TSRM: TWO-SIDED RELIABILITY MODELING FOR COLLABORATIVE REINFORCEMENT POST-TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Collaborative reinforcement post-training allows large language models to learn from distributed small-model evaluations without collecting their local training data. Existing approaches turn client evaluations of generated reasoning into rewards, making both the reasoning candidates and their evaluators part of the training signal. However, without reference-answer rewards, the resulting supervision is subject to uncertainty in both candidate reasoning quality and the suitability of client evaluators for different subjects. We propose TSRM, a two-sided reliability modeling framework that addresses these uncertainties by coordinating candidate selection and evaluator routing within a shared feedback loop. To guide candidate selection without correctness labels, we design Multi-Granularity Candidate Confidence Filtering (MG-CCF), which captures overall confidence and local low-confidence patterns through multi-granularity aggregation of probability- and entropy-based token scores. To adapt evaluator selection to subject-dependent feedback, Stability-Aware Subject-Profile Routing (SA-SPR) maintains client–subject profiles through smoothed agreement estimates and variability penalties. Full-group feedback updates the profiles for subsequent evaluator selection, whereas feedback on retained candidates drives policy learning and thus shapes the next round of candidates. Experiments across six mathematical-reasoning benchmarks show that TSRM achieves the highest average accuracy under both Non-IID and identity-homogenized evaluator settings, outperforming FedGRPO-U by 2.7 and 2.2 points, respectively, without reference-answer rewards or predefined expert assignments.

1 INTRODUCTION

Reinforcement learning can improve the reasoning of large language models by using evaluations of generated solutions to guide policy updates Shao et al. (2024). Other models can supply these evaluations, allowing their knowledge to guide learning without pooling the data on which they were trained. Collaborative post-training uses this idea to improve a large server model through evaluations from smaller models held by distributed clients Zhu et al. (2026). These client models score the server's candidate solutions, so both the solutions admitted to policy learning and the clients providing their rewards shape the resulting training signal.

Candidate selection seeks to improve reasoning accuracy or efficiency by controlling which generated solutions contribute to the final prediction or the training update. At inference time, confidence-based filtering excludes less confident traces from answer aggregation Fu et al. (2025), whereas training-time filtering uses properties such as length or reward per token to favor concise reasoning Shrivastava et al. (2026). Evaluator selection addresses a different source of variation, since the suitability of a reward model can depend on the input it evaluates. Reward-model routing learns this matching from preference data and online feedback Wu & Lu (2026), while client selection in collaborative learning can estimate domain competence from labeled auxiliary examples Zhu et al. (2026).

Using client evaluations as rewards does not by itself establish a reliable basis for policy learning. A candidate may have high overall confidence yet contain locally uncertain reasoning, and clients can differ in their suitability for evaluating different subjects. These uncertainties affect the same learning signal because the policy is updated from client scores on selected candidates. When train-

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.

ing uses neither reference-answer rewards nor predefined expert assignments, the problem is how to select both candidates and feedback sources so that distributed evaluations can effectively improve the policy.

In this paper, we propose TSRM, a Two-Sided Reliability Modeling framework that coordinates candidate selection and evaluator routing in collaborative post-training. To select candidates for policy learning without correctness labels, Multi-Granularity Candidate Confidence Filtering (MG-CCF) constructs a weighted admission score from probability- and entropy-based token scores. By aggregating confidence over the full trace, its tail, and low-confidence windows, MG-CCF allows locally uncertain regions to affect candidate ranking even when overall confidence is high. Stability-Aware Subject-Profile Routing (SA-SPR) adapts evaluator participation to different subjects without predefined expert assignments. Its client profiles summarize historical agreement between client and candidate answers within each subject, using smoothed estimates and variability penalties to guide routing as feedback accumulates while preserving exploration.

In TSRM, candidates retained by MG-CCF are compared using rewards from clients selected by SA-SPR, so both mechanisms shape the supervision for the same policy update. Client profiles, however, are learned from feedback on each client's complete assigned candidate group, preserving agreement observations even for candidates excluded from policy optimization. These profiles guide subsequent evaluator selection for new candidates generated by the updated policy.

Our contributions are as follows.

- We propose TSRM to address uncertainty in both candidate reasoning and evaluator suitability during collaborative policy learning. The framework coordinates MG-CCF and SA-SPR to select training candidates and reward sources for the same policy update, while retaining feedback from complete candidate groups for learning client profiles.
- We show that TSRM achieves higher average accuracy across six mathematical reasoning benchmarks than uniform participation and either component alone, under both persistent Non-IID client identities and an identity-homogenized control. These results support considering candidate selection and evaluator routing together when learning from distributed feedback without reference-answer rewards or predefined expert assignments.

2 Related Work

2.1 Distributed Feedback for Post-Training

Collaborative post-training seeks to improve a server model using information held by distributed clients without collecting their training data. One family of methods performs policy optimization locally and communicates model updates. FedRLHF incorporates local human feedback into client policy learning, while FedMOA and FGRPO address heterogeneous rewards or data in federated GRPO training Fan et al. (2025); Wang et al. (2026); Chen et al. (2026). A different interface keeps policy optimization at the server and asks client models to evaluate its generated reasoning candidates. FedGRPO uses these evaluations to construct group-relative rewards and selects clients according to competence estimated from nearby labeled auxiliary questions Zhu et al. (2026). Thus, the choice of evaluator affects the rewards available to the server, while the use of those rewards still depends on which generated candidates enter the policy update.

2.2 Selection of Reasoning Candidates

Generating several reasoning paths for one question makes selection possible both when producing an answer and when learning from the generated paths. At inference time, self-consistency aggregates answers across sampled paths Wang et al. (2023), whereas DeepConf uses token-derived confidence to filter traces before voting Fu et al. (2025). DeepConf's sequence-level and local measures also illustrate why a trace-wide average can miss a weak span. During training, the retained paths determine which responses contribute to a policy update. GFPO samples larger groups and filters responses by length or reward per token to reduce unnecessarily long reasoning under verifiable rewards Shrivastava et al. (2026). For a group-relative update, the rewards of the retained responses

also determine the group mean subtracted from each reward. Removing one response can therefore change the advantage of another retained response even when the latter's score stays the same.

2.3 Routing of Evaluators

When several models can evaluate a response, their differing judgments raise the question of how to obtain a useful reward signal. One approach combines model judgments. Coste et al. use conservative reward-model ensembles to mitigate overoptimization of imperfect rewards Coste et al. (2024). Another approach chooses an evaluator for each input. LASeR uses a contextual bandit to adaptively select reward models and updates its selection using feedback from subsequent preference-based policy training Nguyen et al. (2025). BayesianRouter learns reward-model strengths from labeled preference data before adapting its selection online as the policy changes Wu & Lu (2026). These approaches differ not only in whether judgments are combined or routed, but also in what evidence supports the evaluator choice. In a system where clients score generated candidate groups, routing determines whose scores can enter the group's reward calculation. If predefined competence labels are absent, historical client feedback offers a possible basis for this choice, although agreement with generated answers does not establish evaluator accuracy.

In collaborative group-relative training, evaluator participation determines whose judgments are observed, while candidate admission determines which responses enter the advantage calculation. TSRM studies both decisions without reference-answer rewards or predefined client competence labels.

3 Problem Formulation

3.1 Collaborative Setting

We study a server policy π_θ trained from feedback supplied by N clients $\mathcal{U} = \{1, \dots, N\}$, without collecting their local training data. Each question x_i has an observed subject label $s_i \in \mathcal{S}$, but no client–subject expertise assignment is given. Reference answers are used only for benchmark evaluation, never for policy optimization or profile learning.

For question x_i, the server generates an indexed collection of candidate reasoning traces

$$\mathcal{Y}_i = \{y_{i,j}\}_{j=1}^{M}, \qquad y_{i,j} \sim \pi_\theta(\cdot \mid x_i). \tag{1}$$

The $M = Gm$ candidates form G disjoint groups of m traces. The server selects G distinct clients $\mathcal{U}_i \subseteq \mathcal{U}$ and assigns one group to each; $\mathcal{I}_i(u)$ denotes the indices assigned to client u. Each candidate receives feedback only from its assigned client, and comparisons remain within that group.

3.2 Client Feedback

For mathematical reasoning, client u independently generates a solver answer $a_{u,i}^{\text{client}}$ to question x_i. Let $A(\cdot)$ extract and normalize a final mathematical answer. The client evaluates each assigned candidate through answer agreement:

$$r_{u,i,j}^{\text{client}} = \mathbf{1}\left[A(y_{i,j}) \equiv A(a_{u,i}^{\text{client}})\right], \tag{2}$$

where $\equiv$ indicates equivalence between normalized mathematical answers. This feedback measures agreement with the client's answer, not reference-answer correctness. The potential feedback matrix $\mathbf{R}_i \in \{0,1\}^{N \times M}$ has entries $[\mathbf{R}_i]_{u,j} = r_{u,i,j}^{\text{client}}$; only assigned client–candidate pairs are observed.

3.3 Learning Objective and Selection Decisions

Let $\widetilde{\mathcal{J}}_i(u) \subseteq \mathcal{J}_i(u)$ be the nonempty subset of candidates retained for policy learning, where $\widetilde{\mathcal{J}}_i(u)$ denotes the retained candidate indices after candidate selection. The assigned client's feedback is used directly as the training reward. For each $j \in \widetilde{\mathcal{J}}_i(u)$, the group-relative advantage is defined as

$$\begin{aligned} r_{i,j}^{\text{train}} &= r_{u,i,j}^{\text{client}}, \\ A_{i,j} &= r_{i,j}^{\text{train}} - \frac{1}{|\widetilde{\mathcal{J}}_i(u)|} \sum_{k \in \widetilde{\mathcal{J}}_i(u)} r_{i,k}^{\text{train}}, \end{aligned} \tag{3}$$

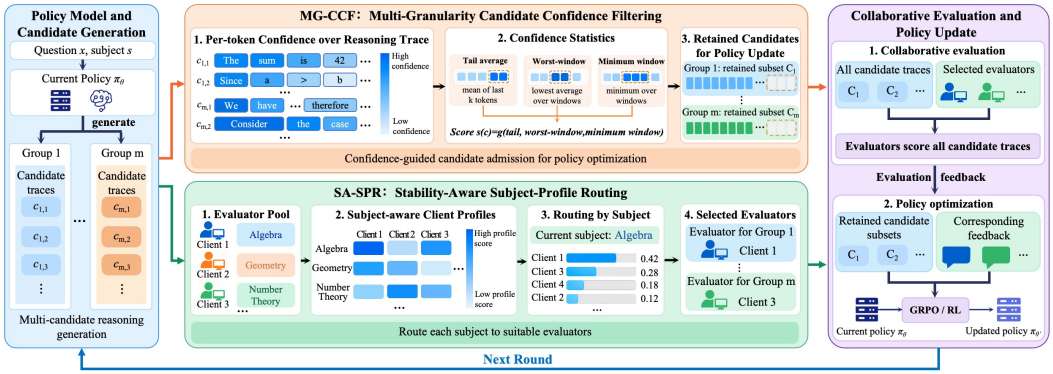

Figure 1: Overview of TSRM. MG-CCF selects candidates using generation-confidence statistics, while SA-SPR selects evaluators using historical subject-specific agreement. Each selected evaluator provides feedback on its complete assigned group. Full-group observations support profile learning, while retained candidates and their rewards support policy optimization. Reference answers enter neither update.

where k indexes candidates within the retained subset. Each reward is centered by the mean over the same client's retained group. Thus, evaluator selection chooses reward sources, while candidate selection determines the group baseline and updated trajectories; TSRM uses generation statistics and past client feedback for these choices.

4 Two-Sided Reliability Modeling

TSRM combines MG-CCF, which selects candidates using generation confidence, and SA-SPR, which routes evaluators using historical subject-specific agreement. Full-group feedback updates routing profiles, while retained candidates and their rewards drive policy optimization (Figure 1).

4.1 Multi-Granularity Candidate Confidence Filtering

MG-CCF ranks candidates using next-token confidence, combining whole-trace, tail, and low-confidence window statistics so that local uncertainty is not hidden by a high sequence average. These scores use neither client rewards nor correctness labels.

For a fixed question, write candidate trace j as $y_j = (v_{j,1}, \ldots, v_{j,T_j})$, where T_j is its length and $v_{j,\ell}$ is the token at position ℓ. Let $p_{j,\ell}^{(k)}$ denote the k-th largest next-token probability conditioned on the question and the generated prefix. At each position, we compute the top-K probability mass $m_{j,\ell}$, the normalized top-K probabilities $\widetilde{p}_{j,\ell}^{(k)}$, and the entropy-based concentration factor $d_{j,\ell}$. Combining this mass and concentration with the largest next-token probability gives the token-level confidence score $c_{j,\ell}$,

$$
\begin{aligned}
m_{j,\ell} &= \min\left(1, \sum_{k=1}^{K} p_{j,\ell}^{(k)}\right), \\
\widetilde{p}_{j,\ell}^{(k)} &= \frac{p_{j,\ell}^{(k)}}{\sum_{a=1}^{K} p_{j,\ell}^{(a)}}, \\
d_{j,\ell} &= 1 + \frac{\sum_{k=1}^{K} \widetilde{p}_{j,\ell}^{(k)} \log \widetilde{p}_{j,\ell}^{(k)}}{\log K}, \\
c_{j,\ell} &= 0.65 p_{j,\ell}^{(1)} + 0.35 m_{j,\ell} d_{j,\ell},
\end{aligned}
\tag{4}
$$

where $K > 1$ is the number of leading tokens considered and $0 \log 0 = 0$. The concentration factor decreases as the normalized top-K distribution becomes more uniform, while multiplication by the top-K probability mass accounts for how much of the full distribution these alternatives cover. The resulting score describes confidence in the next-token distribution along the trace, not the correctness of its final answer.

For a nonempty trace, let $L_j = \min(L, T_j)$ and $W_j = \min(W, T_j)$. We compute the whole-trace mean C_j^{avg}, final-L_j mean C_j^{tail}, minimum W_j-window mean $G_j^{\min}$, and mean of the lowest-scoring 10% of window means G_j^{bottom} (at least one window). Their weighted admission score is

$$q_j = 0.40C_j^{\text{tail}} + 0.30G_j^{\text{bottom}} + 0.20C_j^{\text{avg}} + 0.10G_j^{\min}. \tag{5}$$

The whole-trace mean contributes an overall confidence estimate, while the tail and window statistics allow localized low confidence to influence candidate ranking. After an initial training period without filtering, MG-CCF ranks candidates separately within each evaluator's group and retains a prescribed number of the highest-scoring traces. Their indices form the retained-candidate set $\widetilde{\mathcal{J}}_i(u)$ used for group-relative policy optimization. Before filtering is activated, all assigned candidates are retained.

4.2 Stability-Aware Subject-Profile Routing

SA-SPR selects evaluators for each subject from their observed agreement with the policy's answers, without predefined expert assignments. A high agreement rate on one candidate group does not establish whether similar agreement will persist on other questions in that subject. SA-SPR therefore maintains a separate profile for every client and subject, smoothing the agreement level across observations and penalizing its variability. Routing uses these profiles to favor clients with high historical agreement and low variability, while uniform exploration preserves opportunities to observe other clients.

For a selected client u evaluating question x_i of subject s_i at round t, the profile observation is the full-group agreement rate $h_{u,s_i}^{(t)}$. It averages the binary answer-agreement feedback $r_{u,i,j}^{\text{client}}$ over the complete assigned candidate group,

$$h_{u,s_i}^{(t)} = \frac{1}{|\mathcal{J}_i(u)|} \sum_{j \in \mathcal{J}_i(u)} r_{u,i,j}^{\text{client}}. \tag{6}$$

The sum runs over the full assigned set $\mathcal{J}_i(u)$, so for a fixed group and evaluator, the observation is unchanged by the subset that MG-CCF retains for policy optimization.

For each observed client and subject pair, we maintain a smoothed agreement level $\mu_{u,s}^{(t)}$ and a residual-variability estimate $\nu_{u,s}^{(t)}$. These states are updated from the new full-group agreement observation,

$$\begin{aligned} \mu_{u,s}^{(t)} &= (1-\beta)\mu_{u,s}^{(t-1)} + \beta h_{u,s}^{(t)}, \\ \nu_{u,s}^{(t)} &= (1-\beta)\nu_{u,s}^{(t-1)} + \beta\big(h_{u,s}^{(t)} - \mu_{u,s}^{(t)}\big)^2. \end{aligned} \tag{7}$$

Here β controls the weight of the new observation. The variability estimate smooths the squared deviation of the observed agreement rate from the updated agreement mean. We use the smoothed agreement level and this variability estimate to construct the client-subject profile score $p_{u,s}^{(t)}$,

$$p_{u,s}^{(t)} = \frac{n_{u,s}^{(t)}\mu_{u,s}^{(t)} + \alpha p_{u,s}^{(0)}}{n_{u,s}^{(t)} + \alpha} \exp\left(-\lambda_{\text{var}}\sqrt{\nu_{u,s}^{(t)} + \delta}\right). \tag{8}$$

The count $n_{u,s}^{(t)}$ records profile observations, and α controls shrinkage toward the initial score $p_{u,s}^{(0)}$. With few observations, the initial score has greater influence. As observations accumulate, the

Algorithm 1 TSRM Training

Require: Policy π_θ, N clients, question distribution, G evaluators per question, m candidates per group, filtering schedule and retention count

1: Initialize agreement, variability, count, and profile states
2: **for** each training round t **do**
3: Read the available profile state $p_{u,s}^{(t-1)}$
4: Sample question x_i with subject s_i and generate G groups of m candidates
5: Compute MG-CCF scores using Eqs. (4) and (5)
6: Sample G evaluators using Eq. (9)
7: Collect candidate feedback and full-group observations using Eq. (6)
8: Apply MG-CCF within each group when filtering is active, otherwise retain all candidates
9: Compute retained-group advantages using Eq. (3) and update π_θ using the clipped GRPO objective
10: Incorporate stored observations into the profile states using Eqs. (7) and (8)
11: **end for**

smoothed agreement estimate contributes more strongly. The exponential factor downweights variable agreement, with $\lambda_{\text{var}} \geq 0$ controlling the penalty and $\delta > 0$ providing numerical stabilization. Because agreement can include shared incorrect answers, the profile score encodes a routing preference rather than a calibrated probability of evaluator correctness.

For a new question with subject s, we convert the available profile scores into profile-based sampling weights $P_{\text{prof}}^{(t)}(u \mid s)$ through a softmax, then mix these weights with uniform exploration to obtain the routing weights $P^{(t)}(u \mid s)$,

$$P_{\text{prof}}^{(t)}(u \mid s) = \frac{\exp(p_{u,s}^{(t-1)}/\tau)}{\sum_{v=1}^{N} \exp(p_{v,s}^{(t-1)}/\tau)}, \qquad P^{(t)}(u \mid s) = (1-\varepsilon)P_{\text{prof}}^{(t)}(u \mid s) + \frac{\varepsilon}{N}. \tag{9}$$

The temperature $\tau > 0$ controls how strongly profile differences affect routing, and $\varepsilon \in [0, 1]$ controls uniform exploration. The server samples G distinct evaluators without replacement using these weights and assigns one to each candidate group. For $G > 1$, the weights are not the final probabilities that clients are included in the selected set. Profile scores affect participation only, and each selected client's answer-agreement feedback is used directly as its candidates' reward.

4.3 Policy Optimization and Profile Updates

Once client feedback is available, TSRM uses retained-candidate rewards to optimize the server policy and full-group observations to update the client profiles. For each selected evaluator u, policy optimization uses the retained traces indexed by $\widetilde{\mathcal{J}}_i(u)$ within its assigned group $\mathcal{J}_i(u)$. The assigned client's answer-agreement feedback supplies their training rewards directly, while the confidence and profile scores are used for selection rather than for rescaling this client's rewards.

Policy optimization uses retained traces and their client rewards with the group-relative advantage in Eq. (3). Confidence and profile scores select traces and evaluators but do not rescale rewards; the clipped GRPO objective remains unchanged. TSRM only determines the reward sources and the candidate subset contributing to the GRPO update.

Before filtering, we store each selected client's full-group agreement observation $h_{u,s_i}^{(t)}$ for profile learning. After the policy update, these stored observations update the corresponding agreement and variability estimates and hence the profile scores used for subsequent routing. The next training round therefore uses the updated policy to generate candidates and the updated profiles to choose their evaluators.

5 EXPERIMENTS

We evaluate whether TSRM improves mathematical reasoning through collaborative post-training without reference-answer rewards or predefined expert assignments. Across both evaluator constructions, TSRM achieves higher average accuracy over six benchmarks than uniform participation or either mechanism alone. We further examine the answer correctness of retained candidates and subject-dependent evaluator participation to characterize the behavior of the two mechanisms.

5.1 EXPERIMENTAL SETUP

Model and training data. We use Qwen2.5-Math-1.5B Yang et al. (2024) as the server model and perform collaborative reinforcement post-training on a fixed set of 2,000 problems from MATH Hendrycks et al. (2021). Each of eight evaluator clients provides a cached answer to every training problem. For each problem, two clients are selected, and each is assigned a separate group of eight candidates generated by the server model, yielding 16 candidates before filtering. Feedback is computed from answer agreement between each candidate and its assigned client.

Evaluator constructions. We compare two constructions using the same training problems and cached client answers. In Non-IID, the eight source-client identities remain fixed across problems. In IID-control, the same answers are redistributed across eight virtual client identities, with source composition balanced within each subject. This control changes how feedback is associated with client identities while keeping the underlying answer pool fixed. IID-control denotes identity homogenization rather than strictly identical client distributions. Subject labels are available for routing, but no predefined subject-to-expert mapping is supplied. Appendix A describes the evaluator construction.

Baselines and ablations. We implement TSRM and all collaboratively trained comparison variants within the collaborative training framework of FedGRPO Zhu et al. (2026). To assess candidate filtering and evaluator routing, we compare TSRM with variants using only MG-CCF or only SA-SPR, and with a uniform-participation reference denoted as FedGRPO-U. FedGRPO-U replaces FedGRPO's auxiliary-data competence selector with uniform evaluator selection and uses no candidate filtering. The MG-CCF variant retains uniform evaluator selection, whereas the SA-SPR variant retains all candidates. All collaboratively trained variants use the same client-answer reward definition and our within-group advantage definition. In routing-enabled variants, profile scores affect evaluator selection but do not reweight rewards. Neither benchmark reference-answer rewards nor gold-derived client-correctness labels are used for policy or profile updates. The Base Model is evaluated without collaborative post-training.

Training configuration. We use a learning rate of 3×10^{-6}, a KL coefficient of 10^{-4}, and BNPO loss without reward scaling. Training candidates are sampled with temperature 1.0, top-$p = 0.95$, and a maximum length of 2,048 tokens. For variants with MG-CCF, filtering starts after 40% of the training schedule and retains seven of eight candidates in each group. Additional optimization and confidence-score parameters are listed in Appendix B.

Evaluation. All collaboratively trained models are evaluated at update step 250. This common checkpoint was selected through monitoring training dynamics, including reward and loss, rather than the final benchmark test results, and was not selected separately for individual methods or seeds. We use the evaluation pipeline released with LUFFY Yan et al. (2025), with vLLM Kwon et al. (2024) for generation and the FedGRPO prompt template Zhu et al. (2026). All methods use the same inference configuration, with a maximum generation length of 2,048 tokens. We report final-answer accuracy on MATH Hendrycks et al. (2021), Minerva Lewkowycz et al. (2022), AMC, OlympiadBench He et al. (2024), AIME 2024, and AIME 2025, using answer extraction and mathematical equivalence matching against reference answers. Results are reported as the mean and sample standard deviation across training seeds 42, 43, and 44. For each seed, Avg. is the unweighted mean of the six benchmark accuracies, and its reported statistics are computed across seeds. The Base Model is evaluated once.

Setting	Method	MATH	Minerva	AMC	Olympiad	AIME24	AIME25	Avg.
–	Base Model	47.0	13.2	30.2	23.7	6.8	3.2	20.7
IID-control	FedGRPO-U	64.0 ± 1.0	23.4 ± 1.7	38.0 ± 3.0	30.6 ± 0.9	9.0 ± 1.8	5.7 ± 1.1	28.5 ± 0.6
	MG-CCF	67.3 ± 4.4	23.2 ± 3.6	39.3 ± 1.3	30.2 ± 1.8	9.6 ± 2.5	5.8 ± 0.7	29.2 ± 1.8
	SA-SPR	67.7 ± 4.3	20.6 ± 1.7	38.2 ± 1.9	31.1 ± 2.0	9.8 ± 1.5	5.5 ± 0.9	28.8 ± 1.4
	TSRM	$\mathbf{69.1 \pm 4.2}$	$\mathbf{23.8 \pm 2.6}$	$\mathbf{41.1 \pm 1.0}$	$\mathbf{32.4 \pm 0.9}$	$\mathbf{11.6 \pm 2.6}$	$\mathbf{6.4 \pm 0.4}$	$\mathbf{30.7 \pm 1.2}$
Non-IID	FedGRPO-U	61.4 ± 1.1	21.1 ± 2.8	38.0 ± 1.3	29.5 ± 1.2	8.7 ± 2.4	4.2 ± 1.1	27.1 ± 0.6
	MG-CCF	65.9 ± 2.3	20.8 ± 2.1	39.3 ± 0.7	29.6 ± 0.9	$\mathbf{10.5 \pm 2.0}$	$\mathbf{5.7 \pm 0.2}$	28.6 ± 0.7
	SA-SPR	65.7 ± 0.8	21.1 ± 1.3	39.1 ± 0.8	30.4 ± 1.7	9.6 ± 1.8	5.4 ± 1.1	28.5 ± 0.6
	TSRM	$\mathbf{67.9 \pm 2.7}$	$\mathbf{23.7 \pm 2.8}$	$\mathbf{40.5 \pm 1.0}$	$\mathbf{31.1 \pm 1.1}$	10.0 ± 1.7	$\mathbf{5.7 \pm 0.7}$	$\mathbf{29.8 \pm 0.9}$

Table 1: Main results on Qwen2.5-Math-1.5B under IID-control and Non-IID evaluator settings. Collaborative results are reported as mean $\pm$ sample standard deviation over seeds 42, 43, and 44. MG-CCF and SA-SPR denote the candidate-side and evaluator-side single-module variants, respectively. Bold marks the largest displayed mean within each column and setting, with displayed ties marked jointly.

5.2 Overall Performance and Component Contributions

We first examine whether selecting both candidates and evaluators improves learning from the available client feedback. In Table 1, TSRM achieves the highest average accuracy among the evaluated variants, reaching 30.7% under IID-control and 29.8% under Non-IID, exceeding FedGRPO-U by 2.2 and 2.7 percentage points, respectively. Its mean accuracy exceeds FedGRPO-U on all six benchmarks in both constructions. These gains are obtained with the same training questions, client-answer pool, and candidate generation budget, without adding reference-answer supervision. The comparison therefore supports improving how existing feedback is selected and used, rather than requiring additional feedback sources.

Under Non-IID, TSRM reaches 29.8% average accuracy, compared with 28.6% for MG-CCF and 28.5% for SA-SPR; under IID-control, its margins are 1.5 and 1.9 points. MG-CCF achieves higher accuracy than TSRM on Non-IID AIME 2024, while the two methods achieve the same accuracy on AIME 2025. Routing contributes 0.3 and 1.5 percentage points without and with MG-CCF under IID-control, respectively, compared with 1.4 and 1.2 percentage points under Non-IID. These results indicate that the incremental effects of routing vary with evaluator construction.

5.3 Candidate Selection and Answer Correctness

We evaluate MG-CCF on a fixed candidate pool to examine whether generation confidence provides useful candidate prioritization without changing the generating policy. We generate two groups of eight responses for each of 500 problems sampled from the MATH training cache using a frozen reference model. MG-CCF ranks responses within each group and retains the seven highest-scoring candidates, while reference answers are used only for measuring final-answer correctness. As shown in Figure 2, MG-CCF selects a retained subset with 69.4% accuracy from the original candidate pool with 66.1% accuracy, while the excluded responses achieve 43.2% accuracy.

The retained subset exceeds the full-pool accuracy by 3.3 percentage points under the same seven-of-eight retention budget. Uniform randomretention has an expected accuracy equal to that of the full candidate pool (66.1%), indicating that the improvement comes from confidence-based ranking rather than simply reducing the group size.

5.4 Subject-Dependent Evaluator Participation

To examine how SA-SPR allocates evaluations across subjects, we analyze the full-training selection records from its Non-IID single-mechanism runs with seeds 42, 43, and 44. Figure 3 shows that Client 0 receives the largest share of selections for Number Theory but falls below the uniform frequency of 12.5% for Precalculus, where Client 4 is selected most frequently. The participation ranking of Clients 0 and 4 thus reverses between the two subjects. SA-SPR can represent such

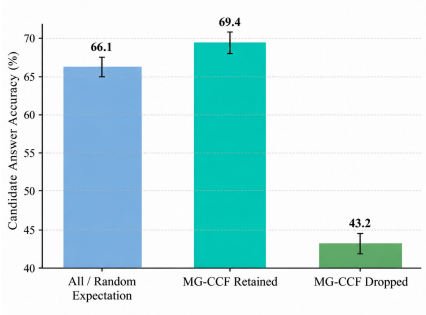

Figure 2: Final-answer accuracy of responses generated by the frozen IID-control FedGRPO-U model for 500 MATH training problems. The three bars report the full response pool, the subset retained by MG-CCF, and the excluded subset. MG-CCF retains seven of eight responses in each group. The full-pool bar also represents the expected accuracy of uniformly retaining seven of eight responses. Error bars indicate 95% confidence intervals obtained by prompt-level bootstrap. The appendix provides the candidate-generation protocol.

differences because a client's feedback in one subject updates that subject's profile rather than a single pooled client score. Each profile records the level and variability of historical answer agreement, allowing routing to assign different selection weights to the same client for different subjects. The observed reversal in client preference is consistent with this subject-specific design, which uses accumulated feedback to adapt evaluator participation without predefined expert assignments.

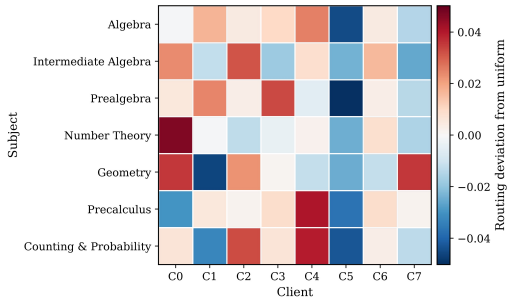

Figure 3: Evaluator participation by subject in the SA-SPR-only variant under Non-IID. Within each run, a subject with Q_s routing events contributes $2Q_s$ selection slots because each event selects two clients. A client's selection frequency is its selection count divided by $2Q_s$. The heatmap summarizes deviations from the uniform eight-client reference of $1/8$ across training seeds 42, 43, and 44, using all recorded selection events throughout training. Positive and negative values indicate selection frequencies above and below $1/8$, respectively.

6 Conclusion

In this paper, we proposed TSRM to jointly address candidate quality and evaluator suitability in collaborative reinforcement post-training. Multi-Granularity Candidate Confidence Filtering (MG-CCF) ranks reasoning traces using both overall confidence and local low-confidence patterns. Stability-Aware Subject-Profile Routing (SA-SPR) adapts evaluator participation through subject-specific profiles that track the level and variability of historical answer agreement. These profiles use feedback from each evaluator's complete candidate group, including feedback on candidates excluded by MG-CCF. This allows TSRM to filter training trajectories while preserving feedback for later evaluator selection. Across six mathematical-reasoning benchmarks, TSRM achieves higher average accuracy than uniform participation and either mechanism alone without reference-answer rewards. In future work, we will investigate how heterogeneous models learn from one another through mutual evaluation.

Reproducibility Statement

The equations and Algorithm 1 describe the update order, and Section 5 and Appendix B summarize the experimental protocol and implementation details. The reported results are based on the configurations, evaluation settings, and diagnostic procedures described in the paper.

AI Use Statement

Large language model assistants supported language editing, manuscript restructuring, code debugging, and experiment planning. The authors reviewed all methods, experiments, numbers, and interpretations and take responsibility for the paper.

References

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, 2026.

Thomas Coste, Usman Anwar, Robert Kirk, and David Krueger. Reward model ensembles help mitigate overoptimization. In *International Conference on Learning Representations*, 2024.

Flint Xiaofeng Fan, Cheston Tan, Yew-Soon Ong, Roger Wattenhofer, and Wei-Tsang Ooi. FedRLHF: A convergence-guaranteed federated framework for privacy-preserving and personalized RLHF. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, pp. 713–721. International Foundation for Autonomous Agents and Multiagent Systems, 2025.

Yichao Fu, Xuewei Wang, Yuandong Tian, and Jiawei Zhao. Deep think with confidence, 2025.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen 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 (Volume 1: Long Papers)*, pp. 3828–3850, Bangkok, Thailand, 2024. Association for Computational Linguistics.

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 *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2021.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 611–626. Association for Computing Machinery, 2024.

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, pp. 3843–3857, 2022.

Duy Nguyen, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. Laser: Learning to adaptively select reward models with multi-armed bandits. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024.

Vaishnavi Shrivastava, Ahmed Awadallah, Vidhisha Balachandran, Shivam Garg, Harkirat Behl, and Dimitris Papailiopoulos. Sample more to think less: Group filtered policy optimization for concise reasoning. In *International Conference on Learning Representations*, 2026.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023.

Ziyao Wang, Daeun Jung, Yexiao He, Guoheng Sun, Zheyu Shen, Myungjin Lee, and Ang Li. Fed-MOA: Federated GRPO for personalized reasoning LLMs under heterogeneous rewards, 2026.

Xinle Wu and Yao Lu. Reward model routing in alignment. In *International Conference on Learning Representations*, 2026.

Jianhao Yan, Yafu Li, Zican Hu, Zhi Wang, Ganqu Cui, Xiaoye Qu, Yu Cheng, and Yue Zhang. Learning to reason under off-policy guidance. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-Math technical report: Toward mathematical expert model via self-improvement, 2024.

Gongxi Zhu, Hanlin Gu, Lixin Fan, Qiang Yang, and Yuxing Han. FedGRPO: Privately optimizing foundation models with group-relative rewards from domain clients. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(24):19487–19495, 2026.

A Evaluator Construction

The evaluator pool contains eight persistent client solvers instantiated from eight separately maintained checkpoints. These checkpoints originate from four mathematical training-subset families: Algebra, Intermediate Algebra/Prealgebra, Number Theory/Geometry, and Precalculus/Counting & Probability, with two client checkpoints associated with each family. These provenance labels only describe how the evaluator pool was constructed. They are not treated as ground-truth expertise labels and are never provided to SA-SPR during evaluator routing or profile updates. Each client provides one cached answer for every question in the fixed 2,000-question training subset, resulting in 16,000 question–client answer records. Each candidate trace is evaluated by comparing its normalized final answer with the cached client answer, following the answer-agreement rule defined in Eq. (2). In the Non-IID construction, the original source-client identity is preserved across questions. In the identity-homogenized control, the same source-answer pool is redistributed across virtual client identities so that source composition is balanced within each subject. No additional solver training or answer generation is performed during this redistribution. Importantly, SPR is not given the source-training family, a predefined subject-to-client mapping, or any gold-derived client-correctness signal. Every client remains eligible for routing on every subject. The resulting subject-dependent routing frequencies therefore reflect profiles learned from historical agreement rather than a manually specified expert assignment.

B Hyperparameter and Implementation Details

Table 2 summarizes the common training configuration and the hyperparameters used for MG-CCF and SA-SPR. The training settings are shared across all collaborative variants, while the MG-CCF and SA-SPR parameters correspond to the candidate filtering and evaluator routing procedures described in Sections 4.1 and 4.2, respectively.

Hyperparameter	Value
Backbone	Qwen2.5-Math-1.5B
Training questions	2,000
Evaluator clients	8
Evaluators sampled per question	2
Candidates per evaluator group	8
Total candidates per question	16
Maximum completion length	2,048
Learning rate	3×10^{-6}
KL coefficient	10^{-4}
Top-K confidence size	12
Sliding-window length W	32
Tail length L	96
Bottom-window fraction	0.10
Candidates retained per evaluator group	7 of 8
CCF activation ratio	0.40
CCF score weights	$(0.40, 0.30, 0.20, 0.10)$
Gold-answer-free reward coefficient γ	1.0
Generation sampling temperature	1.0
Generation top-p	0.95
Profile smoothing factor β	0.10
Prior weight α	4.0
Variability penalty λ_{var}	0.25
Routing temperature τ	0.50
Exploration coefficient ϵ	0.15
GPU devices	4
Per-device batch size	2
Gradient accumulation steps	12
GRPO iterations	3
Loss type	Clipped GRPO
Reward scaling	Disabled
Random seeds	42, 43, 44

Table 2: Common training, MG-CCF, and SA-SPR configurations for Qwen2.5-Math-1.5B.

C Diagnostic Protocols and Supplementary Routing Results

C.1 Candidate-generation and evaluation protocol

For the MG-CCF diagnostic in Figure 2, we keep the candidate generator fixed so that the retained and excluded responses come from the same policy. The generator is the checkpoint-250 server model from the IID-control FedGRPO-U run with training seed 42. We sample 500 problems from the fixed 2,000-problem Hendrycks MATH training cache and generate two groups of eight responses for each problem. MG-CCF ranks responses within each group and retains the seven highest-scoring candidates. Reference answers are used only after selection to measure final-answer accuracy for the full, retained, and excluded candidate sets. Error bars in Figure 2 are 95% confidence intervals obtained by prompt-level bootstrap.

Uniform random retention provides a comparison with the same candidate pool and retention budget as MG-CCF. Selecting seven of eight responses uniformly gives every response a retention probability of $7/8$, so the expected retained-set accuracy equals the full-pool accuracy of 66.1%. MG-CCF instead retains a subset with 69.4% accuracy, exceeding that expectation by 3.3 percentage points. Because the generated responses and the number retained are unchanged, the difference measures the value of selecting responses by confidence rather than uniformly at random.

C.2 Evaluator-participation statistics

The heatmap in Figure 3 is based on the SA-SPR-only Non-IID training runs with seeds 42, 43, and 44. For each run, we count actual evaluator-selection events recorded in `routing_trace.jsonl` over the full training process. Within each subject, each event contributes two selected clients, and

we normalize each client's selection count by the total number of selection slots for that subject. Repeated selections for the same problem contribute separately, so the denominator counts selection slots rather than distinct problems. Each heatmap entry represents a client's selection-frequency deviation from $1/8$ for a given subject, summarizing the records from training seeds 42, 43, and 44. Positive values indicate that the client receives more than the uniform share of evaluation opportunities for that subject, while negative values indicate a smaller share.

C.3 Routing results without candidate filtering

To assess evaluator routing independently of candidate filtering, Figure 4 compares SA-SPR with uniform participation while retaining all generated candidates. SA-SPR increases six-benchmark average accuracy from 28.5% to 28.8% under IID-control and from 27.1% to 28.5% under Non-IID. The corresponding gains are 0.3 and 1.4 percentage points. Under Non-IID, each client identifier refers to the same source evaluator across questions. Under IID-control, the same answer pool is redistributed across virtual client identities, with source composition balanced within each subject.

The larger gain under persistent identities is consistent with the use of client-specific subject histories, since a client's past and current feedback then comes from the same source evaluator. With candidate retention unchanged, the comparison supports adapting evaluator participation through feedback history to improve policy learning.

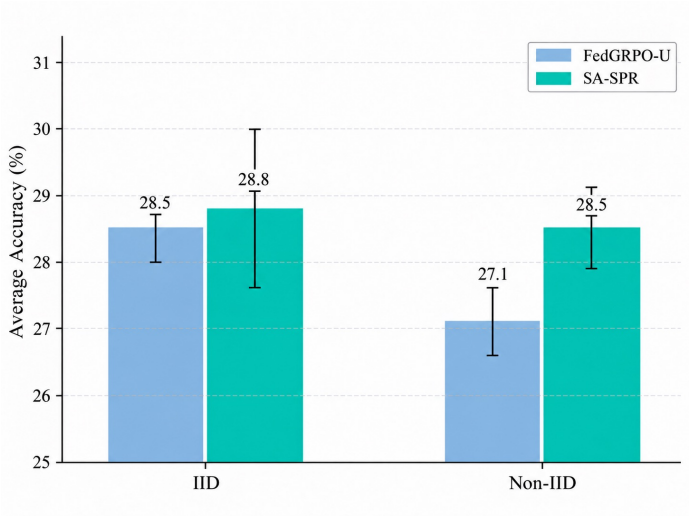

Figure 4: Six-benchmark average accuracy of FedGRPO-U and the SA-SPR-only variant with MG-CCF disabled. Bars show the mean and error bars show the sample standard deviation across training seeds 42, 43, and 44. The corresponding gains are 0.3 percentage points under IID-control and 1.4 percentage points under Non-IID.