is small due to aggregation. □

C.4 Discussion of the Bound

The convergence bound reveals several important properties of FedPeel:

1. **Impact of Layer Stability**: The term $\sum_{l=1}^{L} \rho_l(T)$ decreases as training progresses, since Theorem 3.1 ensures that layers stabilize over time ($\rho_l(T) \to 0$ as $T \to \infty$). This explains the *acceleration* effect observed in experiments.

2. **Frequency-domain Aggregation Efficiency**: The constant C_1 contains $(1-\gamma)^2$, which is small when $\gamma = 90$ as in our experiments. This shows that the spectral approach introduces minimal error.

3. **Adaptive Distillation**: The Δ_{STKD} term is minimized by our temperature adaptation, which reduces gradient misalignment when the teacher is uncertain.

4. **Linear Speedup**: The dominant term $O(1/T)$ shows standard federated learning convergence, while the additional terms quantify the benefits of FedPeel's novel mechanisms.

C.5 Practical Convergence Behavior

In practice, FedPeel exhibits enhanced convergence compared to standard federated learning methods due to:

1. **Early Acceleration**: As shallow layers stabilize quickly (Theorem 3.1), they contribute to stable knowledge sharing early in training, reducing model oscillation.
2. **Adaptive Collaboration**: The spectral decoupling adapts to layer depth, allowing shallow layers (with domain-invariant features) to share more aggressively while deep layers retain client-specific knowledge.
3. **Robust Knowledge Transfer**: STKD's temperature adaptation prevents negative transfer when the proxy model is uncertain on client data, maintaining convergence stability under severe heterogeneity.

The convergence analysis demonstrates that FedPeel maintains theoretical convergence guarantees while enabling personalized adaptation through its novel mechanisms of stable layer aggregation, spectral knowledge decoupling, and adaptive knowledge distillation.

D Additional Experiments

D.1 Model settings

D.1.1 CNN Model for CV task

In terms of model architecture configuration, to simulate realistic model heterogeneity in federated learning, we employ multiple mainstream CNN architectures as client models for image classification tasks. These include the ResNet series (specifically ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152), GoogleNet (a simplified version adapted for the CIFAR-10/100 dataset with only three Inception modules instead of the full architecture), and MobileNet (also a simplified version adapted for CIFAR-10/100 to ensure compatibility with task complexity [34]). These client models exhibit substantial differences in parameter count, network depth, and architectural design. Meanwhile, all clients share a lightweight 5-layer CNN as the proxy model, which consists of three convolutional layers and two linear layers. The parameter size of this proxy model is considerably smaller than that of the heterogeneous client models.

D.1.2 Transformer Model for AGNews task

The hyperparameter configuration of the model used in the experiment (AGNews task) is as follows

D.1.3 DCT for convolutional layers.

We detail how the DCT is applied to convolutional weights. For a convolutional weight tensor $\mathbf{W}_l \in \mathbb{R}^{C_{\text{out}} \times C_{\text{in}} \times H \times W}$, the DCT is applied independently to each 2D kernel slice: for every output–input channel pair $(c_{\text{out}}, c_{\text{in}})$, the corresponding $H \times W$ kernel is transformed using a full 2D DCT

Table 7: Transformer model specifications ($d_{\text{model}} = 64$, layers$= 2$)

ID	Heads (h)	Vocab	Max len	Layers	Params
T-1 (proxy)	1	98 635	200	4	$\sim$6.33 M
T-2	2	98 635	200	6	$\sim$6.34 M
T-4	4	98 635	200	8	$\sim$6.36 M
T-8	8	98 635	200	2	$\sim$6.40 M
T-16	16	98 635	200	4	$\sim$6.48 M

Note: Parameter counts include embeddings, positional encoding, Transformer blocks and classification head, rounded to two decimals.

with orthogonal normalization. The resulting spectrum preserves the original 4D tensor shape, and its frequency coordinates follow the standard DCT layout, in which the top-left component is the lowest-frequency component and higher-frequency components are located farther from that corner. For fully connected layers, the $d_l \times d_{l-1}$ weight matrix is treated as a 2D signal and transformed in the same way.

D.2 Other Settings

All experiments are conducted using PyTorch 2.1.0 with CUDA 12.1 acceleration, running on NVIDIA RTX 4090 GPUs. The system runs on Ubuntu operating system, with additional low-level optimizations via NVIDIA drivers and cuDNN libraries, ensuring high efficiency and reproducibility of all experimental results. And the random seed is 1.

D.3 Additional Results

D.3.1 Final-round evaluation

As shown in Table 8, we additionally report the *final-round* test accuracy, averaged over three independent runs, instead of the maximum accuracy over rounds. This provides a more conservative and standard evaluation measure in federated learning.

Table 8: Final-round test accuracy (%) under different datasets in heterogeneous scenarios.

Dataset	Clients (K)	FedKD	FedTGP	FedMRL	FedSA	Moss	Ours
CIFAR-10	20	75.88 $\pm$1.19	68.23 $\pm$1.79	81.11 $\pm$0.98	82.55 $\pm$0.93	81.41 $\pm$1.09	**83.34** $\pm$0.89
	50	69.13 $\pm$1.76	66.21 $\pm$2.11	78.81 $\pm$1.23	78.87 $\pm$1.88	76.88 $\pm$1.76	**80.26** $\pm$1.07
	100	67.62 $\pm$1.77	67.88 $\pm$2.01	76.13 $\pm$1.67	76.79 $\pm$2.24	74.67 $\pm$2.06	**78.99** $\pm$1.58
CIFAR-100	20	41.62 $\pm$1.68	37.87 $\pm$2.03	40.63 $\pm$1.23	40.46 $\pm$1.06	40.64 $\pm$1.38	**45.43** $\pm$1.36
	50	31.04 $\pm$2.06	31.76 $\pm$1.93	36.01 $\pm$1.56	32.87 $\pm$2.16	35.42 $\pm$1.95	**43.52** $\pm$1.61
	100	29.33 $\pm$2.12	34.82 $\pm$2.35	35.52 $\pm$1.89	31.32 $\pm$1.47	34.91 $\pm$2.17	**40.36** $\pm$2.13
AGNEWS	20	88.36 $\pm$0.77	88.43 $\pm$0.55	88.68 $\pm$0.56	93.44 $\pm$0.52	88.21 $\pm$0.74	**93.85** $\pm$0.45
	50	91.97 $\pm$0.85	92.34 $\pm$0.87	92.47 $\pm$0.69	92.56 $\pm$0.87	87.32 $\pm$1.05	**93.78** $\pm$0.65
	100	91.08 $\pm$0.71	90.41 $\pm$1.12	91.61 $\pm$0.87	90.68 $\pm$1.35	85.37 $\pm$1.33	**91.69** $\pm$0.92
GSC	20	77.35 $\pm$1.19	86.53 $\pm$0.99	87.12 $\pm$0.92	85.76 $\pm$1.08	86.49 $\pm$0.82	**88.93** $\pm$0.88
	50	65.48 $\pm$1.63	80.86 $\pm$1.09	81.59 $\pm$1.08	81.39 $\pm$1.27	81.98 $\pm$1.36	**83.04** $\pm$1.13
	100	59.30 $\pm$1.93	70.12 $\pm$2.18	73.21 $\pm$1.55	71.94 $\pm$2.04	73.87 $\pm$2.11	**76.51** $\pm$1.66

D.3.2 Sensitivity analysis of the number of stable layers uploaded

Table 9: The impact of the number of stable layers on accuracy.

stable layer	Acc(%)
1	45.57
2	45.65
3	45.73

The experimental (Table 9) results indicate that increasing the number of stable layers uploaded will not result in a significant improvement in accuracy, but as the number of uploaded parameters increases, communication consumption will significantly increase. The reason why increasing the number of stable layers will not improve accuracy is that the multiple uploaded stable layers $l-1$ have already been aggregated $max(E, T_l(\epsilon) - T_{l-1}(\epsilon))$ times in the previous round. According to the theory of stable layers, these layers will not undergo significant changes in subsequent rounds and cannot provide effective knowledge.

D.3.3 Computational Cost Analysis

The computational cost of FedPeel is acceptable. On the client side, the additional overhead primarily stems from stability detection, which requires approximately 0.00078 GFLOPs—a negligible amount relative to the cost of model training. On the server side, the main computational burden comes from frequency-domain transformation, consuming at most 0.1 GFLOPs per communication round. This overhead is well within the capacity of typical server resources and does not introduce a significant bottleneck in the overall training process.

The local model training takes $\sim$0.09s, proxy training takes $\sim$0.03s, STKD computation takes $\sim$0.07s, entropy estimation and norm tracking each introduce only $\sim$0.001s overhead, and server-side frequency aggregation takes 0.01–0.07s. These measurements show that the additional costs introduced by stabilization detection, frequency decoupling, and STKD account for only a small fraction of the overall per-round computation, demonstrating that FedPeel achieves the desired improvements with limited extra overhead.

D.3.4 Additional Performance Evaluation

As shown in Table 10, we have revised our evaluation protocol to report the final-round test accuracy averaged over three independent runs, instead of the maximum accuracy over rounds. This provides a more conservative and standard evaluation measure in federated learning. To assess statistical reliability, we conducted paired two-sided t-tests for cases where the improvement over the strongest baseline was within one standard deviation. The results confirm that these improvements remain statistically significant ($p < 0.05$).

Table 10: Final-round test accuracy (%) under different datasets in heterogeneous scenarios.

Dataset	Clients (K)	FedKD	FedTGP	FedMRL	FedSA	Moss	Ours
CIFAR-10	20	75.88 $\pm$1.19	68.23 $\pm$1.79	81.11 $\pm$0.98	<u>82.55</u> $\pm$0.93	81.41 $\pm$1.09	**83.34** $\pm$0.89
	50	69.13 $\pm$1.76	66.21 $\pm$2.11	78.81 $\pm$1.23	<u>78.87</u> $\pm$1.88	76.88 $\pm$1.76	**80.26** $\pm$1.07
	100	67.62 $\pm$1.77	67.88 $\pm$2.01	76.13 $\pm$1.67	<u>76.79</u> $\pm$2.24	74.67 $\pm$2.06	**78.99** $\pm$1.58
CIFAR-100	20	<u>41.62</u> $\pm$1.68	37.87 $\pm$2.03	40.63 $\pm$1.23	40.46 $\pm$1.06	40.64 $\pm$1.38	**45.43** $\pm$1.36
	50	31.04 $\pm$2.06	31.76 $\pm$1.93	<u>36.01</u> $\pm$1.56	32.87 $\pm$2.16	35.42 $\pm$1.95	**43.52** $\pm$1.61
	100	29.33 $\pm$2.12	34.82 $\pm$2.35	<u>35.52</u> $\pm$1.89	31.32 $\pm$1.47	34.91 $\pm$2.17	**40.36** $\pm$2.13
AGNEWS	20	88.36 $\pm$0.77	88.43 $\pm$0.55	88.68 $\pm$0.56	<u>93.44</u> $\pm$0.52	88.21 $\pm$0.74	**93.85** $\pm$0.45
	50	91.97 $\pm$0.85	92.34 $\pm$0.87	92.47 $\pm$0.69	<u>92.56</u> $\pm$0.87	87.32 $\pm$1.05	**93.78** $\pm$0.65
	100	91.08 $\pm$0.71	90.41 $\pm$1.12	<u>91.61</u> $\pm$0.87	90.68 $\pm$1.35	85.37 $\pm$1.33	**91.69** $\pm$0.92
GSC	20	77.35 $\pm$1.19	86.53 $\pm$0.99	<u>87.12</u> $\pm$0.92	85.76 $\pm$1.08	86.49 $\pm$0.82	**88.93** $\pm$0.88
	50	65.48 $\pm$1.63	80.86 $\pm$1.09	81.59 $\pm$1.08	81.39 $\pm$1.27	<u>81.98</u> $\pm$1.36	**83.04** $\pm$1.13
	100	59.30 $\pm$1.93	70.12 $\pm$2.18	73.21 $\pm$1.55	71.94 $\pm$2.04	<u>73.87</u> $\pm$2.11	**76.51** $\pm$1.66

D.3.5 Additional Sensitivity Analysis

Window length E. We further evaluate the sensitivity of stability detection to the window length E. As shown in Table 11, we tested $E \in \{5, 10, 20\}$ under otherwise identical settings. The results demonstrate that accuracy varies by less than $\pm 1.4\%$ for $E \geq 10$, with $E = 10$ selected as the default configuration. In contrast, a shorter window length ($E = 5$) increases false positives and leads to slightly degraded performance. This indicates that the method is robust to window length choices as long as the window is sufficiently large to suppress noise. Additionally, the window length should be set smaller than the total number of training rounds divided by the number of model layers, ensuring that under ideal conditions, all layers can be stabilized and participate in aggregation.

Table 11: Sensitivity analysis of window length E on stability detection.

E	Final Accuracy (%)	Variation
5	44.21	-1.36%
10	45.57	—
20	45.12	-0.45%

Hyperparameter b. In order to study the influence of hyperparameter b on segmentation performance through accuracy, the experimental results are shown in Table 12. The highest accuracy is achieved when b is equal to 0.05, which proves that this bias will directly affect the final effect. Appropriate bias can effectively change the energy threshold according to the depth of the layer to achieve accurate layer-wise information segmentation.

A larger hyperparameter b implies a larger low-frequency range in the early training stage, when shallow networks are mainly aggregated, thereby enabling more shared knowledge to be transferred among shallow networks. In the later training stage, however, when deep networks are mainly aggregated, the shareable knowledge is significantly reduced.

Table 12: Sensitivity Response of Model Accuracy to Energy-bias Coefficient b.

b	0.05	0.1	0.15
Acc	45.57	42.86	42.63

D.3.6 Additional Ablation Study

We have conducted additional ablation experiments on CIFAR-100 with $K = 50$ clients in Table 13, representing a larger-scale federated setting. The results remain consistent: removing any individual component leads to performance degradation, while the full FedPeel achieves the best performance among all variants. These additional results further validate the effectiveness of each proposed component beyond the original setting.

Table 13: Ablation studies on the key mechanisms of FedPeel on CIFAR-100 with K = 50.

STKD	Stabilized Layer Detection	Adaptive Frequency-domain Decoupling	Max Acc(%)
✓	✓	✓	43.84
×	✓	✓	38.09
✓	×	✓	39.58
✓	✓	×	39.52
×	×	×	37.90

D.4 Empirical Experiments of key mechanisms

D.4.1 Empirical Verification of Gradient Variance Reduction

To address whether STKD effectively reduces gradient variance, we conduct systematic experiments. Specifically, we partition samples into three groups (Low, Medium, and High) according to the teacher model's uncertainty R_t, and perform experiments on CIFAR-10 using a 5-layer CNN. As shown in Table 14, compared with vanilla knowledge distillation (Vanilla KD), STKD consistently reduces gradient variance across all uncertainty groups, with reductions ranging from 7.9% to 11.9%. The most significant reduction (11.9%) occurs in the low-uncertainty group, where teacher predictions are most reliable.

To further demonstrate the advantage of STKD in promoting optimization stability during continuous training, we analyze gradient cosine similarity and gradient norm. Results are presented in Table 15. STKD increases the average gradient cosine similarity between consecutive iterations from 0.2308 to 0.2663, corresponding to a relative improvement of 15.4%. Meanwhile, the average gradient norm decreases from 1.2001 to 1.0668, representing an 11.1% reduction. These results indicate that

STKD not only enhances the directional consistency of gradients but also mitigates abrupt changes in gradient magnitude, leading to smoother and more stable optimization.

Table 14: Gradient variance reduction on CIFAR-10 with a 5-layer CNN.

R_t Group	Vanilla KD	STKD	Reduction
Low	0.001345	0.001185	**11.9%**
Medium	0.001308	0.001204	**7.9%**
High	0.001391	0.001266	**9.0%**

Table 15: Gradient smoothness analysis.

Metric	Vanilla KD	STKD
Avg Cosine Similarity	0.2308	0.2663
Avg Gradient Norm	1.2001	1.0668

D.4.2 Empirical Verification of frequency-domain decoupling

To provide direct empirical evidence supporting the assumption that low-frequency components correspond to shared knowledge while high-frequency components capture personalized information, we design a comparative experiment. Three settings were evaluated: Baseline (standard federated averaging with full parameter transmission), Low (retaining and uploading only low-frequency parameters via DCT, preserving the top 30% coefficients), and High (retaining and uploading only high-frequency parameters via DCT, preserving the bottom 70% coefficients). As shown in Table 16, the Low setting achieves 81.4% accuracy, which corresponds to 97.4% of the baseline performance (83.6%), indicating that low-frequency parameters alone suffice for effective cross-client learning and encode the majority of shared semantic information. In contrast, the High setting attains only 16.3% accuracy, confirming that high-frequency parameters alone cannot support effective classification and do not capture cross-client shared knowledge. Thus, this experiment provides direct empirical evidence that low-frequency components correspond to cross-client shared knowledge, whereas high-frequency components do not encode such information.

Table 16: Performance comparison of low-frequency vs. high-frequency parameters on CIFAR-10.

Setting	**Final Accuracy (%)**	**Convergence**
Baseline (Full)	83.6	Stable
Low (30% low-freq)	81.4	Stable
High (70% high-freq)	16.3	Unstable

D.4.3 Empirical Verification of Layer-wise Stabilization

To empirically verify the claim in Theorem 3.1 that shallow layers stabilize earlier than deeper ones, we recorded the number of communication rounds required for each layer of CNN-style models to reach a stable state. As shown in Table 17, the results demonstrate that shallow layers typically stabilize within the first 20–30 rounds, whereas middle and deep layers require substantially more rounds. Specifically, Layer 1 stabilizes at round 9, Layer 2 at round 14, Layer 3 at round 21, Layer 4 at round 39, and Layer 5 at round 47. Analysis of the distribution of stable layers uploaded by clients per round reveals that early rounds are dominated by shallow layers; as training proceeds, middle and deep layers progressively become stable. This layer-wise stabilization trend aligns closely with the theoretical prediction of Theorem 3.2 and provides strong empirical support for our layer-wise aggregation strategy.

Table 17: Number of rounds required for different layers to reach stability on the CIFAR-100 dataset.

	Layer 1	Layer 2	Layer 3	Layer 4	Layer 5
Rounds	9	14	21	39	47

E Discussion

E.1 Contributions

In this work, we address the key obstacles in heterogeneous federated learning (HtFL): model oscillation caused by aggregating unstable layers, capacity mismatch between lightweight proxy models and heterogeneous local models, and the difficulty of separating shared knowledge from client-specific patterns. To tackle these issues, we introduce FedPeel, a novel HtFL framework that aggregates only theoretically stabilized layers to prevent divergence, employs sample-wise temperature-controlled knowledge distillation (STKD) to dynamically regulate knowledge transfer based on teacher uncertainty, and leverages adaptive frequency-domain decoupling via DCT to separate low-frequency shared patterns from high-frequency personalized details. FedPeel operates without requiring public datasets and achieves robust knowledge exchange through a lightweight homogeneous proxy. The effectiveness of our approach is supported by theoretical convergence analysis and extensive experiments on image (CIFAR-10/100), text (AGNews), and audio (GSC) datasets, demonstrating consistent improvements over state-of-the-art methods in both accuracy and communication efficiency.

E.2 Limitations

Limitations remain, particularly regarding deployment on extremely large-scale federations and resource-constrained edge devices. While FedPeel has shown strong performance under simulated heterogeneous conditions, its practical implementation in real-world systems with dynamic client participation, heterogeneous hardware, and varying network conditions may reveal additional engineering challenges, offering valuable directions for future scalability and robustness improvements.

E.3 Impacts

Heterogeneous Federated Learning enables collaborative AI training across siloed, non-identical data sources, directly advancing privacy-preserving applications in healthcare and finance. It also democratizes access to machine learning for resource-limited institutions and provides open educational tools to train a diverse workforce in decentralized AI.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We have faithfully stated our contributions in both the abstract and introduc tion.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We give the limitations of the work in the Appendix.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an impor tant role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: The core theoretical results are stated in Section 3. Due to space limitations, we are unable to present all the missing assumptions, proofs and intermediate results in the main text. They are deferred to the Appendix. Please refer to Appendix B and C for details.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide detailed experimental and hyperparameter setups in Section 4 and Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Our evaluations are based on open-accessed datasets that are available. An official implementation code is provided in supplemental material.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: Experimental setting/details are important parts of reproducing our results. We provide the details in Section 4 and Appendix D to the best of our ability.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Our results are averaged over multiple random seeds.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please find the hardware specifications in Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The NeurIPS code of ethics is strictly enforced throughout our research.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have discussed broader impacts in Section 1. We are unaware of any negative impacts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The existing assets used in this paper have been adequately cited or credited to.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We have documented the experiment details in Section 4 and Appendix D. In addition, we provide our code with clear details and examples.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: the core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.