FED-GRPO: REWARD-SIGNAL-DRIVEN FEDERATED GROUP RELATIVE POLICY OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Language Models (LLMs) have shown strong reasoning capabilities when fine-tuned with reinforcement learning (RL), particularly through Group Relative Policy Optimization (GRPO). However, existing GRPO methods assume centralized access to training data, which may not hold in practice due to privacy or regulatory constraints. To this end, we propose Fed-GRPO, a federated GRPO training framework that addresses these privacy constraints by enabling collaborative reasoning training without sharing raw data, which leverages the reward statistics naturally produced during GRPO training as zero-cost signals to guide aggregation, local training, and communication. Fed-GRPO contains three reward-signal-driven mechanisms: (i) *signal-weighted aggregation* that weights clients by their reward standard deviation, prioritizing clients with stronger learning signals; (ii) *global reward calibration* that re-weights per-prompt objectives based on the local-global reward gap, steering each client toward its relative weaknesses; and (iii) *adaptive sparse communication* that allocates bandwidth based on the informativeness of each client's update. Extensive experiments on mathematical reasoning tasks demonstrate that Fed-GRPO achieves the best performance among all federated methods, clearly outperforms FedAvg and approaches centralized training performance, while losslessly reducing communication by $32\times$ and supporting up to $621\times$ compression under tight bandwidth budgets with only graceful accuracy degradation.

1 INTRODUCTION

Large Language Models (LLMs) have shown strong reasoning capabilities when fine-tuned with reinforcement learning (RL), in particular through Group Relative Policy Optimization (GRPO) Shao et al. (2024), which computes advantages from group-relative reward comparisons without a separate critic model. Building on this, DeepSeek-R1 Guo et al. (2025a) demonstrates that GRPO with simple rule-based rewards can produce strong reasoning behavior. Since then, a growing line of work has improved various aspects of GRPO-style training, including training stability Yu et al. (2025), advantage estimation granularity Liu et al. (2025); Sun et al. (2025); Yu et al. (2026), and multi-reward optimization Liu et al. (2026). However, these methods almost universally assume centralized access to training data. This assumption is increasingly unrealistic in high-value reasoning domains, where data is distributed across institutions, platforms, or jurisdictions. For example, hospitals, financial firms, educational platforms, and enterprise coding environments may each possess valuable reasoning data, but cannot pool it due to privacy, ownership, or regulatory constraints.

Federated Learning (FL) McMahan et al. (2017), which enables collaborative model training across institutions without sharing raw data, offers a natural solution. However, federated GRPO is not as simple as running GRPO inside a standard FL loop. The core difficulty is that, in GRPO, a client's usefulness is governed by its *reward signal*. This makes federated GRPO fundamentally different from conventional supervised FL. In conventional supervised FL, client importance is usually determined by sample counts or controlled through parameter-, gradient-, or representation-space drift correction McMahan et al. (2017); Li et al. (2020a); Karimireddy et al. (2020); Li et al. (2021). In GRPO, however, a large client may be nearly saturated and provide little learning signal, whereas a smaller client with high reward dispersion may produce much more informative policy gradients. Thus, the server must answer a different question: *which clients are currently useful for improving reasoning?*

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.

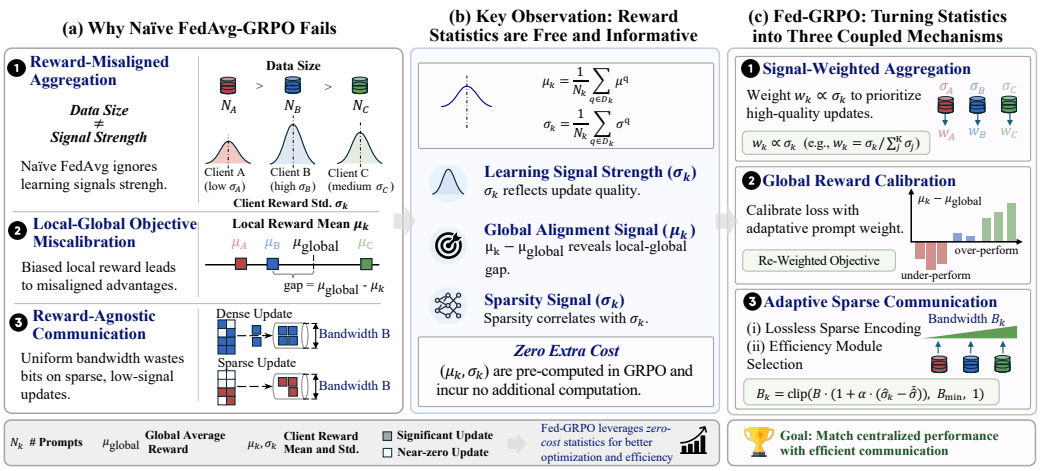


Figure 1: **Motivation of Fed-GRPO.** Naïve FedAvg-GRPO suffers from three coupled issues: (1) reward-misaligned aggregation, (2) local–global objective miscalibration, and (3) reward-agnostic communication. We observe that the reward statistics (μ_k, σ_k), computed by GRPO at *zero extra cost*, are precisely the signals needed to address all three: σ_k guides both signal-weighted aggregation and adaptive bandwidth allocation, while the local–global gap on μ_k calibrates per-prompt objectives.

Notably, a closer examination shows that this reward-dependent notion of usefulness extends far beyond simple client aggregation; it fundamentally alters the entire federated training loop (see Figure 1(a)). First, **aggregation is reward-misaligned**. FedAvg McMahan et al. (2017) and its follow-up work Li et al. (2020a); Karimireddy et al. (2020); Li et al. (2021) weight clients by dataset size or uniformly, at most mitigating drift in parameter-, gradient-, or representation-space. These methods do not capture the *reward-space signal* in GRPO, making the resulting global update misaligned with the clients that are most useful for RL training. Second, **local objective is reward-miscalibrated**. Each client performs GRPO using only local reward statistics, and therefore lacks a global view of prompt difficulty. This can cause clients to under-train globally lagging prompts and over-train prompts already mastered locally. Third, **communication remains reward-agnostic**. Each federated round requires transmitting billion-parameter models or updates, creating a severe bandwidth bottleneck. Existing compression methods Lin et al. (2018); Guastella et al. (2025); Sarfi et al. (2025) usually rely on parameter-space heuristics, which may over-compress reward-informative clients while wasting bandwidth on uninformative ones.

To investigate these challenges, we conduct a diagnostic experiment by training Qwen2.5-3B-Instruct Yang et al. (2025b) on the MATH dataset Hendrycks et al. (2021) under standard FedAvg-GRPO with a non-IID partition (Section 3). We observe three properties that connect directly back to the three challenges above: (1) per-client reward standard deviations σ_k vary substantially across clients within the same round, suggesting that learning signal strength—not dataset size—should drive aggregation; (2) per-client reward means μ_k diverges significantly from the global average, revealing a persistent local-global gap that biases local advantage estimation and misallocates training effort; and (3) GRPO updates are naturally sparse, with over 96.2% of parameters unchanged after local training, and the update density correlates with σ_k, opening the door to adaptive sparse communication that gives more bandwidth to clients with stronger learning signals.

These observations point to a simple but powerful principle: *the reward statistics computed by GRPO contain precisely the information needed to guide federated aggregation, local objective calibration, and communication allocation*. Importantly, these statistics are available at essentially **zero additional training cost**. Building on this insight, we propose **Fed-GRPO**, a federated GRPO training framework that turns these reward statistics into three coupled mechanisms: (i) **signal-weighted aggregation** weights clients by σ_k instead of dataset size, so that clients with stronger learning signals contribute more to the global update; (ii) **global reward calibration** re-weights per-prompt objectives based on the local-global reward gap, steering each client toward prompts where it lags behind and away from those it has already mastered relative to the global average; and (iii) **adaptive sparse communication** allocates bandwidth based on σ_k and compresses updates by exploiting both the natural sparsity of GRPO updates and module-level update efficiency. Our

motivation is illustrated in Figure 1. Extensive experiments across several mathematical reasoning benchmarks demonstrate that Fed-GRPO outperforms all federated baselines by a clear margin and closely matches the centralized upper bound.

We summarize our contributions as: i) We identify a fundamental mismatch between standard FL and GRPO, and reveal a powerful insight: the reward statistics (μ_k, σ_k) inherently computed by GRPO provide zero-cost, perfect signals for federated coordination. ii) We propose **Fed-GRPO**, seamlessly integrating *signal-weighted aggregation*, *global reward calibration*, and *adaptive sparse communication* to resolve the aggregation, optimization, and communication bottlenecks in federated GRPO training. iii) We provide a convergence analysis showing that Fed-GRPO converges at rate $\mathcal{O}(1/\sqrt{T})$ to a neighborhood of a stationary point of the calibrated objective, with the neighborhood characterized by two interpretable terms—a *drift floor* from global reward calibration and a *compression floor* from adaptive sparse communication. iv) We empirically validate Fed-GRPO on five math reasoning benchmarks, where it outperforms all federated baselines and approaches centralized GRPO training, losslessly reduces communication by $32\times$, and supports up to $621\times$ compression under tight bandwidth budgets with a graceful accuracy–bandwidth trade-off.

2 Preliminary

2.1 Group Relative Policy Optimization

Group Relative Policy Optimization (GRPO) Shao et al. (2024) is a critic-free LLM post-training method that computes advantages via group-relative reward comparisons, eliminating the memory-intensive value network required by PPO Schulman et al. (2017). Given a question q, GRPO samples a group of rollouts $\{o_1, o_2, \ldots, o_{G_{\text{rol}}}\}$ from the old policy $\pi_{\theta_{\text{old}}}$ and optimizes the policy π_θ by maximizing the following objective:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim\mathcal{D},\{o_i\}_{i=1}^{G_{\text{rol}}}\sim\pi_{\theta_{old}}(\cdot|q)} \left[\frac{1}{G_{\text{rol}}} \sum_{i=1}^{G_{\text{rol}}} \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left(\min\Big(\rho_{i,t}(\theta)\hat{A}_i,\ \text{clip}\left(\rho_{i,t}(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}_i \Big) - \beta\, D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}}) \right) \right], \tag{1}$$

where $\rho_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t}|q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q,o_{i,<t})}$, ϵ is the clipping parameter, and β controls the KL penalty strength. The advantage $\hat{A}_i$ is shared across all tokens in rollout o_i and is computed by normalizing rewards across the group:

$$\hat{A}_i = \frac{r_i - \mu}{\sigma}, \tag{2}$$

where r_i is the reward for rollout o_i, and μ and σ are the mean and standard deviation of rewards within the group.

2.2 Federated Learning

Federated Learning (FL) McMahan et al. (2017) enables collaborative model training across K clients without sharing raw data. In the standard FedAvg McMahan et al. (2017) framework, each communication round t consists of three steps: (1) the server broadcasts the global model $\theta^{(t)}$ to all clients; (2) each client k performs local training on its private dataset $\mathcal{D}_k$ to obtain updated parameters $\theta_k^{(t)}$ and sends them back to the server; (3) the server aggregates client updates via weighted averaging:

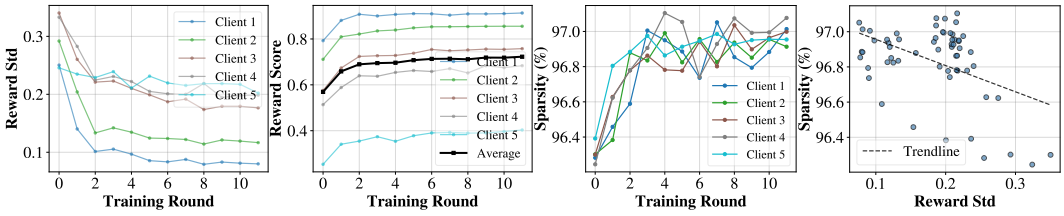
$$\theta^{(t+1)} = \sum_{k=1}^{K} w_k\, \theta_k^{(t)}, \tag{3}$$

where w_k is the aggregation weight for client k, typically set to $\frac{|\mathcal{D}_k|}{\sum_{j=1}^{K}|\mathcal{D}_j|}$ or $\frac{1}{K}$ for uniform weighting. This process repeats until convergence or until reaching the maximum number of rounds.

3 Analysis on Federated GRPO Training

To motivate our design, we conduct a preliminary experiment by training Qwen2.5-3B-Instruct Yang et al. (2025b) on the MATH dataset Hendrycks et al. (2021) with a non-IID partition across 5 clients

based on problem difficulty (one client per difficulty level). We run standard FedAvg with GRPO as the local training algorithm for 12 rounds, with 5 local steps per round. Throughout training, we track the reward mean μ_k and standard deviation σ_k, averaged over all prompts of client k, as well as the overall model update sparsity. Figure 2 summarizes the results, revealing three key observations that directly motivate our design.



(a) (b) (c) (d)

Figure 2: FedAvg-GRPO training dynamics on the MATH dataset with 5 clients split by difficulty. (a) Per-client reward standard deviation σ_k varies by up to $4\times$ across clients within the same round, indicating large differences in learning signal strength. (b) Per-client reward mean μ_k diverges significantly across clients, revealing a persistent local-global gap. (c) Average model update sparsity across clients: over 96.2% of parameters remain unchanged after local GRPO training. (d) Correlation between σ_k and model update sparsity: clients with stronger learning signals produce denser updates.

Observation 1: Learning signals vary drastically across clients. Figure 2a shows that σ_k varies by up to $4\times$ across clients within the same round. Since σ_k reflects the overall strength of a client's learning signal in GRPO, such variation implies that actual gradient contributions also differ substantially. However, standard FedAvg ignores this and weights all clients equally by dataset size. *This motivates signal-weighted aggregation*, which weights clients by σ_k to prioritize those with stronger learning signals.

Observation 2: Local reward means diverge from the global average. Figure 2b shows persistent gaps between per-client reward means μ_k and the global average throughout training. A client specializing in easy problems maintains a high μ_k but contributes little to global improvement, whereas a client on hard problems has a low μ_k yet holds the most room for improvement. Under standard GRPO, each client only sees its local reward statistics and cannot distinguish "locally hard but globally easy" prompts from "locally easy but globally hard" ones, resulting in misallocated training effort. *This motivates global reward calibration*, which uses the gap between μ_k and the global reward mean to re-weight per-prompt objectives so that each client focuses on its relative weaknesses.

Observation 3: GRPO updates are naturally sparse and correlate with σ_k. Figure 2c shows that over 96.2% of parameters remain unchanged after local GRPO training, and this high sparsity is remarkably stable across all training rounds. Moreover, Figure 2d reveals a strong negative correlation between σ_k and model update sparsity—clients with stronger learning signals leave fewer parameters unchanged and thus produce denser updates. *This motivates adaptive sparse communication*, which gives more bandwidth to high-σ_k clients whose updates carry more information, while compressing low-σ_k clients more aggressively, creating a natural alignment between update quality and communication cost.

Taken together, these observations show that the reward statistics μ_k and σ_k, computed by GRPO at *no extra cost*, carry rich information about the aggregation, training, and communication needs of each client. In the next section, we describe how Fed-GRPO leverages these statistics to build a unified framework with three key mechanisms that jointly address the three challenges above.

4 METHODOLOGY

Guided by the observations in Section 3, we design **Fed-GRPO**, a framework that leverages the reward statistics naturally produced during GRPO training to improve aggregation, local training, and communication for federated GRPO training. Fed-GRPO contains three key mechanisms: (i) **signal-weighted aggregation** weights clients by σ_k instead of dataset size, so that clients with stronger learning signals contribute more; (ii) **global reward calibration** re-weights per-prompt objectives

based on the local-global reward gap, steering each client toward prompts where it lags behind; and (iii) **adaptive sparse communication** allocates bandwidth based on σ_k and compresses updates by exploiting sparsity and efficiency-based module selection. We detail each mechanism below.

4.1 Signal-Weighted Aggregation

Standard FedAvg (Eq. (3)) weights clients by dataset size (or uniformly), which ignores the *quality* of each client's update during GRPO training. In GRPO training, dataset size is a poor proxy for update quality while the reward standard deviation σ_k is a direct measure of the learning signal strength: a client with high σ_k produces more informative policy gradients and thus should have a larger influence on the global update. To this end, we propose to weight clients by their reward standard deviation σ_k, which reflects the diversity of rewards and hence the strength of the learning signal. Specifically, the aggregation weight for client k used in Eq. (3) is replaced by:

$$w_k = \frac{\sigma_k}{\sum_{j=1}^{K} \sigma_j}, \tag{4}$$

where σ_k is the reward standard deviation from client k's local GRPO training in the current round.

This weighting has two properties: (1) it is **zero-cost**, since σ_k is already computed during GRPO training for advantage normalization; and (2) it is **adaptive**, meaning the aggregation weights adjust dynamically throughout the training process as each client's σ_k naturally evolves (see Figure 3).

4.2 Global Reward Calibration

In non-IID federated settings, clients have varying data distributions and varying levels of mastery. Under standard GRPO, each client optimizes its policy based solely on local reward outcomes, lacking any awareness of the global training progress. As a result, a client cannot objectively evaluate its own performance—it does not know whether a locally low reward implies an inherently hard problem or a severe lag behind the global model. This lack of global perspective leads to inefficient local training: clients waste valuable computation over-training on prompts they have already mastered relative to the global average, while under-training on their relative weaknesses.

To address this, we introduce **global reward calibration**, which uses the global reward statistics to re-weight per-prompt objectives during local training. After each aggregation round, the server computes global reward statistics from the clients' reported metrics and broadcasts them back to all clients:

$$\mu_{\text{global}} = \frac{1}{K}\sum_{k=1}^{K} \mu_k, \qquad \sigma_{\text{global}} = \sqrt{\frac{1}{K-1}\sum_{k=1}^{K} (\mu_k - \mu_{\text{global}})^2}. \tag{5}$$

Each client then applies a per-prompt weight based on the gap between its local reward mean and the global mean, encouraging it to focus on prompts where it falls behind and reducing effort on prompts it has already mastered relative to others. Formally, for a prompt q with local mean reward $\bar{r}_q$ (averaged over G_{rol} rollouts), the calibration weight is:

$$\omega_q = 1 + \lambda \cdot \tanh\left(\frac{\mu_{\text{global}} - \bar{r}_q}{\sigma_{\text{global}} + \varepsilon_\sigma}\right), \tag{6}$$

where $\lambda \in [0, 1]$ controls the calibration strength, and ε_σ is a small constant to prevent division by zero. Then, the local GRPO training objective will be replaced by a weighted version that incorporates ω_q:

$$\mathcal{L}_k = \frac{\sum_q \omega_q \cdot \ell_q}{\sum_q \omega_q}, \tag{7}$$

where k denotes the client index, and $\ell_q := \frac{1}{G_{\text{rol}}}\sum_{i=1}^{G_{\text{rol}}} \frac{1}{|o_i|}\sum_t \big[\min(\rho_{i,t}\hat{A}_i, \text{clip}(\rho_{i,t}, 1-\epsilon, 1+\epsilon)\hat{A}_i) - \beta D_{\text{KL}}\big]$ is the standard GRPO objective for prompt q.

The intuition behind this design is straightforward: when a client's local performance on a prompt is below the global average, the weight ω_q increases, amplifying the gradient contribution from that prompt and encouraging the client to focus on its weaknesses; conversely, when a client already exceeds the global average on a prompt, ω_q decreases, reducing unnecessary optimization on that prompt. The $\tanh$ function ensures that weights are bounded within $[1-\lambda, 1+\lambda]$, providing a smooth and stable calibration effect while preventing extreme values. In the first round, since global statistics are unavailable, all weights default to 1 and the method reduces to standard GRPO.

4.3 Adaptive Sparse Communication

The communication cost of transmitting full model parameters at each round is a major bottleneck in federated GRPO training, especially as model sizes grow. Fortunately, the natural sparsity of GRPO updates (over 96.2% unchanged parameters, see Figure 2c) and the correlation between σ_k and update density (see Figure 2d) suggest an opportunity for **adaptive sparse communication** that allocates bandwidth based on the informativeness of each client's update. Specifically, clients with higher σ_k—whose updates carry more learning signal and receive larger aggregation weights—are given more communication bandwidth, while low-σ_k clients (whose updates are mostly zero) are compressed more aggressively. This creates a natural alignment: low-signal clients are both compressed more and weighted less, minimizing the impact of compression-induced information loss.

Formally, given a target average per-client communication budget ratio $B \in (0, 1]$ (defined as the fraction of changed parameters to retain), we determine the adaptive budget ratio B_k for client k based on its normalized signal strength $\hat{\sigma}_k = \sigma_k / \max_j \sigma_j$ as follows:

$$B_k = \text{clip}\left(B \cdot \left(1 + \alpha \cdot (\hat{\sigma}_k - \bar{\hat{\sigma}})\right),\ B_{\min},\ 1\right), \tag{8}$$

where $\bar{\hat{\sigma}} = \frac{1}{K}\sum_j \hat{\sigma}_j$ is the mean normalized value across clients, $B_{\min}$ is a minimum budget to ensure that even low-signal clients can transmit some updates, and α controls the adaptation strength.

To meet the target retention ratio B_k, we apply a two-stage compression strategy that exploits the structural properties of GRPO updates. First, we apply a *lossless* sparse encoding that extracts only the parameters that changed after local training, leveraging the fact that most parameters remain unchanged during GRPO training (see Figure 2c). This alone achieves significant compression (approximately $32\times$) without any loss in accuracy, keeping 100% of the updated parameters ($B_k = 1.0$). When $B_k < 1.0$, we further apply lossy compression by selecting a subset of modules to transmit based on their *update efficiency*, ensuring that we preserve the most informative updates while dropping parameters until reaching the target fraction B_k. The update efficiency of a module is defined as the update energy per transmitted parameter:

$$E_m = \frac{\|\Delta_m\|_F^2}{\|\Delta_m\|_0}, \tag{9}$$

where Δ_m is the update for module m and $\|\Delta_m\|_0$ is the number of changed parameters in module m. We then sort modules by efficiency E_m in increasing order and greedily remove modules until the final transmitted parameter count roughly equals the fraction B_k of the changed parameters. This efficiency-based selection ensures that we drop only the least informative updates, maximizing the retained learning signal under tight communication constraints. Importantly, since B_k is adaptive to σ_k, clients with stronger learning signals are more likely to retain more of their updates, while clients with weaker signals (which contribute less to the global update) are compressed more aggressively, creating a natural synergy between update quality and communication cost.

4.4 Fed-GRPO Algorithm

Fed-GRPO integrates the above three mechanisms into a unified training procedure. Specifically, each round begins with the server broadcasting the current global model and global reward statistics to all clients. Each client then performs local GRPO training with global reward calibration, computes its reward statistics, and applies adaptive sparse communication to transmit its compressed update back to the server. The server aggregates the received updates using signal-weighted aggregation and updates the global reward statistics for the next round. This iterative process continues until convergence or until reaching the maximum number of rounds. Algorithm 1 in Appendix A provides a detailed summary.

5 Convergence Analysis

We make the following assumptions, some of which are commonly used in the FL literature Li et al. (2020b); Guo et al. (2025b); Karimireddy et al. (2019), while Assumption 4 adapts a tool from non-stationary stochastic optimization Besbes et al. (2015); Jadbabaie et al. (2015).

Assumption 1. *(L-smoothness) Each local objective $\mathcal{L}_k$ is L-smooth: for any parameters θ, θ',*

$$\mathcal{L}_k(\theta) \le \mathcal{L}_k(\theta') + \langle \theta - \theta', \nabla \mathcal{L}_k(\theta') \rangle_F + \frac{L}{2} \|\theta - \theta'\|_F^2,$$

where $\langle \cdot, \cdot \rangle_F$ and $\| \cdot \|_F$ denote the Frobenius inner product and norm, respectively.

Assumption 2. *(Bounded stochastic gradient) Let $\xi_{k,s}^{(t)}$ be sampled from the k-th client's local data uniformly at random at the s-th training step of round t. The expected squared norm of stochastic gradients is uniformly bounded: $\mathbb{E}\|\nabla \mathcal{L}_k^{(t)}(\theta_{k,s}^{(t)}; \xi_{k,s}^{(t)})\|_F^2 \le G^2$, for all $k = 1, \cdots, K$, $t = 0, \cdots, T-1$, and $s = 1, \cdots, S$.*

Assumption 3. *(Bounded compression error) The adaptive sparse communication operator $\mathcal{C}_k(\cdot)$ in Section 4.3 is contractive path-wise: $\|\mathcal{C}_k(\Delta) - \Delta\|_F^2 \le (1 - B_k)\|\Delta\|_F^2$ for any Δ, where B_k is the budget ratio assigned to client k.*

Assumption 4. *(Bounded average calibration drift) The average round-to-round drift of the global objective along the iterate path is bounded in expectation:*

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[\Delta_{\mathcal{L}}^{(t)}] \;\le\; R, \qquad \Delta_{\mathcal{L}}^{(t)} \;:=\; \big|\mathcal{L}^{(t+1)}(\theta^{(t+1)}) - \mathcal{L}^{(t)}(\theta^{(t+1)})\big|.$$

Here $\mathcal{L}^{(T)}$ denotes any uniformly bounded extension of the round-statistics (e.g., reusing the round-$(T-1)$ weights and calibration); the specific choice does not affect the bound.

Then, we present the convergence rate for Fed-GRPO, with the proof provided in Appendix B.

Theorem 1. *Let Assumptions 1–4 hold and L, G, R be defined therein. For a learning rate η, we have:*

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\big\|\nabla \mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2 \;\le\; 2\sqrt{\frac{D\,M}{T}} \;+\; 2\sqrt{R\,M} \;+\; 4\sqrt{K(1-\bar{B})}\,G^2, \tag{10}$$

where $\mathcal{L}^{(t)}(\theta) = \sum_{k=1}^{K} w_k^{(t)} \mathcal{L}_k^{(t)}(\theta)$ is the global objective at round t, $D := \mathcal{L}^{(0)}(\theta^{(0)}) - \inf_{0 \le t \le T,\, \theta} \mathcal{L}^{(t)}(\theta)$, $\bar{B} = \frac{1}{KT} \sum_{t=0}^{T-1} \sum_{k=1}^{K} B_k^{(t)}$ is the per-client budget ratio averaged over both clients and rounds (which equals the user-specified target B in Eq. (8) when clipping is inactive), and M is a constant depending on L, G, S, K.

According to Theorem 1, Fed-GRPO converges to a neighborhood of a stationary point of the time-varying calibrated objective at rate $\mathcal{O}(1/\sqrt{T})$, where the neighborhood size is determined by two irreducible terms: a *drift floor* of size $\mathcal{O}(\sqrt{RM})$ from the round-to-round change of the calibrated objective, and a *compression floor* of size $\mathcal{O}(\sqrt{K(1-\bar{B})}\,G^2)$ from adaptive sparse communication. When the time-varying global objective is stationary across rounds ($R = 0$) and the encoding is lossless ($\bar{B} = 1$), both floors vanish and Fed-GRPO recovers the standard $\mathcal{O}(1/\sqrt{T})$ rate of FedAvg in the non-convex setting Li et al. (2020b).

6 Experiments

6.1 Experimental Setup

For mathematical reasoning tasks tasks, we train on the MATH dataset Hendrycks et al. (2021), split across K=5 clients by difficulty to form a naturally non-IID partition, and evaluate on five math reasoning benchmarks: GSM8K Cobbe et al. (2021), MATH-500 Hendrycks et al. (2021); Lightman et al. (2023), AIME 2024, AIME 2025, and AMC 2023. The base models are Qwen2.5-3B-Instruct Yang et al. (2025b) and Qwen3-4B-Instruct Yang et al. (2025a), optimized with AdamW Loshchilov & Hutter (2019) (lr 3×10^{-6}) under GRPO with a binary correctness reward, prompt batch size 64, and 8 rollouts per prompt. For coding tasks, we use Qwen2.5-3B-Instruct as the base model, train on the MBPP dataset Austin et al. (2021) ("train" + "validation") and evaluate on the held-out MBPP "test" split and the HumanEval dataset Chen et al. (2021). To construct a non-IID federation setting, we partition the MBPP dataset by the reference-solution length, bucket it into 5 quantile bins, and partition the data across clients with a Dirichlet distribution ($\alpha = 0.5$) over

these buckets. All federated methods run for $T{=}12$ communication rounds with $S{=}5$ local steps; centralized and local-only baselines are trained for an equivalent number of gradient updates. We compare Fed-GRPO against the base model, centralized GRPO (upper bound), local-only GRPO, sample- and uniform-weighted FedAvg McMahan et al. (2017) (FedAvg-SW / FedAvg-UW), two sparse-communication FL methods, SparsyFed Guastella et al. (2025) and SparseLoCo Sarfi et al. (2025), and one Federated GRPO method, FGRPO Chen et al. (2026). Following previous GRPO literature Yu et al. (2025); Guo et al. (2025a), we evaluate the best checkpoint of each method and report avg@32 (32 samples per prompt at temperature 1.0, top-$p{=}0.7$). More details are deferred to Appendix C.

6.2 Main Results

Table 1: Main results on five math reasoning benchmarks using Qwen2.5-3B-Instruct as the base model. FedAvg-SW and FedAvg-UW denote sample-weighted and uniform-weighted FedAvg, respectively. Budget B is the target fraction of changed parameters retained for transmission; Comm. (MB) is the average per-round per-client uplink payload (BF16 values plus 4-byte indices for sparse encodings); Comp. ($\times$) reports the compression ratio relative to dense full-model transmission as *parameter ratio / byte ratio*. Best result for federated methods in **bold**. †For SparsyFed and SparseLoCo, Budget denotes the kept fraction of *all* parameters.

Method	Budget	Comm. (MB)	Comp. ($\times$)	GSM8K	MATH-500	AIME24	AIME25	AMC	Avg.
Qwen2.5-3B-Instruct	-	-	-	59.47	46.54	2.50	1.46	27.11	27.42
Centralized	-	-	-	84.75	63.42	7.50	3.44	41.95	40.21
Local-Only	-	-	-	82.45	58.16	5.88	2.94	38.28	37.54
FedAvg-SW	-	6,794.2	$1\times$ / $1\times$	79.55	56.84	6.56	3.02	39.53	37.10
FedAvg-UW	-	6,794.2	$1\times$ / $1\times$	80.00	55.73	5.42	**3.12**	37.66	36.39
SparsyFed	$5\%^{\dagger}$	1,019.1	$20\times$ / $7\times$	79.10	56.61	6.56	2.92	39.53	36.94
SparseLoCo	$3\%^{\dagger}$	451.2	$32\times$ / $15\times$	78.69	56.05	6.04	2.81	39.30	36.58
FGRPO	-	6,794.2	$1\times$ / $1\times$	83.11	59.79	6.83	3.02	39.73	38.49
Fed-GRPO (Ours)	100%	645.5	$32\times$ / $11\times$	**83.21**	60.82	7.08	**3.12**	**41.09**	**39.06**
	50%	321.4	$63\times$ / $21\times$	83.11	**60.84**	**7.19**	2.71	41.25	39.02
	20%	126.9	$161\times$ / $54\times$	81.16	56.14	5.31	2.81	38.67	36.82
	10%	62.2	$327\times$ / $109\times$	79.69	55.21	5.42	3.12	36.25	35.94
	5%	32.8	$621\times$ / $207\times$	77.29	54.83	5.62	3.23	34.45	35.08

Table 1 reports the main results achieved by each method across the five benchmarks. From this table, we can draw several key conclusions. First, Fed-GRPO achieves the best performance among all federated methods, and approaches the centralized upper bound, demonstrating that it effectively mitigates the challenges of non-IID data and aggregation mismatch in federated GRPO training. Second, the fact that FedAvg-SW and FedAvg-UW perform worse than Local-Only highlights the pitfalls of naive aggregation strategies that ignore update quality, underscoring the importance of our signal-weighted aggregation mechanism. Third, Fed-GRPO offers a flexible communication–accuracy trade-off across a wide range of bandwidth budgets. At $B{=}100\%$, our adaptive sparse communication is mathematically lossless and already shrinks the per-round uplink from $6{,}794.2$ MB to 645.5 MB—a $32\times$ parameter ($11\times$ byte) compression with *zero* accuracy loss. Halving the budget to $B{=}50\%$ doubles the compression to $63\times$ ($21\times$ in bytes) while preserving essentially the same average accuracy (39.02 vs. 39.06), and even at $B{=}20\%$ ($161\times$, $54\times$ in bytes) Fed-GRPO still matches FedAvg's accuracy at a small fraction of the bandwidth. Pushing further to $B{=}10\%$ and $B{=}5\%$ delivers extreme $327\times$ and $621\times$ parameter compression ($109\times$ and $207\times$ in bytes; 32.8 MB per round at $B{=}5\%$), at the cost of a moderate accuracy drop of $\sim$3–4 points relative to lossless transmission. We emphasize that this $621\times$ regime is intended for severely bandwidth-constrained deployments rather than as the default operating point: as the lower rows of Table 1 show, the degradation is gradual rather than abrupt, allowing practitioners to dial B along a smooth Pareto frontier and select an operating point—e.g., $B{=}50\%$ as a near-lossless sweet spot, or $B{=}5\%$ when bandwidth dominates—according to their bandwidth–accuracy preferences.

6.3 More Results on Other Base Models and Tasks

Tables 2 and 3 report additional results on math reasoning benchmarks using Qwen3-4B-Instruct as the base model and on coding tasks using Qwen2.5-3B-Instruct as the base model, respectively. The results show that Fed-GRPO consistently outperforms all other federated methods across different base models and tasks, demonstrating its generality and effectiveness in various settings. More results on different data heterogeneity and analysis on learning dynamics are provided in Appendix D.

Table 2: Results on five math reasoning benchmarks using Qwen3-4B-Instruct as the base model.

Method	GSM8K	MATH-500	AIME24	AIME25	AMC	Avg.
Qwen3-4B-Instruct	77.56	84.55	32.7	29.79	77.5	60.42
Centralized	93.76	86.90	36.67	32.50	81.95	66.36
Local-Only	93.45	86.18	33.75	30.04	78.55	64.39
FedAvg-SW	93.66	86.76	35.62	29.79	82.50	65.66
FedAvg-UW	93.73	86.73	34.90	30.52	81.02	65.38
SparsyFed	93.48	86.53	35.02	31.22	81.13	65.47
SparseLoCo	93.28	86.32	34.93	31.03	81.07	65.33
FGRPO	93.77	86.75	35.23	31.78	81.62	65.83
Fed-GRPO (Ours)	**93.93**	**86.85**	**35.94**	**32.29**	**82.73**	**66.35**

Table 3: Results on coding tasks using Qwen2.5-3B-Instruct as the base model.

Method	GSM8K	MATH-500
Qwen2.5-3B-Instruct	57.40	60.41
Centralized	62.80	62.85
Local-Only	59.60	60.80
FedAvg-SW	60.70	61.63
FedAvg-UW	60.82	61.82
SparsyFed	60.66	61.23
SparseLoCo	60.03	60.98
FGRPO	61.42	61.94
Fed-GRPO (Ours)	**62.23**	**62.44**

6.4 Ablation Studies

To isolate the contribution of each mechanism in Fed-GRPO, we conduct ablation studies on the math tasks using Qwen2.5-3B-Instruct as the base model by starting from FedAvg-SW (i.e., Fed-GRPO with all three mechanisms disabled) and progressively adding each component. As shown in Table 4, each mechanism contributes complementarily. Starting from FedAvg-SW (37.10%), *signal-weighted aggregation* (SWA) lifts the average to 38.01% (+0.91) with consistent gains across all five benchmarks, confirming that σ_k better reflects update quality than dataset size. Adding *global reward calibration* (GRC) further improves the average to 39.07% (+1.06), with the largest gains on harder benchmarks (e.g., MATH-500: $58.73 \rightarrow 61.26$), matching GRC's design intent of steering clients toward globally lagging prompts. Finally, *adaptive sparse communication* (ASC) at $B{=}1.0$ preserves accuracy (39.06% vs. 39.07%) while shrinking the per-client uplink from $6{,}794.2$ MB to 645.5 MB—a $11\times$ reduction at zero accuracy cost. Overall, SWA and GRC drive the accuracy gains, while ASC contributes orthogonal communication savings.

Table 4: Ablation studies on Fed-GRPO components. +SWA: add signal-weighted aggregation. +GRC: add global reward calibration. +ASC: add adaptive sparse communication (at $B{=}1.0$, lossless). Comm. (MB) refers to the average per-round per-client uplink payload.

Configuration	GSM8K	MATH-500	AIME24	AIME25	AMC	Avg.	Comm. (MB)
FedAvg-SW (no component)	79.55	56.84	6.56	3.02	39.53	37.10	6,794.2
+ SWA (§4.1)	81.24	58.73	7.02	3.05	40.03	38.01	6,794.2
+ SWA + GRC (§4.2)	82.75	61.26	7.26	3.13	40.94	39.07	6,794.2
+ SWA + GRC + ASC (Fed-GRPO)	83.21	60.82	7.08	3.12	41.09	39.06	645.5

7 Related Work

Several works also explore federated GRPO training but most from angles orthogonal to ours. Fed-GRPO Zhu et al. (2026) guides privacy-preserving server updates using GRPO's "Group Relative" rewards among experts based on clients' domain knowledge. FedMOA Wang et al. (2026) tackles multi-objective RL via hypergradient-based online weighting and task-aware server aggregation. Concurrently, FGRPO Chen et al. (2026) introduces an adaptive aggregation mechanism that dynamically prioritizes effective learning trajectories based on each client's performance gain relative to its historical baseline. Differently, Fed-GRPO targets the standard federated GRPO reasoning with a shared accuracy-style reward and non-IID prompt distribution. It is the first to identify that the reward statistics (μ_k, σ_k) produced "for free" during GRPO advantage normalization carry rich information about each client's aggregation, training, and communication needs—turning them into a unified, zero-cost signal that drives all three mechanisms.

8 Conclusion

We present Fed-GRPO, a novel FL framework for training large language models with GRPO. Fed-GRPO introduces three key mechanisms that leverage GRPO's unique reward statistics to address the challenges of naïve Fedavg GRPO training: *signal-weighted aggregation* that dynamically weights client updates based on their reward standard deviation, *global reward calibration* that reweights local objectives based on local-global reward gaps, and *adaptive sparse communication* that allocates bandwidth based on update quality. Experiments on several mathematical reasoning tasks demonstrate that Fed-GRPO significantly outperforms federated baselines and approaches centralized performance while achieving up to $32\times$ lossless communication compression. This work opens up new possibilities for collaborative reasoning training across distributed data sources without sharing raw data.

AI USE STATEMENT

In this work, we used generative AI tools for assist in the writing of proofs, implement methods, assist with translation. We have not used generative AI tools for help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, propose or refine hypotheses, design or provide feedback on research methodology or experiments, support qualitative and thematic data analysis, interpret results, and generate synthetic data sets, clean and reformat dataset are not applicable to this work. Additionally, we used generative AI tools for create or edit software code, sourcing/searching for information, edit a research paper to improve readability, identify relevant literature. We have reviewed all AI-assisted work. We checked LLM-generated words, LLM-generated code was verified and tested for correctness by 3 authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Our code is available at anonymous.4open.science/r/Fed-GRPO-34FD for the reproducibility of our work.

REFERENCES

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Omar Besbes, Yonatan Gur, and Assaf Zeevi. Non-stationary stochastic optimization. *Operations research*, 63(5):1227–1244, 2015.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021.

Pengyu Chen, Shaowei Li, Kai Wang, Yunsheng Yuan, Kai Han, Jun Luo, and Feng Li. Fgrpo: Federated grpo with adaptive aggregation on non-iid data. *arXiv preprint arXiv:2606.03094*, 2026.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Adriano Guastella, Lorenzo Sani, Alex Iacob, Alessio Mora, Paolo Bellavista, and Nicholas Donald Lane. Sparsyfed: Sparse adaptive federated learning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=OBUQNASaWw.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025a.

Pengxin Guo, Shuang Zeng, Yanran Wang, Huijie Fan, Feifei Wang, and Liangqiong Qu. Selective aggregation for low-rank adaptation in federated learning. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=iX3uESGdsO.

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 *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=7Bywt2mQsCe.

Ali Jadbabaie, Alexander Rakhlin, Shahin Shahrampour, and Karthik Sridharan. Online optimization: Competing with dynamic comparators. In *Artificial Intelligence and Statistics*, pp. 398–406. PMLR, 2015.

Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian Stich, and Martin Jaggi. Error feedback fixes signsgd and other gradient compression schemes. In *International conference on machine learning*, pp. 3252–3261. PMLR, 2019.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pp. 5132–5143. PMLR, 2020.

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

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020a.

Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. In *International Conference on Learning Representations*, 2020b. URL https://openreview.net/forum?id=HJxNAnVtDS.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The twelfth international conference on learning representations*, 2023.

Yujun Lin, Song Han, Huizi Mao, Yu Wang, and Bill Dally. Deep gradient compression: Reducing the communication bandwidth for distributed training. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=SkhQHMW0W.

Shih-Yang Liu, Xin Dong, Ximing Lu, Shizhe Diao, Peter Belcak, Mingjie Liu, Min-Hung Chen, Hongxu Yin, Yu-Chiang Frank Wang, Kwang-Ting Cheng, et al. Gdpo: Group reward-decoupled normalization policy optimization for multi-reward rl optimization. *arXiv preprint arXiv:2601.05242*, 2026.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=5PAF7PAY2Y.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

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

Amir Sarfi, Benjamin Thérien, Joel Lidin, and Eugene Belilovsky. Communication efficient llm pre-training with sparseloco. *arXiv preprint arXiv:2508.15706*, 2025.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Wei Sun, Wen Yang, Pu Jian, Qianlong Du, Fuwei Cui, Shuo Ren, and Jiajun Zhang. KTAE: A model-free algorithm to key-tokens advantage estimation in mathematical reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=yqQVRNdmKJ.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. TRL: Transformers Reinforcement Learning, 2020. URL https://github.com/huggingface/trl.

Ziyao Wang, Daeun Jung, Yexiao He, Guoheng Sun, Zheyu Shen, Myungjin Lee, and Ang Li. Fedmoa: Federated grpo for personalized reasoning llms under heterogeneous rewards. *arXiv preprint arXiv:2602.00453*, 2026.

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*, 2025a.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025b.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, YuYue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, LingJun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=2a36EMSSTp.

Zhiqi Yu, Zhangquan Chen, Mengting Liu, Heye Zhang, and Liangqiong Qu. Unveiling implicit advantage symmetry: Why grpo struggles with exploration and difficulty adaptation. *arXiv preprint arXiv:2602.05548*, 2026.

Gongxi Zhu, Hanlin Gu, Lixin Fan, Qiang Yang, and Yuxing Han. Fedgrpo: Privately optimizing foundation models with group-relative rewards from domain clients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 19487–19495, 2026.

A FED-GRPO ALGORITHM

Algorithm 1 provides a detailed summary of the Fed-GRPO method.

Algorithm 1 Fed-GRPO

Input: Initial model $\theta^{(0)}$, local datasets $\{\mathcal{D}_k\}_{k=1}^{K}$, communication rounds T, local training steps S, target communication budget ratio B
Output: Trained global model $\theta^{(T)}$

Server initializes: $\mu_{\text{global}} \leftarrow$ None, $\sigma_{\text{global}} \leftarrow$ None
for $t = 0, 1, \ldots, T-1$ **do**
  Server broadcasts global model $\theta^{(t)}$ and statistics $(\mu_{\text{global}}, \sigma_{\text{global}})$ to all clients
  for each client $k \in \{1, \ldots, K\}$ **in parallel do**
    $\theta_k \leftarrow \theta^{(t)}$ ▷ *Initialize local model*
    for $s = 1, \ldots, S$ **do**
      Sample a batch of prompts from $\mathcal{D}_k$; generate G_{rol} rollouts per prompt
      Compute rewards and advantages $\{\hat{A}_i\}$ for the group (Eq. (2))
      if $\mu_{\text{global}} \neq$ None **then**
        Compute calibration weights $\{\omega_q\}$ based on local–global reward gap (Eq. (6))
      else
        $\omega_q \leftarrow 1$ for all prompts
      end if
      Update θ_k via the calibrated GRPO loss $\mathcal{L}_k$ (Eq. (7))
    end for
    Compute local reward mean μ_k and standard deviation σ_k over the round
    Compute local model update $\Delta_k \leftarrow \theta_k - \theta^{(t)}$
    Determine adaptive budget ratio B_k (Eq. (8)) ▷ *Allocate bandwidth*
    Let $\tilde{\Delta}_k$ be the lossless sparse encoding of Δ_k
    if $B_k < 1.0$ **then**
      Further compress $\tilde{\Delta}_k$: retain top modules by efficiency E_m (Eq. (9)) until we only retain a fraction B_k of the changed parameters
    end if
    Client sends compressed update $\tilde{\Delta}_k$ and reward statistics (μ_k, σ_k) to server
  end for
  Server computes signal-weighted aggregation weights $\{w_k\}$ (Eq. (4))
  Server decompresses each $\tilde{\Delta}_k$ back to a dense delta Δ_k
  Server applies aggregated updates: $\theta^{(t+1)} = \theta^{(t)} + \sum_{k=1}^{K} w_k \, \Delta_k$
  Server updates global statistics μ_{global} and σ_{global} (Eq. (5))
end for
return $\theta^{(T)}$

B PROOF OF THEOREM 1

Proof. Let $\theta^{(t)}$ be the global model at the beginning of round t, and let $\theta_k^{(t,s)}$ be the local model of client k after s local steps within round t, with $\theta_k^{(t,0)} = \theta^{(t)}$. Within round t, the one-round update of the global model can be described as the following two sub-steps:

$$\theta^{(t)} \xrightarrow[\text{local update of each client}]{} \left\{\theta_k^{(t,S)}\right\}_{k=1}^{K} \xrightarrow[\text{server aggregation}]{} \theta^{(t+1)}.$$

For convenience, we denote the intermediate quantities as follows:

$$\theta_k^{(t,s+1)} = \theta_k^{(t,s)} - \eta g_k^{(t,s)}, \qquad s = 0, 1, \dots, S-1,$$

$$\Delta_k^{(t)} = \theta_k^{(t,S)} - \theta^{(t)} = -\eta \sum_{s=0}^{S-1} g_k^{(t,s)},$$

$$U^{(t)} = \theta^{(t)} + \sum_{k=1}^{K} w_k^{(t)} \Delta_k^{(t)},$$

$$\theta^{(t+1)} = U^{(t)} + \sum_{k=1}^{K} w_k^{(t)} \big(\tilde{\Delta}_k^{(t)} - \Delta_k^{(t)}\big),$$

where $g_k^{(t,s)}$ is the stochastic gradient of the round-t calibrated local objective $\mathcal{L}_k^{(t)}$ on a mini-batch $\xi_{k,s}^{(t)}$, with $\mathbb{E}\big[g_k^{(t,s)}\big] = \nabla \mathcal{L}_k^{(t)}\big(\theta_k^{(t,s)}\big)$ and $\mathbb{E}\big\|g_k^{(t,s)}\big\|_F^2 \le G^2$ by Assumption 2; $\tilde{\Delta}_k^{(t)} = \mathcal{C}_k(\Delta_k^{(t)})$ is the compressed update produced by the adaptive sparse communication operator in Section 4.3; and $w_k^{(t)} = \sigma_k^{(t)} / \sum_j \sigma_j^{(t)}$ is the signal-weighted aggregation weight in Eq. (4). Here, $U^{(t)}$ represents the immediate result obtained after each client transmits its uncompressed local update, and $\theta^{(t+1)}$ represents the parameter obtained after the actual compressed-communication step.

Since each $\mathcal{L}_k^{(t)}$ is L-smooth (Assumption 1) and $\sum_k w_k^{(t)} = 1$, the global objective $\mathcal{L}^{(t)} = \sum_{k=1}^K w_k^{(t)} \mathcal{L}_k^{(t)}(\theta)$ is also L-smooth. By the smoothness of $\mathcal{L}^{(t)}$,

$$\mathcal{L}^{(t)}(U^{(t)}) \le \mathcal{L}^{(t)}(\theta^{(t)}) + \big\langle U^{(t)} - \theta^{(t)}, \nabla \mathcal{L}^{(t)}(\theta^{(t)}) \big\rangle_F + \frac{L}{2} \big\|U^{(t)} - \theta^{(t)}\big\|_F^2. \tag{11}$$

For the squared-norm term in Eq. equation 11, since $U^{(t)} - \theta^{(t)} = -\eta \sum_k w_k^{(t)} \sum_s g_k^{(t,s)}$, by Cauchy-Schwarz inequality,

$$\begin{aligned}\big\|U^{(t)} - \theta^{(t)}\big\|_F^2 =& \eta^2 \Big\| \sum_k w_k^{(t)} \sum_s g_k^{(t,s)} \Big\|_F^2 \le \eta^2 \Big(\sum_k (w_k^{(t)})^2 \Big) \Big(\sum_k \Big\| \sum_{s=0}^{S-1} g_k^{(t,s)} \Big\|_F^2 \Big) \\ \le& \eta^2 \Big(\sum_k w_k^{(t)} \Big) \Big(\sum_k S \sum_{s=0}^{S-1} \|g_k^{(t,s)}\|_F^2 \Big) \le \eta^2 S \sum_{k=1}^{K} \sum_{s=0}^{S-1} \big\|g_k^{(t,s)}\big\|_F^2,\end{aligned}$$

where we use the fact that $w_k^{(t)} \le 1$ for the last second step, and $\sum_k w_k^{(t)} = 1$ for the last step. Taking expectation,

$$\mathbb{E}\big\|U^{(t)} - \theta^{(t)}\big\|_F^2 \le \eta^2 S \sum_{k=1}^{K} \sum_{s=0}^{S-1} \mathbb{E}\big\|g_k^{(t,s)}\big\|_F^2 \le K \eta^2 S^2 G^2. \tag{12}$$

For the inner-product term in Eq. equation 11,

$$\big\langle U^{(t)} - \theta^{(t)}, \nabla \mathcal{L}^{(t)}(\theta^{(t)}) \big\rangle_F = -\eta \big\langle \sum_k w_k^{(t)} \sum_s g_k^{(t,s)}, \nabla \mathcal{L}^{(t)}(\theta^{(t)}) \big\rangle_F,$$

taking expectation over the stochastic gradients in current round (Denote with $\mathbb{E}_t$), and using $\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y] + \mathrm{Cov}(X, Y)$, we get

$$\begin{aligned}&\mathbb{E}_t \big[\big\langle \sum_k w_k^{(t)} \sum_s g_k^{(t,s)}, \nabla \mathcal{L}^{(t)}(\theta^{(t)}) \big\rangle_F \big] \\ =& \big\langle \sum_k \mathbb{E}_t[w_k^{(t)}] \sum_s \mathbb{E}_t[g_k^{(t,s)}], \nabla \mathcal{L}^{(t)}(\theta^{(t)}) \big\rangle_F + \sum_k \sum_s \big\langle \mathrm{Cov}_t\big(w_k^{(t)}, g_k^{(t,s)}\big), \nabla \mathcal{L}^{(t)}(\theta^{(t)}) \big\rangle \\ =& \sum_k \mathbb{E}_t[w_k^{(t)}] \sum_s \big\langle \nabla \mathcal{L}_k^{(t)}(\theta_k^{(t,s)}), \nabla \mathcal{L}^{(t)}(\theta^{(t)}) \big\rangle_F + \sum_k \sum_s \big\langle \mathrm{Cov}_t\big(w_k^{(t)}, g_k^{(t,s)}\big), \nabla \mathcal{L}^{(t)}(\theta^{(t)}) \big\rangle\end{aligned}$$

If we assume $\big\| \mathrm{Cov}_t\big(w_k^{(t)}, g_k^{(t,s)}\big) \big\|_F \le \delta_t$ for any k, s, then

$$\begin{aligned}&\sum_k \sum_s \big\langle \mathrm{Cov}_t\big(w_k^{(t)}, g_k^{(t,s)}\big), \nabla \mathcal{L}^{(t)}(\theta^{(t)}) \big\rangle \\ \ge& - \sum_k \sum_s \big\| \mathrm{Cov}_t\big(w_k^{(t)}, g_k^{(t,s)}\big) \big\|_F \big\| \nabla \mathcal{L}^{(t)}(\theta^{(t)}) \big\|_F \ge -KS\delta_t \big\| \nabla \mathcal{L}^{(t)}(\theta^{(t)}) \big\|_F\end{aligned}$$

Combing the above facts,

$$\mathbb{E}\big[\big\langle U^{(t)}-\theta^{(t)},\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\rangle_F\big]\le-\eta\mathbb{E}\Big[\sum_{k=1}^{K}w_k^{(t)}\sum_{s=0}^{S-1}\big\langle\nabla\mathcal{L}^{(t)}(\theta^{(t)}),\nabla\mathcal{L}_k^{(t)}(\theta_k^{(t,s)})\big\rangle_F\Big]$$
$$+\eta KS\delta_t\mathbb{E}\big[\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F\big]. \tag{13}$$

We split the gradient as $\nabla\mathcal{L}_k^{(t)}(\theta_k^{(t,s)})=\nabla\mathcal{L}_k^{(t)}(\theta^{(t)})+\big(\nabla\mathcal{L}_k^{(t)}(\theta_k^{(t,s)})-\nabla\mathcal{L}_k^{(t)}(\theta^{(t)})\big)$. For the first part, summing over k with weights $w_k^{(t)}$,

$$\sum_{k=1}^{K}w_k^{(t)}\big\langle\nabla\mathcal{L}^{(t)}(\theta^{(t)}),\nabla\mathcal{L}_k^{(t)}(\theta^{(t)})\big\rangle_F=\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2. \tag{14}$$

Summing $s=0,\dots,S-1$, this contributes $S\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\|_F^2$. For the second (drift) part, by Young's inequality with parameter $\nu=1$ and the L-smoothness of $\mathcal{L}_k^{(t)}$,

$$\big\langle\nabla\mathcal{L}^{(t)}(\theta^{(t)}),\nabla\mathcal{L}_k^{(t)}(\theta_k^{(t,s)})-\nabla\mathcal{L}_k^{(t)}(\theta^{(t)})\big\rangle_F\ge-\tfrac{1}{2}\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2-\tfrac{L^2}{2}\big\|\theta_k^{(t,s)}-\theta^{(t)}\big\|_F^2. \tag{15}$$

Since $w_k^{(t)}\le1$ and $\sum_{k=1}^{K}w_k^{(t)}=1$, we know

$$\sum_{k=1}^{K}w_k^{(t)}\big\|\theta_k^{(t,s)}-\theta^{(t)}\big\|_F^2\le\sum_{k=1}^{K}\big\|\theta_k^{(t,s)}-\theta^{(t)}\big\|_F^2,\quad\sum_{k=1}^{K}w_k^{(t)}\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2=\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2.$$

Multiplying by $w_k^{(t)}$ and summing over k, then summing over s to both sides of Eq. equation 15, we get

$$\sum_{k=1}^{K}w_k^{(t)}\sum_{s=0}^{S-1}\big\langle\nabla\mathcal{L}^{(t)}(\theta^{(t)}),\nabla\mathcal{L}_k^{(t)}(\theta_k^{(t,s)})-\nabla\mathcal{L}_k^{(t)}(\theta^{(t)})\big\rangle_F$$
$$\ge-\tfrac{S}{2}\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2-\tfrac{L^2}{2}\sum_{k=1}^{K}\sum_{s=0}^{S-1}\big\|\theta_k^{(t,s)}-\theta^{(t)}\big\|_F^2.$$

Taking expectation and using $\mathbb{E}\|\theta_k^{(t,s)}-\theta^{(t)}\|_F^2\le\eta^2s^2G^2$ (per-client analogue of Eq. equation 12), together with $\sum_{s=0}^{S-1}s^2\le S^3/3$, we get

$$\mathbb{E}\Big[\sum_{k=1}^{K}\sum_{s=0}^{S-1}\big\|\theta_k^{(t,s)}-\theta^{(t)}\big\|_F^2\Big]\le\frac{K\eta^2S^3G^2}{3}. \tag{16}$$

Then,

$$\mathbb{E}\Big[\sum_{k=1}^{K}w_k^{(t)}\sum_{s=0}^{S-1}\big\langle\nabla\mathcal{L}^{(t)}(\theta^{(t)}),\nabla\mathcal{L}_k^{(t)}(\theta_k^{(t,s)})-\nabla\mathcal{L}_k^{(t)}(\theta^{(t)})\big\rangle_F\Big]$$
$$\ge-\tfrac{S}{2}\mathbb{E}\big[\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2\big]-\frac{K\eta^2L^2S^3G^2}{6}.$$

Combining with Eq. equation 14 and substituting back into Eq. equation 13, we get

$$\begin{aligned}&\mathbb{E}\big[\big\langle U^{(t)}-\theta^{(t)},\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\rangle_F\big]\\&\le-\frac{\eta S}{2}\mathbb{E}\big[\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2\big]+\frac{K\eta^3L^2S^3G^2}{6}+\eta KS\delta_t\mathbb{E}\big[\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F\big]\\&\le-\frac{\eta S}{4}\mathbb{E}\big[\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2\big]+\frac{K\eta^3L^2S^3G^2}{6},\end{aligned} \tag{17}$$

where we assume $\delta_t\le\mathbb{E}\big[\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F\big]/(4K)$ to derive the last step. Plugging Eqs. (12) and (17) into Eq. (11),

$$\mathbb{E}\big[\mathcal{L}^{(t)}(U^{(t)})\big]\le\mathbb{E}\big[\mathcal{L}^{(t)}(\theta^{(t)})\big]-\frac{\eta S}{4}\mathbb{E}\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2+\frac{K\eta^3L^2S^3G^2}{6}+\frac{KL\eta^2S^2G^2}{2}. \tag{18}$$

By the smoothness of $\mathcal{L}^{(t)}$ at $U^{(t)}$,

$$\mathcal{L}^{(t)}(\theta^{(t+1)}) \le \mathcal{L}^{(t)}(U^{(t)}) + \big\langle \theta^{(t+1)} - U^{(t)}, \nabla\mathcal{L}^{(t)}(U^{(t)})\big\rangle_F + \frac{L}{2}\big\|\theta^{(t+1)} - U^{(t)}\big\|_F^2. \tag{19}$$

To continue the proof, we first justify Assumption 3 for our efficiency-based module selection. By construction (Section 4.3), the operator $\mathcal{C}_k$ drops modules in increasing order of efficiency $E_m = \|\Delta_m\|_F^2/\|\Delta_m\|_0$ until only a fraction B_k of the changed parameters is retained. Let $\bar{E} := \|\Delta\|_F^2/\|\Delta\|_0 = \sum_m E_m\|\Delta_m\|_0 / \sum_m \|\Delta_m\|_0$ denote the overall efficiency averaged over all modules with weights $\|\Delta_m\|_0$. Since dropped modules carry the smallest E_m values, their weighted average E_m (with the same weights $\|\Delta_m\|_0$) is no larger than $\bar{E}$, so the total dropped energy satisfies

$$\big\|\mathcal{C}_k(\Delta) - \Delta\big\|_F^2 = \sum_{m\in\text{dropped}} E_m \cdot \|\Delta_m\|_0 \le \bar{E}\cdot(1-B_k)\|\Delta\|_0 = (1-B_k)\|\Delta\|_F^2, \tag{20}$$

which validates Assumption 3. Now, since $\theta^{(t+1)} - U^{(t)} = \sum_k w_k^{(t)}(\tilde{\Delta}_k^{(t)} - \Delta_k^{(t)})$, applying Jensen's inequality together with $w_k^{(t)} \le 1$, we get

$$\begin{aligned}\mathbb{E}\big\|\theta^{(t+1)} - U^{(t)}\big\|_F^2 =& \mathbb{E}\Big[\Big\|\sum_{k=1}^K w_k^{(t)}(\tilde{\Delta}_k^{(t)} - \Delta_k^{(t)})\Big\|_F^2\Big] \le \mathbb{E}\Big[\sum_{k=1}^K w_k^{(t)}\|\tilde{\Delta}_k^{(t)} - \Delta_k^{(t)}\|_F^2\Big] \\ \le& \mathbb{E}\Big[\sum_{k=1}^K \|\tilde{\Delta}_k^{(t)} - \Delta_k^{(t)}\|_F^2\Big] \le \mathbb{E}\Big[\sum_{k=1}^K (1-B_k^{(t)})\|\Delta_k^{(t)}\|_F^2\Big]\end{aligned}$$

where we use Eq. equation 20 in the last step. Applying $\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y] + \mathrm{Cov}(X,Y)$, this further imples

$$\mathbb{E}\big\|\theta^{(t+1)} - U^{(t)}\big\|_F^2 = \sum_{k=1}^K \mathbb{E}[(1-B_k^{(t)})]\mathbb{E}[\|\Delta_k^{(t)}\|_F^2] - \sum_{k=1}^K \mathrm{Cov}(B_k^{(t)}, \|\Delta_k^{(t)}\|_F^2) \le \frac{1}{2}K(1-\bar{B}^{(t)})\eta^2S^2G^2, \tag{21}$$

where $\bar{B}^{(t)} := \frac{1}{K}\sum_{k=1}^K B_k^{(t)}$ is the per-client average budget ratio at round t. Here we used $\mathbb{E}\big\|\Delta_k^{(t)}\big\|_F^2 \le \eta^2S^2G^2$ (analogue to Eq. equation 12) together with the assumption that $\mathrm{Cov}(B_k^{(t)}, \|\Delta_k^{(t)}\|_F^2) \ge -\frac{1}{2}(1-\bar{B}^{(t)})\eta^2S^2G^2$. For the inner-product term in Eq. (19), we first note

$$\big\|\nabla\mathcal{L}_k^{(t)}(\theta)\big\|_F^2 = \big\|\mathbb{E}\nabla\mathcal{L}_k^{(t)}(\theta;\xi_{k,s}^{(t)})\big\|_F^2 \le \mathbb{E}\big\|\nabla\mathcal{L}_k^{(t)}(\theta;\xi_{k,s}^{(t)})\big\|_F^2 \le G^2 \tag{22}$$

Hence by Jensen's inequality,

$$\big\|\nabla\mathcal{L}^{(t)}(U^{(t)})\big\|_F^2 = \Big\|\sum_{k=1}^K w_k^{(t)}\nabla\mathcal{L}_k^{(t)}(U^{(t)})\Big\|_F^2 \le \sum_{k=1}^K w_k^{(t)}\big\|\nabla\mathcal{L}_k^{(t)}(U^{(t)})\big\|_F^2 \le G^2. \tag{23}$$

Now, by Young's inequality with parameter $\eta_0 > 0$ chosen below,

$$\begin{aligned}&\mathbb{E}\big[\big\langle\theta^{(t+1)} - U^{(t)}, \nabla\mathcal{L}^{(t)}(U^{(t)})\big\rangle_F\big] \\ &\le \frac{\eta_0 G^2}{2} + \frac{1}{2\eta_0}\mathbb{E}\big\|\theta^{(t+1)} - U^{(t)}\big\|_F^2 \\ &\le \frac{\eta_0 G^2}{2} + \frac{K(1-\bar{B}^{(t)})\eta^2S^2G^2}{2\eta_0}.\end{aligned} \tag{24}$$

We choose $\eta_0 = \sqrt{K(1-\bar{B}^{(t)})}\eta S$ (when $\bar{B}^{(t)} < 1$), which is the AM–GM-optimal value that balances the two terms, making both equal to $\sqrt{K(1-\bar{B}^{(t)})}\eta SG^2/2$. (When $\bar{B}^{(t)} = 1$, we have $\theta^{(t+1)} = U^{(t)}$ a.s. and Eq. (24) trivially holds.) Plugging Eqs. (21) and (24) into Eq. (19),

$$\mathbb{E}\big[\mathcal{L}^{(t)}(\theta^{(t+1)})\big] \le \mathbb{E}\big[\mathcal{L}^{(t)}(U^{(t)})\big] + \sqrt{K(1-\bar{B}^{(t)})}\eta SG^2 + \frac{LK(1-\bar{B}^{(t)})\eta^2S^2G^2}{4}. \tag{25}$$

Combining Eq. (18) and Eq. (25), and rearranging,

$$\frac{\eta S}{4}\mathbb{E}\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2 \le \mathbb{E}\big[\mathcal{L}^{(t)}(\theta^{(t)})\big] - \mathbb{E}\big[\mathcal{L}^{(t)}(\theta^{(t+1)})\big] + \underbrace{\sqrt{K(1-\bar{B}^{(t)})}\eta S G^2}_{\text{leading-order residual}} + \underbrace{\frac{K\eta^3 L^2 S^3 G^2}{6} + \frac{KL\eta^2 S^2 G^2(2-\bar{B}^{(t)})}{4}}_{=:\Phi(\eta)\text{(higher-order residuals)}}. \tag{26}$$

There exists a constant $M = M(L, G, S, K, \eta_{\max}) > 0$ such that $\Phi(\eta) \le (M\eta)\cdot(\eta S/4) = M\eta^2 S/4$ for all $\eta \in (0, \eta_{\max}]$, so $\Phi(\eta)/(\eta S/4) \le M\eta$. Concretely, since $\Phi(\eta)/(\eta^2 S/4) = \frac{2K\eta L^2 S^2 G^2}{3} + KLSG^2(2-\bar{B}^{(t)})$, we can take $M = \frac{2K\eta_{\max} L^2 S^2 G^2}{3} + 2KLSG^2$.

The descent inequality (26) involves $\mathcal{L}^{(t)}$ on both sides. To telescope across rounds, we connect $\mathcal{L}^{(t)}(\theta^{(t+1)})$ on the LHS of round t to $\mathcal{L}^{(t+1)}(\theta^{(t+1)})$ which appears on the RHS of round $t+1$. By the definition of $\Delta_{\mathcal{L}}^{(t)}$ in Assumption 4,

$$\mathcal{L}^{(t)}(\theta^{(t+1)}) \ge \mathcal{L}^{(t+1)}(\theta^{(t+1)}) - \Delta_{\mathcal{L}}^{(t)}. \tag{27}$$

Substituting Eq. (27) into Eq. (26),

$$\frac{\eta S}{4}\mathbb{E}\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2 \le \mathbb{E}\big[\mathcal{L}^{(t)}(\theta^{(t)})\big] - \mathbb{E}\big[\mathcal{L}^{(t+1)}(\theta^{(t+1)})\big] + \mathbb{E}[\Delta_{\mathcal{L}}^{(t)}] + \sqrt{K(1-\bar{B}^{(t)})}\eta S G^2 + \Phi(\eta). \tag{28}$$

Before telescoping, we fix the boundary term at $t = T$. Define $\mathcal{L}^{(T)}$ using any uniformly bounded extension of the round-T statistics (e.g., reusing the round-$(T-1)$ weights and calibration) so that $\mathcal{L}^{(T)}(\theta^{(T)}) \ge \inf_{0\le t\le T,\,\theta}\mathcal{L}^{(t)}(\theta)$; the specific choice of extension does not affect the bound, since the only round-T quantity entering the telescoped sum is $\rho^{(T-1)}$, which is absorbed by Assumption 4. With this convention, $D \ge \mathcal{L}^{(0)}(\theta^{(0)}) - \inf_{0\le t\le T,\,\theta}\mathcal{L}^{(t)}(\theta) \ge \mathbb{E}[\mathcal{L}^{(0)}(\theta^{(0)})] - \mathbb{E}[\mathcal{L}^{(T)}(\theta^{(T)})]$.

Summing Eq. (28) over $t = 0, 1, \ldots, T-1$, and using the bound on D above together with $\sum_t \mathbb{E}[\Delta_{\mathcal{L}}^{(t)}] \le RT$ from Assumption 4 and Jensen's inequality $\frac{1}{T}\sum_t \sqrt{1-\bar{B}^{(t)}} \le \sqrt{1-\bar{B}}$ over rounds (where $\bar{B} = \frac{1}{T}\sum_t \bar{B}^{(t)}$ is the budget averaged over both clients and rounds),

$$\frac{\eta S}{4}\sum_{t=0}^{T-1}\mathbb{E}\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2 \le D + RT + T\cdot\left[\sqrt{K(1-\bar{B})}\eta S G^2 + \Phi(\eta)\right]. \tag{29}$$

Dividing both sides by $\eta ST/4$ (and using $\Phi(\eta)/(\eta S/4) \le M\eta$),

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2 \le \frac{4D}{\eta ST} + \frac{4R}{\eta S} + 4\sqrt{K(1-\bar{B})}G^2 + M\eta. \tag{30}$$

Setting $\eta = \sqrt{2(D/T+R)/(MS)}$ to optimally balance the η-dependent terms (i.e., $\frac{2(D/T+R)}{\eta S}$ against $M\eta$) via AM–GM,

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2 \le 2\sqrt{\frac{2M(D/T+R)}{S}} + 4\sqrt{K(1-\bar{B})}G^2. \tag{31}$$

Rescaling the constant M to absorb the factor $S/2$ (i.e., $M' = 2M/S$, and renaming $M' \to M$), and using $\sqrt{a+b} \le \sqrt{a} + \sqrt{b}$ for $a, b \ge 0$, we obtain the form stated in Eq. (10):

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\big\|\nabla\mathcal{L}^{(t)}(\theta^{(t)})\big\|_F^2 \le 2\sqrt{\frac{DM}{T}} + 2\sqrt{RM} + 4\sqrt{K(1-\bar{B})}G^2. \tag{32}$$

□

C EXPERIMENTAL DETAILS

Datasets. For mathematical reasoning tasks, we use the MATH dataset Hendrycks et al. (2021), which contains approximately 7,500 competition-level math problems spanning 5 difficulty levels. To construct a naturally non-IID federated partition, we split the training set across $K = 5$ clients by difficulty level, so that each client specializes in problems of a single difficulty. For evaluation, we report accuracy on five widely-used math reasoning benchmarks: MATH-500 Hendrycks et al. (2021); Lightman et al. (2023), GSM8K Cobbe et al. (2021), AIME 2024, AIME 2025, and AMC 2023. For coding tasks, we train on the MBPP dataset Austin et al. (2021) ("train" + "validation") and evaluate on the held-out MBPP "test" split and the HumanEval dataset Chen et al. (2021). To construct a non-IID federation setting, we partition the MBPP dataset by the reference-solution length, bucket it into 5 quantile bins, and partition the data across clients with a Dirichlet distribution ($\alpha = 0.5$) over these buckets.

Baselines. We compare Fed-GRPO against several baselines that represent different points in the centralized–federated spectrum: i) **Base Model**: base model without any training, serving as base performance; ii) **Centralized**: standard GRPO training on the entire MATH dataset, serving as an upper bound for federated methods; iii) **Local-Only**: each client independently runs GRPO on its local data without any communication or aggregation; iv) **FedAvg-SW / FedAvg-UW**: standard FedAvg McMahan et al. (2017) with GRPO as the local training algorithm. SW uses sample-weighted aggregation ($w_k = n_k/N$), while UW uses uniform weights ($w_k = 1/K$); v) **SparsyFed** Guastella et al. (2025) and **SparseLoCo** Sarfi et al. (2025): communication-effecient federated learning methods that apply fixed-ratio top-k sparsification to the model updates. vi) **FGRPO** Chen et al. (2026): incorporates an adaptive aggregation mechanism based on relative performance gain, characterizing each client's improvement relative to its personalized historical baseline to dynamically prioritize effective learning trajectories regardless of local task difficulty.

Training details. We use Qwen2.5-3B-Instruct Yang et al. (2025b) and Qwen3-4B-Instruct Yang et al. (2025a) as the base model for all methods. We use the AdamW Loshchilov & Hutter (2019) optimizer with learning rate 3×10^{-6}, $\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay 0, a constant learning-rate schedule with warmup ratio 0.1, and gradient clipping at 1.0 for GRPO training. For rollout, the prompt batch size is 64 and we sample 8 responses for each prompt at sampling temperature 1.0 and top-$p = 1.0$. The clip ratio is set to $\epsilon = 0.2$, KL coefficient $\beta_{\text{KL}} = 0$, and we use BF16 mixed precision. The maximum prompt and completion lengths are capped at 1024 and 2048 tokens, respectively. The reward function is a binary accuracy reward that checks whether the extracted answer matches the ground truth or whether the code is right or not. All federated methods run for $T = 12$ communication rounds with $S = 5$ local steps per round. Centralized and local-only baselines are trained for up to 310 steps (matching the total number of gradient updates in the federated setting). For Fed-GRPO, we set the global reward calibration strength $\lambda = 0.5$; for adaptive sparse communication, we vary the budget B to sweep the compression–accuracy trade-off, with minimum budget $B_{\min} = 0.01$. All experiments are implemented on top of TRL von Werra et al. (2020) library and run on 8 NVIDIA H800 GPUs.

Evaluation protocol. We save a model checkpoint at the end of each communication round for federated methods, and every 10 training steps for centralized and local-only baselines, then evaluate each checkpoint on all five benchmarks. Following common practice in GRPO literature Yu et al. (2025); Guo et al. (2025a), we report the best accuracy achieved across checkpoints. For inference, we sample with temperature 1.0, top-$p = 0.7$, and a maximum generation length of 4096 tokens; we generate 32 samples per prompt and report avg@32 (the mean accuracy over the 32 samples) for results stability.

D MORE EXPERIMENTAL RESULTS

D.1 RESULTS ON DIFFERENT DATA HETEROGENEITY

We further evaluate the performance of Fed-GRPO under different data heterogeneity settings. Specifically, we partition the MATH dataset by "subject" (Algebra, Geometry, Number Theory, Combinatorics) using a Dirichlet distribution with $\alpha = 0.5$ to create a non-IID federation across

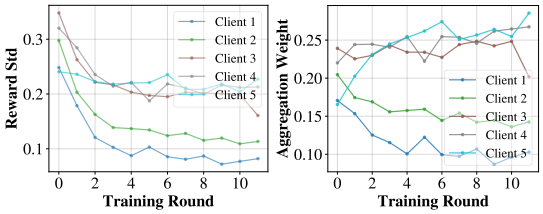
$K = 20$ clients. The results are summarized in Table 5. We observe that Fed-GRPO consistently outperforms all baselines across all benchmarks, demonstrating its robustness and effectiveness in handling heterogeneous data distributions.

Table 5: Results on five math reasoning benchmarks using Qwen2.5-3B-Instruct as the base model under dirichlet partitioning (α=0.5) by "subject" with K=20 clients.

Method	**GSM8K**	**MATH-500**	**AIME24**	**AIME25**	**AMC**	**Avg.**
Qwen2.5-3B-Instruct	59.47	46.54	2.50	1.46	27.11	27.42
Centralized	84.75	63.42	7.50	3.44	41.95	40.21
Local-Only	79.60	53.09	4.43	2.26	33.57	34.59
FedAvg-SW	81.32	56.04	5.04	2.72	35.12	36.05
FedAvg-UW	80.79	56.23	4.97	2.66	34.86	35.90
SparsyFed	80.96	56.02	5.02	2.53	35.12	35.93
SparseLoCo	81.02	55.96	4.89	2.55	34.97	35.22
FGRPO	81.43	56.77	5.48	2.76	38.10	36.91
Fed-GRPO (Ours)	**82.02**	**58.77**	**6.18**	**3.02**	**39.11**	**38.02**

D.2 Dynamics of Signal-Weighted Aggregation

To understand how signal-weighted aggregation adapts during training, we track two quantities across 12 communication rounds on the MATH dataset using Qwen2.5-3B-Instruct as the base model: the per-client reward standard deviation σ_k (Figure 3(a)) and the resulting aggregation weights w_k (Figure 3(b)). As shown in Figure (3a), σ_k evolves differently for each client, reflecting their varying learning signals as training progresses. Some clients experience an increase in σ_k, indicating stronger learning signals, while others see a decrease in σ_k as they master their local data. Figure (3b) shows that the aggregation weights w_k adapt accordingly, shifting more weight toward clients with higher σ_k. This dynamic weighting allows Fed-GRPO to prioritize updates from clients that are currently contributing more to global improvement, effectively mitigating the aggregation mismatch issue in federated GRPO training. The fact that the weights change across rounds also suggests that the method is responsive to the evolving training dynamics, rather than relying on static heuristics. Overall, this analysis confirms that signal-weighted aggregation successfully captures and leverages the varying learning signals across clients to improve the quality of the global update.



(a) (b)

Figure 3: Dynamics of (a) per-client reward standard deviation σ_k and (b) aggregation weights w_k. The aggregation weights shift dynamically across rounds, concentrating on clients with stronger learning signals (higher σ_k) as training progresses.