TRAPPED BY THEIR OWN ROLLOUTS: UNDERSTANDING AGGREGATION–ROLLOUT FEEDBACK IN FEDERATED ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy distillation (OPD) is a promising approach to language-model adaptation, aligning teacher supervision with the student's own generated trajectories. When adaptation prompts are distributed across clients, can this process benefit from federated collaboration? We study federated OPD and find that substantial collaboration gains can be obscured by learning-rate sensitivity: FedAvg can perform no better than independent local training at a small learning rate, yet recover a clear advantage at a larger rate. We explain this phenomenon through the student's dual role as learner and generator of future training data. An aggregation-induced optimization lag can delay access to useful teacher supervision, which in turn slows subsequent learning. Our theory establishes this *aggregation–rollout feedback* in a solvable model with a common optimum and stable updates, and identifies two coupled roles of learning rate: learning from current supervision and reaching future supervision. Guided by this analysis, we propose FedTOPS (Federated Teacher-guided On-Policy Scaling), which reuses teacher feedback on current trajectories to adapt the FedAvg update magnitude under clientwise predictive-change constraints. Across six mathematical reasoning benchmarks, FedTOPS improves macro Avg@8 over FedAvg by 4.56–14.57 percentage points across the evaluated student models and local learning rates.

1 INTRODUCTION

On-policy distillation (OPD) offers an effective way to adapt language models using a stronger teacher (Agarwal et al., 2024; Gu et al., 2024). By learning from teacher feedback on its own generations, a student receives supervision on the states it actually visits. In downstream applications, however, the prompts needed for adaptation may be held by separate clients and cannot be pooled. Federated learning offers a natural way for these clients to collaborate by sharing model updates. This raises a basic question: *can on-policy distillation benefit from federated collaboration?*

At first glance, this appears to be a familiar federated optimization problem: combine local updates while accounting for heterogeneous client objectives (McMahan et al., 2017; Karimireddy et al., 2020). Yet OPD gives the student a second role. It is both *the model being trained* and *the generator of future training data*. In conventional fixed-data federated learning, aggregation changes the learner while the available examples remain fixed. In FedOPD, it also changes the trajectories on which the teacher will provide subsequent supervision. Even when each client's prompt distribution stays fixed, the distribution of visited prefixes evolves with the student. Aggregation therefore couples progress across heterogeneous clients with changes in what those clients will learn from next.

We present, to our knowledge, the first systematic study of FedOPD under a shared teacher. We uncover a counterintuitive pattern: federation can provide essentially no gain over independent local training at a small learning rate, yet become substantially beneficial at a larger one (Figure 1). Learning rate therefore governs more than the pace of optimization: **it can determine whether OPD benefits from federated collaboration within a given training budget**. This sensitivity raises a question specific to the student's dual role: how does the progress made by aggregation affect the supervision available in subsequent rounds?

Our central insight is that **an aggregation-induced optimization lag can become a lag in access to future supervision**. If the aggregated student progresses more slowly toward an ability needed to

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.

reach useful teaching states, those states occur less often in its next rollouts. The student then receives less of the supervision that would support further progress. This creates *aggregation–rollout feedback*: an optimization discrepancy affects the data generated next, which in turn affects subsequent learning. A shared, fixed teacher does not eliminate this feedback, because access to its supervision still depends on the student.

Our theory makes this mechanism precise. In an exactly solvable two-stage model, aggregation delays access to downstream teaching states even though all clients share a realizable teacher and a common optimum, and every local update remains stable. Changing only the rollout source isolates the resulting supervision delay. The analysis reveals two coupled roles of learning rate: *learning from current supervision* and *reaching future supervision*. Their interaction reproduces the observed change in the value of collaboration within a finite budget. A general update decomposition further separates ordinary optimization effects from the propagation of discrepancies through the evolving rollout distribution.

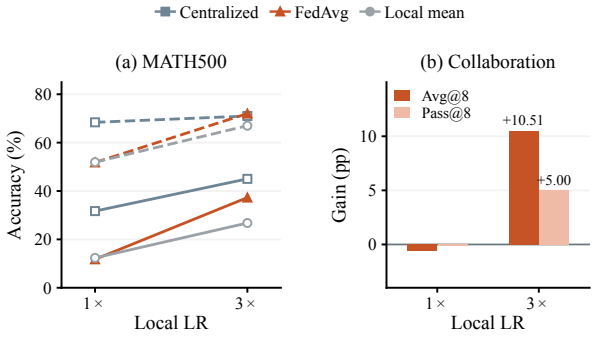


Figure 1: **Learning rate changes collaboration gains.** MATH500, student: Qwen3-0.6B, teacher: Qwen3-8B, matched total rollout budgets. Left: Avg@8 (solid) and Pass@8 (dashed). Right: FedAvg minus Local mean.

This understanding suggests calibrating the aggregate before it generates the next round's trajectories. We propose FedTOPS (Federated Teacher-guided On-Policy Scaling), which uses teacher feedback on current training trajectories to select how far to move along the FedAvg update. Clientwise predictive-change constraints limit the adjustment, and reusing the trajectories avoids additional rollout generation. A controlled server-scaling experiment supports update magnitude as a useful intervention. Across six mathematical reasoning benchmarks, FedTOPS improves macro Avg@8 over FedAvg by 4.56–14.57 percentage points across the evaluated student models and learning rates.

Our contributions are threefold:

- We establish FedOPD as a setting where collaboration can improve distillation, while revealing that its benefits depend strongly on the local learning rate.
- We explain how aggregation-induced optimization lag can delay future supervision and characterize the two roles of learning rate through a solvable construction and a general feedback decomposition.
- We introduce FedTOPS, a simple teacher-guided aggregation rule, and demonstrate improved performance across student models and local learning rates.

2 Phenomenon and Theoretical Analysis

We first establish the learning-rate sensitivity of collaboration gains, then explain it through the student's coupled roles as learner and generator.

2.1 When Does Collaboration Help?

Figure 1 compares FedAvg, independent Local training, and Centralized training from a common 0.6B student under matched total rollout budgets. Local mean averages three separately trained client models. At the reference learning rate, FedAvg offers essentially no collaboration gain; at the larger rate, its MATH500 Avg@8 gain over Local rises from -0.60 to $+10.51$ pp. The six-benchmark macro average follows the same pattern.

The key observation is that learning rate changes whether collaboration is useful within the budget. In OPD, aggregation also sets the generator of the next training trajectories. An optimization discrepancy can therefore change the supervision available in subsequent rounds.

FedOPD protocol. Clients have prompt distributions D_i and positive weights w_i summing to one, and share a fixed teacher q. Each round starts from θ_t: client i generates student trajectories and minimizes

$$\mathcal{L}_i(\theta) = \frac{1}{|\mathcal{H}_i|} \sum_{h \in \mathcal{H}_i} D_{\mathrm{KL}}(p_\theta(\cdot \mid h) \,\|\, q(\cdot \mid h)), \tag{1}$$

where $\mathcal{H}_i$ contains sampled response-token prefixes, held fixed during optimization. The server aggregates the updated parameters as $\theta_{t+1} = \sum_i w_i \theta_{t,i}$. Independent Local training uses the same objective but retains each client's learner and generator without aggregation.

One model, two roles. Distinguish the learner π_θ from the generator π_ϕ. Let d_i^ϕ be its prefix distribution, averaging positions up to horizon L with zero-loss padding after termination (Appendix B). The fixed-source loss and training gradient are

$$L_i(\theta; \phi) = \mathbb{E}_{s \sim d_i^\phi}[\ell(\theta; s)], \qquad g_i(\theta; \phi) = \nabla_\theta L_i(\theta; \phi), \tag{2}$$

where ℓ is a conditional distillation loss, such as forward or reverse token KL. On-policy rounds start at $\phi = \theta$, but the training field $g_i^\circ(\theta) = g_i(\theta; \theta)$ does not differentiate through the sampled-prefix distribution.

Theoretical protocol. We analyze E population gradient steps of size η, freezing the source within each round. Let $\mathcal{F}_\eta(\theta; \phi)$ denote local steps followed by averaging, and $\mathcal{P}_\eta(\theta; \phi)$ denote E synchronous steps on $\sum_i w_i L_i(\theta; \phi)$. This pooled reference isolates the effect of local optimization followed by aggregation. Appendix B states the regularity assumptions.

2.2 How Optimization Lag Delays Supervision

Consider a prerequisite ability u that controls access to teaching states and a downstream ability v learned there. A two-stage Gaussian autoregressive model realizes this setting exactly (Appendix C). For client features $x_i > 0$, let $h_i = x_i^2$, the access threshold be $c < b$, and the precision be $\lambda > 0$. The access probability is

$$\rho_i(u) = \Phi\big(x_i(u - c)\big), \tag{3}$$

where Φ is the standard normal CDF. Summing the two conditional distillation losses gives exactly

$$L_i((u, v); \phi) = \tfrac{h_i}{2}(u - b)^2 + \tfrac{\lambda \rho_i(\phi_u)}{2}(v - 1)^2. \tag{4}$$

Forward and reverse conditional KL agree here. All clients share a realizable teacher and optimum $(b, 1)$, and every fixed-source objective is a convex quadratic. Thus u controls access to supervision for v, without conflicting client targets.

Let $u_0 < c < b$, $0 \le v_0 < 1$, and $0 < \eta \max_i\{h_i, \lambda\} < 1$. Define

$$r_\eta = \sum_i w_i (1 - \eta h_i)^E, \qquad s_\eta(u) = \sum_i w_i (1 - \eta \lambda \rho_i(u))^E. \tag{5}$$

Exact local updates followed by averaging satisfy

$$b - u_{t+1}^F = r_\eta (b - u_t^F), \qquad 1 - v_{t+1}^F = s_\eta(u_t^F)(1 - v_t^F). \tag{6}$$

For synchronous pooled updates, the corresponding factors are $r_\eta^P = (1 - \eta \bar{h})^E$ and $s_\eta^P(u) = (1 - \eta \lambda \bar{\rho}(u))^E$, with weighted means $\bar{h}$ and $\bar{\rho}$.

Theorem 1 (Stable optimization can delay future teaching)**.** *For the construction above and $E \ge 2$, $u_t^F \le u_t^P$ and $v_t^F \le v_t^P$ at every round. Define a source-swapped process that retains federated learner updates but generates each round with u_t^P instead of u_t^F. Then*

$$v_T^F \le v_T^{\mathrm{swap}} \le v_T^P, \qquad \log \frac{1 - v_T^F}{1 - v_T^{\mathrm{swap}}} = \sum_{t=0}^{T-1} \int_{u_t^F}^{u_t^P} \kappa_\eta(z)\, \mathrm{d}z, \tag{7}$$

where $\kappa_\eta(u) = -\partial_u \log s_\eta(u) > 0$. If each ρ_i is made model-independent, the source-swap difference is exactly zero. Under the original Gaussian access law, FedAvg, pooled, and independent Local iterates converge monotonically to the common optimum.

Why this is an additional OPD cost. Jensen's inequality gives $r_\eta \geq r_\eta^P$: averaging delays prerequisite learning, reducing ρ_i and hence downstream supervision. For $E = 2$, the one-round prerequisite lag is exactly $\eta^2 \operatorname{Var}_w(h_i)(b - u)$. Changing only the source in Eq.(7) isolates the additional rollout-mediated cost. This is a finite-budget bottleneck under stable updates, even though all processes converge to the same target.

2.3 The Two Roles of Learning Rate

Learning rate affects both movement toward the teacher and the states available for subsequent learning. The residual downstream error makes these two roles explicit:

$$1 - v_T^F = (1 - v_0)e^{-\mathcal{C}_T(\eta)}, \qquad \mathcal{C}_T(\eta) = -\sum_{t=0}^{T-1} \log s_\eta(u_t^F(\eta)). \tag{8}$$

The quantity $\mathcal{C}_T$ is an *effective supervision clock*: it depends on both the update size and the history of visited teaching states. Its derivative separates these effects:

$$\frac{\mathrm{d}\mathcal{C}_T}{\mathrm{d}\eta} = \underbrace{-\sum_t \partial_\eta \log s_\eta(u_t^F)}_{\text{learning on the same prefixes}} + \underbrace{\sum_t \kappa_\eta(u_t^F)\frac{\mathrm{d}u_t^F}{\mathrm{d}\eta}}_{\text{accessing different supervision}}. \tag{9}$$

Both terms are nonnegative in the stable construction; the second disappears under a common source schedule independent of η. The rate also determines whether substantial supervision is reached within the training budget; Appendix C.6 gives the exact access-time formula.

An explicit sign reversal against Local. Figure 2 instantiates the exact recurrences with two heterogeneous clients and evaluates downstream learning through $S(v) = 1 - (1 - v)^2$ on a common fixed teaching state. At the smaller step size, FedAvg scores below the mean of independently trained Local models; tripling the step size reverses this ordering. All updates remain stable. Swapping the source alone recovers most of the gap to pooled training at both step sizes. This reproduces the change in collaboration gain and isolates a contribution from delayed supervision. Appendix C.7 gives parameters and scores.

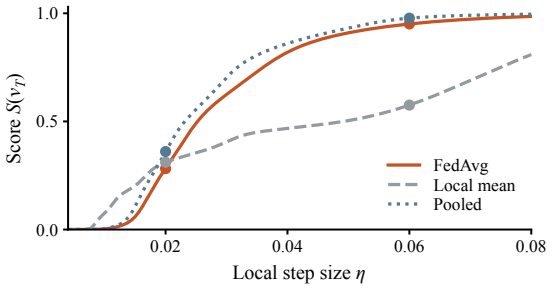


Figure 2: **A sign reversal with stable updates.** Exact recurrences; marked step sizes 0.02 and 0.06. Details: Appendix C.7.

What changes when supervision is fixed? Replace $\rho_i(u)$ by client-specific constants while retaining the protocol and score $S(v)$. Corollary 4 gives a nonnegative FedAvg-minus-Local gain at every stable step size. Thus fixed curvature heterogeneity alone cannot explain this construction's negative gain: model-dependent supervision access is essential.

2.4 Separating the Two Effects in General

For a general OPD update, let $H_i = \partial_\theta g_i$ and $B_i = \partial_\phi g_i$. The on-policy derivative is $Dg_i^\circ = H_i + B_i$: H_i describes learning at a fixed source, while B_i couples teacher gradients to earlier generation decisions. The latter has no predetermined sign. Appendix B derives its score-function expression.

Proposition 2 (Creation and propagation of a discrepancy)**.** *Under the regularity conditions of Appendix B, for fixed E and a frozen source ϕ,*

$$\mathcal{F}_\eta(\theta;\phi) - \mathcal{P}_\eta(\theta;\phi) = \binom{E}{2}\eta^2 \sum_i w_i(H_i - H)(g_i - g) + O(\eta^3), \tag{10}$$

where g_i, H_i are evaluated at $(\theta;\phi)$, $g = \sum_i w_i g_i$, and $H = \sum_i w_i H_i$. For the self-generated processes, write $\delta_t = \theta_t^F - \theta_t^P$ and $e_t = (\mathcal{F}_\eta - \mathcal{P}_\eta)(\theta_t^P;\theta_t^P)$. Then locally

$$\delta_{t+1} = \big[I - \eta E(H_t + B_t)\big]\delta_t + e_t + O\Big(\eta^2 \|\delta_t\| + \|\delta_t\|^2\Big). \tag{11}$$

The matrices in Eq.(11) *are evaluated at the synchronous reference, and* $B_t = \sum_i w_i B_{i,t}$.

Eq.(10) creates an ordinary aggregation discrepancy, which vanishes at $E = 1$. Eq.(11) propagates it through both learning and generation. A common external source removes the first-order $B_t\delta_t$ contribution, motivating the source intervention in Section 4.4.

From explanation to intervention. The aggregate determines both the next learner and its training trajectories. We therefore use current teacher feedback to calibrate the aggregate update magnitude, addressing the coupled roles of learning rate without changing the update direction.

3 Teacher-Guided Magnitude Selection

The analysis suggests a practical design principle: calibrate the aggregated update before it becomes the source of the next training trajectories. Moving farther along a useful update can improve the current student and advance its access to future supervision. Whether a larger move remains useful, however, depends on the current model and local updates. FedTOPS (Federated Teacher-guided On-Policy Scaling) therefore uses the teacher to select the update magnitude in each round.

3.1 From the Analysis to an Aggregation Rule

Let $\bar{\theta}_t$ be the FedAvg result and $\Delta_t = \bar{\theta}_t - \theta_t$ its increment from the round's starting model. The two-stage construction gives a concrete justification for scaling this increment. A constant multiplier satisfying

$$1 \le a \le \min\left\{ \frac{1}{1 - r_\eta}, \frac{1}{1 - (1 - \eta\lambda)^E} \right\}, \tag{12}$$

keeps updates non-overshooting and both coordinates no worse than FedAvg. Moreover, the frozen-source teacher loss decreases along the safe interval (Appendix D). Thus teacher feedback on the current trajectories can help select an update that also advances future learning.

FedTOPS implements this principle with a small set of candidates:

$$\theta_t(a) = \theta_t + a\Delta_t, \qquad a \in \mathcal{A} = \{0.5, 1, 2, 3, 5, 10\}. \tag{13}$$

Here θ contains the trainable LoRA coordinates. The local learning rate η determines the client updates; the server multiplier a scales their aggregate, with $a = 1$ recovering FedAvg. All candidates follow the same increment.

3.2 Selecting an Update from Current Teacher Feedback

Reuse the training trajectories. Before local optimization, each client caches a small sample of the round's student-generated prefixes and their teacher distributions. Every candidate is evaluated on this same cache. This gives a common basis for comparison without generating additional trajectories or querying the teacher for each candidate.

Let $\widetilde{q}$, $\widetilde{\pi}_t$, and $\widetilde{\pi}_{t,a}$ denote the cached teacher, starting-student, and candidate distributions, respectively. The tilde denotes a shared vocabulary partition, specified below. We measure teacher agreement and predictive change by

$$\widehat{J}_t(a) = \sum_i w_i \widehat{\mathbb{E}}_i D_{\mathrm{KL}}(\widetilde{q} \| \widetilde{\pi}_{t,a}), \qquad \widehat{K}_{i,t}(a) = \widehat{\mathbb{E}}_i D_{\mathrm{KL}}(\widetilde{\pi}_t \| \widetilde{\pi}_{t,a}) \le \delta \quad \forall i. \tag{14}$$

Here $\widehat{\mathbb{E}}_i$ averages cached positions on client i, and δ is a predictive-change budget shared across clients within a run. The forward KL in $\widehat{J}_t$ favors candidates closer to the teacher. The constraint is imposed separately on each client, so a small weighted-average change cannot conceal a large change on one client's trajectories.

Choose the magnitude. Among feasible candidates, FedTOPS selects the smallest teacher score. FedAvg is the default when feasible and is replaced only by a strictly better score. If no candidate satisfies the constraint, the server retains the round's starting model. Algorithm 1 in Appendix E summarizes one round and the implementation details.

Table 1: **Avg@8 across benchmarks, learning rates, and student scales.** Scores are percentages; Macro weights all six benchmarks equally. Best/second-best scores within each setting are bold/underlined. Base is repeated for reference; Olympiad denotes OlympiadBench.

Method	MATH500	AMC23	Minerva	Olympiad	AIME24	AIME25	Macro
0.6B student, 1×LR							
Base student	9.07	4.06	2.85	3.12	0.42	0.42	3.32
Centralized	31.70	16.88	7.95	10.31	0.42	<u>0.83</u>	11.35
Local mean	12.35	3.96	3.57	4.32	0.42	0.14	4.13
FedAvg	11.75	3.44	3.95	4.30	0.00	0.00	3.91
Fixed $a = 3$	31.80	14.37	8.04	9.40	0.42	<u>0.83</u>	10.81
Fixed $a = 10$	<u>44.12</u>	<u>21.25</u>	<u>12.87</u>	**15.56**	**2.08**	**1.25**	<u>16.19</u>
FedTOPS	**45.98**	**24.06**	**13.83**	<u>15.50</u>	<u>0.83</u>	<u>0.83</u>	**16.84**
0.6B student, 3×LR							
Base student	9.07	4.06	2.85	3.12	0.42	0.42	3.32
Centralized	45.02	<u>20.94</u>	<u>13.83</u>	15.34	<u>1.67</u>	**2.08**	<u>16.48</u>
Local mean	26.77	13.12	7.08	8.04	1.11	0.69	9.47
FedAvg	37.28	16.56	9.56	11.18	<u>1.67</u>	0.42	12.78
Fixed $a = 3$	<u>45.20</u>	20.00	**14.38**	<u>15.49</u>	<u>1.67</u>	<u>1.67</u>	16.40
Fixed $a = 10$	32.40	14.69	8.73	10.00	0.83	1.25	11.32
FedTOPS	**47.02**	**22.81**	<u>13.83</u>	**17.41**	**2.08**	0.83	**17.33**
1.7B student, 1×LR							
Centralized	<u>62.10</u>	**34.69**	<u>23.30</u>	<u>27.17</u>	**6.67**	<u>3.33</u>	<u>26.21</u>
Local mean	29.41	15.62	10.09	11.99	2.92	1.39	11.90
FedAvg	28.82	15.94	10.20	11.29	2.50	1.67	11.74
FedTOPS	**62.38**	<u>33.75</u>	**23.71**	**27.58**	<u>6.25</u>	**4.17**	**26.31**

A compact scoring cache. Each client caches four training rollouts and at most 16 evenly spaced prediction positions per rollout. The vocabulary partition is the teacher/starting-student top-16 union plus the remaining probability mass, fixed across candidates. Cache details and the full-distribution analysis appear in Appendices E and D.

3.3 Connection to Learning-Rate Robustness

Magnitude selection can offset local learning-rate changes when these mainly rescale the update. For $\Delta(\eta) = \eta d + O(\eta^2)$, write $\beta = a\eta$. The local quadratic teacher-score problem under the predictive-change constraint yields a preferred effective step $\beta^\star$, and hence $a^\star \simeq \beta^\star/\eta$ (Appendix D). This compensates update scale while preserving the FedAvg direction.

4 Experiments

We evaluate collaboration gains across learning rates and student scales (RQ1), adaptive versus fixed aggregate scaling (RQ2), and rollout-source effects on subsequent learning (RQ3).

4.1 Experimental Setup

Models and federated task. We distill Qwen3-8B into Qwen3-0.6B-Base and Qwen3-1.7B-Base students (Yang et al., 2025). Three clients each hold 570 MATH prompts (Hendrycks et al., 2021) from disjoint subject groups: foundational algebra, discrete mathematics, and advanced algebra. All clients participate in 50 rounds, generating 32 responses each per round. We aggregate rank-8 LoRA factor coordinates. The 0.6B local learning rates are 3×10^{-6} (1×LR) and 9×10^{-6} (3×LR); 1.7B uses the reference rate. Each configuration uses one training seed.

Comparisons and evaluation. Local trains clients independently; Centralized pools their prompts. FedAvg averages updates; fixed $a = 3/a = 10$ amplify this increment every round without selection or a trust constraint. FedTOPS selects its magnitude (Algorithm 1), with trust budgets 0.05 for all student models. All trained methods share initialization. Each Local model uses 1,600 rollouts; each federated or Centralized run uses 4,800 system-wide. The adapter-free 0.6B Base is evaluated once. We sample eight answers per problem on MATH500, AMC23, Minerva, OlympiadBench, and

Table 2: **Pass@8 across benchmarks, learning rates, and student scales.** Scores are percentages. Settings, abbreviations, and highlighting follow Table 1.

Method	MATH500	AMC23	Minerva	Olympiad	AIME24	AIME25	Macro
0.6B student, 1×LR							
Base student	46.80	22.50	16.54	16.91	3.33	3.33	18.24
Centralized	68.40	50.00	26.47	33.68	3.33	**6.67**	31.43
Local mean	51.93	22.50	18.14	22.55	2.22	1.11	19.74
FedAvg	51.80	22.50	18.75	21.96	0.00	0.00	19.17
Fixed $a = 3$	69.00	42.50	28.31	31.90	3.33	**6.67**	30.28
Fixed $a = 10$	71.40	52.50	30.15	35.91	**13.33**	**6.67**	**34.99**
FedTOPS	**72.40**	**57.50**	**31.62**	**36.65**	6.67	3.33	34.69
0.6B student, 3×LR							
Base student	46.80	22.50	16.54	16.91	3.33	3.33	18.24
Centralized	71.00	52.50	31.25	36.65	6.67	10.00	34.68
Local mean	67.00	45.00	24.88	30.42	**7.78**	4.44	29.92
FedAvg	72.00	45.00	29.04	33.68	6.67	3.33	31.62
Fixed $a = 3$	**73.00**	**55.00**	31.99	35.76	6.67	**13.33**	**35.96**
Fixed $a = 10$	69.80	42.50	29.04	32.49	3.33	6.67	30.64
FedTOPS	72.20	**55.00**	**32.72**	**39.76**	6.67	6.67	35.50
1.7B student, 1×LR							
Centralized	**84.20**	60.00	40.44	50.74	13.33	13.33	43.67
Local mean	75.20	55.83	35.17	40.26	**14.44**	10.00	38.48
FedAvg	76.20	50.00	34.19	39.47	10.00	10.00	36.64
FedTOPS	83.60	**67.50**	**41.54**	**52.82**	13.33	**23.33**	**47.02**

AIME24/25 (Hendrycks et al., 2021; Lewkowycz et al., 2022; He et al., 2024). Avg@8 measures average correctness; Pass@8 measures whether any answer is correct. Macro weights benchmarks equally; Local mean averages client-model scores. Gains are percentage points (pp). Appendices F and H give settings and client-level results.

4.2 Recovering Collaboration Across Learning Rates and Student Scales

Weak FedAvg performance conceals substantial collaboration gains. At the reference learning rate, FedAvg reaches only 3.91% macro Avg@8, close to the untrained student's 3.32% and below Local's 4.13% (Table 1). With the same local learning rate, FedTOPS reaches 16.84%, improving on FedAvg by $12.93\,\mathrm{pp}$ and exceeding the Centralized reference. This recovery extends across all six benchmarks. Pass@8 improves from 19.17% to 34.69% (Table 2), showing gains in both average correctness and the probability of finding a correct response. Weak FedAvg performance need not imply little potential for collaboration.

The advantage persists at the larger learning rate. At 3×LR, FedAvg itself gains from collaboration, but FedTOPS further improves macro Avg@8 by $4.56\,\mathrm{pp}$ and Pass@8 by $3.88\,\mathrm{pp}$. Its macro Avg@8 is similar at the two evaluated settings: 16.84% and 17.33%, compared with 3.91% and 12.78% for FedAvg. The $0.49\,\mathrm{pp}$ versus $8.87\,\mathrm{pp}$ gap describes these two configurations. Table 1 shows positive collaboration gains on every benchmark at both rates, although FedTOPS ties Centralized on Minerva and trails it on AIME25 at the larger rate.

The benefit extends to a larger student. With the 1.7B student, FedTOPS improves macro Avg@8 over FedAvg by $14.57\,\mathrm{pp}$, reaching the Centralized reference, and achieves the highest Pass@8 among the compared methods (Tables 1 and 2). Its cap-hit rate falls from FedAvg's 38.66% to 9.89%, accompanying the accuracy improvement (Appendix H.3). This experiment tests transfer across student scales at the reference learning rate.

4.3 Aggregate Scaling and Adaptive Selection

Fixed amplification recovers much of the gain. Tables 1 and 2 include fixed $a = 3$ and $a = 10$ at both 0.6B learning rates. At 1×LR, increasing the server multiplier raises macro Avg@8 from FedAvg's 3.91% to 10.81% and 16.19%, respectively, without changing local optimization. Thus a simple magnitude correction already recovers much of the collaboration gain.

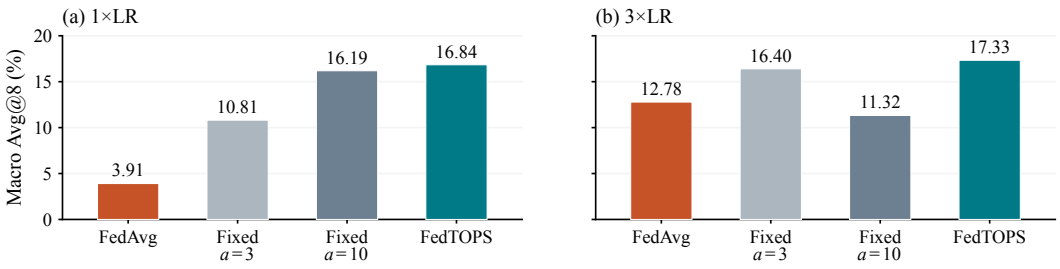


Figure 3: **Fixed amplification versus adaptive selection.** Six-benchmark macro Avg@8 for the 0.6B student. The stronger fixed rule changes from $a = 10$ to $a = 3$ as the local learning rate increases; FedTOPS performs well in both settings. Tables 1 and 2 give benchmark-level accuracy.

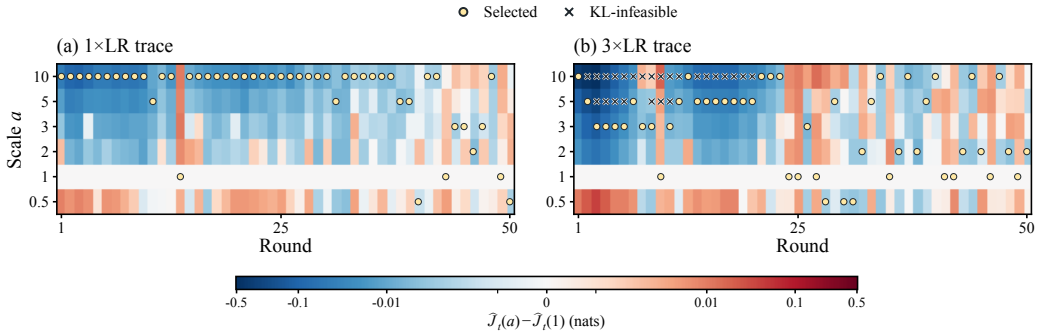


Figure 4: **Teacher-guided selection across rounds.** Candidate-score differences $\widehat{J}_t(a) - \widehat{J}_t(1)$ on a shared symmetric-log scale: blue is better, circles are selected, crosses are infeasible.

The effective multiplier changes with the training setting. The preference between fixed rules reverses at 3×LR (Figure 3): $a = 3$ reaches 16.40% macro Avg@8, whereas $a = 10$ falls to 11.32%, below FedAvg's 12.78%. Always taking the largest candidate is therefore insufficient. FedTOPS achieves the highest macro Avg@8 in both settings, at 16.84% and 17.33%. It does not dominate every metric: fixed $a = 10$ has higher macro Pass@8 at 1×LR (34.99% versus 34.69%), and fixed $a = 3$ does at 3×LR (35.96% versus 35.50%). These controls motivate selecting the magnitude from current teacher feedback.

Selection changes across rounds. The reference-rate trace selects $a = 10$ in 37 of 50 rounds, but also selects every other candidate at least once. The higher-rate trace uses $a = 10$ in 10 rounds and smaller scales in the remaining 40. Within each run, Figure 4 shows how teacher scores and feasibility vary as training evolves. Frequent upper-bound selections coexist with rounds that favor smaller steps, consistent with the need to reassess magnitude during training.

Computational cost. FedTOPS reuses training trajectories and adds forward evaluations for caching and candidate scoring. Recorded caching and round-finalization time accounts for 8.44–15.17% of the training-progress interval across the reported runs. Finalization includes scoring and saving; this fraction is not a measured slowdown relative to FedAvg. Appendix G.3 gives the timing breakdown and forward-pass accounting.

4.4 Rollout Sources and Subsequent Learning

The theory identifies the rollout source as a mediator between aggregation and future learning. Two receiving students start from the same round-9 checkpoint and train at the same local learning rate. Their data come from sources $\phi_1 = \theta_t + \Delta_t$ and $\phi_3 = \theta_t + 3\Delta_t$, constructed from the same round-10 increment. The intervention changes the generator supplying training data while holding the receiving students' starting point fixed (Appendix G.2).

From a common start of 8.97% Avg@8 and 46.20% Pass@8, source $a = 3$ yields a receiver scoring 10.17% and 53.00%; source $a = 1$ yields 9.18% and 44.60% (Figure 5). Changing the source changes subsequent learning from the same receiving model, supporting the proposed rollout-mediated mechanism.

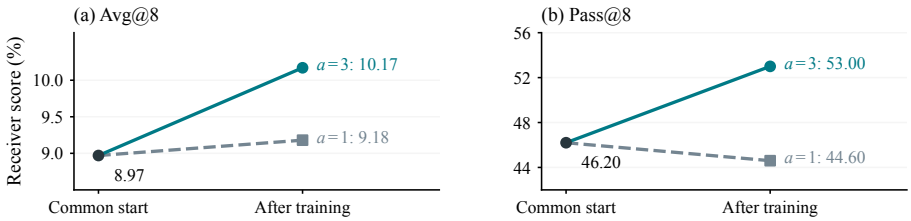


Figure 5: **Rollout source changes subsequent learning.** MATH500 scores before and after short receiver training with source $a = 1$ or $a = 3$. Segments connect the common start to the two measured endpoints; they are not intermediate training trajectories. Both branches share the initial checkpoint and local learning rate.

5 Related Work

Federated low-rank adaptation. FFA-LoRA identifies a mismatch between jointly optimizing LoRA factors locally and averaging them separately, motivating a frozen-factor design (Sun et al., 2024). FLoRA uses stacking to aggregate heterogeneous adapters (Wang et al., 2024), while FedEx-LoRA incorporates the aggregation residual into frozen weights (Singhal et al., 2025). These works address the representation of aggregated updates. FedOPD introduces a further coupling: the aggregated student generates the next training trajectories. We study how optimization discrepancies affect this evolving supervision and use teacher feedback to calibrate the existing update.

On-policy distillation and model-dependent data. Learning from states visited by the learner is central to interactive imitation learning (Ross et al., 2011) and language-model OPD (Agarwal et al., 2024; Gu et al., 2024). Recent work studies teacher compatibility (Li et al., 2026), entropy-aware supervision (Jin et al., 2026), data selection (Hou et al., 2026), and the distinction between state coverage and supervision absorption (Fu et al., 2026). More broadly, performative prediction and performative FL formalize model–data feedback (Perdomo et al., 2020; Jin et al., 2024); federated SARSA studies evolving policies in heterogeneous environments (Zhang et al., 2024; Mangold et al., 2025). Our analysis identifies a specific consequence for FedOPD: aggregation lag can delay access to supervision, giving learning rate two coupled roles in finite-budget learning.

Federated optimization and extrapolation. FedAvg combines local updates (McMahan et al., 2017), and SCAFFOLD addresses heterogeneous client drift (Karimireddy et al., 2020). FedExP selects a server step size from client-update geometry (Jhunjhunwala et al., 2023). In centralized OPD, EffOPD extrapolates checkpoint displacements using validation feedback (Cai et al., 2026). FedTOPS selects the magnitude of the current FedAvg increment using cached teacher distributions and clientwise predictive-change constraints. Its frozen-prefix score and bounded predictive change are related to trust-region policy optimization (Schulman et al., 2015). The motivation here is the effect of aggregation on both current progress and subsequent supervision.

6 Limitations

Our theory explains aggregation–rollout feedback without quantitatively predicting LLM dynamics. Experiments are limited to mathematical reasoning, three fully participating clients, one model family, and one training seed per configuration. FedTOPS scales LoRA factor updates and uses cached teacher agreement as a proxy for future learning benefits. Inexact LoRA aggregation and nonlinear weight-space extrapolation lie outside our analysis; integrating aggregation corrections, more OPD objectives and exploring broader settings remain future work.

7 Conclusion

Federated on-policy distillation couples model aggregation with the generation of future training data. We find that this coupling can make the benefit of collaboration strongly sensitive to learning rate. Our analysis explains how an aggregation-induced optimization lag can delay access to teacher supervision, giving learning rate a second role beyond learning from the current data. This insight motivates FedTOPS, a simple teacher-guided adjustment of the FedAvg update magnitude that improves performance across the evaluated student models and learning rates. More broadly, effective aggregation in FedOPD must account for both the progress made by the current update and the learning opportunities created by the resulting student.

AI USE STATEMENT

We used generative AI tools to polish an existing author-written draft, improving clarity, readability, and linguistic expression. We also used AI tools to assist with code development. The authors reviewed and verified all AI-assisted text and code and take full responsibility for the final content of the paper.

REPRODUCIBILITY STATEMENT

Appendices B–D provide assumptions and complete proofs. The source package includes theory verification, figure-generation code, machine-readable aggregate transcriptions, and table-generation code. Appendix F specifies the training and evaluation protocol, Appendix G reports supplementary analyses, and Appendix H gives complete benchmark results. The bundled experimental figures and tables can be rebuilt from the supplied data and reported summaries. We also upload the code as the supplementary materials for reproduction.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, 2024.

Yuchen Cai, Ding Cao, Liang Lin, Chunxi Luo, Xin Xu, Kai Yang, Weijie Liu, Saiyong Yang, Tianxiang Zhao, Guangzhong Sun, Guiquan Liu, and Junfeng Fang. Learning to foresee: Unveiling the unlocking efficiency of on-policy distillation. *arXiv preprint arXiv:2605.11739*, 2026.

Zixuan Fu, Bingxiang He, Yuxin Zuo, Haohuan Huang, Jinqian Zhang, Ruhang Xiao, Cheng Qian, Qinyu Luo, Huan-ang Gao, Yudong Wang, Zhiyuan Liu, Ning Ding, and Chaojun Xiao. Rethinking on-policy distillation of large language models II: One training example. *arXiv preprint arXiv:2609.04172*, 2026.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *International Conference on Learning Representations*, 2024.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. OlympiadBench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 2024.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *NeurIPS Datasets and Benchmarks*, 2021.

Zhinan Hou, Jiaqi Zhang, Xunliang Cai, and Keyou You. What matters in on-policy distillation? a perspective on data efficiency and data selection. *arXiv preprint arXiv:2609.05198*, 2026.

Divyansh Jhunjhunwala, Shiqiang Wang, and Gauri Joshi. FedExP: Speeding up federated averaging via extrapolation. In *International Conference on Learning Representations*, 2023.

Kun Jin, Tongxin Yin, Zhongzhu Chen, Zeyu Sun, Xueru Zhang, Yang Liu, and Mingyan Liu. Performative federated learning: A solution to model-dependent and heterogeneous distribution shifts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 12938–12946, 2024.

Woogyeol Jin, Taywon Min, Yongjin Yang, Dennis Wei, Yi Zhou, Swanand Ravindra Kadhe, Nathalie Baracaldo, and Kimin Lee. Entropy-aware on-policy distillation of language models. *arXiv preprint arXiv:2603.07079*, 2026.

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 of *Proceedings of Machine Learning Research*, pp. 5132–5143, 2020.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huanang Gao, Wenkai Yang, Zhiyuan Liu, and Ning Ding. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026.

Paul Mangold, Eloïse Berthier, and Eric Moulines. Convergence guarantees for federated SARSA with local training and heterogeneous agents. *arXiv preprint arXiv:2512.17688*, 2025.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera 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 of *Proceedings of Machine Learning Research*, pp. 1273–1282, 2017.

Juan Perdomo, Tijana Zrnic, Celestine Mendler-Dünner, and Moritz Hardt. Performative prediction. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7599–7609, 2020.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635, 2011.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1889–1897, 2015.

Raghav Singhal, Kaustubh Ponkshe, and Praneeth Vepakomma. FedEx-LoRA: Exact aggregation for federated and efficient fine-tuning of large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1316–1336, 2025.

Youbang Sun, Zitao Li, Yaliang Li, and Bolin Ding. Improving LoRA in privacy-preserving federated learning. In *International Conference on Learning Representations*, 2024.

Ziyao Wang, Zheyu Shen, Yexiao He, Guoheng Sun, Hongyi Wang, Lingjuan Lyu, and Ang Li. FLoRA: Federated fine-tuning large language models with heterogeneous low-rank adaptations. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

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.

Chenyu Zhang, Han Wang, Aritra Mitra, and James Anderson. Finite-time analysis of on-policy heterogeneous federated reinforcement learning. In *International Conference on Learning Representations*, 2024.

A Notation and Organization of the Analysis

We distinguish the receiving learner θ from the trajectory generator ϕ throughout the analysis. Table 3 summarizes the notation. Appendix B derives the update decomposition and propagation of learner–generator discrepancies. Appendix C gives the solvable model, its exact recurrences, and the source intervention. Appendix D analyzes aggregate scaling and the relation between full-distribution and finite-cache trust constraints.

Table 3: Main notation. $\mathcal{P}_\eta$ denotes the matched-depth synchronous reference used in the analysis; Centralized denotes the pooled-data training baseline in the experiments.

Symbol	Meaning
K, w_i, D_i	Number of clients, aggregation weight, and fixed prompt distribution
q, π_θ, π_ϕ	Fixed teacher, receiving learner, and trajectory generator
L, d_i^ϕ	Maximum horizon and uniformly position-sampled prefix distribution
$L_i(\theta;\phi), g_i(\theta;\phi)$	Fixed-source loss and its learner gradient
H_i, B_i	Learner response and generator response of the update vector field
E, η, T	Local gradient steps per round, step size, and communication rounds
$\mathcal{F}_\eta, \mathcal{P}_\eta$	Local-then-average map and matched-depth synchronous map
δ_t, e_t	Cross-process state gap and one-round algorithm discrepancy
u, v	Prerequisite and downstream parameters in the construction
h_i, b, c, λ	Client curvature, shared prerequisite target, access threshold, and downstream precision
$\rho_i(u)$	Probability of reaching a downstream teaching state
$r_\eta, s_\eta(u)$	Exact prerequisite and downstream residual factors
$\kappa_\eta, \mathcal{C}_T$	Access sensitivity and effective supervision clock
Δ_t, a, δ	FedAvg increment, server multiplier, and predictive KL budget

B General mechanism: assumptions and complete derivations

B.1 The learner and source are different arguments

For a prompt x and a length-L padded trajectory, write

$$P_i^\phi(x, y_{1:L}) = D_i(x) \prod_{h=1}^{L} \pi_\phi(y_h \mid x, y_{<h}). \tag{15}$$

After termination, both policies use a common absorbing symbol, and the corresponding losses and score derivatives are zero. Drawing h uniformly from $\{1, \dots, L\}$ and retaining $(x, y_{<h}, h)$ defines d_i^ϕ. For continuous outputs, products and expectations refer to densities and integrals. The fixed-source objective is

$$L_i(\theta;\phi) = \frac{1}{L} \sum_{h=1}^{L} \mathbb{E}_{P_i^\phi} \ell(\theta; x, y_{<h}, h). \tag{16}$$

The conditional loss can be either exact forward or reverse KL; the following derivations only require the stated regularity. In particular,

$$\nabla_\theta [L_i(\theta;\theta)] = g_i(\theta;\theta) + \partial_\phi L_i(\theta;\phi)|_{\phi=\theta}\,. \tag{17}$$

Frozen-prefix training uses the first term. It does not generally optimize the diagonal risk by its full gradient.

Assumption 1 (Local regularity and round protocol)**.** The client set and positive weights are fixed. Each round uses all clients, a common starting learner, and E exact gradient steps with one frozen source per client. There is a compact parameter neighborhood containing all points used in the local expansions and their intervening line segments. The functions $g_i(\theta;\phi)$ have bounded continuous derivatives through order two there, with locally Lipschitz second derivatives. The prefix laws have common support on the nonabsorbing states. A dominating integrable function justifies differentiating

each expectation with respect to the learner and source. The bounds hold uniformly for the sufficiently small step sizes under consideration and for fixed E.

This is a sufficient local assumption, not a global boundedness assertion for neural-network KL losses. In the Gaussian construction all relevant expectations are analytic and these bounds hold on compact parameter sets. The fixed teacher supplies no parameter-dependent derivative.

B.2 Proof of the learner–generator decomposition

At the learner–source pair, define $H_i = \partial_\theta g_i$ and $B_i = \partial_\phi g_i$. The on-policy chain rule gives

$$Dg_i^\circ(\theta) = H_i(\theta;\theta) + B_i(\theta;\theta). \tag{18}$$

For $s_h = (x, y_{<h}, h)$ and $z_\theta(s) = \nabla_\theta \ell(\theta; s)$, the source response is

$$B_i(\theta;\theta) = \frac{1}{L}\sum_{h=1}^{L} \mathbb{E}\left[z_\theta(s_h) \left(\sum_{j<h} \nabla_\theta \log \pi_\theta(y_j \mid x, y_{<j}) \right)^\top \right]. \tag{19}$$

Let $z_\theta(s) = \nabla_\theta \ell(\theta; s)$. At a fixed learner,

$$\partial_\phi g_i(\theta;\phi) = \frac{1}{L}\sum_h \int z_\theta(s_h)[\nabla_\phi P_i^\phi(s_h)]^\top \, \mathrm{d}s_h \tag{20}$$

$$= \frac{1}{L}\sum_h \mathbb{E}_{P_i^\phi}\left[z_\theta(s_h) \big(\nabla_\phi \log P_i^\phi(s_h)\big)^\top \right]. \tag{21}$$

The prompt distribution has no parameter dependence, so

$$\nabla_\phi \log P_i^\phi(s_h) = \sum_{j<h} \nabla_\phi \log \pi_\phi(y_j \mid x, y_{<j}). \tag{22}$$

Evaluation at $\phi = \theta$ proves Eq.(19). Applying the chain rule to $g_i(\theta;\theta)$ gives $Dg_i^\circ = H_i + B_i$. The first prediction position has no earlier generated decisions, so its source-score term is zero. None of these expressions implies that B_i is positive semidefinite, symmetric, or harmful.

B.3 Proof of the fixed-source aggregation discrepancy

Fix (θ, ϕ) and suppress those arguments in g_i and H_i. Let $\theta_{i,0} = \theta$ and

$$\theta_{i,e+1} = \theta_{i,e} - \eta g_i(\theta_{i,e}; \phi). \tag{23}$$

For fixed E, bounded gradients give $\theta_{i,e} - \theta = O(\eta)$. Induction and Taylor expansion yield

$$\theta_{i,e} = \theta - e\eta g_i + \binom{e}{2}\eta^2 H_i g_i + O(\eta^3). \tag{24}$$

Indeed, substituting the induction hypothesis into $g_i(\theta_{i,e};\phi) = g_i + H_i(\theta_{i,e} - \theta) + O(\eta^2)$ adds $e\eta^2 H_i g_i$ to the second-order term, and $\binom{e}{2} + e = \binom{e+1}{2}$. Averaging Eq.(24) at $e = E$ gives

$$\mathcal{F}_\eta(\theta;\phi) = \theta - E\eta g + \binom{E}{2}\eta^2 \sum_i w_i H_i g_i + O(\eta^3). \tag{25}$$

The identical expansion for E steps on the pooled frozen-source gradient gives

$$\mathcal{P}_\eta(\theta;\phi) = \theta - E\eta g + \binom{E}{2}\eta^2 H g + O(\eta^3). \tag{26}$$

Finally,

$$\sum_i w_i (H_i - H)(g_i - g) = \sum_i w_i H_i g_i - Hg, \tag{27}$$

which proves Eq.(10). For $E = 1$, both maps are exactly $\theta - \eta \sum_i w_i g_i(\theta;\phi)$, so equality is exact rather than merely second-order. If all client gradient functions agree, the maps also agree for every E. Identical optima alone do not imply identical gradient functions.

Why the source term does not appear here. Within this protocol, ϕ is fixed while each local learner changes. Consequently the within-round curvature is H_i, not $H_i + B_i$. Replacing it by $H_i + B_i$ would analyze a different algorithm that refreshes the source within the local loop. Client drift itself is not a new OPD-specific result (Karimireddy et al., 2020); the next derivation concerns its additional propagation channel.

B.4 Proof of the cross-round propagation formula

Write the diagonal maps as $F^\circ(\theta) = \mathcal{F}_\eta(\theta;\theta)$ and $P^\circ(\theta) = \mathcal{P}_\eta(\theta;\theta)$. The smooth local-step composition and Assumption 1 imply the derivative expansion

$$DF^\circ(\theta) = I - E\eta(H + B) + O(\eta^2), \tag{28}$$

where the remainder is uniform on the neighborhood. This follows either by differentiating the smooth update composition, or by applying Eq.(24) with its uniform differentiated remainder. It does not rely on differentiating an uncontrolled big-O bound.

At θ_t^P, add and subtract $F^\circ(\theta_t^P)$:

$$\delta_{t+1} = F^\circ(\theta_t^P + \delta_t) - F^\circ(\theta_t^P) + F^\circ(\theta_t^P) - P^\circ(\theta_t^P) \tag{29}$$

$$= DF^\circ(\theta_t^P)\delta_t + e_t + O(\|\delta_t\|^2). \tag{30}$$

Substitution of Eq.(28) proves Eq.(11).

If both processes instead use the same externally specified source ϕ_t, differentiation of $\mathcal{F}_\eta(\theta;\phi_t)$ gives $I - E\eta H_t + O(\eta^2)$. The first-order generator contribution is absent because the source does not move with the receiving learner. Static optimization discrepancies, finite-sample noise, and effects of different optimizer states need not disappear. The proposition is a local decomposition, not a global contraction or divergence theorem.

B.5 A general bound on capability-specific supervision

Lemma 3 (Access-limited movement)**.** *Let v be a block of the learner parameters and $\mathcal{S}$ a set of prefixes. Suppose, throughout the relevant local trajectories,*

$$\|\nabla_v \ell(\theta; s)\| \le G\mathbf{1}_{s\in\mathcal{S}} + \epsilon\mathbf{1}_{s\notin\mathcal{S}}, \qquad 0 \le \epsilon \le G. \tag{31}$$

If $\rho_i(\phi_t) = \Pr_{d_i^{\phi_t}}(s \in \mathcal{S})$ and the source is fixed within each round, population local gradient descent followed by averaging satisfies

$$\|v_T - v_0\| \le \eta E \sum_{t=0}^{T-1} \sum_i w_i \left[\epsilon + (G - \epsilon)\rho_i(\phi_t)\right]. \tag{32}$$

Proof. For any local learner iterate, Jensen's inequality for the norm gives $\|\nabla_v L_i(\theta;\phi_t)\| \le \epsilon + (G-\epsilon)\rho_i(\phi_t)$. Summing E steps bounds the local movement. The convexity of the norm bounds the aggregated movement by the weighted local bounds, and the triangle inequality over rounds proves the claim. □

At $\epsilon = 0$, no capability-related movement can be arbitrarily large without accumulating visits to the teaching states. With nonnegative server multipliers, the same argument inserts a_t in each round's bound. The lemma does not apply unchanged to arbitrary momentum states. Its prefix-gradient assumption is capability-specific: it does not state that all incorrect or long trajectories are uninformative.

B.6 An elementary long-horizon extension

Let a trajectory-level teaching event be exactly $\mathcal{S} = \bigcap_{h=1}^H A_h$, where A_h are specified successive events. Define $p_h^j = \Pr_j(A_h \mid A_1, \ldots, A_{h-1})$ for $j \in \{F, P\}$, with positive conditioning probabilities and $p_h^P > 0$. If $p_h^F \le r p_h^P$ for every h, with $0 < r < 1$, then the chain rule gives

$$\frac{\Pr_F(\mathcal{S})}{\Pr_P(\mathcal{S})} = \prod_{h=1}^H \frac{p_h^F}{p_h^P} \le r^H. \tag{33}$$

If the numerator is zero, the bound is immediate without defining conditionals beyond a zero-probability prefix. The equality requires a *complete* event characterization: if $\mathcal{S}$ is only a subset of $\bigcap_h A_h$, an additional conditional success factor is needed, and the prerequisite inequalities alone do not imply Eq.(33).

With N independent trajectories from a fixed generator, the probability of no visit to this event is $(1 - \Pr(\mathcal{S}))^N$. This illustrates how horizon and sampling budget can regulate rare-supervision access. It does not imply domination of all state occupancies or that long reasoning is inherently harmful.

B.7 Relation to practical OPD implementations

An exact conditional reverse KL fits Eq.(2). A clipped importance-weighted surrogate can additionally depend explicitly on the behavior policy through its likelihood ratios, saved advantages, and clipping sets. Its source derivative must include those dependencies, and may require piecewise or generalized derivatives at clipping boundaries. Persistent Adam states require augmenting the state space beyond θ. Likewise, if the rollout sampler uses temperature or top-p truncation, the source-score expression must use that actual sampling law; the raw model softmax is not automatically the generator distribution. The controlled population experiments isolate the mechanism before these implementation effects are introduced.

C A solvable on-policy construction and its consequences

C.1 Deriving the objective from a shared conditional teacher

A client prompt is the deterministic feature $x_i > 0$. The teacher is a single conditional function of (x, Y), not a different target chosen independently for each client. At the first prediction it uses $\mathcal{N}(xb, 1)$, while the student uses $\mathcal{N}(xu, 1)$. At the second prediction, if $Y \geq xc$, the teacher uses $\mathcal{N}(1, \lambda^{-1})$ and the student uses $\mathcal{N}(v, \lambda^{-1})$. Otherwise both use a common distribution independent of (u, v), for example $\mathcal{N}(0, \lambda^{-1})$.

For equal-variance Gaussians,

$$D_{\mathrm{KL}}\big(\mathcal{N}(m_1, \sigma^2) \| \mathcal{N}(m_2, \sigma^2)\big) = \frac{(m_1 - m_2)^2}{2\sigma^2}. \tag{34}$$

The first-position conditional loss is $x_i^2(u - b)^2/2$. Under a source ϕ, the probability of the second teaching state is

$$\Pr(Y \geq x_i c \mid Y \sim \mathcal{N}(x_i \phi_u, 1)) = \Phi(x_i(\phi_u - c)). \tag{35}$$

The second-position conditional loss, whenever reached, is $\lambda(v - 1)^2/2$. Summing both positions proves Eq.(4). This convention differs from the position average in Appendix B by the constant factor two; the step size absorbs that factor. Equality of forward and reverse KL here is an equality of the *conditional* losses. It does not assert equality of the two joint trajectory KLs evaluated under different source laws.

For every finite ϕ_u, $\rho_i(\phi_u) > 0$. Thus the frozen-source Hessian is diagonal with positive entries h_i and $\lambda\rho_i(\phi_u)$, and the unique optimum is $(b, 1)$ for every client. There is no teacher incompatibility and no conflict between client optima. The threshold is fixed in the generated variable; the expected objective is smooth in its parameter arguments.

C.2 Exact iterates for all four processes

Let $h_i = x_i^2$, $u_0 < c < b$, $v_0 \in [0, 1)$, and

$$0 < \eta \max\{h_1, \ldots, h_K, \lambda\} < 1. \tag{36}$$

With the round-t source fixed, the local gradient is

$$g_i((u, v); \phi_t) = \big(h_i(u - b), \lambda\rho_i(\phi_{t,u})(v - 1)\big)^\top. \tag{37}$$

Starting at (u_t, v_t), the E-step residuals on client i are exactly

$$b - u_{i,E} = (1 - \eta h_i)^E (b - u_t), \qquad 1 - v_{i,E} = (1 - \eta\lambda\rho_i(\phi_{t,u}))^E (1 - v_t). \tag{38}$$

For FedAvg, $\phi_{t,u} = u_t^F$. Averaging Eq.(38) gives Eq.(6) and hence

$$u_t^F = b - (b - u_0) r_\eta^t, \qquad 1 - v_T^F = (1 - v_0) \prod_{t=0}^{T-1} s_\eta(u_t^F). \tag{39}$$

For the synchronous reference, each of the E gradient steps uses the mixed frozen-source loss. Its exact factors are r_η^P and s_η^P in the main text, so

$$u_t^P = b - (b - u_0)(r_\eta^P)^t, \qquad 1 - v_T^P = (1 - v_0) \prod_{t=0}^{T-1} s_\eta^P(u_t^P). \tag{40}$$

Independent Local training has

$$u_{i,t}^L = b - (b - u_0)(1 - \eta h_i)^{Et}, \tag{41}$$

$$1 - v_{i,T}^L = (1 - v_0) \prod_{t=0}^{T-1} (1 - \eta\lambda\rho_i(u_{i,t}^L))^E. \tag{42}$$

Finally, the source-swapped federated learner retains the E-step local updates and averaging, but uses the predetermined u_t^P source schedule. The generator's second coordinate is irrelevant to the distribution of prefixes at the second prediction. Therefore

$$1 - v_T^{\text{swap}} = (1 - v_0) \prod_{t=0}^{T-1} s_\eta(u_t^P). \tag{43}$$

The swapped experiment changes supervision generation, not the receiving learner's number of updates or its update rule.

C.3 Proof of Theorem 1

Prerequisite lag. For $E \geq 2$, the second derivative of $f(z) = (1 - \eta z)^E$ is $E(E-1)\eta^2(1 - \eta z)^{E-2} \geq 0$ on the stable interval. Jensen's inequality yields

$$r_\eta = \sum_i w_i f(h_i) \geq f\left(\sum_i w_i h_i\right) = r_\eta^P. \tag{44}$$

Both factors lie in $(0, 1)$, so Eq.(39)–Eq.(40) imply $u_t^F \leq u_t^P$. For $E = 2$ the Jensen gap is exact:

$$r_\eta - r_\eta^P = \eta^2 \left[\sum_i w_i h_i^2 - \left(\sum_i w_i h_i\right)^2\right] = \eta^2 \operatorname{Var}_w(h_i). \tag{45}$$

From a common starting point u, the one-round prerequisite difference is therefore $\eta^2 \operatorname{Var}_w(h_i)(b - u)$.

The two downstream costs. The Gaussian CDF has derivative

$$\rho_i'(u) = \frac{x_i}{\sqrt{2\pi}} \exp\left[-\frac{x_i^2(u - c)^2}{2}\right] > 0. \tag{46}$$

Consequently $s_\eta(u)$ decreases with u. Jensen's inequality applied to $\rho_i(u)$ also gives

$$s_\eta(u_t^F) \;\geq\; s_\eta(u_t^P) \;\geq\; s_\eta^P(u_t^P). \tag{47}$$

All factors and the initial residual $1 - v_0$ are positive. Multiplying over t proves $v_T^F \leq v_T^{\text{swap}} \leq v_T^P$. The first comparison changes only the source; the second compares local-then-average and synchronous learning from the same source schedule.

Exact source attribution. Direct differentiation gives

$$\kappa_\eta(u) = \frac{E\eta\lambda \sum_i w_i \rho_i'(u)(1-\eta\lambda\rho_i(u))^{E-1}}{s_\eta(u)} > 0. \tag{48}$$

Taking the logarithm of the ratio of Eq.(39) and Eq.(43) yields

$$\log \frac{1-v_T^F}{1-v_T^{\text{swap}}} = \sum_t \left[\log s_\eta(u_t^F) - \log s_\eta(u_t^P)\right] \tag{49}$$

$$= \sum_t \int_{u_t^F}^{u_t^P} \kappa_\eta(z)\,\mathrm{d}z. \tag{50}$$

If each ρ_i is replaced by a model-independent constant, s_η no longer depends on u, $\kappa_\eta = 0$, and the source-swapped and original federated v iterates coincide exactly. Constants can still differ across clients, so ordinary Jensen discrepancies between local and synchronous learning can remain.

Monotonic convergence, not divergence. Since $0 < r_\eta < 1$, u_t^F increases to b. Because $u_t^F \geq u_0$,

$$0 < s_\eta(u_t^F) \leq s_\eta(u_0) < 1. \tag{51}$$

The uniform upper bound follows from $\rho_i(u_0) > 0$. Hence $0 \leq 1 - v_T^F \leq (1-v_0)s_\eta(u_0)^T \to 0$. The same argument applies to the pooled and Local recurrences and also to the swapped process. No step overshoots its shared target under Eq.(36). This proves all statements of Theorem 1.

C.4 The coupling is present without an unstable eigenvalue

The exact federated map has Jacobian

$$D\mathcal{F}_\eta(u,v) = \begin{pmatrix} r_\eta & 0 \\ -(1-v)s_\eta'(u) & s_\eta(u) \end{pmatrix}. \tag{52}$$

Both eigenvalues lie in $(0,1)$, and the lower-left entry is nonnegative when $v \leq 1$. An upstream lag can therefore affect downstream learning without an eigenvalue exceeding one. The matrix need not be normal or a Euclidean contraction; monotonic convergence above was established directly, rather than inferred from a local eigenvalue calculation. At the gradient level, the source response is particularly transparent:

$$H_i = \begin{pmatrix} h_i & 0 \\ 0 & \lambda\rho_i(u) \end{pmatrix}, \qquad B_i = \begin{pmatrix} 0 & 0 \\ \lambda\rho_i'(u)(v-1) & 0 \end{pmatrix}. \tag{53}$$

The generator term transmits prerequisite differences to the downstream coordinate.

C.5 A fixed-source contrast against independent Local training

Corollary 4 (No negative gain in the constant-access counterfactual)**.** *Replace the access laws in Eq.*(4) *by client-specific constants $p_i \in [0,1]$, held fixed across rounds, learning rates, and training processes. Under Eq.*(36)*, use the same E, T, η, w_i, and initial downstream parameter v_0 for FedAvg and independent Local training. On the common evaluation score $S(v) = 1-(1-v)^2$,*

$$S(v_T^F) - \sum_i w_i S(v_{i,T}^L) = (1-v_0)^2 \left[\sum_i w_i \zeta_i^{2T} - \left(\sum_i w_i \zeta_i\right)^{2T}\right] \geq 0, \tag{54}$$

where $\zeta_i = (1-\eta\lambda p_i)^E$.

Proof. The constant-access residuals are $1 - v_T^F = (1-v_0)(\sum_i w_i\zeta_i)^T$ and $1 - v_{i,T}^L = (1-v_0)\zeta_i^T$. Substitute them into S. Since $T \geq 1$ and $\zeta_i \in [0,1]$, convexity of $z \mapsto z^{2T}$ proves the inequality by Jensen's inequality. □

Thus the negative low-step federated gain in this construction cannot be reproduced by merely retaining different constant downstream curvatures. Its model-dependent access law matters. This contrast is specific to the scalar common-state score, shared initialization, and time-constant access rates; it is not a general superiority theorem for FedAvg on static data. An arbitrary time-varying source schedule need not satisfy the same comparison.

C.6 Bounds and sensitivity of the supervision clock

Let $\gamma = \eta\lambda \in (0,1)$. From Eq.(39), $1 - v_T^F = (1 - v_0)\exp(-\mathcal{C}_T)$ with $\mathcal{C}_T = -\sum_t \log s_\eta(u_t^F)$. For any $\rho \in [0,1]$,

$$1 - (1-\gamma\rho)^E = \int_0^{\gamma\rho} E(1-z)^{E-1}\,\mathrm{d}z \geq E\gamma\rho(1-\gamma)^{E-1}. \tag{55}$$

Using $-\log s \geq 1 - s$ gives the lower bound below. For the upper bound, convexity of $-\log$ gives

$$-\log s_\eta(u) \leq -E\sum_i w_i \log(1-\gamma\rho_i(u)) \tag{56}$$

$$\leq E\sum_i w_i \frac{\gamma\rho_i(u)}{1-\gamma\rho_i(u)} \leq \frac{E\gamma}{1-\gamma}\bar{\rho}(u). \tag{57}$$

Consequently,

$$E\gamma(1-\gamma)^{E-1}\sum_t \bar{\rho}(u_t^F) \leq \mathcal{C}_T \leq \frac{E\gamma}{1-\gamma}\sum_t \bar{\rho}(u_t^F). \tag{58}$$

The history of accessible supervision therefore regulates the effective amount of downstream learning.

For the derivative, first observe

$$r_\eta' = -E\sum_i w_i h_i (1-\eta h_i)^{E-1} < 0, \qquad \frac{\mathrm{d}u_t^F}{\mathrm{d}\eta} = -(b-u_0)t r_\eta^{t-1} r_\eta' \geq 0, \tag{59}$$

where the derivative for $t = 0$ is zero. At fixed u,

$$-\partial_\eta \log s_\eta(u) = \frac{E\lambda\sum_i w_i\rho_i(u)(1-\eta\lambda\rho_i(u))^{E-1}}{s_\eta(u)} \geq 0. \tag{60}$$

The chain rule then proves Eq.(9); its second term is nonnegative by Eq.(48). If every learning rate is assigned the same external source schedule, independent of η, that second term is absent. This comparison requires fixing the source across learning rates, not merely freezing each rate's own source within a round.

The first source index at which every client has access probability at least $1/2$ is

$$T_{\mathrm{hit}} = \left\lceil \frac{\log((b-c)/(b-u_0))}{\log r_\eta} \right\rceil \tag{61}$$

The first source index t with $u_t^F \geq c$ satisfies $r_\eta^t \leq (b-c)/(b-u_0)$, which proves Eq.(61). Since $\rho_i(c) = 1/2$ for every client, this is a half-access threshold, not the onset of nonzero supervision: $\rho_i(u) > 0$ for every finite u. A run of T rounds uses sources $u_0, \ldots, u_{T-1}$, so the number of rounds at or above this threshold is $\max\{T - T_{\mathrm{hit}}, 0\}$. A crossing at $t = T$ cannot influence that run's preceding updates. This is a finite-budget access boundary, not a singular phase transition. Increasing η outside the non-overshooting interval has not been analyzed by these inequalities. Nor does the theory predict that a vanishing step size increases one-round aggregation error.

C.7 Numerical Illustration

All illustrations use the fixed parameters

$$\begin{gathered} K = 2, \quad w_1 = w_2 = \tfrac{1}{2}, \quad (h_1, h_2) = (1, 9), \quad E = 4, \quad T = 5, \\ b = 10, \quad c = 7.25, \quad \lambda = 2, \quad u_0 = v_0 = 0. \end{gathered} \tag{62}$$

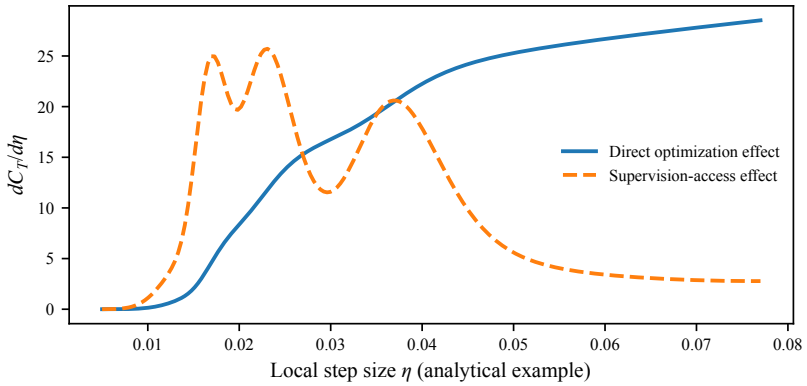


Figure 6: **Two contributions to step-size sensitivity.** Both curves are exact derivatives of the construction's supervision clock. The mediated contribution changes the source along with the learner; it is absent under a source schedule held fixed across learning rates.

The common evaluation state is a downstream teaching prefix, and its score is $S(v) = 1 - (1 - v)^2$. Thus the evaluation distribution is fixed independently of the training generator. Local mean is $\sum_i w_i S(v_{i,T}^L)$, not $S(\sum_i w_i v_{i,T}^L)$. These scores are not percentages or fitted language-model accuracy estimates.

Table 4: Exact-recursion scores for the same construction. All update factors satisfy the stated stability condition.

η	FedAvg	Local mean	Pooled	Source-swapped FedAvg
0.02	0.281532	0.311221	0.360625	0.360494
0.06	0.950782	0.575594	0.977725	0.977670

The federated gain changes from -0.029689 to $+0.375187$. At the smaller step, the faster client can independently reach downstream teaching states before the repeatedly averaged student does. At the larger step, the federated student reaches those states sufficiently early to share the acquired downstream capability. The comparison is finite-budget; all processes eventually reach the shared optimum.

At $\eta = 0.02$, the direct and mediated derivatives are approximately 8.368607 and 19.770051; at $\eta = 0.06$, they are 26.682896 and 3.410686. The mediated contribution dominates at the smaller step size, whereas the direct contribution is larger at the higher step size, illustrating how their relative importance changes with access to downstream teaching states.

D Magnitude Calibration: Constructive and Local Guarantees

D.1 Pure Server Scaling in the Construction

Proposition 5 (A safe amplitude intervention)**.** *Under the construction's stability conditions, define*

$$a_{\max} = \min\left\{\frac{1}{1 - r_\eta}, \frac{1}{1 - (1 - \eta\lambda)^E}\right\}. \tag{63}$$

For a constant $a \in [1, a_{\max}]$, scaling the FedAvg increment by a yields non-overshooting updates and, from the same initial state,

$$u_t^{(a)} \geq u_t^{(1)}, \qquad v_t^{(a)} \geq v_t^{(1)} \quad \textit{for all } t. \tag{64}$$

Moreover, the population teacher loss evaluated on the current frozen source is nonincreasing in a throughout this safe interval.

Proof. Let $\Delta u = (1 - r_\eta)(b - u)$ and $\Delta v = (1 - s_\eta(u))(1 - v)$. The scaled residuals are

$$b - u^+ = [1 - a(1 - r_\eta)](b - u), \qquad 1 - v^+ = [1 - a(1 - s_\eta(u))](1 - v). \tag{65}$$

Because $\rho_i(u) \le 1$, we have $s_\eta(u) \ge (1-\eta\lambda)^E$. The bound on a therefore places both bracketed factors in $[0,1]$. For the prerequisite, the scaled factor is no larger than r_η, so $u_t^{(a)} \ge u_t^{(1)}$ follows directly by induction. For the downstream coordinate, write $q_a(u) = 1-a(1-s_\eta(u))$. If $u^{(a)} \ge u^{(1)}$, monotonicity of s_η gives

$$0 \le q_a(u^{(a)}) \le s_\eta(u^{(a)}) \le s_\eta(u^{(1)}). \tag{66}$$

Multiplication by the ordered nonnegative residuals in Eq.(65) completes the induction.

On the frozen round-start source, the mixed teacher loss at the candidate is exactly

$$\begin{aligned} J_t(a) = {} & \tfrac{\bar{h}}{2}(b-u_t)^2[1-a(1-r_\eta)]^2 \\ & + \tfrac{\lambda\bar{\rho}(u_t)}{2}(1-v_t)^2[1-a(1-s_\eta(u_t))]^2. \end{aligned} \tag{67}$$

Each squared term is nonincreasing while its bracket is nonnegative. Thus $J_t'(a) \le 0$ on the safe interval. □

For the numerical instance at $\eta = 0.02$, the fixed-state scores at $a = 1, 2, 3$ are 0.281532, 0.884641, and 0.991698. No new direction is introduced. Proposition 5 does not guarantee the behavior of a finite-cache selector, which can choose $a < 1$ and does not know this construction's safe interval. It demonstrates that correcting a progress deficit can be sufficient without correcting the direction.

In this construction, a full predictive-change score on a client's frozen prefixes has a particularly simple form:

$$K_{i,t}(a) = \tfrac{a^2}{2}\left[h_i(\Delta u_t)^2 + \lambda\rho_i(u_t)(\Delta v_t)^2\right]. \tag{68}$$

This uses the sum of both prediction-position KLs; dividing by two gives the position average. It makes explicit why a larger beneficial step can nevertheless be bounded by a behavior-change budget.

D.2 Local scale compensation in a general parameterization

At a fixed learner and optimizer state, suppose $\Delta(\eta) = \eta d + \eta^2\xi_\eta$ with uniformly bounded ξ_η. For $\beta = a\eta$, the actual candidate displacement is

$$a\Delta(\eta) = \beta d + \beta\eta\xi_\eta. \tag{69}$$

Thus extrapolation amplifies the original $O(\eta^2)$ direction error to $O(|\beta|\eta)$, rather than leaving it $O(\eta^2)$. For a smooth fixed-source teacher score along the limiting ray,

$$J(\theta+\beta d) = J(\theta) - b_0\beta + \tfrac{1}{2}c_0\beta^2 + o(\beta^2), \tag{70}$$

where $b_0 = -\nabla J(\theta)^\top d$ and $c_0 = d^\top\nabla^2 J(\theta)d$. In a favorable local model, $b_0, c_0 > 0$. Expanding the old-to-candidate KL yields

$$K_i(\theta+\beta d) = \tfrac{1}{2}\beta^2 f_i + o(\beta^2), \qquad f_i = d^\top F_i d \ge 0, \tag{71}$$

where F_i is the conditional Fisher information averaged over the fixed source. These KL expansions require differentiable, normalized conditionals with common support and justified exchange of differentiation and expectation. For the actual candidate in Eq.(69), the score expansion acquires an additional $O(|\beta|\eta)$ term and the KL expansion an $O(\beta^2\eta)$ term, locally for small η and $|\beta|$; the latter uses the zero KL gradient at the starting policy. Consequently, the quadratic model discards both displacement and direction remainders. If $f_{\max} = \max_i f_i > 0$, its continuous constrained problem over $\beta \ge 0$ has the solution

$$\beta^\star = \min\left\{\frac{b_0}{c_0}, \sqrt{\frac{2\delta}{f_{\max}}}\right\}. \tag{72}$$

Hence the idealized compensating multiplier is $a^\star \simeq \beta^\star/\eta$. This is a local quadratic design calculation, not an exact optimizer for the nonlinear problem. In particular, if a grows like $1/\eta$, the candidate displacement must still remain in the neighborhood where the expansion is accurate; making η small does not alone ensure that. A finite candidate grid, changed directions, clipping, and persistent optimizer states all limit compensation. If $b_0 \le 0$, a positive multiple of the proposed direction need not improve the teacher objective at all.

D.3 What an ideal full-distribution trust constraint controls

The use of a frozen-state surrogate with a bounded policy change is related to trust-region policy optimization (Schulman et al., 2015). Here we provide the precise statement needed for the distillation setting, separately from its empirical implementation. Throughout, logarithms are natural and $\mathrm{TV}(P, Q) = \sup_A |P(A) - Q(A)|$, equivalently one half of the L^1 distance between densities.

Proposition 6 (An ideal source-transfer bound)**.** *Let θ be the old policy and θ' a candidate, with the same prompt distribution and padded horizon L. Define the full-vocabulary population constraint*

$$K_i(\theta') = \mathbb{E}_{s\sim d_i^\theta} D_{\mathrm{KL}}\big(\pi_\theta(\cdot \mid s)\|\pi_{\theta'}(\cdot \mid s)\big). \tag{73}$$

Assume the same policies are used for sampling and for this KL, the trajectory laws have the necessary absolute continuity, and $0 \le \ell(\theta'; s) \le M$ on the relevant states. Then

$$|L_i(\theta'; \theta') - L_i(\theta'; \theta)| \le M\sqrt{\frac{LK_i(\theta')}{2}}. \tag{74}$$

If $K_i(\theta') \le \delta$ for every client, then

$$L(\theta'; \theta') \le L(\theta'; \theta) + M\sqrt{\frac{L\delta}{2}}. \tag{75}$$

Proof. The KL chain rule, with unchanged prompts, gives

$$D_{\mathrm{KL}}(P_i^\theta \| P_i^{\theta'}) = \sum_{h=1}^{L} \mathbb{E}_{P_i^\theta} D_{\mathrm{KL}}(\pi_\theta(\cdot \mid s_h)\|\pi_{\theta'}(\cdot \mid s_h)) = LK_i(\theta'). \tag{76}$$

Sampling a uniform position and taking its prefix is a common Markov transformation of the trajectory. Data processing and Pinsker's inequality therefore imply

$$\mathrm{TV}(d_i^\theta, d_i^{\theta'}) \le \mathrm{TV}(P_i^\theta, P_i^{\theta'}) \le \sqrt{LK_i(\theta')/2}. \tag{77}$$

The expectation of any function taking values in $[0, M]$ changes by at most M times the total variation distance. Apply this to $\ell(\theta'; \cdot)$, then average over clients. □

A corresponding change-from-start inequality is

$$L(\theta'; \theta') - L(\theta; \theta) \le \underbrace{L(\theta'; \theta) - L(\theta; \theta)}_{\text{fixed-source score change}} + M\sqrt{L\delta/2}. \tag{78}$$

The bounded-loss assumption is additional and nontrivial for KL; a suitable probability floor or restricted domain can imply it, but the deployed neural model does not automatically satisfy it. For long sequences the bound can also be vacuous. The result bounds a distillation-loss distribution shift, not task accuracy or access to useful reasoning states.

D.4 Finite-Cache and Vocabulary Approximations

Fix a prefix and a set S of retained token IDs, and combine the complement T into one tail bucket. Let $p_T = \sum_{v\in T} p_v$, $q_T = \sum_{v\in T} q_v$, and let $\widetilde{p}, \widetilde{q}$ be the partitioned distributions. Expanding the tail sum gives the exact identity

$$D_{\mathrm{KL}}(p\|q) = D_{\mathrm{KL}}(\widetilde{p}\|\widetilde{q}) + p_T D_{\mathrm{KL}}\big(p(\cdot \mid T)\|q(\cdot \mid T)\big), \tag{79}$$

when $p_T, q_T > 0$, with extended-real KL if some conditional support is missing. If $p_T = 0$, the tail term contributes zero. If $p_T > 0 = q_T$, both full and coarsened KL are infinite, and no undefined tail conditional is needed to assert the lower bound. Thus the coarsened KL is a lower bound. Its being below δ cannot certify that the full KL is below δ, and changes within the tail are invisible to the coarsened score. Comparing candidates requires the same fixed partition for every candidate at that prefix.

The implemented cache also differs from a population prefix average: it uses few trajectories, regularly selected positions, and training-dependent candidates. Positions within a trajectory are

correlated. Because the same round's data help create the candidates, a naive independent-validation finite-candidate bound is inapplicable.

Finally, the current rule does not always compare a feasible update with $a = 0$. Even an accurately measured score better than the ordinary FedAvg candidate need not be better than the round start. Proposition 6 thus bounds source-distribution shift under its stated population assumptions; it does not establish monotonic improvement for the finite-cache selection rule.

E Algorithm and Implementation Details

Algorithm 1 gives one round of FedTOPS. Local OPD determines the update direction, and the teacher-guided selector chooses its magnitude using a fixed cache of that round's training trajectories.

Algorithm 1 FedTOPS: teacher-guided magnitude selection (one round)

Require: Starting model θ_t, client weights w_i, teacher q, budget δ, candidates $\mathcal{A}$
for each client i in parallel **do**
  Generate local training trajectories from θ_t; cache sampled prefixes and distributions
  Perform local OPD with learning rate η to obtain $\theta_{i,t}^{+}$
end for
$\Delta_t \leftarrow \sum_i w_i \theta_{i,t}^{+} - \theta_t$
for $a \in \mathcal{A}$ **do**
  Form $\theta_t(a) \leftarrow \theta_t + a\Delta_t$
  Evaluate $\widehat{J}_t(a)$ and each $\widehat{K}_{i,t}(a)$ on the fixed caches using Eq.(14)
end for
$\mathcal{A}_f \leftarrow \{a \in \mathcal{A} : \max_i \widehat{K}_{i,t}(a) \le \delta\}$
if $\mathcal{A}_f = \varnothing$ **then**
  $a_t \leftarrow 0$
else if $1 \in \mathcal{A}_f$ **then**
  $a_t \leftarrow 1$
  for $a \in \mathcal{A}_f$ in fixed order **do**
    if $\widehat{J}_t(a) < \widehat{J}_t(a_t)$ **then**
      $a_t \leftarrow a$
    end if
  end for
else
  $a_t \leftarrow \arg\min_{a \in \mathcal{A}_f} \widehat{J}_t(a)$
end if
return $\theta_{t+1} = \theta_t + a_t \Delta_t$

E.1 Trajectory Cache and Candidate Scores

Every client begins a round from the same global student. Before local optimization, it randomly selects four training responses and at most 16 evenly spaced prediction positions per response. All candidates use the same selected prefixes. Reusing the training responses avoids additional trajectory generation; candidate evaluation requires forward passes but no optimizer steps.

A fixed vocabulary partition. At each selected prefix, let S be the union of the teacher's and starting student's top-16 token IDs. The cache stores their probabilities on S and a tail bucket containing the remaining probability mass. For any distribution p, this tail is $1 - \sum_{v \in S} p_v$, where p_v is the full-softmax probability; the retained tokens are not renormalized. The same partition is used for every candidate.

Teacher agreement is measured by the teacher-to-candidate forward KL, averaged within each client and then combined using the aggregation weights. Predictive change is measured by the starting-student-to-candidate forward KL, with a separate constraint on each client's cache mean. Both quantities refer to these compressed distributions.

E.2 Selection Rule

The candidate set is $\{0.5, 1, 2, 3, 5, 10\}$. If $a = 1$ is feasible, it is the incumbent and is replaced only by a feasible candidate with a strictly lower teacher score; equal scores therefore preserve FedAvg. If

$a = 1$ is infeasible, the server chooses the lowest-score feasible candidate. If no candidate is feasible, the server retains the round's starting model. Score-comparison and feasibility tolerances are 10^{-9} and 10^{-8}, respectively; Algorithm 1 omits these tolerances for clarity.

E.3 Coordinate-Level Meaning for LoRA

Aggregation and scaling operate on the uploaded LoRA factors, which share an initialization. We use rank 8, $\alpha = 16$, zero dropout, and all-linear target modules. If a layer has factors B_t, A_t and increments $\Delta B, \Delta A$, then

$$(B_t + a\Delta B)(A_t + a\Delta A) - B_t A_t = a(\Delta B A_t + B_t \Delta A) + a^2 \Delta B \Delta A. \tag{80}$$

A scaled factor increment therefore need not equal a scaled merged dense update. The local-response analysis can be expressed in trainable coordinates, while the scalar construction abstracts away this factor-product geometry. FedTOPS uses no additional correction direction. We use PEFT's default LoRA initialization (Kaiming-uniform A and zero-initialized B), shared across clients, and target all attention and MLP linear projections (`q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj`, and `down_proj`), excluding the language-model head. Aggregation and server scaling operate directly on the LoRA factors without canonicalization or re-factorization, and evaluation loads the resulting adapter through vLLM without merging it into the frozen base weights.

E.4 Execution and Data Locality

Our experiments simulate logical clients on one GPU. In a distributed implementation, each client could evaluate candidates on its own cache and return scalar scores and trust statistics. We do not evaluate that deployment here. Each candidate forward pass processes the cached prefix, with the language-model head evaluated only at selected positions. Timing and forward-pass counts are given in Appendix G.3.

F EXPERIMENTAL SETUP

This appendix specifies the data partition, training procedure, and evaluation protocol used in Section 4.

F.1 MODELS AND CLIENT DATA

We distill Qwen3-8B into Qwen3-0.6B-Base and Qwen3-1.7B-Base, using the 0.6B model for the main study (Yang et al., 2025). Client prompts are drawn from MATH (Hendrycks et al., 2021), using three disjoint subject groups: C1 contains prealgebra and algebra; C2 contains number theory and counting and probability; C3 contains intermediate algebra and precalculus. Each client receives 570 prompts. The partition is fixed with seed 42 and shared across methods. Client weights are proportional to their prompt counts, giving $w_i = 1/3$. Although the prompts remain fixed, students generate new responses during training. We use the official training split of `EleutherAI/hendrycks_math`, partitioned into three subject-based clients, with 570 OPD prompts per client selected without replacement using proportional subject–difficulty stratification (seed 42), disjoint from the reserved warm-up and probe subsets. Preprocessing removes malformed examples, examples without an extractable and classifiable boxed answer, and duplicate questions across splits; during training, chat-formatted prompts exceeding 2,048 tokens are filtered rather than truncated, and each client draws batches of 32 prompts from a persistent shuffled data loader, generating one response per prompt with temperature 1, top-$p = 1$, and a 4,096-token response limit. Details can be found in the anonymous codes.

F.2 TRAINING AND BASELINES

Within each student size, all trained methods optimize rank-8 LoRA adapters from the same initialization with AdamW (Table 5). We denote $\eta = 3 \times 10^{-6}$ by 1×LR and 9×10^{-6} by 3×LR. FedTOPS uses Algorithm 1, with trust budget 0.05 for all student models.

Table 5: Training configuration. The local learning rate η controls client optimization; the server multiplier a scales the aggregated increment.

Setting	Value
Participation / rounds	All 3 clients / 50 rounds
Rollouts per round	32 per client, 1 response per prompt
Mini-batch / micro-batch	8 / 4; one optimization epoch
Local learning rate	0.6B: 3×10^{-6} or 9×10^{-6}; 1.7B: 3×10^{-6}
Optimizer	AdamW, $\beta = (0.9, 0.999)$, weight decay 0.01, gradient clipping 1
Scheduler	Cosine; no warmup
LoRA	Rank 8, $\alpha = 16$, dropout 0, all-linear target modules
Aggregation	Equal-weight averaging in shared LoRA factor coordinates
Training decoding	Non-thinking, temperature 1, top-$p = 1$
Training length limits	Prompt 2,048 tokens; response 4,096 tokens
Data / training seed	42 / 42

Training budgets. Each Local model uses 1,600 rollouts and 200 AdamW mini-batch steps. FedAvg and FedTOPS use the same per-client budget: 4,800 rollouts and 600 client steps over 50 rounds system-wide. Centralized uses 4,800 rollouts and 600 consecutive steps on pooled prompts. These totals match across methods, while sequential update depth differs. Token counts and training time depend on response length (Appendix G.3).

Reference models. Local mean averages the three independently trained client-model scores. Base is an untrained, adapter-free Qwen3-0.6B-Base model, evaluated once and repeated in both learning-rate blocks. The 1.7B study compares Centralized, Local, FedAvg, and FedTOPS.

F.3 EVALUATION

Benchmarks. We evaluate on six mathematical reasoning benchmarks, totaling 1,546 problems and 12,368 sampled responses per model (Table 6). Client-private test sets are excluded. All methods use the same evaluation prompt format, sampling protocol, and answer grader.

Table 6: Evaluation benchmarks. Each problem receives eight sampled responses.

Benchmark	Problems	Responses
MATH500	500	4,000
AMC23	40	320
Minerva	272	2,176
OlympiadBench	674	5,392
AIME24	30	240
AIME25	30	240
Total	1,546	12,368

Evaluation prompt. Evaluation is zero-shot. Each problem is supplied as a single user message, with the following instruction appended:

```
Please reason step by step, and put your final answer within
\boxed{}.
```

Chat template and sampling. No separate system message is supplied to the tokenizer. We render the message with its chat template using `add_generation_prompt=True` and `enable_thinking=False`. Each problem receives eight responses sampled with temperature 1, top-$p = 0.8$, top-$k = -1$, minimum probability 0, seed 0, and a maximum response length of 8,192 tokens. Answers are extracted with `qwen_extraction_math_verify_0p8_latex_v2` and checked with `math-verify` 0.8.0.

Metrics. Let $c_{bjm} \in \{0, 1\}$ indicate whether response m to problem j in benchmark b is correct. For a benchmark with n_b problems,

$$\mathrm{Avg@8}_b = \frac{100}{8n_b} \sum_{j=1}^{n_b} \sum_{m=1}^{8} c_{bjm}, \qquad \mathrm{Pass@8}_b = \frac{100}{n_b} \sum_{j=1}^{n_b} \mathbf{1}\left\{ \sum_{m=1}^{8} c_{bjm} > 0 \right\}. \tag{81}$$

Avg@8 measures average response correctness, whereas Pass@8 measures the fraction of problems solved at least once. Cap-hit rate is the percentage of responses reaching the generation length limit. Macro scores average the six benchmark scores with equal weights, including for cap-hit rate. Aggregation precedes rounding, and all reported scores are percentages.

Statistical units. Each configuration is trained once with seed 42. The eight responses per problem measure sampling variability within an evaluation, and Local mean averages client models rather than independent training repetitions. For the paired source intervention, bootstrap resampling uses the same question indices for all three evaluated models and retains all eight responses to each sampled question. These intervals describe evaluation uncertainty conditional on the trained models.

F.4 Resources and Software

Each experiment runs on one RTX PRO 6000 Blackwell Server GPU with 96 GB memory. The three clients are simulated on a single machine; additional GPUs are used for independent experiments. Table 7 identifies the model and training-data versions. The grading dependencies are `latex2sympy2_extended` 1.10.2, `sympy` 1.14.0, and ANTLR runtime 4.9.3. Our implementation builds on verl; the audited environment uses PyTorch 2.8.0 (CUDA 12.8), Transformers 4.57.1, PEFT 0.20.0, FlashAttention 2.8.3, and vLLM 0.11.0 for rollout generation and evaluation.

Table 7: Model and data revisions for training. Exact identifiers pin the resources independently of later repository updates.

Resource	Revision
Qwen3-0.6B-Base	`da87bfb608c14b7cf20ba1ce41287e8de496c0cd`
Qwen3-1.7B-Base	`ea980cb0a6c2ae4b936e82123acc929f1cec04c1`
Qwen3-8B	`b968826d9c46dd6066d109eabc6255188de91218`
EleutherAI/hendrycks_math	`7811dc4cf37bd3a0384adc84f33da0695629b4dc`

The Base reference uses the same sampling protocol and prompt template, with different engine batching. Its checkpoint revision was not recorded. Generation settings and available artifact identifiers are provided in the accompanying reproducibility materials.

G Additional Experimental Analyses

We examine the effect of aggregate scaling, the behavior of the selector, and the influence of rollout sources on subsequent learning. We then characterize computational cost and response length under the common evaluation protocol.

G.1 Selection Across Communication Rounds

Table 8 complements the round-level traces in Figure 4. Candidate and cache settings are given in Table 5; Appendix E specifies scoring and tie handling.

Table 8: Candidate selection over 50 rounds with the 0.6B student. The trust budget is fixed within each run.

Quantity	1×LR	3×LR
Trust budget δ	0.05	0.05
Selected $a = 0.5$	2	3
Selected $a = 1$	3	9
Selected $a = 2$	1	7
Selected $a = 3$	3	8
Selected $a = 5$	4	13
Selected $a = 10$	37	10
Infeasible candidates (of 300)	0	25
Rounds with infeasible candidates	0	18
Holds	0	0

At 1×LR, all candidates are feasible throughout training, yet the selected multiplier changes across rounds. Thus teacher-score comparisons alone can favor different scales. At 3×LR, 25 of the 300 candidates are infeasible across 18 rounds, and $a = 10$ is selected less often. The traces characterize adaptation within each setting.

Figure 4 displays each candidate's teacher score relative to $a = 1$ in the same round. Both panels use a shared symmetric-logarithmic color scale, linear within ± 0.002 nats, without clipping or smoothing. This makes the selected scales and feasibility boundaries comparable over training.

G.2 Rollout-Source Intervention

The intervention in Section 4.4 uses a common round-9 receiver checkpoint and frozen sources constructed from the same round-10 aggregate increment. We evaluate the receiving students on MATH500. Their cap-hit rates are 51.85% with source $a = 3$ and 53.30% with source $a = 1$, compared with 52.38% at the common start.

Paired evaluation. Table 9 reports Avg@8 differences and paired 95% bootstrap intervals. We resample the 500 questions 5,000 times, retaining each question's eight responses together and using the same question indices for all three models. These intervals characterize evaluation uncertainty for the saved models, not variation across training seeds. Differences use unrounded statistics and may differ from subtraction of the displayed endpoint scores.

Table 9: Paired differences in receiver Avg@8 on MATH500 (pp).

Comparison	Difference	95% paired interval
Source $a = 3$ receiver − common start	1.20	[0.27, 2.10]
Source $a = 1$ receiver − common start	0.20	[−0.70, 1.07]
Source $a = 3$ receiver − source $a = 1$ receiver	1.00	[0.18, 1.82]

G.3 Computational Cost

The auxiliary forwards operate on at most 12 cached responses and 192 prediction positions per round. Table 10 reports caching and finalization time separately.

Forward computation. Cache construction adds 12 student and 12 teacher prefix forwards. Scoring a reference and six candidates adds 84 student forwards: 96 student and 12 teacher forwards in total per round. Teacher distributions are reused across candidates. Each pass processes the actual prefix, with only the LM head restricted to selected positions; cost therefore depends on prefix length.

Table 10: Auxiliary computation for FedTOPS. Cache construction and candidate scoring/saving are timed over the training loop. Share is their sum divided by the training-loop interval; initialization and evaluation are excluded.

Student / LR	Cache (s)	Score/save (s)	Sum (s)	Training (s)	Share (%)
0.6B / 1×	188.69	979.36	1168.05	13837	8.44
0.6B / 3×	180.56	764.11	944.67	6227	15.17
1.7B / 1×	171.22	754.80	926.03	7598	12.19

Measured time and training tokens. Finalization includes scoring and saving; timing shares use the training-progress interval, excluding initialization and evaluation. Equal rollout counts need not imply equal training time because response lengths vary. The response-mask token totals are 6,445,612 for 0.6B at 1×LR, 5,164,606 for 0.6B at 3×LR, and 3,748,317 for 1.7B. These counts cover all 150 client-round batches and exclude prompt tokens and auxiliary forwards. The single-machine experiments characterize computation; they do not measure network traffic or comparative peak memory.

G.4 Response Length and Generation Limits

Table 11 summarizes generation length. FedTOPS reduces cap hits at the reference learning rate for both student sizes, while the 0.6B high-rate run has a similar macro cap-hit rate to FedAvg despite higher accuracy. Cap hits characterize output behavior, not supervision quality.

Table 11: Mean response tokens / cap-hit rate (%), equally averaged across benchmarks. The evaluation cap is 8,192 tokens. The 0.6B Base result is repeated for reference.

Method	0.6B, 1×LR	0.6B, 3×LR	1.7B, 1×LR
Base student	4366.95 / 50.27	4366.95 / 50.27	—
Centralized	3675.20 / 38.18	4535.74 / 48.62	1537.92 / 8.98
Local mean	4807.97 / 54.58	4120.63 / 44.23	3664.13 / 38.94
FedAvg	4821.79 / 54.88	3164.08 / 31.16	3644.79 / 38.66
FedTOPS	3830.62 / 38.31	3334.40 / 31.23	1617.09 / 9.89

H CLIENT-LEVEL ACCURACY AND GENERATION STATISTICS

The following tables complement the main accuracy comparisons with individual Local models and benchmark-level cap-hit rates. Local C1, C2, and C3 correspond to the subject partitions in Appendix F; Local mean averages their scores. Avg@8 and Pass@8 tables contain only these client results, while cap-hit tables compare all original training methods and the available Base reference.

All scores are percentages; Macro weights benchmarks equally. Best and second-best values among the rows in each table are bold and underlined. MATH, AMC, and Olympiad denote MATH500, AMC23, and OlympiadBench.

H.1 0.6B STUDENT AT THE REFERENCE LEARNING RATE

Table 12: 0.6B student, 1×LR: individual Local models, Avg@8 (↑).

Method	MATH	AMC	Minerva	Olympiad	AIME24	AIME25	Macro
Local C1	**12.93**	4.06	3.58	**4.58**	**0.83**	**0.42**	**4.40**
Local C2	11.47	**4.69**	**3.68**	4.08	0.42	0.00	4.06
Local C3	12.65	3.12	3.45	4.30	0.00	0.00	3.92
Local mean	12.35	3.96	3.57	4.32	0.42	0.14	4.13

Table 13: 0.6B student, 1×LR: individual Local models, Pass@8 (↑).

Method	MATH	AMC	Minerva	Olympiad	AIME24	AIME25	Macro
Local C1	52.40	**27.50**	18.38	**23.00**	**3.33**	**3.33**	**21.32**
Local C2	50.40	25.00	**19.49**	22.11	**3.33**	0.00	20.05
Local C3	**53.00**	15.00	16.54	22.55	0.00	0.00	17.85
Local mean	51.93	22.50	18.14	22.55	2.22	1.11	19.74

Table 14: 0.6B student, 1×LR: cap-hit rate (↓).

Method	MATH	AMC	Minerva	Olympiad	AIME24	AIME25	Macro
Base student	50.68	46.88	48.85	51.91	52.08	51.25	50.27
Centralized	30.12	35.31	41.13	**37.11**	**42.50**	**42.92**	**38.18**
Local C1	53.12	57.19	50.69	53.36	55.00	49.17	53.09
Local C2	55.85	55.31	52.53	56.31	59.17	53.75	55.49
Local C3	54.43	57.81	51.98	54.75	55.83	56.25	55.17
Local mean	54.47	56.77	51.73	54.80	56.67	53.06	54.58
FedAvg	54.27	55.62	51.79	54.27	58.33	55.00	54.88
FedTOPS	**18.75**	**30.31**	**10.48**	38.22	69.17	62.92	38.31

H.2 0.6B STUDENT AT THE LARGER LEARNING RATE

Table 15: 0.6B student, 3×LR: individual Local models, Avg@8 (↑).

Method	MATH	AMC	Minerva	Olympiad	AIME24	AIME25	Macro
Local C1	**33.70**	**16.56**	**8.32**	**10.65**	0.83	0.42	**11.75**
Local C2	22.15	10.94	6.16	6.36	0.83	**1.67**	8.02
Local C3	24.45	11.88	6.76	7.10	**1.67**	0.00	8.64
Local mean	26.77	13.12	7.08	8.04	1.11	0.69	9.47

Table 16: 0.6B student, 3×LR: individual Local models, Pass@8 (↑).

Method	MATH	AMC	Minerva	Olympiad	AIME24	AIME25	Macro
Local C1	**69.20**	**47.50**	**26.47**	**35.01**	6.67	3.33	**31.36**
Local C2	64.80	42.50	23.16	27.60	6.67	**10.00**	29.12
Local C3	67.00	45.00	25.00	28.64	**10.00**	0.00	29.27
Local mean	67.00	45.00	24.88	30.42	7.78	4.44	29.92

Table 17: 0.6B student, 3×LR: cap-hit rate (↓).

Method	MATH	AMC	Minerva	Olympiad	AIME24	AIME25	Macro
Base student	50.68	46.88	48.85	51.91	52.08	51.25	50.27
Centralized	29.62	46.88	20.08	51.41	70.00	73.75	48.62
Local C1	25.00	29.06	35.80	33.10	40.83	38.33	33.69
Local C2	44.85	50.94	50.14	51.19	53.75	56.25	51.19
Local C3	40.88	43.44	48.39	49.59	53.33	51.25	47.81
Local mean	36.91	41.15	44.78	44.63	49.31	48.61	44.23
FedAvg	21.93	**26.25**	32.26	29.88	**40.42**	**36.25**	**31.16**
FedTOPS	**13.68**	28.44	**7.90**	**29.47**	53.75	54.17	31.23

H.3 1.7B STUDENT AT THE REFERENCE LEARNING RATE

Table 18: Qwen3-1.7B-Base student after distillation: individual Local models, Avg@8 (↑).

Method	MATH	AMC	Minerva	Olympiad	AIME24	AIME25	Macro
Local C1	29.43	15.31	9.65	**12.31**	2.50	1.25	11.74
Local C2	**30.03**	**17.19**	**10.62**	12.18	**3.33**	0.83	**12.36**
Local C3	28.78	14.37	10.02	11.48	2.92	**2.08**	11.61
Local mean	29.41	15.62	10.09	11.99	2.92	1.39	11.90

Table 19: Qwen3-1.7B-Base student after distillation: individual Local models, Pass@8 (↑).

Method	MATH	AMC	Minerva	Olympiad	AIME24	AIME25	Macro
Local C1	74.60	55.00	33.82	**40.80**	**16.67**	10.00	38.48
Local C2	**76.40**	**65.00**	35.66	40.06	13.33	6.67	**39.52**
Local C3	74.60	47.50	**36.03**	39.91	13.33	**13.33**	37.45
Local mean	75.20	55.83	35.17	40.26	14.44	10.00	38.48

Table 20: Qwen3-1.7B-Base student after distillation: cap-hit rate (↓).

Method	MATH	AMC	Minerva	Olympiad	AIME24	AIME25	Macro
Centralized	**4.17**	**6.25**	2.90	**8.46**	20.42	**11.67**	**8.98**
Local C1	37.05	38.75	32.72	39.52	40.42	42.08	38.42
Local C2	37.77	39.69	32.31	39.47	39.58	43.75	38.76
Local C3	37.50	40.31	32.86	40.49	44.17	42.50	39.64
Local mean	37.44	39.58	32.63	39.82	41.39	42.78	38.94
FedAvg	38.15	38.44	33.13	40.13	40.42	41.67	38.66
FedTOPS	**4.17**	9.06	**2.57**	8.55	**19.17**	15.83	9.89