BEYOND SINGLE-POLICY FEDERATED LLM ALIGNMENT UNDER CONFLICTING PREFERENCES

Anonymous authors
Paper under double-blind review

ABSTRACT

Federated LLM alignment seeks to reconcile global consensus with diverse user preferences while keeping private data on clients. Online iterative methods improve alignment by fine-tuning models on freshly sampled responses, but conflicting client preferences complicate their use in federated learning. We present FedDePO, a framework for federated online preference optimization that separates shared preferences from client-specific preferences. To improve local supervision, FedDePO uses a preference transitivity metric to filter unstable labels based on the logical self-consistency of judgments. It then compares implicit reward margins on public probes to detect cross-client preference divergence without transmitting raw training data. A dual low-rank adapter design routes consensus pairs to a shared adapter and contested pairs to personal adapters. Policy Timestamp Alignment adaptively limits local updates to preserve data freshness between communication rounds. Experiments across multiple model scales show that FedDePO outperforms the evaluated federated baselines and approaches centralized online training. Capacity-matched controls and evaluation on human survey preferences further support the benefit of preference decomposition.

1 INTRODUCTION

Preference alignment steers large language models (LLMs) toward human values and quality expectations (Ouyang et al., 2022). Among post-training methods, Direct Preference Optimization (DPO) (Rafailov et al., 2023) has become especially popular because it replaces the two-stage RLHF pipeline with a single classification loss that implicitly encodes rewards in the policy. Yet offline DPO can degrade as the policy drifts away from the fixed preference dataset (Guo et al., 2024; Tajwar et al., 2024). Online iterative DPO (Xiong et al., 2024; Dong et al., 2024) addresses this by repeatedly sampling fresh responses from the current policy, relabeling them, and updating on the resulting on-policy data. The self-rewarding variant (Yuan et al., 2024) further removes external reward models by letting the LLM itself act as the Judge.

Federated learning (FL) (McMahan et al., 2017) offers a natural way to align LLMs without sharing private data. OpenFedLLM (Ye et al., 2024) extends DPO to this setting through FedDPO, where clients optimize local preference data and the server aggregates parameter updates. Offline FedDPO reuses fixed preference pairs and aggregates a single global policy, leaving both online data refresh and conflicting client preferences unresolved.

Beyond online data refresh, the central obstacle is preference conflict. When the Judge is assigned different personas such as *concise*, *detailed*, or *direct*, different clients may prefer opposite responses to the same prompt. Recent analyses (Gölz et al., 2025; Shirali et al., 2025) show that this heterogeneity induces irreducible distortion in a single global DPO policy, and because its root cause is the single-policy assumption rather than the aggregation rule, it cannot be removed by reweighting clients alone. A practical federated online DPO framework must therefore address both on-policy data generation and the need to decouple conflicting preferences during aggregation.

Motivated by these challenges, we propose **FedDePO**, a framework for federated online DPO that decouples consensus and personal preference learning under conflicting preferences. We identify three tensions specific to this setting, namely unreliable local Judges, the lack of ground-truth anchors for subjective preferences, and stale on-policy data caused by multi-step local updates. FedDePO addresses them with four coupled components. Preference Transitivity Consistency measures the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

logical self-consistency of local judgments to filter noisy labels. Implicit Reward Divergence detects cross-client preference conflict in reward space. A dual LoRA design decouples consensus learning from personalization, which directly targets preference conflict and relaxes the single-policy assumption. Policy Timestamp Alignment adaptively limits local updates to preserve data freshness under communication constraints.

For example, a concise-answer client and a detailed-answer client may disagree on which of two correct responses is better. Both can be internally consistent. PTC checks the stability of each local judgment, while IRD compares their learned reward margins on common public pairs. Routing separates agreement useful to the federation from disagreement useful for personalization. PTA controls how long either branch trains on the same generated responses.

Our contributions are threefold.

- We formalize federated online DPO and identify three tensions arising from the interaction of local self-judging, conflicting preferences, and multi-step federated updates.
- We propose FedDePO, which integrates Preference Transitivity Consistency, Implicit Reward Divergence with a dual LoRA decomposition, and Policy Timestamp Alignment to jointly address Judge reliability, preference conflicts, and data freshness.
- Experiments across model scales show that FedDePO outperforms federated baselines and approaches centralized online DPO. Capacity-matched controls and human survey preferences support the benefit of reward-space routing beyond added adapter capacity.

2 PRELIMINARIES

Reinforcement learning from human feedback. Aligning LLMs with human preferences typically follows a two-stage pipeline (Ouyang et al., 2022; Christiano et al., 2017). First, a reward model $r_\phi(x, y)$ is trained on a dataset of human preference comparisons to predict which response is preferred for a given prompt. Second, the LLM policy π_θ is optimized via reinforcement learning (e.g., PPO (Schulman et al., 2017)) to maximize the learned reward while staying close to a reference policy π_{ref}:

$$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D},\, y\sim\pi_\theta(\cdot|x)}\big[r_\phi(x,y)\big] - \beta\, D_{\text{KL}}\big[\pi_\theta(y|x)\,\|\,\pi_{\text{ref}}(y|x)\big], \tag{1}$$

Here $\beta > 0$ controls KL regularization. Training a reward model and running on-policy RL add computation and optimization complexity (Bai et al., 2022; Touvron et al., 2023).

Direct preference optimization. DPO (Rafailov et al., 2023) eliminates the need for explicit reward modeling by exploiting a closed-form mapping between the optimal policy and the reward function. Given a preference dataset $\mathcal{D} = \{(x, y_w, y_l)\}$ where y_w is preferred over y_l, the optimal policy of Eq. (1) satisfies $r^*(x, y) = \beta \log \frac{\pi^*(y|x)}{\pi_{\text{ref}}(y|x)} + \beta \log Z(x)$. Substituting this into the Bradley-Terry preference model and canceling the partition function yields a loss defined directly over the policy:

$$\mathcal{L}_{\text{DPO}}(\theta) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta\log\frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}\right)\right]. \tag{2}$$

A property central to our framework is the *implicit reward*. For any DPO-trained policy π_θ, the quantity

$$\hat{r}_\theta(x, y) = \beta \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} \tag{3}$$

serves as a pointwise reward estimate without requiring a separate reward model. We will use this structure extensively in Section 3.2.

Online iterative DPO. Training on a fixed offline preference dataset can lead to policy degradation as the learned policy moves away from the data distribution (Guo et al., 2024; Tajwar et al., 2024). Online iterative DPO (Xiong et al., 2024; Dong et al., 2024) addresses this by repeating a cycle of (i) generating new responses from the current policy $\pi_\theta^{(t)}$, (ii) obtaining preference labels from a reward model or an LLM-as-a-Judge (Zheng et al., 2023; Yuan et al., 2024), and (iii) performing DPO updates on the freshly generated data. This keeps the training distribution aligned with the evolving policy and has been shown to significantly outperform offline DPO both theoretically and

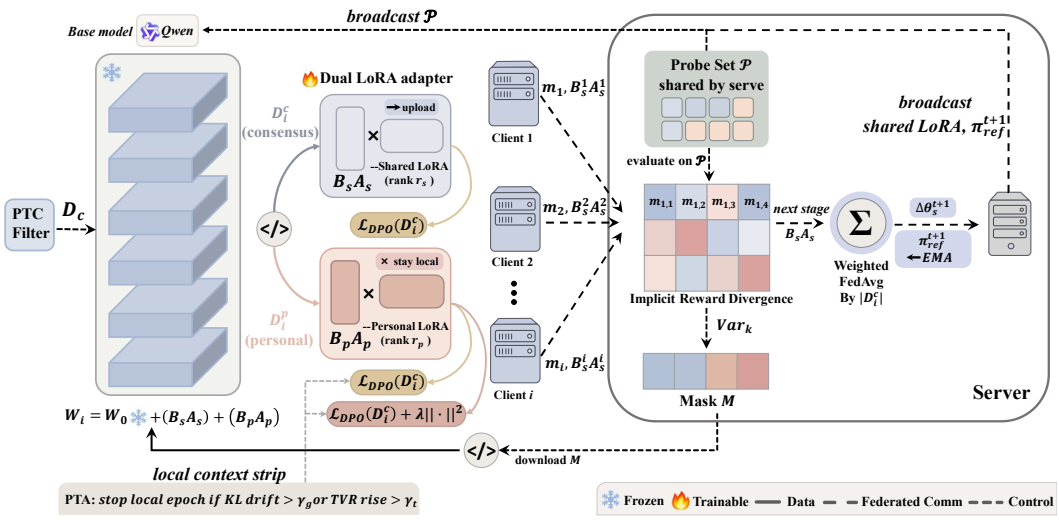

Figure 1: FedDePO. Clients filter local comparisons with PTC and upload reward margins on public probes. The server returns an IRD consensus mask. Local routing directs consensus pairs to the shared adapter and contested pairs to the personal adapter. PTA bounds local training, and only shared adapters are aggregated.

empirically (Xiong et al., 2024). In the self-rewarding variant (Yuan et al., 2024), the LLM itself acts as the Judge, producing preference labels via prompted evaluation, which removes the dependence on an external annotator.

Preference heterogeneity. When preferences follow a homogeneous Bradley–Terry model, DPO recovers the optimal policy in closed form. Under heterogeneous preferences, this guarantee no longer holds. Gölz et al. (2025) formalize the resulting utility loss through *distortion*, measured in the worst case by the ratio of maximum achievable average utility to the average utility of the learned policy π,

$$\text{distortion} = \frac{\max_{\pi^*} \text{AvgUtil}(\pi^*)}{\text{AvgUtil}(\pi)} \tag{4}$$

DPO implicitly aggregates heterogeneous pairwise preferences through a Borda-type rule, which can incur substantial distortion. Complementarily, Shirali et al. (2025) characterize when direct alignment under heterogeneous preferences requires additional annotator information. The implication for federated settings is clear. Simply embedding online DPO into FedAvg is inadequate when clients hold conflicting preferences. The framework must structurally separate shared and personal preferences directly at the optimization target level.

3 Method

3.1 Problem Formulation

We consider N clients, each holding a private prompt set $\mathcal{D}_i$, who collaboratively train a global LLM policy π_θ without sharing raw data. Let $\pi_\theta(\cdot|x)$ denote the conditional distribution over responses given a prompt x, and let π_{ref} denote a reference policy that is frozen within each round and refreshed between rounds. At each federated round t, every client receives the current global policy $\pi_\theta^{(t)}$ from the server. Client i generates on-policy responses $y \sim \pi_\theta^{(t)}(\cdot|x)$ for prompts $x \in \mathcal{D}_i$ and constructs preference pairs (y_w, y_l) from these responses.

Preference labels can come from local human annotators, local reward models, or an LLM judge. We use local self-judging to avoid requiring external annotation or transmitting private prompts and responses to a centralized evaluator. This choice makes label stability part of the learning problem.

In this self-rewarding setup (Yuan et al., 2024), each client uses its local LLM policy as a preference judge. Prompted evaluation produces pairwise labels under the client's assigned persona. Let $J_i(\cdot,\cdot|x)$ denote the pairwise judgment function of client i, which outputs a preference ordering over two candidate responses for prompt x. The locally constructed preference dataset is $\mathcal{D}_i^{\text{pref}} = \{(x, y_w, y_l) : x \in \mathcal{D}_i,\ (y_w, y_l) \sim J_i\}$, and the single-policy federated objective is

$$\min_{\theta} \sum_{i=1}^{N} w_i \, \mathcal{L}_{\text{DPO}}\big(\pi_\theta;\, \mathcal{D}_i^{\text{pref}}\big), \tag{5}$$

where $w_i = |\mathcal{D}_i| / \sum_j |\mathcal{D}_j|$ is the weight of client i.

This formulation introduces two challenges. First, each client's LLM judge J_i receives no ground-truth supervision, so its judgments may be noisy, inconsistent, or biased. Agreement with external annotations can assess label quality when such annotations are available, but local self-judging must operate without them. A useful filter must therefore assess stability from the client's own judgments while respecting its preference direction. Second, heterogeneous preferences across clients can cause irreducible distortion in single-policy optimization. DPO aggregates conflicting preferences through a Borda-type rule, which can yield substantial utility loss (Gölz et al., 2025; Shirali et al., 2025). Reliable local labels alone do not remove this conflict. Federated preference optimization must address both the stability of each client's supervision and the compatibility of preferences across clients.

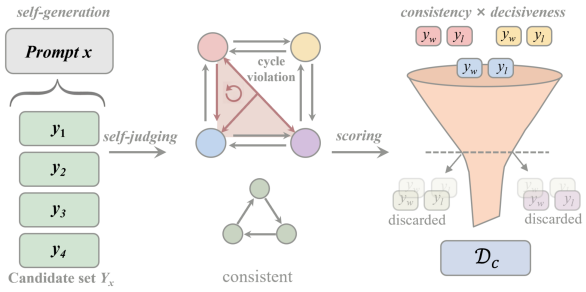

Figure 2: PTC combines graph consistency and pairwise decisiveness to select training pairs.

3.2 FedDePO

FedDePO addresses these challenges through four coupled components. **PTC** filters unreliable local judgments via logical self-consistency (addressing tensions 1 and 2), **IRD** detects cross-client preference conflict in reward space, **Dual LoRA** separates consensus and personal learning, and **PTA** preserves data freshness under federated communication constraints (addressing tension 3).

Preference Transitivity Consistency.

The self-rewarding design makes each client's LLM serve simultaneously as both the policy being optimized and the judge producing training labels. Since these labels are subjective and lack ground-truth references (tensions 1 and 2), we cannot apply standard quality metrics such as inter-annotator agreement or accuracy against gold labels. We therefore need an intrinsic criterion that evaluates judge reliability from the judgments themselves.

Under a fixed prompt, persona, and judging instruction, transitivity provides a consistency criterion. If $y_a \succ y_b$ and $y_b \succ y_c$, a transitive ordering also has $y_a \succ y_c$. We use violations to identify unstable supervision. Systematic bias can remain transitive, while valid multi-attribute trade-offs can be cyclic. PTC therefore selects stable, decisive comparisons, with independent label-agreement checks reported in Appendix C.14.

For a prompt x the current policy generates a candidate set $\mathcal{Y}_x = \{y_1, \ldots, y_K\}$ and evaluates all $\binom{K}{2}$ pairs bidirectionally. We construct a directed preference graph $G_x = (\mathcal{Y}_x, E)$ where edge $(y_a, y_b) \in E$ indicates y_a is judged preferable to y_b. A transitivity violation occurs whenever a directed 3-cycle $y_a \succ y_b \succ y_c \succ y_a$ exists. We define the transitivity violation rate as

$$\text{TVR}(x) = \frac{1}{\binom{K}{3}} \sum_{1 \le a < b < c \le K} \mathbf{1}[\{y_a, y_b, y_c\} \text{ forms a directed cycle}]\,. \tag{6}$$

For each preference pair (y_w, y_l) extracted from G_x, we compute a sample-level confidence score combining the logical consistency of the overall graph and the decisiveness of the individual pairwise

verdict. Denoting the forward and reverse judgment probabilities as $P_j(y_w \succ y_l)$ and $P_j(y_l \succ y_w)$ respectively, we define

$$c(x, y_w, y_l) = \big(1 - \mathrm{TVR}(x)\big)\,\big|P_j(y_w \succ y_l) - P_j(y_l \succ y_w)\big|. \tag{7}$$

The first factor penalizes logically inconsistent judges while the second factor measures the decisiveness of the pairwise verdict, reaching zero when the judge is maximally uncertain and its maximum when the judgment is deterministic. Only pairs exceeding a threshold $c_{\min}$ enter the filtered training set $\mathcal{D}_i^{\text{clean}}$, ensuring that parameter updates rely on logically consistent and decisive verdicts.

Implicit Reward Divergence. PTC filters unstable local judgments, yet two clients can produce internally consistent judgments that favor opposite responses. Such conflicts arise naturally when clients represent user populations with different values or tastes. Detecting them requires comparing preference signals across clients, using common response pairs so that differences reflect how clients evaluate the same alternatives.

We use the implicit reward structure of DPO (Eq. 3) to compare clients in reward space. The server maintains a small public probe set $\mathcal{P} = \{(x_k, y_k^{(1)}, y_k^{(2)})\}_{k=1}^{|\mathcal{P}|}$, and before local optimization client i evaluates its current policy on $\mathcal{P}$ to compute a margin vector

$$\mathbf{m}_i = \Big[\hat{r}_{\theta_i}(x_k, y_k^{(1)}) - \hat{r}_{\theta_i}(x_k, y_k^{(2)})\Big]_{k=1}^{|\mathcal{P}|}, \tag{8}$$

where $\hat{r}_{\theta_i}(x, y) = \beta \log \frac{\pi_{\theta_i}(y|x)}{\pi_{\text{ref}}(y|x)}$ is the implicit reward. Each entry $m_{i,k}$ measures the direction and strength of client i's preference on the k-th probe. The server then computes the cross-client variance for each probe dimension

$$v_k = \mathrm{Var}\big(\{m_{1,k},\, m_{2,k},\, \ldots,\, m_{N,k}\}\big). \tag{9}$$

A binary consensus mask follows from thresholding, $\mathcal{M}_k = \mathbf{1}[v_k \le \tau]$. Low-variance probes indicate consensus, while high-variance probes indicate potential conflict. Margin scale also affects variance. Appendix C.15 reports controls for direction, calibration, and training state.

Each client transfers this probe-level signal to its local pairs through probe-anchor routing, detailed in Appendix E and Algorithm 1. For each clean pair $z = (x, y_w, y_l) \in \mathcal{D}_i^{\text{clean}}$, we embed z with a frozen reference encoder, retrieve its top-J nearest probes, and form similarity-based routing weights

$$a_k(z) = \frac{\exp\big(\cos(\mathbf{q}(z), \mathbf{q}_k)/T_a\big)}{\sum_{\ell \in \mathcal{N}_J(z)} \exp\big(\cos(\mathbf{q}(z), \mathbf{q}_\ell)/T_a\big)}, \qquad k \in \mathcal{N}_J(z), \tag{10}$$

where $T_a > 0$ is a temperature. We then assign consensus and personal scores by

$$s_i^c(z) = \sum_{k \in \mathcal{N}_J(z)} a_k(z)\, \mathcal{M}_k, \qquad s_i^p(z) = 1 - s_i^c(z). \tag{11}$$

Pairs with $s_i^c(z) \ge 1/2$ are routed to the consensus branch and the rest to the personal branch. This turns the probe mask into an actionable partition of local data without transmitting raw pairs off-device.

Preference Decomposition via Dual LoRA. The consensus mask $\mathcal{M}$ distinguishes shared preferences from client-specific preferences. A single adapter combines their updates even when preference directions oppose each other, forcing a common policy to compromise (Gölz et al., 2025). We allocate separate adaptation capacity to consensus and contested pairs. This parameter separation allows each client to retain its preferred behavior while contributing shared updates to the federation.

Following the LoRA formulation (Hu et al., 2022), for a pretrained weight matrix W_0, we introduce two adapters for each target layer. The *shared adapter* (B_s, A_s) with rank r_s parameterizes the consensus update $\Delta\theta_s = B_s A_s$ and participates in federated aggregation. The *personal adapter* (B_p^i, A_p^i) with rank $r_p \le r_s$ parameterizes the local deviation $\Delta\theta_p^i = B_p^i A_p^i$ and remains local, with a smaller rank used for client-specific adaptation. The effective weight at client i becomes

$$W_i = W_0 + \Delta\theta_s + \Delta\theta_p^i. \tag{12}$$

Each client uses Eqs. (10)–(11) to partition its filtered data $\mathcal{D}_i^{\text{clean}}$ into a consensus subset $\mathcal{D}_i^c$ and a personal subset $\mathcal{D}_i^p$. The local objective combines two branch-specific DPO losses with a regularization term

$$\mathcal{L}_i = \mathcal{L}_{\text{DPO}}(\pi_{W_0+\Delta\theta_s};\, \mathcal{D}_i^c) + \mathcal{L}_{\text{DPO}}(\pi_{W_i};\, \mathcal{D}_i^p) + \lambda \|\Delta\theta_p^i\|_F^2. \tag{13}$$

The consensus loss updates only $\Delta\theta_s$ while keeping $\Delta\theta_p^i$ frozen, and the personal loss updates only $\Delta\theta_p^i$ while keeping $\Delta\theta_s$ frozen. The Frobenius norm penalty prevents excessive personal deviation that could compromise generalization. Consensus preferences thus benefit from the full statistical strength of the federation through $\Delta\theta_s$, while conflicting personal preferences are absorbed by $\Delta\theta_p^i$ without directly aggregating their personal updates.

Policy Timestamp Alignment. Online DPO stays on-policy by refreshing responses after every update, whereas federated training necessarily performs multiple local steps between synchronizations. As these local updates accumulate, the policy that generated the data drifts away from the policy being optimized, degrading both learning quality and the reliability of self-judging (addressing tension 3).

To control this mismatch, we adopt an adaptive stopping rule. Let $\theta_i^{(0)}$ denote the client parameters at the start of round t and $\theta_i^{(e)}$ the parameters after e local epochs. Client i stops local training once

$$\mathbb{E}_{x\sim\mathcal{D}_i}\left[D_{\mathrm{KL}}\big[\pi_{\theta_i^{(e)}}(\cdot|x)\,\big\|\,\pi_{\theta_i^{(0)}}(\cdot|x)\big]\right] > \gamma_g \quad \text{or} \quad \mathrm{TVR}^{(e)} > \mathrm{TVR}^{(0)} + \gamma_t, \tag{14}$$

where γ_g bounds policy drift and γ_t bounds judge degradation. The KL term keeps optimization close to the generation distribution, while the TVR term halts training when the judge's logical consistency begins to deteriorate.

We also update the reference policy after each round using an exponential moving average,

$$\theta_{\mathrm{ref}}^{(t+1)} = \alpha\,\theta_{\mathrm{ref}}^{(t)} + (1-\alpha)\,\bar{\theta}_s^{(t+1)}, \tag{15}$$

where $\bar{\theta}_s^{(t+1)}$ is the aggregated shared adapter and $\alpha \in (0,1)$ is the decay rate. This keeps the KL anchor aligned with the evolving global model.

3.3 Federated Update Protocol

At round t, the server broadcasts the shared adapter, round-frozen reference, and public probes to all N clients. Every client uploads its current probe-margin vector, including clients not selected for training. The server computes the consensus mask from these N vectors. The training subset S_t generates on-policy responses, applies PTC, and routes retained pairs using the mask. Shared and personal updates follow Eq. 13 until PTA stops the local phase or the local-step cap is reached. Only training clients upload shared adapter parameters, aggregated as

$$(B_s^{(t+1)}, A_s^{(t+1)}) = \sum_{i\in S_t} \frac{|\mathcal{D}_i^c|}{\sum_{j\in S_t}|\mathcal{D}_j^c|}(B_s^i, A_s^i). \tag{16}$$

The server then refreshes the reference by Eq. 15. Raw pairs and personal adapters remain local. Shared updates and probe margins can still reveal client preferences, and the evaluated protocol does not apply differential privacy. The participation sweep varies training participation while retaining full diagnostic participation (Appendix C.8). Algorithm 2 specifies the complete procedure.

4 Experiments

4.1 Experimental Setup

Datasets and models. We evaluate FedDePO on UltraFeedback (Cui et al., 2024) for general helpfulness alignment and HH-RLHF (Bai et al., 2022) for safety alignment, the latter containing helpful and harmless subsets with structurally opposing optimization directions. Three Qwen3 models (Yang et al., 2025) at 0.6B, 1.7B and 4B parameters serve as base policies, covering three parameter scales. Cross-architecture results on Llama-3.2-3B-Instruct (Meta, 2024) and Mistral-7B-Instruct-v0.3 (Mistral AI, 2024) are provided in Appendix C.2.

Online versus offline training. Prior federated DPO (Ye et al., 2024) consumes a fixed annotated set $\mathcal{D}_i^{(0)}$ throughout training. FedDePO instead generates responses $y \sim \pi_{\theta_i}^{(t)}(\cdot|x)$ on policy at every round and labels them via the local Judge, so $\mathcal{D}_i^{(t)}$ coevolves with the policy. Appendix C.11 separates adapter capacity and routing from the online configuration. The online single-adapter control does

Table 1: Main results on UltraFeedback and HH-RLHF across three Qwen3 model scales under moderate heterogeneity. MT: MT-Bench; AE: AlpacaEval; RB: RewardBench. Subscripts are standard deviations over three seeds; best federated results are **bold**.

Dataset	Model	Method	MT	AE	HHH	RB	Avg
UltraFeedback	Qwen3-0.6B	Zero-shot	$3.82_{\pm 0.03}$	$5.13_{\pm 0.13}$	$58.27_{\pm 0.11}$	$53.71_{\pm 0.19}$	39.04
		FedAvg+SFT	$4.25_{\pm 0.09}$	$8.24_{\pm 0.38}$	$60.12_{\pm 0.42}$	$55.83_{\pm 0.29}$	41.40
		FedDPO	$4.56_{\pm 0.08}$	$11.37_{\pm 0.43}$	$61.82_{\pm 0.28}$	$57.94_{\pm 0.34}$	43.71
		FedDePO	$\mathbf{5.02}_{\pm 0.05}$	$\mathbf{15.68}_{\pm 0.27}$	$\mathbf{64.21}_{\pm 0.19}$	$\mathbf{60.33}_{\pm 0.23}$	**46.74**
		Central online	$5.14_{\pm 0.04}$	$16.83_{\pm 0.22}$	$64.92_{\pm 0.16}$	$61.25_{\pm 0.18}$	47.67
	Qwen3-1.7B	Zero-shot	$4.93_{\pm 0.02}$	$8.62_{\pm 0.12}$	$66.41_{\pm 0.09}$	$59.85_{\pm 0.13}$	44.96
		FedAvg+SFT	$5.31_{\pm 0.06}$	$13.24_{\pm 0.31}$	$68.53_{\pm 0.33}$	$62.17_{\pm 0.22}$	47.98
		FedDPO	$5.72_{\pm 0.07}$	$17.85_{\pm 0.28}$	$70.72_{\pm 0.23}$	$64.53_{\pm 0.27}$	51.03
		FedDePO	$\mathbf{6.24}_{\pm 0.04}$	$\mathbf{22.61}_{\pm 0.21}$	$\mathbf{73.42}_{\pm 0.17}$	$\mathbf{67.35}_{\pm 0.16}$	**54.46**
		Central online	$6.38_{\pm 0.03}$	$23.94_{\pm 0.18}$	$74.13_{\pm 0.13}$	$68.04_{\pm 0.14}$	55.37
	Qwen3-4B	Zero-shot	$5.82_{\pm 0.02}$	$12.43_{\pm 0.08}$	$72.15_{\pm 0.07}$	$64.32_{\pm 0.10}$	49.63
		FedAvg+SFT	$6.21_{\pm 0.05}$	$17.83_{\pm 0.26}$	$75.04_{\pm 0.21}$	$66.92_{\pm 0.25}$	53.26
		FedDPO	$6.47_{\pm 0.06}$	$22.65_{\pm 0.23}$	$77.82_{\pm 0.19}$	$69.43_{\pm 0.22}$	56.63
		FedDePO	$\mathbf{6.95}_{\pm 0.03}$	$\mathbf{28.34}_{\pm 0.17}$	$\mathbf{81.63}_{\pm 0.14}$	$\mathbf{72.85}_{\pm 0.13}$	**60.94**
		Central online	$7.08_{\pm 0.03}$	$30.12_{\pm 0.14}$	$82.41_{\pm 0.11}$	$73.52_{\pm 0.11}$	62.02
HH-RLHF	Qwen3-0.6B	Zero-shot	$3.82_{\pm 0.04}$	$5.13_{\pm 0.16}$	$58.27_{\pm 0.15}$	$53.71_{\pm 0.21}$	39.04
		FedAvg+SFT	$4.08_{\pm 0.10}$	$6.42_{\pm 0.36}$	$63.75_{\pm 0.47}$	$58.93_{\pm 0.35}$	43.03
		FedDPO	$4.22_{\pm 0.08}$	$7.91_{\pm 0.41}$	$66.94_{\pm 0.38}$	$61.73_{\pm 0.33}$	45.53
		FedDePO	$\mathbf{4.51}_{\pm 0.06}$	$\mathbf{10.32}_{\pm 0.25}$	$\mathbf{70.82}_{\pm 0.24}$	$\mathbf{65.24}_{\pm 0.22}$	**48.79**
		Central online	$4.63_{\pm 0.04}$	$11.25_{\pm 0.23}$	$71.63_{\pm 0.19}$	$65.92_{\pm 0.17}$	49.60
	Qwen3-1.7B	Zero-shot	$4.93_{\pm 0.03}$	$8.62_{\pm 0.14}$	$66.41_{\pm 0.12}$	$59.85_{\pm 0.16}$	44.96
		FedAvg+SFT	$5.08_{\pm 0.08}$	$10.13_{\pm 0.37}$	$72.34_{\pm 0.32}$	$65.42_{\pm 0.24}$	49.30
		FedDPO	$5.24_{\pm 0.06}$	$12.63_{\pm 0.27}$	$75.84_{\pm 0.29}$	$68.92_{\pm 0.26}$	52.46
		FedDePO	$\mathbf{5.67}_{\pm 0.04}$	$\mathbf{16.32}_{\pm 0.20}$	$\mathbf{79.63}_{\pm 0.17}$	$\mathbf{72.41}_{\pm 0.16}$	**56.12**
		Central online	$5.78_{\pm 0.03}$	$17.04_{\pm 0.19}$	$80.21_{\pm 0.15}$	$72.93_{\pm 0.12}$	56.73
	Qwen3-4B	Zero-shot	$5.82_{\pm 0.02}$	$12.43_{\pm 0.10}$	$72.15_{\pm 0.09}$	$64.32_{\pm 0.12}$	49.63
		FedAvg+SFT	$6.02_{\pm 0.07}$	$14.23_{\pm 0.31}$	$78.52_{\pm 0.27}$	$70.14_{\pm 0.21}$	54.30
		FedDPO	$6.18_{\pm 0.05}$	$17.12_{\pm 0.22}$	$82.42_{\pm 0.24}$	$73.73_{\pm 0.19}$	57.76
		FedDePO	$\mathbf{6.56}_{\pm 0.03}$	$\mathbf{21.83}_{\pm 0.15}$	$\mathbf{86.24}_{\pm 0.13}$	$\mathbf{77.14}_{\pm 0.12}$	**61.74**
		Central online	$6.71_{\pm 0.02}$	$22.72_{\pm 0.13}$	$86.92_{\pm 0.10}$	$77.83_{\pm 0.10}$	62.49

not improve on offline FedDPO at rank 16, whereas IRD routing remains beneficial at matched total rank.

Data and preference heterogeneity. We control two dimensions of heterogeneity. For *data heterogeneity*, UltraFeedback prompts are allocated using a symmetric Dirichlet distribution with concentration κ. HH-RLHF's helpful and harmless subsets are assigned to separate clients, introducing differences in alignment objectives as well as data. For *preference heterogeneity*, each client receives a distinct Judge persona (concise, detailed, direct, elaborated, or neutral) on top of this partition. These personas can prefer different responses to the same prompt. On a held-out probe set $\mathcal{Q}$, we measure pairwise preference disagreement and its average as

$$\mathrm{PD}_{ij} = \mathbb{E}_{(x,y_a,y_b)\sim\mathcal{Q}}[\mathbf{1}\{\succ_i \neq \succ_j\}], \qquad H = \frac{2}{N(N-1)} \sum_{i<j} \mathrm{PD}_{ij}, \tag{17}$$

with $H = 0$ meaning full consensus. For N deterministic binary rankings, the per-pair maximum is $\lfloor N^2/4 \rfloor / \binom{N}{2}$, approaching $1/2$ as N grows. The main experiments use $\kappa = 0.1$ and five distinct Judge personas, with measured disagreement $H \approx 0.28$.

Baselines. We compare against Zero-shot (no post-training), FedAvg+SFT (McMahan et al., 2017) as a non-preference federated baseline, offline FedDPO (Ye et al., 2024) as the direct counterpart, and Centralized Online DPO (Xiong et al., 2024) as a centralized reference. All federated methods share the base policy, client partition ($N = 5$, three training clients per round), and 20-round budget. Single-adapter baselines use rank 16. FedDePO uses shared rank 16 and personal rank 8, with all

Table 2: Ablation results on Qwen3-1.7B. We report performance when individually removing PTC, IRD & Dual LoRA, or PTA. Subscripts denote standard deviation over three seeds. Avg represents the mean of AlpacaEval, HHH, and RewardBench.

Dataset	Variant	MT-Bench	AlpacaEval	HHH	RewardBench	Avg
UltraFeedback	**Full FedDePO**	**6.24**$_{\pm 0.04}$	**22.61**$_{\pm 0.21}$	**73.42**$_{\pm 0.17}$	**67.35**$_{\pm 0.16}$	**54.46**
	w/o PTC	6.01$_{\pm 0.07}$	20.15$_{\pm 0.32}$	71.88$_{\pm 0.22}$	65.42$_{\pm 0.26}$	52.48
	w/o IRD & Dual LoRA	5.85$_{\pm 0.09}$	18.73$_{\pm 0.34}$	69.51$_{\pm 0.38}$	63.88$_{\pm 0.25}$	50.71
	w/o PTA	6.12$_{\pm 0.05}$	21.04$_{\pm 0.23}$	72.10$_{\pm 0.27}$	66.21$_{\pm 0.18}$	53.12
HH-RLHF	**Full FedDePO**	**5.67**$_{\pm 0.04}$	**16.32**$_{\pm 0.20}$	**79.63**$_{\pm 0.17}$	**72.41**$_{\pm 0.16}$	**56.12**
	w/o PTC	5.50$_{\pm 0.08}$	14.88$_{\pm 0.31}$	77.21$_{\pm 0.26}$	70.85$_{\pm 0.23}$	54.31
	w/o IRD & Dual LoRA	5.28$_{\pm 0.10}$	13.15$_{\pm 0.39}$	74.55$_{\pm 0.33}$	68.12$_{\pm 0.29}$	51.94
	w/o PTA	5.55$_{\pm 0.06}$	15.41$_{\pm 0.28}$	78.05$_{\pm 0.19}$	71.33$_{\pm 0.21}$	54.93

clients providing IRD diagnostics each round. Main experiments use three random seeds. Runtime is measured separately.

Implementation and metrics. FedDePO uses DPO temperature 0.1 and reference-policy EMA decay $\alpha = 0.95$. Appendices C.10 and E provide the remaining hyperparameters and implementation details. Table 1 reports results at $\kappa = 0.1$ and $H \approx 0.28$. We evaluate alignment using MT-Bench (Zheng et al., 2023) scores (1–10) assigned by GPT-4, AlpacaEval 2.0 length-controlled win rate (Dubois et al., 2024), HHH alignment accuracy, and RewardBench accuracy for implicit reward fidelity. Avg is the arithmetic mean of AlpacaEval, HHH, and RewardBench. It summarizes three percentage-valued metrics whose scales are not calibrated to a common utility. We therefore report each component alongside Avg and keep MT-Bench separate. Main-table entries are means over three independent runs, with standard deviations shown for individual metrics.

4.2 Comparison with Baselines

Table 1 reports every component metric across six dataset–model settings. FedDePO outperforms the federated baselines on each metric and approaches centralized online DPO at every scale. On Qwen3-1.7B, it improves Avg over FedDPO by 3.43 points on UltraFeedback and 3.66 on HH-RLHF. The improvements hold separately for AlpacaEval, HHH, and RewardBench, so the ordering does not depend on their arithmetic average. Cross-architecture results appear in Appendix C.2, and comparisons with six personalized FL baselines appear in Appendix C.7.

4.3 Ablation Study

Module contribution. Table 2 evaluates PTC, IRD with Dual LoRA, and PTA on Qwen3-1.7B. Removing PTC lowers general helpfulness scores on UltraFeedback, supporting the use of consistent judgments for local supervision. Replacing IRD-guided Dual LoRA with a single shared adapter produces the largest decline, particularly on HH-RLHF. Because this ablation changes both capacity and routing, Appendix C.11 compares single and dual adapters at matched total rank. Additional rank and random splitting both fall short of IRD routing, indicating that the allocation of preference pairs matters beyond parameter count. Disabling PTA causes a smaller, consistent decline across the evaluated metrics. Together, these results support filtering local labels, separating conflicting preferences, and controlling how long each client trains on generated responses.

Effect of preference heterogeneity. Figure 3 varies preference disagreement and compares FedDPO, a single-adapter control, and full FedDePO. Both single-policy variants decline as H increases, while FedDePO preserves higher performance. The widening gap supports separating contested preferences from shared updates. Appendix C.10 reports the full persona-count sweep, and Appendix C.12 extends the analysis to opposed value preferences.

Robustness to Judge source. Figure 4 replaces the local self-judge with GPT-5 or released gold annotations while retaining the five-persona preference structure. FedDePO remains the strongest federated method in each condition. PTC's contribution declines as labels improve, while the gap to the no-Dual-LoRA control persists. PTC and decomposition thus address complementary problems

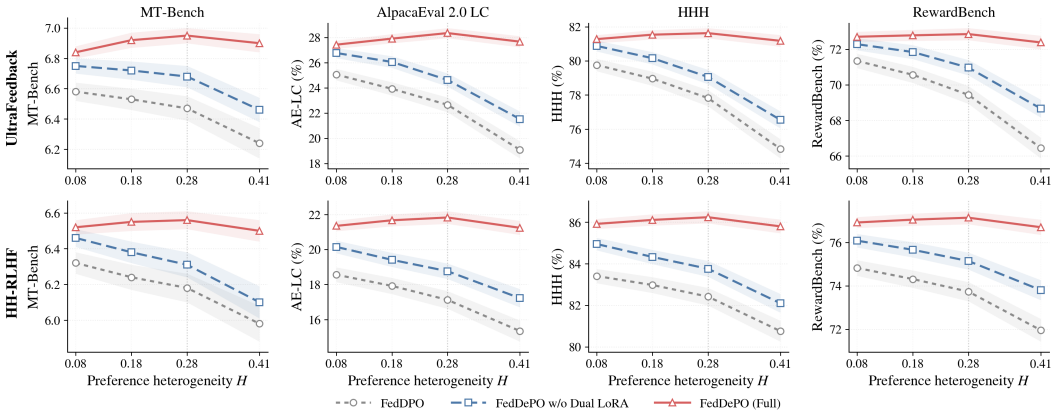

Figure 3: Effect of preference heterogeneity on Qwen3-1.7B across four alignment benchmarks. Curves report the mean and shaded bands cover ± 1 standard deviation.

of unstable supervision and preference conflict. Their operations remain coupled through the filtered training pairs and evolving policy. Independent label-agreement and controlled training-state analyses appear in Appendices C.14 and C.15.

Beyond stylistic personas. Appendix C.12 evaluates safety-first versus information-freedom preferences, and Appendix C.13 uses native country-level survey distributions from GlobalOpinionQA. On GlobalOpinionQA, FedDePO reduces mean JSD from 0.398 for FedDPO to 0.213 under balanced training. Its advantage persists under imbalanced training. These human survey labels provide an evaluation independent of the self-rewarding judge. Appendices C.16 and C.17 further report personal-adapter adaptation and dispersion across personas. Two local adaptation rounds reduce the seen-to-unseen gap by over 70% on both datasets, supporting rapid personalization while the shared adapter remains frozen.

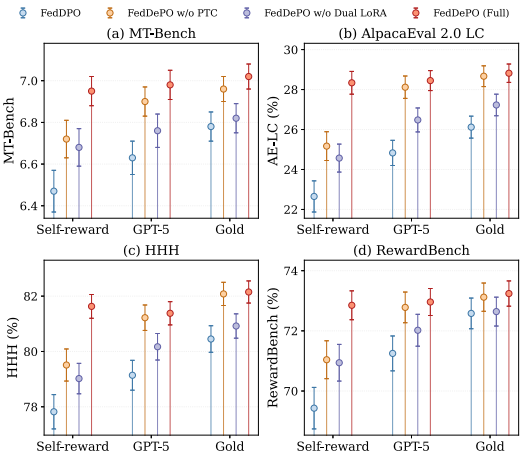

Figure 4: Module contribution under three Judge conditions (self-reward, external GPT-5, gold annotations), at $H \approx 0.28$. Error bars denote ± 1 standard deviation.

Shared and personal alignment. Appendix C.11 distinguishes shared-adapter quality from the benefit of personalization. With the personal adapter disabled, FedDePO still exceeds FedDPO by 1.78 Avg points on UltraFeedback and 1.77 on HH-RLHF. Restoring the personal adapter adds 1.65 and 1.89 points, respectively. These comparisons show that the gains include improvements to both the shared policy and the client-specific model. The rank-32 single-adapter control remains below FedDePO despite communicating twice its shared rank, further supporting the role of routing in how adaptation capacity is used.

5 Conclusion

We introduced FedDePO for federated online alignment under conflicting preferences. Reward-space routing directs consensus pairs to a shared adapter and contested pairs to local personal adapters, supported by consistency filtering and drift control. Experiments across model scales and datasets, together with matched-capacity controls and native human preference evaluation, support this decomposition. The resulting protocol improves shared and personalized alignment while keeping raw data local. Personalization adds local capacity without increasing the rank communicated for aggregation.

AI USE STATEMENT

Generative AI tools assisted with language polishing and literature lookup. The authors take responsibility for the final manuscript.

ETHICS STATEMENT

FedDePO keeps private prompts, generated responses, and personal adapters on clients. Shared adapter updates and public-probe margins are communicated to the server and may reveal client preferences; the evaluated protocol does not provide differential privacy. GlobalOpinionQA uses released aggregate survey distributions. Country-level distributions describe population aggregates and should not be treated as preferences shared by every individual. Further discussion of privacy and societal impacts appears in Appendices E and H.

REPRODUCIBILITY STATEMENT

Section 4.1 specifies the datasets, models, metrics, and federated setting. Appendices D and E give the data partition, probe construction, optimization settings, and comparison protocol. Appendix F provides the algorithms. Appendix B states the analytical assumptions and derivations. Additional experimental protocols and results appear in Appendix C.

REFERENCES

Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. Federated Learning with Personalization Layers. *arXiv preprint arXiv:1912.00818*, 2019. URL https://arxiv.org/abs/1912.00818.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Rémi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238, pp. 4447–4455. PMLR, 2024. URL https://proceedings.mlr.press/v238/gheshlaghi-azar24a.html.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback. *arXiv preprint arXiv:2204.05862*, 2022. URL https://arxiv.org/abs/2204.05862.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/d5e2c0adad503c91f91df240d0cd4e49-Abstract.html.

Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 2089–2099. PMLR, 2021. URL https://proceedings.mlr.press/v139/collins21a.html.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. UltraFeedback: Boosting Language Models with Scaled AI Feedback. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 9722–9744, 2024. URL https://proceedings.mlr.press/v235/cui24f.html.

Canh T. Dinh, Nguyen H. Tran, and Tuan Dung Nguyen. Personalized federated learning with moreau envelopes. In *Advances in Neural Information Processing Systems*, volume 33, pp. 21394–21405, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/f4f1f13c8289ac1b1ee0ff176b56fc60-Paper.pdf.

Hanze Dong, Wei Xiong, Bo Pang, Haoxiang Wang, Han Zhao, Yingbo Zhou, Nan Jiang, Doyen Sahoo, Caiming Xiong, and Tong Zhang. RLHF Workflow: From Reward Modeling to Online RLHF. *arXiv preprint arXiv:2405.07863*, 2024. URL https://arxiv.org/abs/2405.07863.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. Length-Controlled AlpacaEval: A Simple Way to Debias Automatic Evaluators. In *First Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2404.04475.

Esin Durmus, Karina Nguyen, Thomas I. Liao, Nicholas Schiefer, Amanda Askell, Anton Bakhtin, Carol Chen, Zac Hatfield-Dodds, Danny Hernandez, Nicholas Joseph, Liane Lovitt, Sam McCandlish, Orowa Sikder, Alex Tamkin, Janel Thamkul, Jared Kaplan, Jack Clark, and Deep Ganguli. Towards Measuring the Representation of Subjective Global Opinions in Language Models. *arXiv preprint arXiv:2306.16388*, 2023. URL https://arxiv.org/abs/2306.16388.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Model alignment as prospect theoretic optimization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 12634–12651, 2024. URL https://proceedings.mlr.press/v235/ethayarajh24a.html.

Paul Gölz, Nika Haghtalab, and Kunhe Yang. Distortion of AI Alignment: Does Preference Optimization Optimize for Preferences? In *Advances in Neural Information Processing Systems*, volume 38, pp. 25969–26007, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/2526c5e8110bc6bc8b462ba95198161e-Abstract-Conference.html.

Pengxin Guo, Shuang Zeng, Yanran Wang, Huijie Fan, Feifei Wang, and Liangqiong Qu. Selective aggregation for low-rank adaptation in federated learning. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/f53a37f820d5be5930415d964f4a0187-Abstract-Conference.html.

Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Ramé, Thomas Mesnard, Yao Zhao, Bilal Piot, Johan Ferret, and Mathieu Blondel. Direct Language Model Alignment from Online AI Feedback. *arXiv preprint arXiv:2402.04792*, 2024. URL https://arxiv.org/abs/2402.04792.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2106.09685.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 5132–5143. PMLR, 2020. URL https://proceedings.mlr.press/v119/karimireddy20a.html.

Weirui Kuang, Bingchen Qian, Zitao Li, Daoyuan Chen, Dawei Gao, Xuchen Pan, Yuexiang Xie, Yaliang Li, Bolin Ding, and Jingren Zhou. FederatedScope-LLM: A comprehensive package for fine-tuning large language models in federated learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 5260–5271, 2024. URL https://arxiv.org/abs/2309.00363.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems*, volume 2, pp. 429–450, 2020. URL https://proceedings.mlsys.org/paper_files/paper/2020/hash/1f5fe83998a09396ebe6477d9475ba0c-Abstract.html.

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *Proceedings of the 38th International Conference on Machine*

Learning, volume 139, pp. 6357–6368. PMLR, 2021. URL https://proceedings.mlr.press/v139/li21h.html.

Yao Lu, Jiaxing Qi, Zhongzhi Luan, Shaohan Huang, Carol Fung, Hailong Yang, and Depei Qian. FDLoRA: Personalized Federated Learning of Large Language Model via Dual LoRA Tuning. *arXiv preprint arXiv:2406.07925*, 2024. URL https://arxiv.org/abs/2406.07925.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54, pp. 1273–1282. PMLR, 2017. URL https://proceedings.mlr.press/v54/mcmahan17a.html.

Yu Meng, Mengzhou Xia, and Danqi Chen. SimPO: Simple preference optimization with a reference-free reward. In *Advances in Neural Information Processing Systems*, volume 37, pp. 124198–124235, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/e099c1c9699814af0be873a175361713-Abstract-Conference.html.

Meta. Llama 3.2-3B-Instruct model card, 2024. URL https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct.

Mistral AI. Mistral-7B-Instruct-v0.3 model card, 2024. URL https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3.

Fatemeh Nourzad, Amirhossein Roknilamouki, Eylem Ekici, Jia Liu, and Ness Shroff. FIRM: Federated In-client Regularized Multi-objective Alignment for Large Language Models. *arXiv preprint arXiv:2511.16992*, 2025. URL https://arxiv.org/abs/2511.16992.

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://arxiv.org/abs/2106.06042.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022. URL https://proceedings.neurips.cc/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html.

Chanwoo Park, Mingyang Liu, Dingwen Kong, Kaiqing Zhang, and Asuman Ozdaglar. RLHF from Heterogeneous Feedback via Personalization and Preference Aggregation. *arXiv preprint arXiv:2405.00254*, 2024. URL https://arxiv.org/abs/2405.00254.

Sriyash Poddar, Yanming Wan, Hamish Ivison, Abhishek Gupta, and Natasha Jaques. Personalizing reinforcement learning from human feedback with variational preference learning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 52516–52544, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/5e1c255653eb98cef13f45b2d337c882-Abstract-Conference.html.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. In *Advances in Neural Information Processing Systems*, volume 36, pp. 53728–53741, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/a85b405ed65c6477a4fe8302b5e06ce7-Paper-Conference.pdf.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms. *arXiv preprint arXiv:1707.06347*, 2017. URL https://arxiv.org/abs/1707.06347.

Ali Shirali, Arash Nasr-Esfahany, Abdullah Alomar, Parsa Mirtaheri, Rediet Abebe, and Ariel Procaccia. Direct Alignment with Heterogeneous Preferences. In *Advances in Neural Information Processing Systems*, volume 38, pp. 69858–69898, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/6537833d11b4bc2f2cd150780b3b168c-Abstract-Conference.html.

Mahmoud Srewa, Tianyu Zhao, and Salma Elmalaki. PluralLLM: Pluralistic alignment in LLMs via federated learning. In *Proceedings of the 3rd International Workshop on Human-Centered Sensing, Modeling, and Intelligent Systems*, pp. 64–69, 2025. URL https://arxiv.org/abs/2503.09925.

Youbang Sun, Zitao Li, Yaliang Li, and Bolin Ding. Improving LoRA in privacy-preserving federated learning. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/4e243e95c913b367775d71d7182b99d9-Abstract-Conference.html.

Fahim Tajwar, Anikait Singh, Archit Sharma, Rafael Rafailov, Jeff Schneider, Tengyang Xie, Stefano Ermon, Chelsea Finn, and Aviral Kumar. Preference fine-tuning of LLMs should leverage suboptimal, on-policy data. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 47441–47474, 2024. URL https://proceedings.mlr.press/v235/tajwar24a.html.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open Foundation and Fine-Tuned Chat Models. *arXiv preprint arXiv:2307.09288*, 2023. URL https://arxiv.org/abs/2307.09288.

Feijie Wu, Xiaoze Liu, Haoyu Wang, Xingchen Wang, Lu Su, and Jing Gao. Towards Federated RLHF with Aggregated Client Preference for LLMs. In *International Conference on Learning Representations*, 2025a. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/124dde499d62b58e97e42a45b26d7369-Abstract-Conference.html.

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason E. Weston, and Sainbayar Sukhbaatar. Meta-Rewarding Language Models: Self-Improving Alignment with LLM-as-a-Meta-Judge. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 11537–11554. Association for Computational Linguistics, 2025b. doi: 10.18653/v1/2025.emnlp-main.583. URL https://aclanthology.org/2025.emnlp-main.583/.

Wei Xiong, Hanze Dong, Chenlu Ye, Ziqi Wang, Han Zhong, Heng Ji, Nan Jiang, and Tong Zhang. Iterative preference learning from human feedback: Bridging theory and practice for RLHF under KL-constraint. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 54715–54754, 2024. URL https://proceedings.mlr.press/v235/xiong24a.html.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Yiyuan Yang, Guodong Long, Tao Shen, Jing Jiang, and Michael Blumenstein. Dual-personalizing adapter for federated foundation models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 39409–39433, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/45a30141c6719e9cfedfb51f1c665a37-Abstract-Conference.html.

Rui Ye, Wenhao Wang, Jingyi Chai, Dihan Li, Zexi Li, Yinda Xu, Yaxin Du, Yanfeng Wang, and Siheng Chen. OpenFedLLM: Training large language models on decentralized private data via federated learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 6137–6147, 2024. URL https://arxiv.org/abs/2402.06954.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason E Weston. Self-Rewarding Language Models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 57905–57923, 2024. URL https://proceedings.mlr.press/v235/yuan24d.html.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In

Advances in Neural Information Processing Systems, volume 36, pp. 46595–46623, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91f18a1287b398d378ef22505bf41832-Abstract-Datasets_and_Benchmarks.html.

APPENDIX

We organize the appendix as follows.

Background and theoretical support.

- Appendix A reviews prior work on federated preference optimization, online alignment, and personalization under heterogeneous preferences.
- Appendix B provides modular theoretical support for FedDePO, including concentration of the transitivity violation rate (B.1), IRD conflict detection under estimation error (B.2), the idealized utility benefits of preference decomposition (B.3), and conditional analysis of the shared update under adaptive local stopping (B.4).

Extended empirical results.

- Appendix C reports additional experiments on training dynamics (Appendix C.1), cross-architecture validation (Appendix C.2), sensitivity to data heterogeneity (Appendix C.3) and Judge source (Appendix C.4), held-out persona transfer (Appendix C.5), comparisons with alternative Judge filters (Appendix C.6) and personalized federated learning baselines (Appendix C.7), client scaling and participation (Appendix C.8), communication and computation overhead (Appendix C.9), and hyperparameter robustness (Appendix C.10).
- The same section continues with capacity-matched routing controls (Appendix C.11), polarized value preferences (Appendix C.12), native human preference evaluation (Appendix C.13), independent PTC and IRD validation (Appendices C.14 and C.15), cold-start adaptation (Appendix C.16), population-imbalance dispersion (Appendix C.17), and response-budget analysis (Appendix C.18).

Data, implementation, and algorithmic details.

- Appendix D describes the construction of client partitions and controlled preference heterogeneity.
- Appendix E details implementation choices, probe-set construction, baseline tuning, the protocol used for fair comparison, and a privacy discussion of the uploaded IRD statistics.
- Appendix F presents the full FedDePO pseudocode, with the probe-anchor routing procedure detailed in Appendix E.
- Appendix G discusses practical limitations and scope.
- Appendix H briefly discusses broader societal impacts.

A Related Work

Reinforcement Learning and Preference Optimization. Aligning large language models with human preferences typically relies on reinforcement learning from human feedback (Ouyang et al., 2022; Christiano et al., 2017; Bai et al., 2022), where a reward model trained on pairwise comparisons guides policy optimization via PPO (Schulman et al., 2017). Direct preference optimization (Rafailov et al., 2023) replaces this two-phase pipeline with a single classification loss that implicitly encodes the reward inside the policy, and subsequent variants further relax its modeling assumptions (Azar et al., 2024; Ethayarajh et al., 2024; Meng et al., 2024). A well-documented limitation of training on a fixed offline dataset is that the learned policy gradually drifts away from the data distribution, causing performance degradation (Guo et al., 2024; Tajwar et al., 2024). Online iterative DPO addresses this by regenerating on-policy responses at each iteration and collecting fresh preference labels (Xiong et al., 2024; Dong et al., 2024), with the self-rewarding variant (Yuan et al., 2024) and its meta-rewarding extension (Wu et al., 2025b) further removing the need for an external reward model by letting the LLM serve as its own judge. Our work embeds this online self-rewarding loop into a federated framework so that the iterative improvement cycle can operate under data isolation constraints.

Federated Large Language Model Alignment. Federated learning enables collaborative model training without exposing private local data (McMahan et al., 2017; Li et al., 2020; Karimireddy et al., 2020). Early federated LLM efforts focus on supervised fine-tuning and static preference optimization: OpenFedLLM (Ye et al., 2024) introduces FedDPO by applying offline DPO within FedAvg, and FederatedScope-LLM (Kuang et al., 2024) provides a modular toolkit for federated parameter-efficient fine-tuning. More recent work tackles the preference dimension directly. FedBiscuit (Wu et al., 2025a) encodes client preferences via binary selectors and uses clustered aggregation to handle heterogeneity; FIRM (Nourzad et al., 2025) balances multiple alignment objectives through in-client regularization; PluralLLM (Srewa et al., 2025) trains per-group reward modules as distributed preference generators under federated PPO. These methods differ in how preferences are represented and aggregated. Our focus is an online DPO loop whose shared-versus-personal training partition is recomputed from implicit-reward margins. A parallel line of theoretical work reveals a more fundamental barrier: Gölz et al. (2025) prove that single-policy DPO under heterogeneous preferences incurs an irreducible distortion that grows with preference divergence, and Shirali et al. (2025) show that the standard DPO closed-form linkage fails to express the true likelihood when preferences are mixed. Centralized studies on heterogeneous RLHF propose clustering (Park et al., 2024) or variational inference over latent user types (Poddar et al., 2024), but neither has been adapted to a federated protocol. Our framework couples online iterative DPO with reward-space routing between shared and personal preference updates.

Personalized Federated Learning and Dual Adapter Architecture. Personalized federated learning resolves data heterogeneity by maintaining client-specific parameters alongside a shared global model. Classical approaches include proximal regularization (Li et al., 2020), Moreau-envelope personalization (Dinh et al., 2020), regularized personal copies (Li et al., 2021), and shared-representation methods that separate base layers from personal heads (Arivazhagan et al., 2019; Collins et al., 2021). Adapting these ideas to large language models naturally leads to LoRA-based designs (Hu et al., 2022). FFA-LoRA (Sun et al., 2024) freezes the randomly initialized A matrices and aggregates only B matrices to halve communication cost and stabilize training under differential privacy. FedSA-LoRA (Guo et al., 2025) reverses this choice by showing that A matrices capture general knowledge while B matrices encode client-specific patterns, and accordingly shares only A for aggregation. FDLoRA (Lu et al., 2024) maintains separate global and personal LoRA modules on each client, uploading only the global branch and fusing both at inference via an adaptive optimizer. FedDPA (Yang et al., 2024) similarly deploys a global and a local adapter but targets test-time distribution shifts rather than preference heterogeneity, weighting the two via an instance-wise mechanism during inference. A common thread in all these methods is that the partition between shared and personal parameters is determined by the *architectural* structure of the adapter, whether by matrix role, network depth, or fixed functional assignment, without reference to the semantic content being learned. Our Dual LoRA design departs from this paradigm: the consensus mask produced by Implicit Reward Divergence partitions preference gradients in *reward semantic space*, so the shared adapter learns preference pairs classified as consensus while the personal adapter absorbs contested ones, and the partition is re-estimated each round as the preference landscape evolves.

B Theoretical Analysis

We analyze TVR concentration, the estimation error in IRD, the potential benefit of personalized policies, and the shared update under adaptive local stopping. The first two results concern diagnostics; the last two use explicitly specified optimization models. We use $\beta > 0$ for the KL regularization coefficient, consistently with Eq. 1. Prompt arguments are suppressed where convenient.

B.1 Concentration of the Transitivity Violation Rate

Assumption 1 (Conditional judging model)**.** *Fix a prompt x, its $K \geq 3$ candidate responses, and the judge parameters. The random judging outputs associated with distinct unordered response pairs are independent. The two directional evaluations of the same pair may be dependent and are treated as one block. Each triple has probability at most $\eta_i(x)$ of forming a directed cycle.*

Proposition 1 (TVR concentration)**.** *Under Assumption 1, for any $\delta \in (0, 1)$, with conditional probability at least $1 - \delta$,*

$$|\mathrm{TVR}(x) - \mathbb{E}[\mathrm{TVR}(x)]| \leq 3\sqrt{\frac{\log(2/\delta)}{K(K-1)}}, \qquad \mathbb{E}[\mathrm{TVR}(x)] \leq \eta_i(x). \tag{18}$$

Proof. Write $\mathrm{TVR}(x) = \binom{K}{3}^{-1} \sum_t X_t$, where X_t is the cycle indicator for triple t. Changing the judging block for one pair affects at most $K - 2$ indicators. The bounded-difference constant for each of the $\binom{K}{2}$ independent blocks is therefore

$$c = \frac{K-2}{\binom{K}{3}} = \frac{6}{K(K-1)}, \qquad \binom{K}{2} c^2 = \frac{18}{K(K-1)}.$$

McDiarmid's inequality gives $\Pr(|\mathrm{TVR} - \mathbb{E}\mathrm{TVR}| > \epsilon) \leq 2\exp\{-\epsilon^2 K(K-1)/9\}$, which yields the bound. Linearity of expectation gives the second claim without requiring independence between triple indicators. □

For $K = 8$ and $\delta = 0.1$, this general bound is approximately 0.694. It does not certify precise per-prompt reliability at this sample size. TVR measures internal consistency: a deterministic but biased ordering has zero TVR. Its usefulness as a filter is evaluated empirically, and concentration under shared judging randomness requires a different dependence model.

B.2 Implicit Reward Divergence as a Conflict Detector

For a probe $z = (x, y^{(1)}, y^{(2)})$, define the latent reward gap $d_i(z) = r_i^*(x, y^{(1)}) - r_i^*(x, y^{(2)})$ and the observed implicit margin $m_i(z)$ from Eq. 8. All clients use the same reference policy and reward scale.

Lemma 2 (Margin error decomposition)**.** *Define $\xi_i(z) := m_i(z) - d_i(z)$. Then $m_i(z) = d_i(z) + \xi_i(z)$. If a client policy equals the exact KL-regularized optimum*

$$\pi_i^*(y \mid x) = \frac{\pi_{\mathrm{ref}}(y \mid x) \exp(r_i^*(x, y)/\beta)}{Z_i(x)}, \tag{19}$$

then $\xi_i(z) = 0$ wherever the probabilities are positive.

Proof. At this optimum, $\beta \log(\pi_i^*/\pi_{\mathrm{ref}}) = r_i^* - \beta \log Z_i(x)$. Subtracting the two response rewards cancels the prompt-dependent normalizer. For an approximate policy, ξ_i retains the resulting approximation and estimation error. □

Proposition 3 (IRD with persistent estimation error)**.** *Condition on the trained client policies. Suppose the probes $z_1, \ldots, z_M$ are i.i.d. from a distribution Q and $d_i, d_k, \xi_i, \xi_k \in L^2(Q)$. Set*

$$\Delta_{ik}^* = \mathbb{E}_Q[(d_i - d_k)^2], \qquad \epsilon_{ik} = \|\xi_i - \xi_k\|_{L^2(Q)}.$$

Then, almost surely as $M \to \infty$,

$$\mathrm{IRD}(i,k) := \frac{1}{M}\sum_{j=1}^{M}(m_i(z_j) - m_k(z_j))^2 \longrightarrow D_{ik} := \mathbb{E}_Q[(m_i - m_k)^2], \tag{20}$$

and

$$\left|\sqrt{D_{ik}} - \sqrt{\Delta^*_{ik}}\right| \le \epsilon_{ik}, \qquad |D_{ik} - \Delta^*_{ik}| \le 2\sqrt{\Delta^*_{ik}}\,\epsilon_{ik} + \epsilon^2_{ik}. \tag{21}$$

Proof. The squared observed gap is integrable by the L^2 assumptions, so the strong law proves Eq. 20. The reverse triangle inequality in $L^2(Q)$ proves the first error bound. Expanding $(d_i - d_k + \xi_i - \xi_k)^2$ and applying Cauchy–Schwarz proves the second. □

Increasing the probe count reduces sampling error around D_{ik}; recovering Δ^*_{ik} additionally requires $\epsilon_{ik} \to 0$. This is a separate requirement on reward estimation, not a consequence of increasing M. For the fixed, selected probe set used in the experiments, the following deterministic statement applies without an i.i.d. assumption.

Effect on the consensus mask. For probe j, let $v^*_j = N^{-1}\sum_i (d_i(z_j) - \bar{d}_j)^2$ be the latent cross-client variance and let $q_j = N^{-1}\sum_i \xi_i(z_j)^2$. Using the population-variance convention for v_j in Eq. 9,

$$|\sqrt{v_j} - \sqrt{v^*_j}| \le \sqrt{q_j}. \tag{22}$$

Indeed, centering is an orthogonal projection in $\mathbb{R}^N$, so its operator norm is one; the reverse triangle inequality gives Eq. 22. Thresholding at τ agrees with the latent variance decision whenever $|\sqrt{v^*_j} - \sqrt{\tau}| > \sqrt{q_j}$. Thus reliable mask recovery needs both small margin error and separation from the threshold. Agreement of just one client pair is insufficient to establish consensus across all clients. The variance measures disagreement in reward gaps, including their scale, rather than only disagreement in their signs.

B.3 Idealized Benefit of Preference Decomposition

We isolate the benefit of permitting a separate policy for each client under a common KL regularizer. This policy-space calculation does not assume that additive adapter weights induce additive policy expectations.

Assumption 2 (Ideal policy space)**.** *For a fixed prompt, the response space $\mathcal{A}$ is finite and $\pi_{\mathrm{ref}}(y) > 0$ for every y. Rewards $r^*_i(y)$ are finite. Define $r^*_s(y) = N^{-1}\sum_i r^*_i(y)$ and $r^p_i(y) = r^*_i(y) - r^*_s(y)$, so $\sum_i r^p_i(y) = 0$. Each policy may range over the full probability simplex.*

The centered reward decomposition always exists; the substantive idealization is access to the full policy simplex and its exact optima. For any reward r, let

$$J_r(\pi) = \mathbb{E}_\pi[r] - \beta D_{\mathrm{KL}}(\pi\|\pi_{\mathrm{ref}}), \qquad Z(r) = \sum_y \pi_{\mathrm{ref}}(y)e^{r(y)/\beta}.$$

Write $J_i = J_{r^*_i}$ and let π_s maximize $J_{r^*_s}$ and π^*_i maximize J_i.

Lemma 4 (KL-regularized reward identity)**.** *For any policy π and $\pi_r(y) = \pi_{\mathrm{ref}}(y)e^{r(y)/\beta}/Z(r)$,*

$$J_r(\pi) = \beta \log Z(r) - \beta D_{\mathrm{KL}}(\pi\|\pi_r). \tag{23}$$

Thus π_r is the unique maximizer of J_r.

Proof. Substitute $\log \pi_r = \log \pi_{\mathrm{ref}} + r/\beta - \log Z(r)$ into the KL divergence. Nonnegativity of KL and its equality condition establish optimality and uniqueness. □

Theorem 5 (Idealized personalization gain)**.** *Under Assumption 2, the gain in average KL-regularized utility over the best common policy is exactly*

$$\begin{aligned} G &:= \frac{1}{N}\sum_i J_i(\pi^*_i) - \max_\pi \frac{1}{N}\sum_i J_i(\pi) \\ &= \beta\left(\frac{1}{N}\sum_i \log Z(r^*_i) - \log Z(r^*_s)\right) = \frac{\beta}{N}\sum_i D_{\mathrm{KL}}(\pi_s\|\pi^*_i) \ge 0. \end{aligned} \tag{24}$$

The gain is zero if and only if every r_i^p is constant over responses. If client policies $\widetilde{\pi}_i$ satisfy $J_i(\pi_i^*) - J_i(\widetilde{\pi}_i) \le \varepsilon_i$, then

$$\frac{1}{N}\sum_i J_i(\widetilde{\pi}_i) - \frac{1}{N}\sum_i J_i(\pi_s) \ge G - \frac{1}{N}\sum_i \varepsilon_i. \tag{25}$$

Proof. Linearity of the reward expectation gives $N^{-1}\sum_i J_i(\pi) = J_{r_s^*}(\pi)$ for a common policy π. Lemma 4 therefore identifies its optimum as π_s and its value as $\beta \log Z(r_s^*)$. Applying Eq. 23 separately to each J_i at π_s gives $J_i(\pi_i^*) - J_i(\pi_s) = \beta D_{\mathrm{KL}}(\pi_s \| \pi_i^*)$. Averaging proves Eq. 24. Because the reference has full support, equality holds precisely when $\pi_i^* = \pi_s$ for all i, equivalently when $r_i^* - r_s^*$ is response-independent. Finally, subtract the average suboptimality of $\widetilde{\pi}_i$ from G. □

The theorem identifies the policy-space opportunity that motivates personalization. It does not assert that finite-rank Dual LoRA, routed DPO losses, or FedDPO attain these optima. It also does not give a raw-utility improvement or a universal gain proportional to personal reward variance: response-independent reward offsets can have positive variance across clients while leaving every optimal policy unchanged. The empirical comparisons assess the implemented method.

B.4 Conditional Analysis of the Shared Update

We first analyze a fixed shared-loss surrogate. Fix client weights $w_i > 0$, $\sum_i w_i = 1$, consensus data, and the reference policy, and define $F(\theta) = \sum_i w_i F_i(\theta)$. Here θ concatenates the shared LoRA factors that are actually averaged. Consider full participation and local SGD,

$$\theta_{i,e+1} = \theta_{i,e} - \eta g_{i,e}, \quad \theta_{i,0} = \theta_t, \qquad \theta_{t+1} = \theta_t - \eta d_t, \quad d_t = \sum_i w_i \sum_{e=0}^{E_i^{(t)}-1} g_{i,e}. \tag{26}$$

The random stopping times satisfy $0 \le E_i^{(t)} \le E_{\max}$. Let $\mathcal{F}_t$ contain the state before round t and write $\mathbb{E}_t = \mathbb{E}[\,\cdot \mid \mathcal{F}_t]$. These assumptions describe an SGD surrogate; the experiments use AdamW, refresh data and the reference, and train a subset of clients each round.

Assumption 3 (Smooth fixed objective)**.** *Each F_i has an L_g-Lipschitz gradient and $F \ge F_*$.*

Assumption 4 (Predictable stopping and gradient noise)**.** *The decision to execute step e is measurable before drawing its stochastic gradient. Given that pre-step history, $g_{i,e}$ has mean $\nabla F_i(\theta_{i,e})$ and noise second moment at most σ^2. The stopped updates have finite second moments.*

Assumption 5 (Residual heterogeneity)**.** *At the global iterates, $\sum_i w_i \|\nabla F_i(\theta_t) - \nabla F(\theta_t)\|^2 \le \zeta^2$.*

Assumption 6 (Local drift moment)**.** *Let $a_t = \sum_i w_i \mathbb{E}_t[E_i^{(t)}] > 0$. The normalized local drift satisfies*

$$\frac{1}{a_t}\sum_i w_i \mathbb{E}_t\left[\sum_{e<E_i^{(t)}} \|\theta_{i,e} - \theta_t\|^2\right] \le \rho_t^2. \tag{27}$$

Assumption 6 is a condition in parameter space, not an implication of the policy KL threshold. Neural parameterizations need not be identifiable. A parameter bound would require an additional local inequality such as $D_{\mathrm{KL}}(\pi_\theta \| \pi_{\theta_t}) \ge c\|\theta - \theta_t\|^2$ with $c > 0$. Moreover, the post-update stopping test in Eq. 14 can admit a threshold-crossing final step. Its actual drift, including any overshoot, must be accounted for.

Define $g_t = \nabla F(\theta_t)$ and the normalized update bias and variance

$$b_t = \frac{\mathbb{E}_t[d_t]}{a_t} - g_t, \qquad V_t = \frac{\mathbb{E}_t\|d_t - \mathbb{E}_t[d_t]\|^2}{a_t^2}, \qquad h_t = \eta a_t. \tag{28}$$

Lemma 6 (Conditional one-round descent)**.** *Under Assumption 3, if $h_t \le 1/L_g$, then*

$$\mathbb{E}_t[F(\theta_{t+1})] \le F(\theta_t) - \frac{h_t}{2}\|g_t\|^2 + \frac{h_t}{2}\|b_t\|^2 + \frac{L_g h_t^2}{2} V_t. \tag{29}$$

Under Assumptions 4–6, set $c_{i,t} = w_i \mathbb{E}_t[E_i^{(t)}]/a_t$ and $\chi_t^2 = \sum_i (c_{i,t} - w_i)^2 / w_i$. Then

$$\|b_t\|^2 \le 2\chi_t^2 \zeta^2 + 2L_g^2 \rho_t^2. \tag{30}$$

Proof. Smoothness and Eq. 28 give

$$\mathbb{E}_t[F(\theta_{t+1})] \le F(\theta_t) - h_t\langle g_t, g_t + b_t\rangle + \frac{L_g h_t^2}{2}\big(\|g_t + b_t\|^2 + V_t\big).$$

Use $2\langle g_t, g_t + b_t\rangle = \|g_t\|^2 + \|g_t + b_t\|^2 - \|b_t\|^2$ and $L_g h_t \le 1$ to obtain Eq. 29.

Predictable stopping and conditional unbiasedness give zero mean for each included gradient-noise increment. Consequently,

$$b_t = \sum_i (c_{i,t} - w_i)(\nabla F_i(\theta_t) - g_t) + R_t, \quad R_t = \frac{1}{a_t}\sum_i w_i \mathbb{E}_t \sum_{e<E_i^{(t)}} \big(\nabla F_i(\theta_{i,e}) - \nabla F_i(\theta_t)\big).$$

Weighted Cauchy–Schwarz bounds the squared norm of the first term by $\chi_t^2\zeta^2$. Jensen's inequality over the normalized expected local-step weights, followed by smoothness and Assumption 6, gives $\|R_t\|^2 \le L_g^2\rho_t^2$. Applying $\|u+v\|^2 \le 2\|u\|^2 + 2\|v\|^2$ proves Eq. 30. □

Theorem 7 (Fixed-surrogate stationarity bound)**.** *Under Lemma 6, the iterates satisfy*

$$\sum_{t<T} \mathbb{E}[h_t\|g_t\|^2] \le 2(F(\theta_0) - F_*) + \sum_{t<T} \mathbb{E}[h_t\|b_t\|^2 + L_g h_t^2 V_t]. \tag{31}$$

In particular, if $a_t \ge a_{\min} > 0$, $\|b_t\|^2 \le B^2$, $V_t \le V$, and $\eta E_{\max} \le 1/L_g$, then

$$\frac{1}{T}\sum_{t<T} \mathbb{E}\|g_t\|^2 \le \frac{2(F(\theta_0) - F_*)}{\eta a_{\min} T} + \frac{E_{\max}}{a_{\min}} B^2 + \frac{L_g \eta E_{\max}^2}{a_{\min}} V. \tag{32}$$

Proof. Take total expectations in Eq. 29, sum over rounds, and use $F(\theta_T) \ge F_*$. For the second display, use $\eta a_{\min} \le h_t \le \eta E_{\max}$ and the assumed moment bounds. □

The descent coefficient depends on the actual expected number of executed steps a_t, so shorter local training reduces both drift and available progress. Unequal stopping times also change the effective client weights through χ_t. Neither a smaller ρ_t nor a smaller ζ is guaranteed by filtering alone. For the special case of one local step and independent client gradients, $b_t = 0$ and $V_t \le \sigma^2 \sum_i w_i^2$; uniform weights give σ^2/N. Adaptive multistep updates need their own variance control. Finally, applying a telescoping argument to the full online protocol would require accounting for changing objectives, reference updates, client sampling, and adaptive optimization. The theorem establishes the conditional optimization terms that PTA and decomposition are intended to influence.

C Additional Experiments

This appendix examines training dynamics, robustness, personalization, and computational cost. It also reports capacity-matched controls, independent validation of PTC and IRD, and native human preference evaluation. Unless stated otherwise, the default configuration follows the main paper with Qwen3-1.7B, UltraFeedback, $N = 5$ (3 training clients per round), 20 rounds, $\kappa = 0.1$, $H \approx 0.28$, $r_s = 16$, $r_p = 8$, $\beta = 0.1$, $\lambda = 10^{-3}$, $\gamma_g = 0.05$, $\gamma_t = 0.03$, $\alpha = 0.95$, and $|\mathcal{P}| = 256$. Experiments use three random seeds, and reported numbers are means over these seeds. Individual subsections specify any changes to this configuration. UF and HH denote UltraFeedback and HH-RLHF.

C.1 Training Dynamics

Figure 5 traces loss, win rate, TVR, consensus ratio, and per-client personalization gap across 20 rounds. The full model dominates every panel. Panels (a)(b) show faster and deeper convergence; removing any module slows the descent, with Dual LoRA ablation being the most harmful on win rate. Panel (c) reproduces the Judge-reliability picture from the ablation study: only PTC drives TVR down meaningfully, and disabling it leaves the Judge roughly as noisy as at initialization. Panel (d) rises monotonically for the full model once IRD stabilizes around round six, while the w/o IRD variant plateaus at a lower level because its mask is driven by raw gradient variance rather than reward-space divergence. Panel (e) shows that Dual LoRA is the sole contributor to the personalization gap; ablating it collapses every client's gap to near zero regardless of the other modules. The five panels together match the responsibility map argued in Section 4.3.

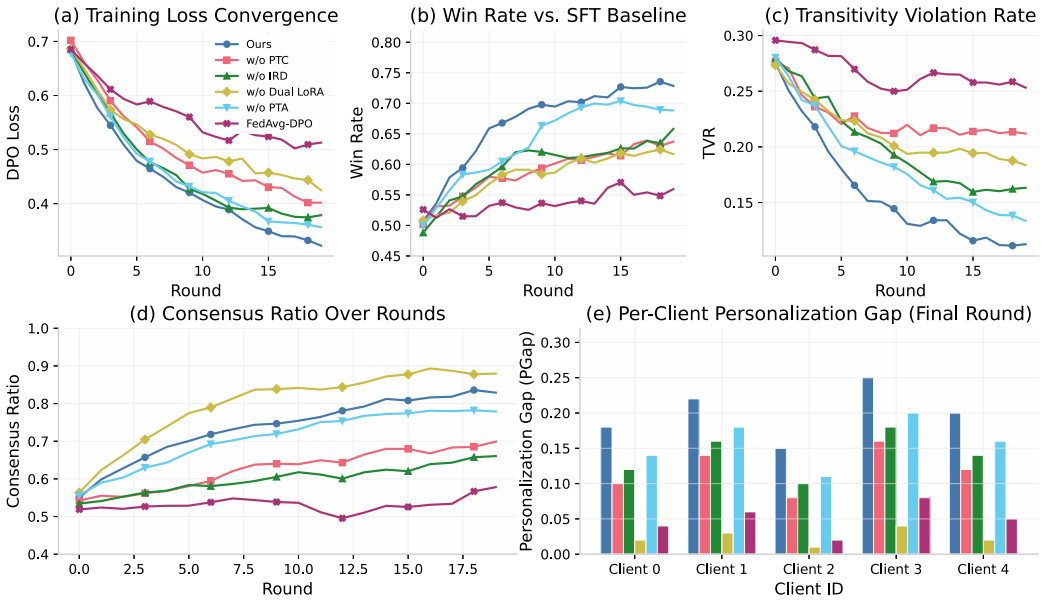

Figure 5: Training dynamics on UltraFeedback with Qwen3-1.7B. (a) DPO loss. (b) Win rate vs. SFT. (c) Judge TVR. (d) IRD consensus ratio. (e) Per-client personalization gap.

C.2 Cross-Architecture Validation

We replace Qwen3 with Llama-3.2-3B-Instruct (Meta, 2024) and Mistral-7B-Instruct-v0.3 (Mistral AI, 2024) (Table 3). The method ordering persists across both families and corpora. On UltraFeedback, the Mistral gain over FedDPO is 3.97 Avg points, using AlpacaEval, HHH, and RewardBench; Qwen3-4B gains 4.31 points. On HH-RLHF, FedDePO is within one Avg point of centralized online DPO on both backbones. These results support transfer across the evaluated architectures without implying that a larger backbone necessarily yields a larger gain.

Table 3: Cross-architecture results. Best federated number in each block in **bold**.

		UltraFeedback				HH-RLHF			
Backbone	Method	MT-Bench	AlpacaEval	HHH	RewardBench	MT-Bench	AlpacaEval	HHH	RewardBench
Llama-3.2-3B	Zero-shot	5.43	10.27	69.34	61.52	5.43	10.27	69.34	61.52
	FedAvg+SFT	5.81	15.42	72.18	64.25	5.62	11.83	75.41	67.29
	FedDPO	6.12	20.13	74.62	66.81	5.83	14.62	78.54	70.43
	FedDePO (Ours)	**6.58**	**25.47**	**77.93**	**69.48**	**6.21**	**18.74**	**82.31**	**73.32**
	Central Online DPO	6.71	26.83	78.64	70.15	6.34	19.61	82.95	73.94
Mistral-7B-v0.3	Zero-shot	6.12	14.83	73.52	65.14	6.12	14.83	73.52	65.14
	FedAvg+SFT	6.45	19.72	76.31	67.93	6.28	16.42	79.18	70.85
	FedDPO	6.78	24.65	78.84	70.52	6.49	19.31	82.42	73.81
	FedDePO (Ours)	**7.24**	**30.18**	**82.13**	**73.62**	**6.91**	**23.85**	**86.24**	**76.73**
	Central Online DPO	7.38	31.42	82.84	74.35	7.02	24.71	86.82	77.32

C.3 Data Heterogeneity Sensitivity

We fix the Judge personas and sweep the Dirichlet concentration κ (Table 4). Two trends are visible. Offline FedDPO loses over six Avg points between $\kappa = 0.05$ and the near-i.i.d. limit because its consensus gradients are increasingly drowned by client-specific label skew, while FedDePO loses under three points in the same range. The gap from FedDePO to Central Online DPO narrows from roughly 2.5 points under sharp skew to within 0.1 points as $\kappa \to 10$, which is expected once data heterogeneity becomes negligible and only the fixed preference heterogeneity remains. The w/o Dual LoRA row sits roughly midway between FedDPO and the full model at every κ, indicating that Dual LoRA absorbs most of the skew-induced gradient conflict while PTC and PTA contribute the rest.

Table 4: Data heterogeneity sweep on Qwen3-1.7B. Each cell: MT-Bench / Avg. Best federated per column in **bold**.

Dataset	Method	$\kappa = 0.05$	$\kappa = 0.10$	$\kappa = 0.30$	$\kappa = 1.00$	$\kappa = 10.0$
UltraFeedback	FedAvg+SFT	4.93 / 44.72	5.31 / 47.98	5.48 / 49.15	5.62 / 50.23	5.71 / 50.86
	FedDPO	5.12 / 47.41	5.72 / 51.03	5.93 / 52.74	6.08 / 53.82	6.14 / 54.21
	FedDePO w/o Dual LoRA	5.34 / 49.23	5.85 / 52.42	6.04 / 53.67	6.15 / 54.52	6.19 / 54.78
	FedDePO (Ours)	**5.94 / 52.86**	**6.24 / 54.46**	**6.32 / 55.12**	**6.37 / 55.41**	**6.39 / 55.53**
	Central Online DPO	6.38 / 55.37	6.38 / 55.37	6.38 / 55.37	6.38 / 55.37	6.38 / 55.37
HH-RLHF	FedAvg+SFT	4.82 / 45.83	5.08 / 49.30	5.24 / 50.72	5.36 / 51.65	5.43 / 52.14
	FedDPO	4.95 / 48.12	5.24 / 52.46	5.41 / 53.92	5.52 / 54.83	5.58 / 55.24
	FedDePO w/o Dual LoRA	5.12 / 49.85	5.38 / 53.78	5.52 / 54.84	5.60 / 55.52	5.63 / 55.81
	FedDePO (Ours)	**5.42 / 54.31**	**5.67 / 56.12**	**5.74 / 56.73**	**5.76 / 56.92**	**5.78 / 57.04**
	Central Online DPO	5.78 / 56.73	5.78 / 56.73	5.78 / 56.73	5.78 / 56.73	5.78 / 56.73

C.4 Robustness to Judge Source

We replace the local Judge with three external LLMs and with the gold annotations released by the two datasets (Table 5). FedDePO sits at the top envelope of every column across both model scales and both corpora. The PTC contribution shrinks from roughly two Avg points under Self to 0.21–0.31 points under Gold, because PTC is designed to filter Judge-produced logical inconsistencies that higher-quality annotators rarely introduce. The Dual LoRA contribution stays within a narrow band of 3.5 to 4.1 Avg points across all five Judge sources, because it targets preference conflict rather than Judge noise. Together the two trends trace a clean responsibility map: PTC scales with Judge unreliability and Dual LoRA scales with preference heterogeneity, and the two remain non-redundant under any Judge.

Table 5: Avg under different Judge sources. Self: local policy; GPT-5, Gemini-3-Pro, Claude-S4: external LLMs; Gold: released annotations.

Dataset	Model	Variant	Self	GPT-5	Gemini-3-Pro	Claude-S4	Gold
UltraFeedback	Qwen3-1.7B	FedDePO (Full)	**54.46**	**55.83**	**56.21**	**56.04**	**56.74**
		w/o PTC	52.48	54.72	55.64	55.32	56.53
		w/o IRD & Dual LoRA	50.71	52.14	52.48	52.25	53.02
		w/o PTA	53.12	54.93	55.32	55.12	55.94
	Qwen3-4B	FedDePO (Full)	**60.94**	**62.35**	**62.81**	**62.58**	**63.24**
		w/o PTC	59.12	61.47	62.13	61.82	63.02
		w/o IRD & Dual LoRA	57.28	58.74	59.13	58.92	59.61
		w/o PTA	59.83	61.62	62.04	61.84	62.58
HH-RLHF	Qwen3-1.7B	FedDePO (Full)	**56.12**	**57.41**	**57.84**	**57.63**	**58.25**
		w/o PTC	54.31	56.25	57.12	56.82	58.02
		w/o IRD & Dual LoRA	51.94	53.42	53.81	53.58	54.32
		w/o PTA	54.93	56.52	56.92	56.71	57.43
	Qwen3-4B	FedDePO (Full)	**61.74**	**63.12**	**63.58**	**63.34**	**64.03**
		w/o PTC	59.84	62.14	62.82	62.52	63.72
		w/o IRD & Dual LoRA	57.92	59.32	59.75	59.52	60.21
		w/o PTA	60.52	62.24	62.73	62.51	63.32

C.5 Held-out Persona Transfer

Following federated preference-alignment evaluation in OpenFedLLM (Ye et al., 2024) and recent work on heterogeneous preferences (Park et al., 2024; Poddar et al., 2024), we train exactly as in the main experiments on the original five Judge personas, then evaluate the final model on the same test prompts with a disjoint set of held-out personas sampled from the same attribute pool. We report only Avg on the seen-persona split, the unseen-persona split, and their gap. The purpose is to verify that the decomposition does not simply memorize the training personas.

Table 6: Held-out persona transfer. Train on the original five personas; test on seen vs. unseen personas; report Avg and the gap.

Dataset	Method	Seen Avg	Unseen Avg	Gap
UltraFeedback	FedDPO	51.03	50.24	0.79
	FedDePO w/o Dual LoRA	50.71	49.68	1.03
	FedDePO (Ours)	**54.46**	**52.93**	1.53
HH-RLHF	FedDPO	52.46	51.84	0.62
	FedDePO w/o Dual LoRA	51.94	50.82	1.12
	FedDePO (Ours)	**56.12**	**54.38**	1.74

C.6 PTC AGAINST OTHER JUDGE FILTERS

We keep this comparison minimal and vary only the filtering rule under the same Judge prompt and total Judge budget (Zheng et al., 2023; Yuan et al., 2024). For a pair $z = (x, y_w, y_l)$, define the Judge margin

$$m(z) = \left| P_j(y_w \succ y_l) - P_j(y_l \succ y_w) \right|.$$

Margin threshold. Keep z iff $m(z) \geq \delta$, where δ is tuned on the same held-out split as other hyperparameters.

Self-consistency vote. Re-judge a subsampled pair three times, obtain votes $v^{(1)}, v^{(2)}, v^{(3)} \in \{-1, +1\}$, and use

$$\hat{v}(z) = \operatorname{sign}\left(\sum_{r=1}^{3} v^{(r)} \right).$$

Under a fixed Judge budget, this baseline labels fewer unique pairs because each retained pair consumes three Judge calls. We report only the retained fraction and final Avg.

Table 7: PTC vs. simple Judge filters. All methods use the same total Judge budget.

Dataset	Filter	Retained (%)	Avg↑
UltraFeedback	Margin threshold	68.24	53.12
	Self-consistency vote	29.83	52.47
	PTC (Ours)	**57.42**	**54.46**
HH-RLHF	Margin threshold	65.73	55.04
	Self-consistency vote	28.62	54.58
	PTC (Ours)	**54.83**	**56.12**

C.7 COMPARISON WITH PERSONALIZED FEDERATED LEARNING

Generic personalized FL methods are an obvious alternative to Dual LoRA and deserve a direct comparison. We adapt six widely used schemes to the online DPO setting (Table 8): FedPer (Arivazhagan et al., 2019) (shared backbone plus personal head), FedRep (Collins et al., 2021) (shared low-rank representation plus personal classifier), FedBABU (Oh et al., 2022) (body-only federation followed by personal adaptation), Ditto (Li et al., 2021) (regularized personal copy), pFedMe (Dinh et al., 2020) (Moreau-envelope personalization), and FedProx (Li et al., 2020) (proximal regularization without personalization, as a reference). All variants share the FedDePO LoRA budget and the same online DPO loop. We report global Avg, personal Avg, AlpacaEval on mixed prompts (AE-G) and per-client persona prompts (AE-P), personalization gap PGap, and an additional worst-client Avg to quantify fairness across heterogeneous users.

Several patterns emerge. FedProx-DPO improves only slightly over offline FedDPO under the evaluated preferences. FedPer, FedRep, and FedBABU yield PGap values from 0.67 to 0.94 across the two datasets, while retaining lower global and personal Avg than FedDePO. Ditto and pFedMe perform slightly better than FedPer on personal metrics but still trail FedDePO by more than two global-Avg points and four AE-P points. The worst-client column shows a larger gap: generic

methods lose more on under-represented clients, whereas FedDePO keeps the worst client above 52 on UltraFeedback and 54 on HH-RLHF, close to its median client. The picture is consistent with our analysis in Section 3.2: generic personalized FL partitions weights along architectural boundaries, while FedDePO partitions them along the reward-semantic boundary identified by the IRD consensus mask, which is the direction where heterogeneity actually lives.

Table 8: Comparison with personalized federated learning on Qwen3-1.7B. G-Avg / P-Avg: global / personal Avg; AE-G / AE-P: AlpacaEval on mixed / per-client persona prompts; PGap = P-Avg - G-Avg; Worst: minimum per-client Avg. Best per column in **bold**.

Dataset	Method	MT-Bench	AE-G	AE-P	HHH	RewardBench	G-Avg	P-Avg (PGap)	Worst
UltraFeedback	FedDPO	5.72	17.85	18.42	70.72	64.53	51.03	51.22 (+0.19)	48.34
	FedProx-DPO	5.75	18.12	19.04	70.92	64.75	51.26	51.62 (+0.36)	48.63
	FedPer-DPO	5.82	19.42	21.85	70.14	64.52	51.36	52.17 (+0.81)	48.92
	FedBABU-DPO	5.87	19.31	22.12	70.48	64.83	51.54	52.47 (+0.93)	49.08
	FedRep-DPO	5.91	19.83	22.64	70.83	65.12	51.93	52.87 (+0.94)	49.41
	pFedMe-DPO	5.93	19.74	22.83	70.91	65.28	51.98	53.02 (+1.04)	49.52
	Ditto-DPO	5.98	20.12	23.42	71.24	65.42	52.26	53.35 (+1.09)	49.83
	FedDePO (Ours)	**6.24**	**22.61**	**27.85**	**73.42**	**67.35**	**54.46**	**56.21 (+1.75)**	**52.38**
HH-RLHF	FedDPO	5.24	12.63	13.02	75.84	68.92	52.46	52.59 (+0.13)	49.72
	FedProx-DPO	5.27	12.85	13.42	76.12	69.24	52.74	52.93 (+0.19)	50.02
	FedPer-DPO	5.32	13.82	16.14	76.42	69.83	53.36	54.03 (+0.67)	50.58
	FedBABU-DPO	5.38	13.93	16.52	76.84	70.12	53.63	54.39 (+0.76)	50.83
	FedRep-DPO	5.41	14.05	16.82	77.02	70.24	53.77	54.65 (+0.88)	51.02
	pFedMe-DPO	5.43	14.12	16.95	77.18	70.41	53.90	54.78 (+0.88)	51.14
	Ditto-DPO	5.48	14.38	17.52	77.52	70.62	54.17	55.14 (+0.97)	51.43
	FedDePO (Ours)	**5.67**	**16.32**	**21.84**	**79.63**	**72.41**	**56.12**	**58.07 (+1.95)**	**54.21**

C.8 Client Scaling and Participation

Table 9 varies the total client count N and training participation rate ρ. Training clients are the clients whose shared updates enter aggregation, so the sweep contributes ρN clients per round. Increasing N from 5 to 100 raises FedDePO Avg by 1.12 points, compared with 1.14 for FedAvg+SFT and 1.18 for FedDPO. Thus, its advantage is higher attained performance, not a larger absolute scaling gain. At $N = 10$, the lead over FedDPO grows from 3.39 points at $\rho = 1.0$ to 5.09 points at $\rho = 0.2$, where two clients participate in each round. All clients still upload fresh IRD margins every round. These results concern reduced training participation with at least two training clients per round and full diagnostic participation. They do not test clients that are unavailable for probe evaluation.

Table 9: Client scaling and training participation on Qwen3-1.7B / UltraFeedback. Training clients are the clients whose shared updates enter aggregation. Upper: varying N with $\rho = 0.6$. Lower: varying ρ with $N = 10$. Each cell: MT-Bench / Avg.

Method	Total clients N with $\rho = 0.6$				
	$N = 5$	$N = 10$	$N = 20$	$N = 50$	$N = 100$
FedAvg+SFT	5.31 / 47.98	5.38 / 48.52	5.42 / 48.81	5.45 / 49.03	5.46 / 49.12
FedDPO	5.72 / 51.03	5.81 / 51.62	5.86 / 51.93	5.89 / 52.12	5.90 / 52.21
FedDePO w/o Dual LoRA	5.85 / 52.42	5.92 / 52.91	5.96 / 53.18	5.98 / 53.32	5.99 / 53.41
FedDePO (Ours)	**6.24 / 54.46**	**6.31 / 55.02**	**6.36 / 55.34**	**6.39 / 55.52**	**6.40 / 55.58**

Method	Participation ρ with $N = 10$			
	$\rho = 0.2$	$\rho = 0.4$	$\rho = 0.6$	$\rho = 1.0$
FedAvg+SFT	5.25 / 46.21	5.32 / 47.83	5.38 / 48.52	5.39 / 48.74
FedDPO	5.54 / 48.12	5.72 / 50.53	5.81 / 51.62	5.84 / 51.93
FedDePO w/o Dual LoRA	5.74 / 50.12	5.85 / 52.04	5.92 / 52.91	5.94 / 53.21
FedDePO (Ours)	**6.18 / 53.21**	**6.26 / 54.42**	**6.31 / 55.02**	**6.34 / 55.32**

C.9 Communication and Computation Overhead

Online preference optimization costs more per round than offline training. Each training client generates on-policy responses, runs bidirectional self-judging on $\binom{K}{2}$ pairs, and evaluates the probe set before the DPO update. Table 10 decomposes the per-round wall-clock into these four phases and reports total time to reach two quality targets. FedDePO costs roughly $1.6\times$ as much as offline FedDPO per round, dominated by rollout and judging rather than by the update itself. This overhead is offset by two factors on the critical path. First, FedDePO reaches Avg$=51$ in 76.2 fewer minutes of wall-clock time than offline FedDPO. Second, higher-quality targets such as Avg≥ 53 are unreachable by offline methods within the 20-round budget regardless of how much time is spent. The shared-adapter payload is comparable, and each 256-dimensional margin vector contains about 1 KB in float32. Every client sends this diagnostic payload each round, while only training clients upload a shared adapter. The personal adapter remains local.

Table 10: Per-client cost on Qwen3-1.7B / UltraFeedback. Rollout/Judge/Probe/Update: per-round wall-clock of each phase on an A100-80GB. Time→T: cumulative time to reach Avg=T. Dash: target not reached in 20 rounds.

Method	Rollout (min)	Judge (min)	Probe (min)	Update (min)	Per-rd (min)	Time→51 (min)	Time→53 (min)	Upload (MB)
FedAvg+SFT	–	–	–	14.23	14.23	–	–	17.82
FedDPO	–	–	–	16.81	16.81	235.3	–	17.82
FedDePO w/o PTC	2.84	5.12	0.62	16.84	25.42	178.0	406.7	17.84
FedDePO w/o Dual LoRA	2.84	5.28	–	16.82	24.94	174.6	399.0	17.82
FedDePO w/o PTA	2.84	5.25	0.62	18.83	27.54	192.8	440.6	17.84
FedDePO (Ours)	2.84	5.18	0.62	17.88	26.52	**159.1**	**291.7**	17.84
Central Online DPO	2.84	5.14	–	67.66	75.64	302.6	453.8	0

C.10 Hyperparameter Studies

LoRA ranks. Performance saturates beyond $r_s = 16$ and the best r_p stays at or below $r_s/2$, consistent with our design assumption that personal deviations occupy a lower-dimensional subspace than shared ones (Table 11). Two specific cells are informative. The inverted setting $(r_s = 4, r_p = 16)$ inflates the personal branch beyond the consensus branch and drops 3.34 Avg points below the default, because a narrow consensus adapter cannot absorb enough shared signal to be filtered by IRD. Conversely, $(r_s = 64, r_p = 32)$ quadruples the combined rank yet gains nothing over the default, indicating that the bottleneck at the default is not representational capacity but the quality of the preference decomposition itself.

Table 11: Avg on UltraFeedback under LoRA ranks (r_s, r_p). Default in **bold**.

$r_s \setminus r_p$	2	4	8	16	32
$r_s = 4$	49.73	49.85	49.91	51.12	50.84
$r_s = 8$	52.35	52.78	53.12	52.94	52.62
$r_s = 16$	53.64	54.12	**54.46**	54.21	53.82
$r_s = 32$	53.72	54.18	54.52	54.31	53.91
$r_s = 64$	53.54	54.02	54.34	54.13	53.68

The flat 3×3 band of cells around the default spans less than 0.9 Avg points, suggesting that the method is not sensitive to fine rank tuning as long as the relation $r_p \leq r_s/2$ is respected. In practice we recommend $r_s = 16$ for models up to $\sim$2B parameters and scaling r_s mildly with model size, keeping r_p at half of r_s. Reducing the personal rank saves local memory and computation; it does not reduce the shared-adapter upload.

DPO temperature and reference EMA. Figure 6 sweeps β and α. Small β yields noisy implicit rewards; large β squashes the preference margin below the Judge noise floor. A frozen reference ($\alpha = 1$) loses regularization strength as the global model drifts, while an overly reactive reference ($\alpha = 0.5$) tracks the current iterate too closely and weakens the KL anchor. Both curves show a broad

plateau around the defaults $\beta = 0.1$ and $\alpha = 0.95$, and the run-to-run standard deviation is minimized at exactly these values, indicating that the defaults coincide with the low-variance region of the loss landscape.

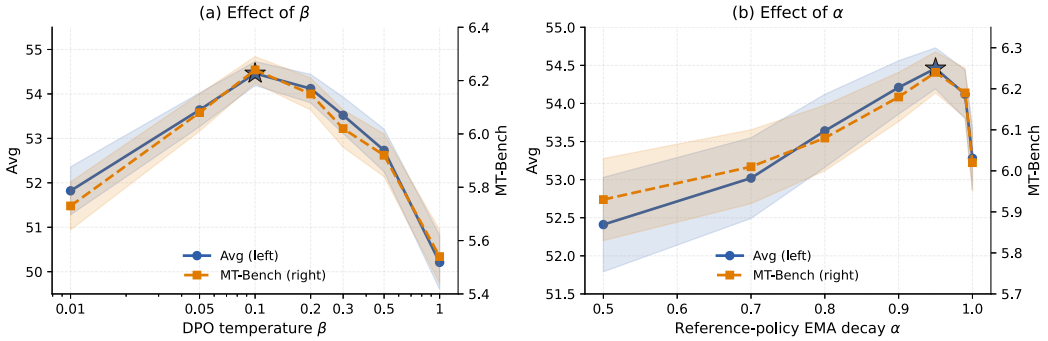

Figure 6: Hyperparameter sensitivity. (a) DPO temperature β. (b) EMA decay α. Shaded bands: ± 1 standard deviation. Defaults marked with a star.

PTA drift thresholds. Figure 7 visualizes the joint effect of the KL drift bound γ_g and the TVR drift bound γ_t. The Avg surface is smooth and concave, with a flat band of cells around the default that stays within 0.4 points of the peak. Loose thresholds admit stale on-policy data and Judge degradation, driving Avg below 52.5 at $(0.5, 0.10)$. Tight thresholds force early stopping before the local optimization has made progress, as seen in the leftmost column. The companion panel shows that the average number of local steps taken grows monotonically in both directions, indicating that PTA correctly translates larger drift budgets into longer local phases.

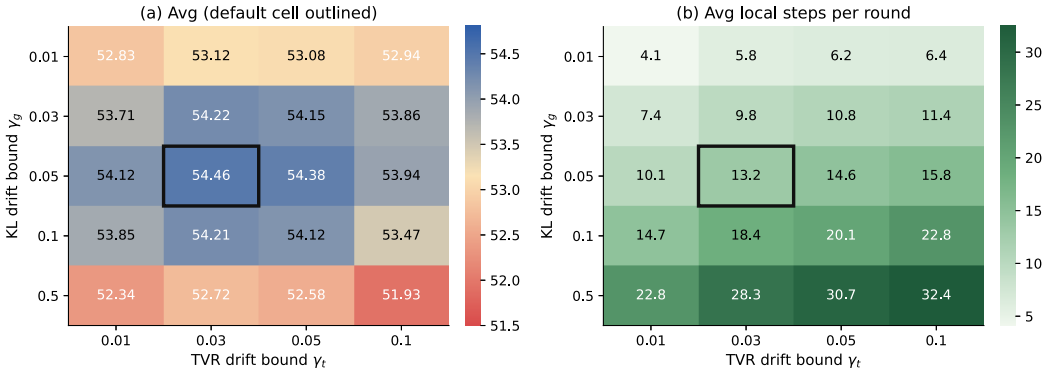

Figure 7: PTA drift thresholds on UltraFeedback (Qwen3-1.7B). (a) Avg over the (γ_g, γ_t) grid. (b) Average local steps per round. Default cell outlined.

Regularization, probe set, and local-step cap. Figure 8 sweeps the three remaining scalars. The Frobenius penalty λ peaks at 10^{-3}; smaller values let the personal adapter memorize Judge noise and larger values collapse it toward zero, recovering the single-policy regime. The probe set size $|\mathcal{P}|$ saturates at 256: mask IoU plateaus at 0.89 and downstream Avg gains less than 0.1 points from further doubling, at a linear cost in upload bandwidth and client-side evaluation time. The local-step cap $E_{\max}$ exhibits an inverted-U: short local phases underfit, and long phases drift away from the sampling policy despite PTA, because the KL budget γ_g becomes saturated well before $E_{\max}$ is reached and the actual number of steps taken plateaus around 63 even when $E_{\max} = 128$.

Preference heterogeneity. Finally we scale the number of Judge personas as a proxy for preference heterogeneity H (Table 12). Two patterns are consistent across both datasets. FedDPO degrades sharply, losing 8.71 Avg points on UltraFeedback and 9.52 on HH-RLHF between 2 and 15 personas, because single-policy aggregation cancels opposing gradient directions. FedDePO only drops by 2.18 and 2.35 points over the same range. The w/o Dual LoRA variant lies consistently between FedDPO

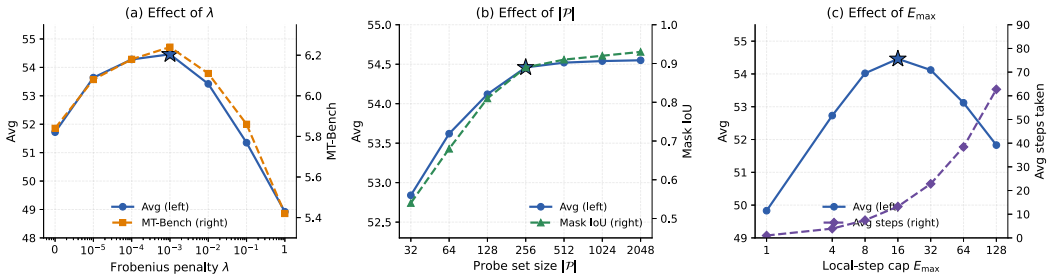

Figure 8: Scalar hyperparameter sweeps. (a) Penalty λ. (b) Probe size $|\mathcal{P}|$ with mask IoU. (c) Local-step cap $E_{\max}$ with actual average steps taken. Defaults marked with a star.

and the full model, and the gap from this variant to the full model widens monotonically with persona count on both datasets, growing from roughly 1.0 point at two personas to roughly 5.1 points at fifteen. The w/o PTC and w/o PTA rows also degrade with heterogeneity but at a much milder rate than FedDPO, confirming that PTC and PTA reduce Judge-level noise but cannot compensate for structural preference conflict on their own. This table supports the empirical benefit of separating shared and personal updates as preference disagreement increases.

Table 12: Preference heterogeneity via Judge persona count on Qwen3-1.7B. H: measured pairwise disagreement on the held-out probe set $\mathcal{Q}$. Best federated per column in **bold**.

Dataset	Method	2 persona ($H=0.12$)	3 persona ($H=0.19$)	**5 persona ($H=0.28$)**	7 persona ($H=0.34$)	10 persona ($H=0.39$)	15 persona ($H=0.43$)
UltraFeedback	FedDPO	53.83	52.84	51.03	49.52	47.64	45.12
	FedDePO w/o PTA	54.32	53.72	53.12	52.31	51.24	49.82
	FedDePO w/o PTC	54.21	53.42	52.48	51.42	50.14	48.53
	FedDePO w/o Dual LoRA	54.12	53.64	52.42	51.24	49.72	47.83
	FedDePO (Ours)	**55.12**	**54.82**	**54.46**	**54.08**	**53.62**	**52.94**
HH-RLHF	FedDPO	55.42	54.38	52.46	50.83	48.74	45.90
	FedDePO w/o PTA	55.83	55.22	54.93	54.08	52.92	51.38
	FedDePO w/o PTC	55.74	54.92	54.31	53.23	51.83	50.02
	FedDePO w/o Dual LoRA	55.68	54.83	53.78	52.52	50.84	48.84
	FedDePO (Ours)	**56.72**	**56.41**	**56.12**	**55.71**	**55.14**	**54.37**

C.11 Capacity-Matched and Routing Controls

We first retain PTC and PTA, remove IRD and the personal adapter, and increase the rank of the single shared adapter from 16 to 32. We then fix the total rank at 24 and compare random assignment, the Ditto-based partition, and IRD routing. Table 13 separates trainable capacity from the rank uploaded for aggregation.

Table 13: Capacity and routing controls. Upload denotes the rank communicated to the server. Single-adapter controls retain PTC and PTA.

Variant	Total rank	Upload rank	UF Avg	HH Avg
Offline FedDPO	16	16	51.03	52.46
Online single LoRA, $r = 16$	16	16	50.71	51.94
Online single LoRA, $r = 24$	24	24	51.62	53.08
Online single LoRA, $r = 32$	32	32	51.89	53.35
Dual LoRA, random split	24	16	51.74	53.18
Ditto-based partition	24	24	52.26	54.17
Dual LoRA, IRD routing	24	16	**54.46**	**56.12**

Increasing the single-adapter rank from 16 to 24 improves Avg by 0.91 on UF and 1.14 on HH; increasing it from 24 to 32 adds 0.27 on each dataset. The rank-32 control still trails FedDePO by 2.57 and 2.77 points while communicating twice its shared rank. At matched total rank, random routing

improves over the single rank-24 adapter by only 0.12 and 0.10 points. IRD routing outperforms the Ditto-based control by 2.20 and 1.95 points. These comparisons support reward-space routing as the source of the improvement beyond added capacity.

The online single rank-16 control scores 50.71/51.94, compared with 51.03/52.46 for offline FedDPO. Thus, online refresh with PTC and PTA does not by itself improve this baseline under the default conflicting preferences. This comparison evaluates the combined online configuration; it does not separately isolate the reference EMA.

Shared and personal inference. Table 14 disables the personal adapter at inference. The shared adapter alone exceeds FedDPO by 1.78 points on UF and 1.77 on HH. Restoring the personal adapter adds 1.65 and 1.89 points. These are within-model inference differences; the persona-conditioned PGap in Table 8 compares different evaluation conditions.

Table 14: Shared-only versus full inference on the same trained FedDePO model.

Inference mode	UF Avg	HH Avg
Shared adapter only, $W_0 + \Delta\theta_s$	52.81	54.23
Full model, $W_0 + \Delta\theta_s + \Delta\theta_p^i$	54.46	56.12

C.12 Polarized Value Preferences

We assign two opposed, non-stylistic personas across five clients: safety-first prefers refusal on borderline queries, whereas information-freedom prefers direct answers. The measured disagreement is $H \approx 0.47$. Table 15 reports results under this stronger conflict.

Table 15: Non-stylistic value conflict with five clients and $H \approx 0.47$.

Method	UF Avg	HH Avg
FedDPO	46.24	44.93
FedDePO w/o Dual LoRA	49.17	48.34
FedDePO	**53.08**	**53.72**

Relative to the default setting, FedDPO decreases by 4.79 points on UF and 7.53 on HH, while FedDePO decreases by 1.38 and 2.40. The full model exceeds the no-Dual-LoRA control by 3.91 and 5.38 points. Preference decomposition therefore remains useful when disagreement concerns refusal and direct answering rather than response style.

C.13 GlobalOpinionQA: Native Human Disagreement

GlobalOpinionQA (Durmus et al., 2023) contains multiple-choice survey questions and country-level response distributions from the Pew Global Attitudes Survey and World Values Survey. We select the USA, China, Nigeria, Germany, and India for geographic coverage and availability of shared questions, before model training. Keeping questions answered in all five countries and removing unanimously agreed items leaves 247 questions, split 80%/20% into training and held-out evaluation. The pairwise disagreement measure in Eq. 17 gives $H = 0.37$.

For question x with answer options $\{a_1, \ldots, a_K\}$, country c supplies distribution $p_c(a_k \mid x)$. Pairwise preferences are obtained from the Plackett–Luce construction with utility $u_{c,k} = \log p_c(a_k \mid x)$:

$$P_c(a_j \succ a_k \mid x) = \frac{\exp(u_{c,j})}{\exp(u_{c,j}) + \exp(u_{c,k})} = \frac{p_c(a_j \mid x)}{p_c(a_j \mid x) + p_c(a_k \mid x)}. \tag{33}$$

The last expression applies to pairs with positive total survey mass. These survey-derived labels replace the self-rewarding judge, providing a test of preference decomposition using human group disagreement.

We obtain a model distribution over answer options from length-normalized log probabilities,

$$s_{c,k}(x) = \frac{\log \pi_c(a_k \mid x)}{|a_k|}, \qquad q_c(a_k \mid x) = \frac{\exp(s_{c,k}(x))}{\sum_j \exp(s_{c,j}(x))}, \tag{34}$$

and evaluate the Jensen–Shannon divergence (JSD) from the country survey distribution:

$$\mathrm{JSD}(p_c \| q_c) = \tfrac{1}{2} D_{\mathrm{KL}}(p_c \| m_c) + \tfrac{1}{2} D_{\mathrm{KL}}(q_c \| m_c), \qquad m_c = (p_c + q_c)/2. \tag{35}$$

Lower JSD indicates closer agreement with the survey distribution. Runs use Qwen3-1.7B, three seeds, and the same LoRA budgets as the main experiments. Balanced training allocates equal mass to each country. A separately trained imbalanced setting allocates 35%, 20%, 18%, 15%, and 12% to the USA, Germany, India, China, and Nigeria, respectively. Evaluation weights countries equally in both settings.

Table 16: GlobalOpinionQA under balanced training. Entries are country-level JSD; lower is better.

Method	USA	China	Nigeria	Germany	India	Mean
FedDPO	.339	.427	.458	.361	.403	.398
Ditto-DPO	.291	.378	.416	.313	.354	.350
FedDePO w/o Dual LoRA	.308	.389	.428	.325	.371	.364
FedDePO	**.182**	**.224**	**.253**	**.196**	**.208**	**.213**

Table 17: GlobalOpinionQA with balanced and imbalanced training. Worst is the largest country JSD. SD and range describe dispersion across countries.

Setting	Method	Mean JSD	Worst JSD	Country SD	Range
Balanced	FedDPO	.398	.458	.043	.119
	Ditto-DPO	.350	.416	.045	.125
	FedDePO w/o Dual LoRA	.364	.428	.043	.120
	FedDePO	**.213**	**.253**	**.025**	**.071**
Imbalanced	FedDPO	.410	.512	.065	.186
	Ditto-DPO	.365	.463	.064	.180
	FedDePO w/o Dual LoRA	.378	.476	.064	.180
	FedDePO	**.225**	**.274**	**.030**	**.085**

FedDePO improves agreement in every evaluated country (Table 16). Its balanced mean JSD is 46.5% lower than FedDPO's, and the worst-country JSD decreases from 0.458 to 0.253. Under imbalanced training, mean JSD increases to 0.225, while the worst-country value remains below all baselines (Table 17). The experiment supports preference decomposition beyond synthetic judge personas.

C.14 Independent Validation of PTC

We use the external GPT-5 judge from Appendix C.4, conditioned on the same persona, as a reference labeler. Table 18 groups prompts by TVR quartile. Agreement decreases from 82.4% in the lowest quartile to 57.3% in the highest. Among pairs rejected by PTC, 72.4% have an external-judge probability margin below 0.15. These observations support TVR as a stability signal while separating consistency from correctness.

Table 18: PTC validation against a persona-conditioned external judge.

TVR quartile	Label agreement (%)	Spearman ρ
First (lowest TVR)	82.4	.71
Second	76.8	.59
Third	68.5	.43
Fourth (highest TVR)	57.3	.26

Complex preferences and injected instability. TVR compares responses under a fixed prompt, persona, and instruction. Nevertheless, a valid near-tie can have a small verdict margin, and a multi-attribute preference can form a cycle. We therefore stratify response triads by preference structure and include an edge-flip control (Table 19). Clear preferences have the highest retention (96.3%); the edge-flip control has the lowest (24.7%). Near-ties have low mean TVR but lower retention because the decisiveness factor in Eq. 7 also enters the filter. PTC is a filter for stable, decisive supervision and can discard valid but weak or cyclic comparisons.

Table 19: Stratified triad analysis with independent persona-conditioned reference labels and controlled edge flips. Reference agreement is reported for retained samples.

Stratum	n	Mean TVR	Retained (%)	Reference agreement (%)
Clear preference	1247	.11	96.3	91.4
Near-tie	384	.17	58.6	73.8
Multi-attribute	276	.29	71.4	76.2
Edge-flip control	341	.52	24.7	43.5

C.15 IRD Direction, Calibration, and Training-State Controls

Direction and scale. For each client, we subtract the mean of its probe margins and divide by their standard deviation, then recompute the mask. A direction flip denotes opposite signs assigned by clients to the same probe margin, checked against persona-conditioned reference rankings. The normalized mask has intersection-over-union (IoU) 0.91 with the original mask; direction-flip AUROC changes from 0.84 to 0.82 (Table 20). This supports directional information in the margins for the evaluated setting. Variance itself measures both direction and magnitude, so the result is an empirical calibration check.

Table 20: IRD calibration control. IoU compares the normalized mask with the original.

Metric	Original	Per-client z-normalization
Mask IoU	–	.91
Direction-flip AUROC	.84	.82

Matched training states. We use 20 clients divided into five groups of four. In the training-state block, preferences are fixed while data fraction, local steps, label noise, and seeds vary. In the preference block, these training conditions are matched while the persona varies. Define

$$V_w = \mathbb{E}_{k,g}[\mathrm{Var}_{i:g_i=g}(m_{i,k})], \quad V_b = \mathbb{E}_k[\mathrm{Var}_g(\mathbb{E}_{i:g_i=g}[m_{i,k}])], \quad S = \frac{V_b}{V_w + \epsilon}. \tag{36}$$

The training-state block is dominated by within-group variance, whereas changing preferences under matched conditions produces larger between-group variance (Table 21). The separation ratio increases from 0.17 to 6.85 and AUROC from 0.53 to 0.87. These controls distinguish systematic preference differences from training-state variation in this experiment.

Table 21: IRD under matched and varied training conditions ($N = 20$, three seeds). Data denotes the percentage retained by subsampling; noise denotes label-flip percentage.

Varied factor	Data (%)	Steps	Noise (%)	Seeds	V_w	V_b	S	AUROC
Training state	5,25,100	1,3,5	0,10,20	3	.041	.007	.17	.53
Preference	100	3	0	3	.013	.089	6.85	.87

Probe domain and threshold. Using HH-RLHF probes during UF training gives Avg 54.12, compared with 54.46 for the default UF probes; the masks have IoU 0.87. Table 22 varies the IRD threshold around its default. Since $\mathcal{M}_k = \mathbf{1}[v_k \leq \tau]$, increasing τ assigns more probes to consensus. The consensus ratio rises from 0.52 to 0.85, with the best Avg at the default threshold.

C.16 Cold-Start Adaptation

FedDePO's seen-to-unseen gaps are 1.53 points on UF and 1.74 on HH (Table 6). Although larger than FedDPO's gaps, its unseen-persona Avg remains higher in absolute terms. We freeze the shared adapter and fine-tune the personal adapter for one to three rounds on the unseen persona. After two rounds the gap is reduced by 71.9% on UF and 70.1% on HH (Table 23), showing that short local adaptation recovers much of the initial loss.

Table 22: Probe-domain and IRD-threshold sensitivity on UltraFeedback.

Probe source	UF Avg	Mask IoU vs. default
UltraFeedback	54.46	–
HH-RLHF	54.12	.87
$\tau/\tau_{\text{default}}$	Consensus ratio	UF Avg
0.5	.52	53.84
1.0	.68	54.46
1.5	.79	54.21
2.0	.85	53.72

Table 23: Remaining seen-to-unseen Avg gap after personal-adapter-only adaptation.

Adaptation rounds	UF gap	HH gap
0	1.53	1.74
1	.87	.98
2	.43	.52
3	.22	.29

C.17 Dispersion under Population Imbalance

We evaluate per-persona Avg across five groups under balanced populations and populations in which one persona has 60% or 80% of the mass. Table 24 reports standard deviation, coefficient of variation, range, and worst-persona Avg. These are dispersion statistics across personas. FedDePO has the lowest dispersion and highest worst-persona score in each setting. At 80% imbalance, its standard deviation is 2.41, compared with 4.83 for FedDPO, and its worst-persona score is 6.41 points higher.

Table 24: Per-persona Avg dispersion on UltraFeedback. SD and CV describe variation across five personas; range is the maximum minus the minimum.

Population (%)	Method	Persona SD	CV (%)	Range	Worst Avg
20/20/20/20/20	FedDPO	2.76	5.4	7.08	48.34
	Ditto-DPO	2.18	4.2	5.83	49.83
	FedDePO	1.42	2.6	3.71	52.38
60/10/10/10/10	FedDPO	3.91	7.7	9.64	45.72
	Ditto-DPO	2.87	5.5	7.42	47.83
	FedDePO	1.93	3.6	5.12	51.24
80/5/5/5/5	FedDPO	4.83	9.6	12.15	43.41
	Ditto-DPO	3.72	7.2	9.58	45.74
	FedDePO	2.41	4.5	6.38	49.82

C.18 Response Budget and Client Cost

Reducing the candidate count from $K = 8$ to $K = 4$ lowers the number of bidirectional comparisons from 56 to 12 per prompt. Table 25 reports the resulting accuracy and runtime tradeoff. Per-round time decreases by 24.2%, with Avg reductions of 0.78 on UF and 0.71 on HH. These measurements use an A100-80GB GPU, matching Appendix C.9.

Training versus diagnostic participation. The participation sweep in Table 9 changes the fraction of clients performing local DPO updates. Every client still evaluates the public probes and uploads fresh margins each round, including clients that do not train. The main experiments use the same diagnostic participation. Thus, the result concerns sparse training with full diagnostics. Client unavailability that also prevents diagnostic uploads changes the IRD information available to the server.

Table 25: Candidate-count tradeoff. Runtime columns are minutes per client per round.

K	UF Avg	HH Avg	Rollout	Judge	Probe	Update	Per-round
8	54.46	56.12	2.84	5.18	.62	17.88	26.52
4	53.68	55.41	1.42	1.24	.62	16.83	20.11

D Data Partition and Heterogeneity Construction

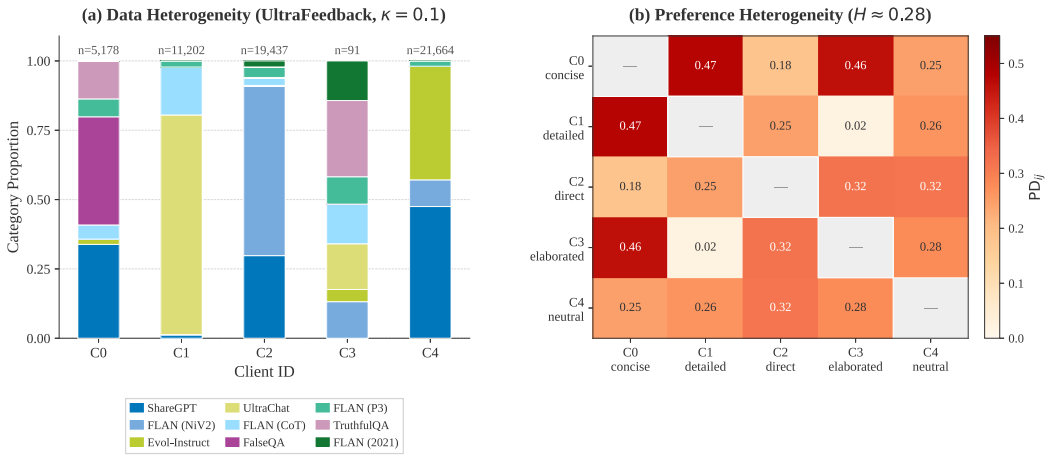

Figure 9: Dirichlet partition of UltraFeedback prompts at $\kappa = 0.1$ and pairwise judgment disagreement PD_{ij} across five Judge personas with average $H \approx 0.28$, on $N = 5$ clients.

Figure 9 visualizes the two heterogeneity dimensions used in the main experiments. Figure 9(a) shows the Dirichlet partition at $\kappa = 0.1$ over the nine UltraFeedback source categories, under which each client is dominated by a small subset of sources and per-client prompt counts span more than two orders of magnitude. Figure 9(b) reports the pairwise judgment disagreement PD_{ij} on the held-out probe set $\mathcal{Q}$ across the five Judge personas, with average $H \approx 0.28$; personas with opposing stylistic tendencies such as *concise* versus *detailed* produce the largest disagreement (0.47), while personas along similar axes such as *detailed* and *elaborated* agree closely (0.02). The two dimensions are induced independently, so a client's data composition and its preference direction vary without any imposed correlation.

E Implementation Details and Fair Comparison

This section collects the practical details that are omitted from the main text for space. We first specify the exact local training and routing pipeline, then describe how the public probe set is constructed, next explain the constraints used to make baseline comparisons fair, and finally discuss the privacy implications of the information uploaded to the server. This ordering mirrors the role of Appendix E in the overall logic of FedDePO: implementation choices determine what each client computes, fair-comparison rules determine how the method should be evaluated, and privacy analysis clarifies what is and is not protected by keeping raw data local.

Implementation details. Experiments are run on NVIDIA A100-80GB GPUs with PyTorch, Hugging Face Transformers, and TRL. LoRA adapters are attached to the query, key, value, and output projections of every attention block, optimized with AdamW at learning rate 5×10^{-5}, batch size 32, and a cosine schedule within each local phase. Rollouts draw $K = 8$ candidates per prompt at temperature 0.8 and top-p 0.95; bidirectional self-judging prompts the policy to return a pairwise verdict, and the normalized token probabilities of the verdict label provide the $P_j(y_w \succ y_l)$ and $P_j(y_l \succ y_w)$ used in Eq. (7). The probe set $\mathcal{P}$ consists of 256 neutral pairs drawn from a held-out public split of UltraFeedback that does not overlap any client's training data, and the thresholds $c_{\min}$ and τ are both set to the median of their respective score distributions at round one and held

fixed thereafter. To map the probe-level mask back to local pairs, we instantiate the routing rule in Eqs. (10)–(11) with a frozen copy of the initial reference model as the encoder $g(\cdot)$, using the mean-pooled last hidden state of the concatenated sequence $[x; y_w; y_l]$. We keep only the top-$J=8$ nearest probes for each local pair and use temperature $T_a=0.1$ in the softmax. This probe-anchor routing is computed locally after PTC filtering and before the two DPO updates, so the server only observes the probe margins $\mathbf{m}_i$ and never any routed local samples.

Probe-anchor routing pseudocode. Algorithm 1 specifies the mask-to-data mapping used by FedDePO. The key design choice is to route a real local pair by semantic proximity to a small number of probe anchors rather than requiring an exact prompt match.

Algorithm 1: Probe-anchor routing from the IRD mask to local training pairs

Input: clean local pairs $\mathcal{D}_i^{\text{clean}}$, probe anchors $\{\mathbf{q}_k\}_{k=1}^{|\mathcal{P}|}$, consensus mask $\mathcal{M}$, top-J, temperature T_a
Output: consensus subset $\mathcal{D}_i^c$, personal subset $\mathcal{D}_i^p$

Initialize $\mathcal{D}_i^c \leftarrow \emptyset$, $\mathcal{D}_i^p \leftarrow \emptyset$
foreach $z = (x, y_w, y_l) \in \mathcal{D}_i^{\text{clean}}$ **do**
  Compute pair embedding $\mathbf{q}(z) = g(x, y_w, y_l)$ using the frozen reference encoder
  Retrieve top-J nearest probes $\mathcal{N}_J(z)$ by cosine similarity
  Compute $a_k(z) = \text{softmax}_{k\in\mathcal{N}_J(z)}\big(\cos(\mathbf{q}(z), \mathbf{q}_k)/T_a\big)$
  $s_i^c(z) \leftarrow \sum_{k\in\mathcal{N}_J(z)} a_k(z)\mathcal{M}_k$
  if $s_i^c(z) \geq 1/2$ **then**
    Add z to $\mathcal{D}_i^c$
  else
    Add z to $\mathcal{D}_i^p$
  end
end

Construction of the Probe Set $\mathcal{P}$. The probe set $\mathcal{P}$ provides a fixed reference grid on which every client evaluates its implicit reward margin (Eq. 8), so its composition directly affects what preference directions IRD can observe. We construct $\mathcal{P}$ with $|\mathcal{P}|=256$ pairs under the following requirements.

- *Public source.* Pairs are drawn from a public UltraFeedback split held out from every client's training partition, so no local data is disclosed through $\mathcal{P}$.
- *Neutrality.* A pair is eligible only if every training persona returns a margin $|P(y^{(1)} \succ y^{(2)}) - P(y^{(2)} \succ y^{(1)})| < 0.15$, discarding pairs with a universally obvious winner that would contribute nothing to Eq. (9).
- *Coverage.* 64 pairs are retained along each of the four persona trait axes (length, directness, formality, caution), preventing $\mathcal{P}$ from concentrating on any single dimension.
- *Disjointness.* Prompts are hashed against every client's training set, so cross-client margin differences reflect learned preferences rather than memorization.

Table 26 summarizes the resulting composition. Applying the same probe-construction procedure to HH-RLHF gives a reported mask Jaccard overlap of 0.91. The separate cross-domain training control in Appendix C.15 gives mask IoU 0.87 and measures the corresponding downstream performance.

Table 26: Composition of the probe set $\mathcal{P}$ ($|\mathcal{P}| = 256$). Lengths are whitespace token counts.

Trait axis	Pairs	Prompt length	$y^{(1)}$ length	$y^{(2)}$ length
Length	64	43.2 ± 18.7	112.5 ± 41.3	108.7 ± 39.8
Directness	64	39.8 ± 16.2	98.4 ± 35.7	101.2 ± 37.4
Formality	64	41.5 ± 17.9	105.3 ± 38.2	103.8 ± 36.9
Caution	64	44.7 ± 19.3	110.8 ± 40.1	109.4 ± 39.5
Overall	256	42.3 ± 18.0	106.8 ± 38.8	105.8 ± 38.4

Fair comparison. Because FedDePO combines a shared adapter with a personal adapter, a fair comparison must separate gains from better decomposition from gains that would arise merely from giving the model more trainable parameters or more communication budget. All federated methods therefore share the same base policy, reference initialization, client partition with $N = 5$ and 3 active per round, and total budget of 20 rounds. Single-adapter baselines (FedAvg+SFT, FedDPO, FedProx-DPO) use LoRA rank $r = r_s = 16$ to match FedDePO's aggregated rank, so the shared-adapter upload size is matched across methods (Table 10); the personalized FL baselines in Table 8 receive a total rank of $r_s + r_p = 24$, split between shared and personal parameters according to each method's own convention. Method-specific hyperparameters, including the DPO temperature, the proximal coefficient in FedProx-DPO, and the personal-copy weight in Ditto-DPO, are tuned independently on a 10% held-out prompt split and then fixed for reporting. Central Online DPO trains on the union of all client prompts with the same total number of DPO updates as the aggregate federated update count, so the gap to FedDePO reflects the federated protocol rather than a compute difference. The main comparison matches uploaded rank rather than total trainable capacity. Appendix C.11 provides the separate capacity-matched controls. Runtime differences are reported in Table 10. All clients provide fresh probe diagnostics in every round, including clients outside the training subset.

Privacy analysis. Raw prompts, candidate responses, and personal adapters remain local. The server receives shared adapter updates and a vector of per-probe margins. This vector is not an average that automatically reduces sensitivity: it reveals a client's response preferences on the public probe set, and repeated releases can expose a preference direction. The evaluated method does not apply differential privacy.

For a possible protected release, let $\widetilde{\mathbf{m}}_i = \mathrm{clip}_C(\mathbf{m}_i)$ have Euclidean norm at most C. Under replacement of a client's entire margin vector, its ℓ_2 sensitivity is bounded by $2C$. The normalized release $\widetilde{\mathbf{m}}_i/\sqrt{M}$, $M = |\mathcal{P}|$, instead has sensitivity at most $2C/\sqrt{M}$:

$$\sup_{D_i, D_i'} \left\| \frac{\mathrm{clip}_C(\mathbf{m}_i(D_i)) - \mathrm{clip}_C(\mathbf{m}_i(D_i'))}{\sqrt{M}} \right\|_2 \leq \frac{2C}{\sqrt{M}}.$$

Thus the $M^{-1/2}$ factor requires explicit norm clipping and normalization; public probes alone do not imply it. Rescaling the vector rescales its signal as well. A differential privacy guarantee for the full protocol would additionally require a calibrated noise mechanism, composition across rounds, and protection of shared-adapter uploads. The reported experiments evaluate the unperturbed protocol.

F Algorithm

Algorithm 2 summarizes the FedDePO procedure. Each round begins with probe evaluation by all clients and mask construction at the server. Training clients then generate on-policy responses and perform bidirectional self-judging, from which PTC retains only pairs whose verdicts are logically consistent and decisive (Eqs. 6–7). IRD then contrasts the resulting implicit reward margins across clients (Eq. 8) to produce a consensus mask $\mathcal{M}$. Each client maps this probe-level mask to its actual local pairs through probe-anchor routing (Eqs. 10–11, Algorithm 1), yielding consensus and personal subsets. Local optimization runs two branch-specific DPO updates on these subsets under the drift-bounded schedule of PTA (Eq. 14), after which the shared adapter is aggregated over training clients and the reference policy is refreshed by EMA.

G Limitations

FedDePO presupposes a base policy capable of producing pairwise judgments with non-trivial signal, and our main experiments accordingly cover models from 0.6B parameters upward. Applying the framework to substantially smaller policies would require an external annotator, since PTC filters noise but cannot recover signal from a judge that has none. Per-round compute is higher than offline FedDPO by roughly $1.6\times$ due to rollout and bidirectional self-judging (Table 10), although fewer rounds are needed to reach a comparable quality target. PTC can discard valid near-ties and multi-attribute cycles, and it does not identify a consistently biased judge. Unseen personas benefit from local personal-adapter adaptation (Appendix C.16). Finally, IRD requires a small shared probe set of public neutral pairs; $|\mathcal{P}| = 256$ is sufficient in our experiments, and the tested protocol also

Algorithm 2: FedDePO

Input: clients $\{\mathcal{D}_i\}_{i=1}^N$, probe set $\mathcal{P}$, rounds T, thresholds $c_{\min}, \tau, \gamma_g, \gamma_t$, EMA decay α
Output: shared adapter (B_s, A_s), personal adapters $\{(B_p^i, A_p^i)\}_{i=1}^N$

Initialize $\Delta\theta_s^{(0)}$, $\{\Delta\theta_p^i\}_{i=1}^N$, $\pi_{\text{ref}}^{(0)}$
for $t = 0, \ldots, T-1$ **do**
  Server broadcasts $\Delta\theta_s^{(t)}$, $\pi_{\text{ref}}^{(t)}$, $\mathcal{P}$
  Select training clients S_t
  foreach *client* $i \in \{1, \ldots, N\}$ ***in parallel*** **do**
    Compute current margin vector $\mathbf{m}_i$ on $\mathcal{P}$ and upload to server
  end
  Server computes $\mathcal{M}_k = \mathbf{1}[v_k \le \tau]$ and broadcasts $\mathcal{M}$ // Eq. (9)
  foreach *training client* $i \in S_t$ ***in parallel*** **do**
    Sample K responses per private prompt and obtain bidirectional verdicts
    Retain $\mathcal{D}_i^{\text{clean}}$ using PTC (Eqs. 6–7)
    Route $\mathcal{D}_i^{\text{clean}}$ into $\mathcal{D}_i^c$ and $\mathcal{D}_i^p$ using $\mathcal{M}$ and Algorithm 1
    while *local-step cap not reached and Eq. (14) not triggered* **do**
      Update $\Delta\theta_s$ on $\mathcal{D}_i^c$ with $\Delta\theta_p^i$ frozen
      Update $\Delta\theta_p^i$ on $\mathcal{D}_i^p$ with $\Delta\theta_s$ frozen // Eq. (13)
    end
    Upload (B_s^i, A_s^i)
  end
  Aggregate $(B_s^{(t+1)}, A_s^{(t+1)})$ and update $\pi_{\text{ref}}^{(t+1)}$ // Eqs. (16), (15)
end

requires every client to provide fresh probe diagnostics. Missing diagnostics and the privacy–utility tradeoff of protected uploads remain outside the evaluated setting.

H Societal Impacts

FedDePO may broaden access to personalized alignment by allowing organizations to improve language models without centralizing sensitive data. This can benefit settings such as education and healthcare, but it can also reinforce the values or biases of dominant client groups when participation is unbalanced or when Judge personas encode narrow norms. Our design reduces raw data exposure, yet practical deployment still requires careful governance, auditing, and user consent because uploaded updates and preference statistics may reveal information about local populations. We therefore view FedDePO as a tool that should be deployed with clear accountability and regular monitoring of fairness, privacy, and misuse risks.