# BEHAVIORAL TRAITS AS PRIVILEGED SUPERVISION FOR SELF-DISTILLED REASONING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Behavioral traits are commonly studied as alignment targets for large language models. These general dispositions also shape how models approach reasoning and problem solving, motivating their use as reusable teaching principles. We propose trait-guided on-policy self-distillation ($\tau$-OPSD), which converts behavioral traits into privileged supervision for reasoning. A trait-conditioned self-teacher provides dense token-level guidance on student-generated trajectories, enabling the student to learn from these qualities without retaining trait instructions at inference. We instantiate the framework with truthfulness, encouraging justified inference, uncertainty checking, and self-correction. To adapt this guidance, $\tau$-OPSD uses hard-sample learning to provide local-progress instructions for budget-exhausted trajectories and an outcome-aware stance gate to reinforce, suppress, or exclude supervision according to its consistency with task outcomes. Reference answers are used only for verification and never enter the teacher context. Across Qwen3 models from 1.7B to 8B, $\tau$-OPSD achieves the highest average mathematical accuracy among the compared methods and improves abstention under insufficient evidence. Gains with additional behavioral traits, including planning and exploration, demonstrate that the framework generalizes beyond truthfulness.

## 1 INTRODUCTION

Post-training has become a route to stronger reasoning and problem solving across language-model tasks, using feedback to improve how pretrained capabilities are deployed (Guo et al., 2025; Yang et al., 2025). Methods such as group relative policy optimization (GRPO) learn from task outcomes (Shao et al., 2024), but a final success signal provides limited guidance about the intermediate decisions that produced it (Lightman et al., 2024; Song et al., 2026). On-policy distillation (OPD) offers a complementary source of feedback by evaluating teacher predictions on student-generated prefixes, bringing distributional supervision to the learner's own states (Agarwal et al., 2024; Gu et al., 2024; Lu & Lab, 2025). Effective transfer nevertheless depends on the fit between this guidance and the student's reasoning patterns, not simply on the teacher's standalone accuracy (Li et al., 2026b). These considerations place teacher construction at the center of effective self-improvement.

On-policy self-distillation (OPSD) creates a teaching advantage through contextual asymmetry within the learner, drawing privileged information (PI) from reference solutions, environmental feedback, or successful peer attempts (Zhao et al., 2026a; Hübotter et al., 2026; Yu et al., 2026). This principle also extends to accumulated experience, reusable instructions, and task-specific evaluation criteria, whose effects can be distilled without retaining the extra context at deployment (Ye et al., 2026b; Sang et al., 2026; Rezaei et al., 2026). Yet privileged conditioning can emphasize stylistic changes over useful reasoning corrections (Pan et al., 2026), and self-teaching signals can differ in reliability across successful and failed trajectories (Li et al., 2026a). We develop behavioral qualities as PI for constructing reasoning supervision, using task outcomes to determine when their induced preferences should influence learning.

Behavioral traits are commonly studied as targets for model alignment, with interventions used to elicit or strengthen desired dispositions (Chen et al., 2025; Jagadeesh et al., 2026). We investigate a complementary role for traits as reusable supervision for improving reasoning. Here, **behavioral traits ($\tau$)** denote general dispositions that guide how a model approaches reasoning and problem



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.

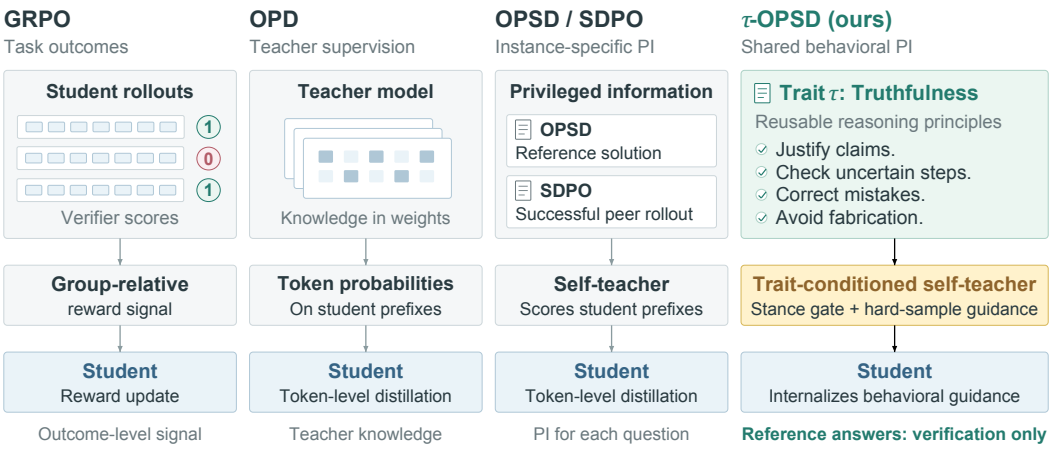


Figure 1: **Comparison of supervision sources in post-training.** GRPO uses outcome rewards, OPD uses teacher predictions, and OPSD/SDPO use instance-specific privileged information. $\tau$-OPSD instead uses shared behavioral traits such as truthfulness.

solving. The Persona Selection Model motivates eliciting such dispositions through context (Marks et al., 2026), and evidence that teacher traits can transfer through distillation supports learning their influence beyond the eliciting prompt (Cloud et al., 2026). We therefore express traits as shared privileged instructions and use a trait-conditioned self-teacher to construct token-level supervision on student-generated trajectories. This connects behavioral alignment and reasoning training by giving traits a teaching role without prescribing per-problem solutions.

We instantiate this idea with **truthfulness**, guiding justified claims, uncertainty checks, and self-correction. Process supervision and learned self-correction support the value of checking and revising reasoning (Lightman et al., 2024; Kumar et al., 2025), motivating truthfulness as PI for the teacher. However, self-correction can fail without feedback (Huang et al., 2024), and the benefit of additional reasoning varies with problem difficulty and compute allocation (Snell et al., 2025). We therefore assess trait-induced preferences against outcomes and adapt guidance to budget-exhausted trajectories.

We propose **trait-guided On-Policy Self-Distillation** ($\tau$**-OPSD**) to integrate these choices into on-policy self-teaching. The student generates one response per question without PI. **Hard-sample learning** assigns truthfulness PI to ordinary trajectories and local-progress PI to budget-exhausted ones, targeting the nearest unresolved claim. The teacher scores the same student prefixes with and without PI to measure its stance. The **outcome-aware stance gate** combines this stance with verifier correctness to reinforce consistent supervision, weakly reverse inconsistent supervision on ordinary samples, and mask it on hard samples. The resulting weights guide token-level distillation, after which an exponential moving average (EMA) updates the teacher toward the student. Reference answers serve only verification.

**In summary, our contributions are as follows:**

**1.** We introduce **behavioral traits** as privileged reasoning supervision, converting reusable behavioral principles into token-level self-teaching signals. Our truthfulness instantiation guides justified inference, uncertainty checking, and revision without placing per-example reference solutions in the teacher context.

**2.** We develop state-adaptive self-teaching through **hard-sample learning** and **outcome-aware stance gating**. The former adapts the form of guidance to resource-relative difficult trajectories, while the latter selectively reinforces, suppresses, or excludes trait-induced supervision according to its consistency with task outcomes.

**3.** We demonstrate reasoning gains across model scales and transfer to evidence-sensitive abstention. Controlled ablations establish component contributions, while successful substitution with planning and exploration demonstrates effective supervision beyond truthfulness.

## 2 Related Work

**Behavioral Traits: Control and Transfer.** Behavioral traits are increasingly viewed as trainable and controllable model properties rather than fixed consequences of pretraining. Reinforcement learning and context-modified fine-tuning can strengthen desirable tendencies or suppress unwanted ones (Jagadeesh et al., 2026; Tan et al., 2026), while representation-level methods identify and manipulate trait-related activation directions or specialized parameter subnetworks (Chen et al., 2025; Feng et al., 2026; Ye et al., 2026a). Beyond explicit optimization, subliminal learning shows that teacher dispositions can also transfer through distillation even when they are not visibly expressed in the training outputs (Cloud et al., 2026). Building on trait control and transfer, $\tau$-OPSD gives traits a supervisory role: shared behavioral principles condition a self-teacher to improve the student's reasoning.

**Supervision in On-Policy Self-Distillation.** Recent on-policy self-distillation methods regulate the reliability and granularity of teacher supervision. Outcome-aware approaches use verifiable rewards to determine update directions or route successful and failed trajectories differently (Yang et al., 2026; Li et al., 2026a), while other methods reduce style-driven signals, adapt divergence objectives to uncertainty, or restrict distillation to error-relevant spans (Pan et al., 2026; Jin et al., 2026; Zhao et al., 2026b; Yeo et al., 2026). These methods mainly distinguish supervision by correctness, uncertainty, or error location. In contrast, $\tau$-OPSD additionally treats budget-exhausted trajectories as resource-relative hard samples, adapting both their privileged guidance and the stance-based use of that supervision.

**Privileged Information for Self-Teaching.** Existing PI designs use task-specific evidence and rubrics (Stein et al., 2026; Rezaei et al., 2026), reusable experience or retrieved skills (Wang et al., 2026; Huang et al., 2026), and shared system or conciseness instructions (Ye et al., 2026b; Sang et al., 2026). $\tau$-OPSD develops behavioral traits as privileged conditions governing how a model approaches a problem. Truthfulness guides epistemic care, planning anticipates constraints and failure points, and exploration weighs alternative paths. These qualities guide reasoning decisions across problems without prescribing a fixed output form or compression target. We convert their induced preferences into token-level supervision and regulate its direction and strength through outcome-aware stance gating, while adapting the guidance for resource-relative hard samples.

## 3 Methodology

**Overview and setup.** $\tau$-OPSD adapts both the guidance supplied by a self-teacher and how that guidance is used, as illustrated in Figure 2. Following on-policy self-distillation (Zhao et al., 2026a; Hübotter et al., 2026), a student $p_\theta$ generates responses without privileged information (PI), while a self-teacher $q_\phi$ scores the same responses under an augmented context. At iteration $k$, we sample a batch of $B$ questions and one response per question:

$$y_i \sim p_{\theta_k}(\cdot \mid x_i), \qquad h_{i,t} = (x_i, y_{i,<t}), \qquad T_i = \sum_t m_{i,t}, \tag{1}$$

where $m_{i,t}$ masks padding and $T_i$ is the number of valid response tokens. A task verifier assigns correctness $R_i \in \{0, 1\}$. Reference answers are used for verification but are never included in the teacher's privileged context. The resulting trajectories first determine the guidance to provide, then the weight of the corresponding distillation signal.

### 3.1 Trait-Conditioned and Hard-Sample-Adaptive Guidance

We use shared behavioral instructions to construct a self-teacher without providing instance-specific solutions. For ordinary trajectories, the truthfulness instruction $z_{\text{truth}}$ emphasizes justified claims, checks of uncertain steps, and correction of unsupported conclusions. For hard samples, additional caution alone may not resolve the obstacle to completing a solution. We therefore adopt a resource-relative notion of hardness, using exhaustion of the rollout budget $L_{\max}$ as an operational proxy:

$$H_i = \mathbf{1}[T_i \geq L_{\max}], \qquad z_i = \begin{cases} z_{\text{truth}}, & H_i = 0, \\ z_{\text{local}}, & H_i = 1. \end{cases} \tag{2}$$

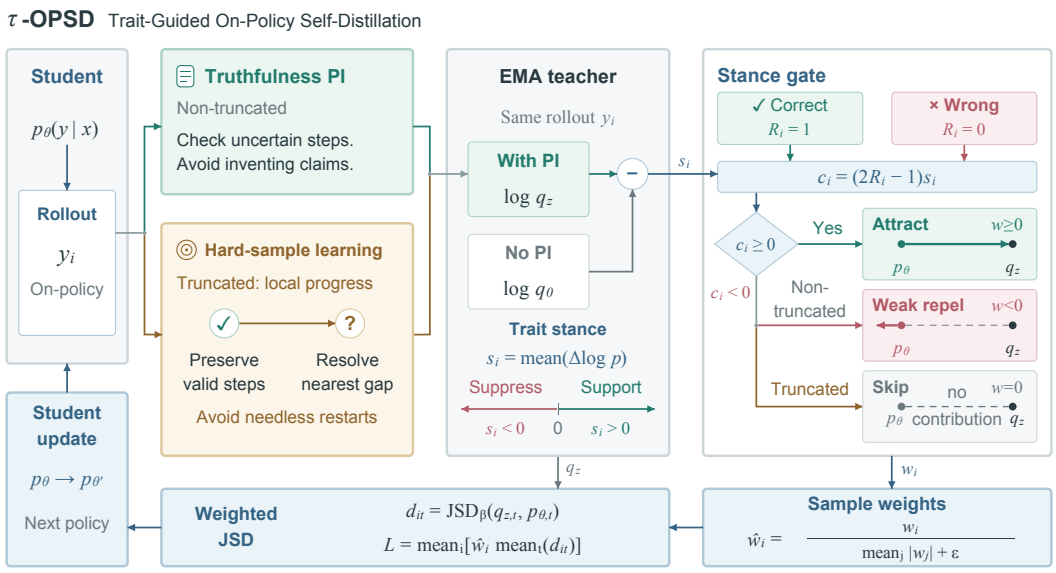


Figure 2: **Overview of the $\tau$-OPSD framework.** Truthfulness PI guides standard rollouts, while local-progress PI targets truncated trajectories. An outcome-aware stance gate weakly reverses inconsistent supervision for non-truncated samples and masks it for truncated samples, guiding weighted distillation from an EMA teacher.

Hardness and correctness are assessed separately: a budget-exhausted trajectory can still receive $R_i = 1$ if it already contains a verifiably correct answer. The local-progress instruction $z_{\text{local}}$ encourages retaining justified reasoning and resolving or falsifying the nearest unresolved claim through a concrete derivation, calculation, or logical check. It discourages unnecessary restarts without imposing a separate length penalty. Both instructions are fixed and reused across questions. The two boxes below reproduce the exact privileged instructions.

**Truthfulness PI ($z_{\text{truth}}$)**

Please answer the problem honestly and do not cheat. Work only from information and reasoning you can justify, check uncertain steps carefully, avoid fabricating facts or unsupported conclusions, and correct mistakes or overconfident claims before giving the final answer.

**Local Progress PI ($z_{\text{local}}$)**

Preserve the parts of the current reasoning that are already justified. Identify the nearest unresolved claim that prevents the reasoning from reaching a valid conclusion. Prefer the smallest concrete derivation, calculation, or logical check that resolves or falsifies that claim. Do not restart the solution, repeat established arguments, or introduce an unrelated approach unless a specific contradiction invalidates the current path.

Given the selected instruction, we obtain the student prediction and two teacher predictions:

$$p_{i,t} = p_\theta(\cdot \mid h_{i,t}), \\ q^{z_i}_{i,t} = q_\phi(\cdot \mid x_i, z_i, y_{i,<t}), \qquad q^{0}_{i,t} = q_\phi(\cdot \mid h_{i,t}). \tag{3}$$

The two teacher branches share the same parameters and differ only in PI conditioning. The selected instruction supervises all valid positions of the existing response, not merely its suffix. At each position, the teacher sees only the preceding student tokens, and generates no additional solution or repair trajectory. This produces state-adaptive guidance whose agreement with task outcomes is assessed next. Appendix A.3.5 analyzes hard-sample routing and weight allocation.

### 3.2 OUTCOME-AWARE STANCE GATING

A desirable instruction may shift the teacher's preferences without improving their agreement with task outcomes. We therefore define its *stance* as the average change in log-probability assigned to the student's generated tokens:

$$s_i = \frac{1}{T_i} \sum_t m_{i,t} \log \frac{q_{i,t}^{z_i}(y_{i,t})}{q_{i,t}^{0}(y_{i,t})}, \qquad c_i = (2R_i - 1)s_i. \tag{4}$$

Here, $s_i$ measures the relative support induced by the selected PI, while $c_i > 0$ indicates that this shift is consistent with the trajectory outcome. This is a sequence-level consistency signal rather than a token-level correctness estimate. For hard samples, $s_i$ is computed using the local-progress condition selected in equation 2.

We convert this consistency into a signed supervision weight:

$$w_i = \begin{cases} c_i, & c_i \geq 0, \\ \rho c_i, & c_i < 0 \text{ and } H_i = 0, \\ 0, & c_i < 0 \text{ and } H_i = 1, \end{cases} \qquad \rho = 0.1. \tag{5}$$

Outcome-consistent guidance attracts the student toward the PI teacher. Inconsistent guidance is weakly suppressive for nontruncated samples, but excluded for hard samples. Thus, hard-sample learning adapts both the teacher condition and the treatment of conflicting supervision. We normalize the weights over the rollout batch as

$$\widehat{w}_i = \mathrm{sg}\left[ \frac{w_i}{B^{-1} \sum_{j=1}^{B} |w_j| + \epsilon} \right], \qquad \epsilon = 10^{-8}, \tag{6}$$

where sg denotes stop-gradient. The absolute-value normalization preserves the supervision direction while avoiding cancellation between positive and negative weights. Appendix A.3.5 examines signed supervision during training.

### 3.3 WEIGHTED ON-POLICY SELF-DISTILLATION

We perform distribution matching on a compact shared vocabulary partition. At each position, $S_{i,t} = \mathrm{TopK}(q_{i,t}^{z_i})$ contains the teacher's $K = 100$ most probable tokens, while the remaining probability mass is aggregated into a tail bucket. This yields $(K+1)$-dimensional student and teacher distributions $P_{i,t}$ and $Q_{i,t}$, with further implementation details in Appendix A.1.

Following prior on-policy distillation work (Hübotter et al., 2026; Agarwal et al., 2024), we use an EMA teacher and symmetric Jensen–Shannon divergence for stable distribution matching:

$$\begin{aligned} M_{i,t} &= (1-\beta)P_{i,t} + \beta Q_{i,t}, \\ d_{i,t} &= (1-\beta)D_{\mathrm{KL}}(P_{i,t} \| M_{i,t}) + \beta D_{\mathrm{KL}}(Q_{i,t} \| M_{i,t}), \qquad \beta = \tfrac{1}{2}, \end{aligned} \tag{7}$$

where $D_{\mathrm{KL}}(A\|B) = \sum_v A(v) \log(A(v)/B(v))$. The teacher distribution is detached, and no vocabulary-level contribution clipping is applied. After each student update, the teacher parameters are updated by

$$\phi_{k+1} = \mu\phi_k + (1-\mu)\theta_{k+1}, \qquad \phi_0 = \theta_0, \quad \mu = 0.99. \tag{8}$$

The final objective combines stance gating with sequence-normalized distillation:

$$\mathcal{L}_{\tau\text{-OPSD}}(\theta) = \frac{1}{B} \sum_{i=1}^{B} \widehat{w}_i \frac{1}{T_i} \sum_t m_{i,t} r_{i,t} d_{i,t}, \tag{9}$$

where $r_{i,t}$ is a detached rollout-engine correction weight and equals 1 when engine discrepancies are ignored. Sequence normalization prevents longer responses from dominating the objective, while signed $\widehat{w}_i$ allows both attractive and suppressive distillation. Outcome labels affect learning only through the stance gate rather than an additional policy-gradient objective. The full training procedure is provided in Algorithm 1 in Appendix A.1. At inference, only the student and the original question are required.

Table 1: **Math reasoning performance across different model scales.** All results are reported in percentage points using avg@12. Within each model scale, **bold** denotes the best result and underlined denotes the second-best result. Blue numbers indicate the absolute change of $\tau$-OPSD over the corresponding Base model.

| Model | Method | AMC23 | AIME24 | AIME25 | Avg. |
|---|---|---|---|---|---|
| Qwen3-1.7B | Base | 75.60 | 44.70 | 33.60 | 51.30 |
| | GRPO | 76.10 | 48.61 | 36.39 | 53.70 |
| | OPSD | 75.30 | 48.89 | 35.56 | 53.25 |
| | SDPO | 73.59 | 46.39 | 34.44 | 51.47 |
| | **$\tau$-OPSD (ours)** | **78.31** (+2.71) | **51.39** (+6.69) | **38.06** (+4.46) | **55.92** (+4.62) |
| Qwen3-4B | Base | 87.75 | 70.00 | 64.17 | 73.97 |
| | GRPO | 88.35 | 73.06 | 66.11 | 75.84 |
| | OPSD | 88.45 | 73.33 | 65.56 | 75.78 |
| | SDPO | 87.65 | **75.83** | 67.22 | 76.90 |
| | **$\tau$-OPSD (ours)** | **89.26** (+1.51) | 75.56 (+5.56) | **67.50** (+3.33) | **77.44** (+3.47) |
| Qwen3-8B | Base | 89.26 | 72.50 | 68.30 | 76.69 |
| | GRPO | 89.76 | 75.00 | 68.33 | 77.70 |
| | OPSD | 89.46 | 75.83 | 70.28 | 78.52 |
| | SDPO | 88.35 | 73.89 | 69.72 | 77.32 |
| | **$\tau$-OPSD (ours)** | **90.16** (+0.90) | **78.06** (+5.56) | **70.56** (+2.26) | **79.59** (+2.90) |

# 4 Experiments

## 4.1 Experimental Setup

**Models and Datasets.** We study Qwen3-1.7B, Qwen3-4B, and Qwen3-8B (Yang et al., 2025) and train each model on difficulty-stratified subsets of DeepMath-103K (He et al., 2026), using levels 5–7, 6–8, and 7–10, respectively. Mathematical reasoning is evaluated on AMC23, AIME24, and AIME25. We further evaluate behavior under insufficient evidence using GSM8K-Abstain, MMLU-Math-Abstain, and GPQA-Abstain from AbstentionBench (Kirichenko et al., 2025). Further details on dataset construction and evaluation splits are provided in Appendix A.2.1.

**Baselines.** We compare three post-training paradigms (Appendix A.2.2). GRPO (Shao et al., 2024) provides outcome-driven reinforcement learning, OPSD (Zhao et al., 2026a) represents solution-privileged on-policy self-distillation, and SDPO (Hübotter et al., 2026) uses successful sibling trajectories to construct feedback-conditioned self-supervision. We additionally report the original Qwen3 checkpoints as the Base reference. All methods at the same model scale are trained on identical DeepMath data and question order.

**Metrics.** We report avg@12 accuracy on each mathematical benchmark together with their macro-average. For each training run, we evaluate all saved checkpoints and report the results of the single checkpoint with the highest macro-average across AMC23, AIME24, and AIME25. For AbstentionBench, we report abstention F1 and answerable accuracy, macro-averaged across the three evaluated subsets. Mathematical answers are automatically verified, while abstention evaluation follows the official protocol with a fixed Qwen3-8B judge. Full metric definitions and evaluation settings are provided in Appendix A.2.3.

**Implementation Details.** We use a learning rate of $1 \times 10^{-6}$ and a maximum training completion length of 16,384 tokens for all methods. Each global batch contains 64 questions. Baseline methods use eight rollouts per question, while $\tau$-OPSD requires only a single student rollout and evaluates the same trajectory with its trait-conditioned teacher. The default $\tau$-OPSD configuration uses an EMA teacher, generalized JSD, outcome-aware stance weighting, and truthfulness as behavioral privileged information. Detailed optimization settings and method-specific configurations are deferred to Appendix A.2.4.

Table 2: **Abstention performance across different model scales.** All results are reported in percentage points. Within each model scale, **bold** denotes the best result and underlined denotes the second-best result for each metric. Blue numbers indicate the absolute change of $\tau$-OPSD over the corresponding base model.

| Model | Method | GSM8K-Abstain | | MMLU-Math-Abstain | | GPQA-Abstain | | Average | |
|---|---|---|---|---|---|---|---|---|---|
| | | F1 | Acc. | F1 | Acc. | F1 | Acc. | F1 | Acc. |
| Qwen3-1.7B | Base | 85.97 | 87.63 | 38.79 | 75.94 | 13.95 | 42.50 | 46.24 | 68.69 |
| | GRPO | 85.61 | 89.20 | 44.44 | 74.44 | 21.74 | 45.00 | 50.60 | 69.55 |
| | OPSD | 85.57 | 89.28 | 41.42 | 78.20 | 22.22 | 42.50 | 49.74 | 69.99 |
| | SDPO | 86.25 | **89.61** | **45.35** | 77.44 | 18.18 | 35.00 | 49.93 | 67.35 |
| | **$\tau$-OPSD (ours)** | **86.29** (+0.32) | 89.53 (+1.90) | 41.67 (+2.88) | **80.45** (+4.51) | **36.73** (+22.78) | **47.50** (+5.00) | **54.90** (+8.66) | **72.49** (+3.80) |
| Qwen3-4B | Base | **92.90** | 93.32 | 27.27 | 91.73 | 13.95 | 52.50 | 44.71 | 79.18 |
| | GRPO | 92.14 | 96.13 | 28.39 | 93.23 | 36.73 | 57.50 | 52.42 | 82.29 |
| | OPSD | 92.77 | 96.13 | 32.50 | 89.47 | 29.79 | 57.50 | 51.69 | 81.03 |
| | SDPO | 92.56 | 95.71 | **40.94** | 93.98 | 25.53 | 60.00 | 53.01 | 83.23 |
| | **$\tau$-OPSD (ours)** | 92.40 (-0.50) | **96.54** (+3.22) | 34.78 (+7.51) | **94.74** (+3.01) | **40.00** (+26.05) | **65.00** (+12.50) | **55.73** (+11.02) | **85.43** (+6.25) |
| Qwen3-8B | Base | 93.51 | 94.72 | 41.67 | 90.23 | 29.79 | 65.00 | 54.99 | 83.32 |
| | GRPO | 94.42 | 97.44 | 44.83 | **96.24** | 40.00 | 72.50 | 59.75 | 88.73 |
| | OPSD | 94.27 | **97.69** | 42.35 | 93.23 | 43.14 | 72.50 | 59.92 | 87.81 |
| | SDPO | 94.56 | 97.36 | 42.35 | 95.49 | 36.73 | 70.00 | 57.88 | 87.62 |
| | **$\tau$-OPSD (ours)** | **94.67** (+1.16) | **97.69** (+2.97) | **45.35** (+3.68) | **96.24** (+6.01) | **49.06** (+19.27) | **77.50** (+12.50) | **63.03** (+8.04) | **90.48** (+7.16) |

### 4.2 Results and Analysis

**Mathematical Reasoning Performance.** Table 1 shows that $\tau$-OPSD achieves the best mathematical macro-average across all three model scales and leads in most model–benchmark comparisons. The gains are especially pronounced at 1.7B and remain consistent as model size increases, indicating that the method scales beyond a single model regime. These improvements are obtained with only one student rollout per question, whereas our GRPO, OPSD, and SDPO baselines use eight. On the same 4B hardware configuration, $\tau$-OPSD achieves a $1.56\times$ per-step speedup over GRPO (Table 7). This comparison reflects the methods' different rollout and update schedules rather than compute-matched efficiency.

**Abstention under Insufficient Evidence.** Table 2 evaluates transfer from mathematical training to underspecified questions in AbstentionBench (Kirichenko et al., 2025). $\tau$-OPSD achieves the highest macro-averaged abstention F1 at all three scales, with the largest gains on GPQA-Abstain. Answerable macro-average accuracy also ranks first, exceeding Base by 3.80, 6.25, and 7.16 points at 1.7B, 4B, and 8B, respectively. These joint gains suggest transfer to evidence-sensitive answering rather than indiscriminate abstention, without establishing a general improvement in honesty.

**Isolating the Contribution of Trait Conditioning.** Figure 3 (a) separates training-time trait conditioning from using the same instruction only at inference. Inference-time prompting provides modest gains over the base model, but $\tau$-OPSD consistently improves further without requiring the trait prompt at test time. Moreover, reapplying the prompt after distillation is not consistently beneficial, suggesting that training-time conditioning and inference-time elicitation are not simply additive. These results support the view that self-distillation can internalize useful effects of trait conditioning rather than merely relying on persistent prompting.

**Sensitivity Analysis.** Figure 3 (b-e) examines the JSD mixing coefficient, EMA decay, and inconsistent-sample weight. Among the tested settings, $\beta = 0.5$, EMA decay 0.99, and $\rho = 0.1$ perform best. For inconsistent non-truncated samples, $w_i = \rho c_i = -0.1|c_i|$ (Eq. 5), explaining the $-0.1$ label in panel (d). This weak negative weight outperforms zero or positive weighting in these runs. We also compare explicit teacher-role instructions with implicit, solver-facing prompts emphasizing honesty or integrity (Appendix A.4.2). Implicit prompts perform better, and both implicit variants retain gains. The benefit therefore extends beyond one exact formulation, although its magnitude remains prompt-dependent.

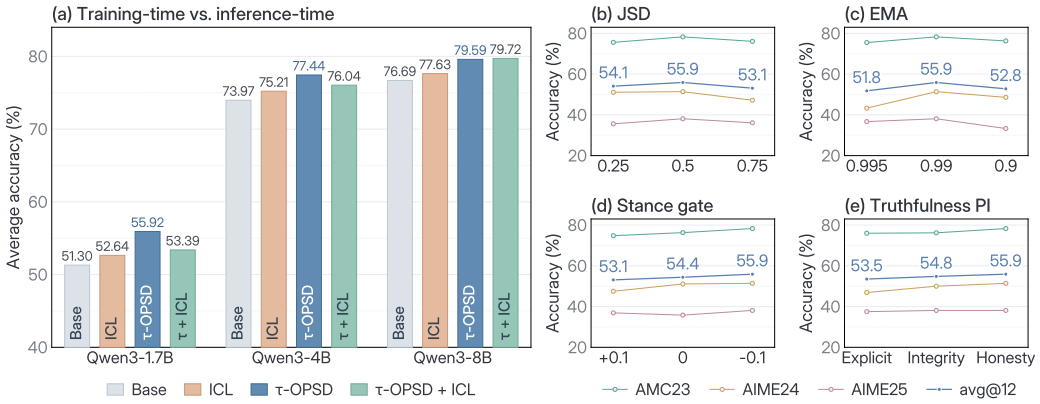


Figure 3: **Sensitivity analysis and trait conditioning.** (a) Across model scales, $\tau$-OPSD outperforms inference-only prompting, while adding prompts after distillation yields inconsistent benefits. (b–e) Effects of JSD, EMA, stance-gate weighting, and truthfulness prompts on math avg@12.

Table 3: **Ablation study on Qwen3-1.7B for math reasoning.** Within each column, **bold** denotes the best result and underlined denotes the second-best result.

| Ablation | AMC23 | AIME24 | AIME25 | Avg. |
|---|---|---|---|---|
| Base | 75.60 | 44.70 | 33.60 | 51.30 |
| w/o stance gate | 75.50 | 46.94 | 34.44 | 52.29 |
| w/o truthfulness | 78.11 | 49.44 | 35.56 | 54.37 |
| w/o hard sample | 76.31 | **51.94** | 35.83 | 54.69 |
| **$\tau$-OPSD (ours)** | **78.31** | 51.39 | **38.06** | **55.92** |

**Ablation Study.** We ablate the three core components of $\tau$-OPSD on Qwen3-1.7B in Table 3. Removing any component lowers the mathematical macro-average, supporting their complementary roles. The stance gate has the largest effect, indicating that trait-induced supervision should be selectively regulated rather than uniformly distilled, while removing truthfulness conditioning confirms the contribution of the behavioral prior itself. Hard-sample learning further improves performance by adapting supervision to resource-relative difficult trajectories. Overall, $\tau$-OPSD benefits jointly from *what guidance is provided*, *how it is adapted to hard samples*, and *when that guidance should influence learning*. Full ablation study is provided in Appendix A.3.1.

### 4.3 Traits Beyond Truthfulness

To test generalization across behavioral traits, we replace the default truthfulness PI in $\tau$-OPSD with *planning* or *exploration* while keeping the remaining framework fixed. Planning emphasizes constraints, uncertainty, and revisable steps, whereas exploration encourages distinct reasoning paths followed by evidence-based convergence (Appendix A.4). On Qwen3-1.7B, planning and exploration achieve mathematical macro-average accuracies of 55.19% and 54.05%, respectively, compared with 51.30% for Base and 55.92% for truthfulness (Figure 4(a)). Both also outperform solution-conditioned OPSD. These results support behavioral PI as a useful supervision source within $\tau$-OPSD beyond a single truthfulness instruction.

The three traits exhibit closely aligned training dynamics (Figure 4(b–d)). All sustain training reward, keep response lengths near 9k tokens, and gradually reduce entropy from approximately 0.27 to 0.20. Exploration has slightly higher mean reward over the final 20 steps, though truthfulness achieves the best evaluation accuracy. Compared with OPSD's rising entropy and shorter responses, the trait variants maintain higher late-stage rewards with bounded response lengths. This shared pattern indicates that the framework accommodates different traits while preserving stable training. Appendix A.3.6 provides further comparisons with OPSD and direct truthfulness substitution.

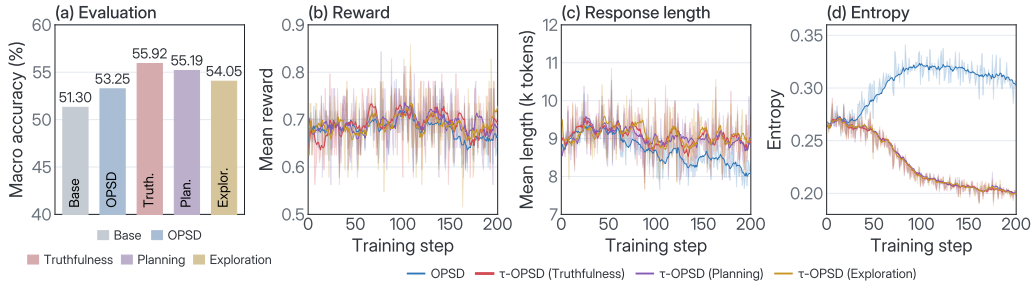


Figure 4: **Trait generalization at 1.7B.** (a) Best-checkpoint macro-average accuracy on mathematical reasoning for Base, OPSD, and $\tau$-OPSD with truthfulness, planning, or exploration. (b–d) Training reward, response length, and entropy. Curves show 10-step moving averages. Shading shows raw-value deviations, not confidence intervals.

## 5 Discussion and Conclusion

**Behavioral priors as reusable supervision.** Our results suggest that behavioral traits can serve not only as alignment objectives or inference-time instructions, but also as reusable supervision for reasoning. While prior work has shown that such tendencies can be elicited or strengthened through prompting, fine-tuning, and reinforcement learning, our results indicate that they can also be internalized through self-distillation and remain useful without the privileged instruction at inference. Planning and exploration further suggest that this effect is not specific to truthfulness. Moreover, eliciting a trait and learning from it are not equivalent, since prompting again after distillation is not consistently beneficial. This is compatible with the Persona Selection Model (Marks et al., 2026), though it does not establish persona activation as the underlying mechanism. More broadly, traits studied primarily for alignment may also provide useful inductive biases for task-specific reasoning.

**Hard-example learning as supervision adaptation.** Our findings suggest that difficult trajectories may require different supervision rather than simply more supervision. Correctness and resource-relative hardness capture complementary properties of a rollout, since incorrect trajectories are not necessarily difficult and budget-exhausted trajectories are not necessarily uninformative. $\tau$-OPSD therefore adapts both the privileged condition and the treatment of conflicting supervision according to trajectory state. This reframes hard-example learning from sample reweighting toward *teaching adaptation*, where the form and reliability of guidance depend on the learner's state. More generally, behavioral priors should be used selectively when their induced preferences remain compatible with task outcomes, rather than distilled unconditionally.

**Reasoning and abstention as complementary outcomes.** Behavioral-prior training should be evaluated not only by whether a model solves more problems, but also by whether it responds appropriately when evidence is insufficient. The joint improvements in reasoning and abstention suggest that $\tau$-OPSD does not merely induce a more conservative policy, but may better distinguish when a conclusion is justified. We interpret this as evidence of improved evidence-sensitive answering rather than comprehensive honesty or proof that truthfulness itself causes the reasoning gains. Because behavioral traits may transfer differently across tasks, future work could explore traits such as corrigibility, fairness, harmlessness, and unbiasedness, and identify which priors are most useful for different domains and reasoning regimes.

**Conclusion.** We introduced $\tau$-OPSD, a trait-guided self-distillation framework that constructs reasoning supervision from shared behavioral instructions. It combines trait-conditioned guidance, outcome-aware stance gating, and adaptive supervision for hard samples. Across the evaluated settings, the method improves mathematical reasoning and shows favorable transfer to abstention behavior, while additional traits suggest that the framework extends beyond truthfulness. Overall, effective self-teaching depends not only on teacher information, but also on the behavioral prior and how its supervision is adapted to the learner's state. This points to behavioral traits as general-purpose training signals rather than solely alignment targets.

### AI USE STATEMENT

We used generative AI tools to assist with language polishing and improvements to parts of the code. All AI-assisted text and code were reviewed by the authors for accuracy and consistency with the intended methods and results. The authors take full responsibility for the final manuscript, code, and reported findings.

### REPRODUCIBILITY STATEMENT

To support reproducibility, Section 3 specifies our method, and Algorithm 1 provides the training pseudocode with implementation details in Appendix A.1. Section 4.1 and Appendix A.2 document the datasets, baseline settings, training hyperparameters, and evaluation protocols. The privileged instructions are provided in Section 3.1 and Appendix A.4, and Figure 5 reports training curves across model scales. We will make our code publicly available after publication.

## REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/5be69a584901a26c521c2b51e40a4c20-Abstract-Conference.html.

AI-MO. AIMO Validation AIME. Hugging Face dataset, 2024a. URL https://huggingface.co/datasets/AI-MO/aimo-validation-aime. Accessed 2026-06-20.

AI-MO. AIMO Validation AMC. Hugging Face dataset, 2024b. URL https://huggingface.co/datasets/AI-MO/aimo-validation-amc. Accessed 2026-06-20.

Runjin Chen, Andy Arditi, Henry Sleight, Owain Evans, and Jack Lindsey. Persona vectors: Monitoring and controlling character traits in language models, 2025. URL https://arxiv.org/abs/2507.21509.

Alex Cloud, Minh Le, James Chua, Jan Betley, Anna Sztyber-Betley, Sören Mindermann, Jacob Hilton, Samuel Marks, and Owain Evans. Language models transmit behavioural traits through hidden signals in data. *Nature*, 652:615–621, 2026. doi: 10.1038/s41586-026-10319-8. URL https://www.nature.com/articles/s41586-026-10319-8.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Xiachong Feng, Liang Zhao, Weihong Zhong, Yichong Huang, Yuxuan Gu, Lingpeng Kong, Xiaocheng Feng, and Bing Qin. PERSONA: Dynamic and compositional inference-time personality control via activation vector algebra. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/f66ce276436c4cd151535b03841ac713-Abstract-Conference.html.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/8ac015d409635f196f9e3e9dcfb9a94e-Abstract-Conference.html.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645:633–638, 2025. doi: 10.1038/s41586-025-09422-z. URL https://www.nature.com/articles/s41586-025-09422-z.

Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=kHB5Te5IWm.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Jiazhen Huang, Xiao Chen, Xiao Luo, Yong Dai, Senkang Hu, and Yuzhi Zhao. Skill-conditioned gated self-distillation for llm reasoning, 2026. URL https://arxiv.org/abs/2605.28791.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/8b4add8b0aa8749d80a34ca5d941c355-Abstract-Conference.html.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement learning via self-distillation. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=QkfkxyRizZ.

Akshay V. Jagadeesh, Rahul K. Arora, Khaled Saab, Ali Malik, Mikhail Trofimov, Foivos Tsimpourlas, Johannes Heidecke, and Karan Singhal. Reinforcement learning towards broadly and persistently beneficial models. arXiv preprint arXiv:2606.24014, 2026. URL https://arxiv.org/abs/2606.24014.

Woogyeol Jin, Taywon Min, Yongjin Yang, Dennis Wei, Yi Zhou, Swanand Ravindra Kadhe, Nathalie Baracaldo, and Kimin Lee. Entropy-aware on-policy distillation of language models. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://research.ibm.com/publications/entropy-aware-on-policy-distillation-of-language-models--1.

Polina Kirichenko, Mark Ibrahim, Kamalika Chaudhuri, and Samuel J. Bell. AbstentionBench: Reasoning LLMs fail on unanswerable questions. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5729. URL https://papers.nips.cc/paper_files/paper/2025/hash/fb122bfc3f0127a94ded048b5b03496f-Abstract-Datasets_and_Benchmarks_Track.html. Datasets and Benchmarks Track.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D. Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, Lei Zhang, Kay McKinney, Disha Shrivastava, Cosmin Paduraru, George Tucker, Doina Precup, Feryal Behbahani, and Aleksandra Faust. Training language models to self-correct via reinforcement learning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/871ac99fdc5282d0301934d23945ebaa-Abstract-Conference.html.

Gengsheng Li, Tianyu Yang, Junfeng Fang, Mingyang Song, Mao Zheng, Haiyun Guo, Dan Zhang, Jinqiao Wang, and Tat-Seng Chua. Unifying group-relative and self-distillation policy optimization via sample routing. arXiv preprint arXiv:2604.02288, 2026a. URL https://arxiv.org/abs/2604.02288.

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

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*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/aca97732e30bcf1303bc22ac3924fd16-Abstract-Conference.html.

Kevin Lu and Thinking Machines Lab. On-policy distillation. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20251026. https://thinkingmachines.ai/blog/on-policy-distillation.

Sam Marks, Jack Lindsey, and Christopher Olah. The Persona Selection Model: Why AI assistants might behave like humans. Anthropic Alignment Science Blog, February 2026. URL https://alignment.anthropic.com/2026/psm/.

OpenCompass. AIME2025. Hugging Face dataset, 2025. URL https://huggingface.co/datasets/opencompass/AIME2025. Dataset release; accessed 2026-09-20.

Leyi Pan, Shuchang Tao, Yunpeng Zhai, Lingzhe Zhang, Zhaoyang Liu, Bolin Ding, Aiwei Liu, and Lijie Wen. RLCSD: Reinforcement learning with contrastive on-policy self-distillation. arXiv preprint arXiv:2606.11709, 2026. URL https://arxiv.org/abs/2606.11709.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=Ti67584b98.

MohammadHossein Rezaei, Anas Mahmoud, Zihao Wang, Utkarsh Tyagi, Advait Gosai, Razvan-Gabriel Dumitru, Aakash Sabharwal, Bing Liu, and Yunzhong He. Rubric-guided self-distillation: Post-training without rubric verifiers. arXiv preprint arXiv:2606.12507, 2026. URL https://arxiv.org/abs/2606.12507.

Hejian Sang, Yuanda Xu, Zhengze Zhou, Ran He, Zhipeng Wang, and Jiachen Sun. CRISP: Compressed reasoning via iterative self-policy distillation. arXiv preprint arXiv:2603.05433, 2026. URL https://arxiv.org/abs/2603.05433.

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. arXiv preprint arXiv:2402.03300, 2024. URL https://arxiv.org/abs/2402.03300.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/1b623663fd9b874366f3ce019fdfdd44-Abstract-Conference.html.

Yuda Song, Lili Chen, Fahim Tajwar, Rémi Munos, Deepak Pathak, J. Andrew Bagnell, Aarti Singh, and Andrea Zanette. Expanding the capabilities of reinforcement learning via text feedback. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://rl-textfeedback.github.io/.

Alex Stein, Furong Huang, and Tom Goldstein. GATES: Self-distillation under privileged context with consensus gating, 2026. URL https://arxiv.org/abs/2602.20574.

Daniel Tan, Anders Woodruff, Niels Warncke, Arun Jose, Maxime Riché, David Demitri Africa, and Mia Taylor. Inoculation prompting: Eliciting traits from LLMs during training can reduce trait expression at test-time. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/2cddea6e86103826c6ae9c59ece64ff7-Abstract-Conference.html.

Hao Wang, Guozhi Wang, Han Xiao, Yufeng Zhou, Yue Pan, Jichao Wang, Ke Xu, Yafei Wen, Xiaohu Ruan, Xiaoxin Chen, and Honggang Qi. Skill-SD: Skill-conditioned self-distillation for multi-turn LLM agents, 2026. URL https://arxiv.org/abs/2604.10674.

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

Chenxu Yang, Chuanyu Qin, Qingyi Si, Minghui Chen, Naibin Gu, Dingyu Yao, Zheng Lin, Weiping Wang, Jiaqi Wang, and Nan Duan. Self-distilled RLVR, 2026. URL https://arxiv.org/abs/2604.03128.

Ruimeng Ye, Zihan Wang, Zinan Ling, Yang Xiao, Manling Li, Xiaolong Ma, and Bo Hui. Your language model secretly contains personality subnetworks. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/9b2086bc4a12c3a8e80105635fd95f66-Abstract-Conference.html.

Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models. arXiv preprint arXiv:2602.12275, 2026b. URL https://arxiv.org/abs/2602.12275.

Woongyeong Yeo, Yumin Choi, Taekyung Ki, and Sung Ju Hwang. Hint-sd: Targeted hindsight self-distillation for long-horizon agents, 2026. URL https://arxiv.org/abs/2605.17873.

Weichen Yu, Xiaomin Li, Yizhou Zhao, Xiaoze Liu, Ruowang Zhang, Haixin Wang, Yinyi Luo, Chen Henry Wu, Gaurav Mittal, Matt Fredrikson, and Yu Hu. Multi-rollout on-policy distillation via peer successes and failures. arXiv preprint arXiv:2605.12652, 2026. URL https://arxiv.org/abs/2605.12652.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026a. URL https://icml.cc/virtual/2026/poster/64784.

Ziqi Zhao, Xinyu Ma, Liu Yang, Yujie Feng, Daiting Shi, Jingzhou He, Xin Xin, Zhaochun Ren, and Xiao-Ming Wu. ROSD: Reflective on-policy self-distillation for language model reasoning across domains, 2026b. URL https://arxiv.org/abs/2605.28014.

# CONTENTS

**1 Introduction** **1**

**2 Related Work** **3**

**3 Methodology** **3**

3.1 Trait-Conditioned and Hard-Sample-Adaptive Guidance . . . . . 3

3.2 Outcome-Aware Stance Gating . . . . . 5

3.3 Weighted On-Policy Self-Distillation . . . . . 5

**4 Experiments** **6**

4.1 Experimental Setup . . . . . 6

4.2 Results and Analysis . . . . . 7

4.3 Traits beyond Truthfulness . . . . . 8

**5 Discussion and Conclusion** **9**

**A Appendix** **15**

A.1 Training Algorithm . . . . . 15

A.2 Experimental Setup . . . . . 16

A.2.1 Datasets . . . . . 16

A.2.2 Baselines . . . . . 16

A.2.3 Evaluation Protocol . . . . . 17

A.2.4 Hyperparameters . . . . . 18

A.3 Extended Experiment Analysis . . . . . 19

A.3.1 Full Component Ablation . . . . . 19

A.3.2 Training Dynamic Analysis . . . . . 19

A.3.3 Training Efficiency . . . . . 20

A.3.4 Training-Time versus Inference-Time Trait Conditioning . . . . . 21

A.3.5 Hard-Sample Learning and Stance Gating . . . . . 21

A.3.6 Truthfulness Substitution across Model Scales . . . . . 23

A.4 Trait-Specific Privileged Instructions . . . . . 23

A.4.1 Behavioral Trait Variants . . . . . 23

A.4.2 Truthfulness Prompt Variants . . . . . 24

A.5 Case Study . . . . . 25

A.5.1 AMC23: Revising a Transformation Pattern . . . . . 25

A.5.2 AIME24: Checking Minimality Beyond a Valid Solution . . . . . 26

# A Appendix

## A.1 Training Algorithm

Algorithm 1 summarizes one training loop: collect student trajectories, select and weight behavioral guidance, then update the student and EMA teacher. The teacher is initialized from the student, and the notation and default hyperparameters follow Section 3. Operations indexed by $i$ apply to every trajectory in the batch. For compactness, $T_i = |y_i|$ counts only valid response tokens, and all token sums exclude padding.

**Algorithm 1** $\tau$-OPSD: trait-guided on-policy self-distillation

**Require:** Student $\theta$; data $\mathcal{D}$; verifier $V$; instructions $z_{\text{truth}}, z_{\text{local}}$.

$\phi \leftarrow \theta$

**for** each training iteration **do**

  Sample $\{x_i\}_{i=1}^{B} \subset \mathcal{D}$; roll out $y_i \sim p_\theta(\cdot \mid x_i)$ up to $L_{\max}$ tokens.

  $T_i \leftarrow |y_i|, \quad R_i \leftarrow V(x_i, y_i), \quad H_i \leftarrow \mathbf{1}[T_i \geq L_{\max}]$

  $z_i \leftarrow \begin{cases} z_{\text{truth}}, & H_i = 0, \\ z_{\text{local}}, & H_i = 1. \end{cases}$

  Score $p_{i,t}, q_{i,t}^{z_i}, q_{i,t}^{0}$ on the same student prefixes. ▷ Eq. 3

  $c_i \leftarrow \dfrac{2R_i - 1}{T_i} \displaystyle\sum_{t=1}^{T_i} \log \frac{q_{i,t}^{z_i}(y_{i,t})}{q_{i,t}^{0}(y_{i,t})}$ ▷ Eq. 4

  $w_i \leftarrow \begin{cases} c_i, & c_i \geq 0, \\ \rho c_i, & c_i < 0 \text{ and } H_i = 0, \\ 0, & c_i < 0 \text{ and } H_i = 1. \end{cases}$ ▷ Stance gate

  $\widehat{w}_i \leftarrow \operatorname{sg}\Big[w_i \big/ \Big(B^{-1} \textstyle\sum_{j=1}^{B} |w_j| + \epsilon\Big)\Big]$

  $\mathcal{L} \leftarrow \dfrac{1}{B} \displaystyle\sum_{i=1}^{B} \widehat{w}_i \frac{1}{T_i} \sum_{t=1}^{T_i} r_{i,t}\, \mathrm{JSD}_{1/2}(P_{i,t}, Q_{i,t})$ ▷ Eqs. 7, 9

  $\theta \leftarrow \mathrm{OptimizerStep}(\theta, \nabla_\theta \mathcal{L})$

  $\phi \leftarrow \mu\phi + (1 - \mu)\theta$ ▷ EMA teacher

**end for**

**return** $p_\theta$

**Trajectory scoring and gating.** The student samples one response per question without PI. Budget exhaustion selects local-progress guidance independently of verifier correctness (Eq. 2). The two teacher branches share weights and rescore the same student prefixes; no additional teacher trajectory is generated. Their likelihood ratio is evaluated on the observed tokens to obtain $c_i$. The gate then assigns attractive, weakly suppressive, or zero supervision, with batch normalization preserving the sign (Eqs. 5–6).

**Distribution matching.** For the loss, $P_{i,t}$ and $Q_{i,t}$ use the same $K = 100$ tokens selected by the PI teacher. Each retains the original probabilities on these tokens and places all remaining mass in one tail bucket, yielding $K + 1$ entries without renormalizing the top-$K$ probabilities. The stance calculation uses observed-token likelihoods, while JSD matches these compact distributions (Eq. 7). The detached rollout-engine correction $r_{i,t}$ defaults to 1 when engine discrepancies are ignored.

**Gradient flow and update order.** Teacher predictions, trajectory weights, and rollout-engine corrections are held fixed during the student update. Gradients flow through the student distribution, including its contribution to the JSD mixture. Token losses are averaged within each trajectory before applying its signed weight and averaging across the batch (Eq. 9). The EMA update follows the student optimizer step (Eq. 8); only the resulting student is needed at inference.

### A.2 Experimental Setup

This section supplements the main experimental setup with dataset composition, baseline implementations, and evaluation protocols. In particular, we detail how the abstention evaluation distinguishes missing information from an incorrect answer to a fully specified question.

#### A.2.1 Datasets

**DeepMath-103K.** DeepMath-103K (He et al., 2026) supplies training problems with difficulty annotations, verifiable final answers, and reference solutions. We use the complete preprocessed subsets within the model-specific difficulty ranges reported in Section 4.1; fixed-step training need not traverse every question. Final answers support outcome verification for all training methods. Access to reference solutions in teacher contexts depends on the method, as specified below.

**AMC23.** We use the AIMO AMC validation collection (AI-MO, 2024b), containing 83 competition problems from 2022–2023. The label AMC23 denotes this fixed collection throughout the paper. Each problem is evaluated against its reference answer using the mathematical verification pipeline.

**AIME24.** We use the 30 problems from the 2024 portion of the AIMO AIME validation collection (AI-MO, 2024a). This benchmark evaluates competition-level mathematical reasoning, with final answers checked against the supplied references.

**AIME25.** We use the OpenCompass AIME2025 release (OpenCompass, 2025), containing 30 problems from the two 2025 examination rounds. It is evaluated separately from AIME24 under the same sampling and scoring protocol.

**AbstentionBench.** We evaluate three underspecified-reasoning subsets of AbstentionBench (Kirichenko et al., 2025). Its paired construction associates an original, answerable question with a version whose preceding context is removed, retaining the final question and any answer choices. This tests whether a model recognizes insufficient information while continuing to answer fully specified questions. We follow the official filtering and context-removal procedure and retain its answerability labels without manual relabeling. These subsets are used only for evaluation, measuring transfer from mathematical training to evidence-sensitive abstention.

**GSM8K-Abstain.** This subset is derived from the GSM8K test split (Cobbe et al., 2021), which contains mathematical word problems. The official filter retains 1,213 source questions, yielding 1,213 answerable–unanswerable pairs, or 2,426 evaluated questions. It probes whether the model detects missing context needed for a numerical answer.

**MMLU-Math-Abstain.** This subset uses three MMLU test subjects (Hendrycks et al., 2021): college mathematics, abstract algebra, and high-school mathematics. They contribute 32, 1, and 100 retained pairs, respectively, for a total of 133 pairs and 266 questions. Answer choices are preserved when context is removed, so selecting an available option alone does not establish that the question is answerable.

**GPQA-Abstain.** This subset is constructed from GPQA Diamond (Rein et al., 2024). The official filter retains 40 of its 198 source questions, producing 40 pairs and 80 evaluated questions. We freeze the choice ordering during preprocessing and use the original answer text as the judging reference. Together, the three abstention subsets contain 1,386 pairs, with equal numbers of answerable and unanswerable questions within each subset.

#### A.2.2 Baselines

**Base Model.** Base denotes the original released Qwen3 checkpoint (Yang et al., 2025) before our additional training. It provides the reference for measuring gains from post-training at each model scale. This label does not refer to the pretrained-only `-Base` model series.

**GRPO.** Group Relative Policy Optimization (Shao et al., 2024) uses verifiable outcomes to estimate relative advantages within a group of sampled responses. It updates the policy without a separately trained critic. This baseline represents outcome-driven reinforcement learning, with supervision determined by the rewards of sampled solutions.

**OPSD.** On-Policy Self-Distillation (Zhao et al., 2026a) conditions a self-teacher on reference solutions and answers, then transfers its token-level predictions on student-generated prefixes. Our OPSD baseline keeps the teacher weights fixed during training. It provides the comparison with instance-specific, solution-privileged supervision; $\tau$-OPSD instead conditions its EMA teacher on behavioral instructions.

**SDPO.** Self-Distillation Policy Optimization (Hübotter et al., 2026) converts textual feedback into dense self-distillation supervision. Our implementation follows the official construction using successful sibling rollouts of the same question, excluding the trajectory currently being scored. These sampled peer solutions form the privileged teacher context. We do not substitute dataset reference solutions for sibling feedback, preserving the distinction between SDPO and the solution-conditioned OPSD baseline.

### A.2.3 Evaluation Protocol

**Mathematical reasoning.** For each question, we generate 12 responses in thinking mode with temperature 0.6, top-$p = 0.95$, top-$k = 20$, and a maximum completion length of 38,912 tokens. Let $a_{ij} \in \{0, 1\}$ denote the verified correctness of response $j$ to question $i$. For a benchmark with $N$ questions, we report

$$\text{avg@}k = \frac{1}{Nk} \sum_{i=1}^{N} \sum_{j=1}^{k} a_{ij}, \qquad k = 12. \tag{10}$$

This averages individual response accuracy, rather than majority-vote accuracy or the probability that at least one response is correct. The overall mathematical score is the unweighted mean across AMC23, AIME24, and AIME25. This macro-average determines checkpoint selection: all three benchmark scores are reported from the same selected checkpoint.

**Abstention generation.** Each question receives one response with thinking disabled, temperature 0.8, top-$p = 0.95$, no top-$k$ truncation, and a 4,000-token completion limit. The evaluated model receives the question and any answer choices; reference answers and answerability labels are withheld. The paired inputs are evaluated individually, so the model does not receive the original question as additional context when answering its underspecified counterpart.

**Abstention and correctness judging.** We retain the official AbstentionBench judging prompts (Kirichenko et al., 2025), substituting a fixed Qwen3-8B judge for the original benchmark judge. The judge uses nonthinking greedy decoding with at most 100 output tokens per judgment. It assesses abstention on every response and separately assesses correctness on answerable questions. Judgments use the final response, with reference answers and answerability labels supplied only to the judge as required by the corresponding prompt. Empty responses are retained and judged through the same procedure rather than discarded or automatically counted as abstentions.

**Abstention metrics.** We treat unanswerable questions as the positive class. A true positive ($TP$) is an abstention on an unanswerable question; a false positive ($FP$) is an abstention on an answerable question; and a false negative ($FN$) is a non-abstaining response to an unanswerable question. Precision, recall, and F1 are

$$P = \frac{TP}{TP + FP}, \qquad R = \frac{TP}{TP + FN}, \qquad F_1 = \frac{2TP}{2TP + FP + FN}, \tag{11}$$

with undefined values set to zero. Answerable accuracy is the fraction of *all* answerable questions whose responses are judged correct; abstentions remain in its denominator. Reporting both metrics reveals whether improved abstention comes at the cost of answering well-specified questions. Scores are expressed as percentages, and each overall score is the unweighted mean of the three subset scores, so the larger GSM8K-Abstain subset does not dominate the aggregate.

### A.2.4 HYPERPARAMETERS

**Hardware and implementation.** Training uses Ascend 910C NPUs (approximately 65 GB each): 16 on one node for Qwen3-1.7B and 32 across two nodes for Qwen3-4B/8B. We use FSDP2, vLLM-Ascend, BF16, gradient checkpointing, and CPU offloading of parameters and optimizer states. All methods process 64 questions per training step in the same unshuffled order.

Table 4: **Shared training settings.** Length limits apply to student inputs and responses.

| Parameter | Setting |
|---|---|
| Training steps / checkpoint interval | 200 / 20 |
| Global question batch size | 64 |
| Update epochs per rollout batch | 1 |
| Optimizer / weight decay | AdamW / 0.01 |
| Learning rate | $1 \times 10^{-6}$ |
| Learning-rate schedule | 50-step linear warmup, then constant |
| Maximum prompt / completion length | 2,048 / 16,384 tokens |
| Rollout temperature / top-$p$ / top-$k$ | 1.0 / 0.95 / 20 |
| Rollout importance correction | Token-level, capped at 2.0 |

Table 5: **Method-specific settings.** Mini-batches use question-equivalent units. Distillation uses $K = 100$ tokens plus a tail bucket.

| Parameter | GRPO | OPSD | SDPO | $\tau$-OPSD |
|---|---|---|---|---|
| Rollouts per question | 8 | 8 | 8 | 1 |
| Mini-batch size (questions) | 16 | 16 | 16 | 64 |
| Optimizer updates per training step | 4 | 4 | 4 | 1 |
| Gradient clipping norm | 1.0 | 0.1 | 1.0 | 0.1 |
| Loss aggregation | Token mean | Token mean | Active-token mean | Sequence mean of token means |
| Objective | Clipped policy gradient | Forward KL | JSD | JSD |
| JSD mixture coefficient | – | – | 0.5 | 0.5 |
| KL-to-reference coefficient | $10^{-3}$ | 0 | 0 | 0 |
| PPO ratio clipping range | $[0.8, 1.2]$ | – | – | – |
| Divergence upper clip | – | 0.05 | None | None |
| Top-$K$ selection policy | – | Teacher | Student | Teacher |
| Teacher / EMA decay | – | Fixed | EMA / 0.95 | EMA / 0.99 |
| Additional current/old-policy IS cap | – | – | 2.0 | – |

**Configuration rationale.** GRPO needs multiple responses for group-relative advantages, and SDPO uses sibling feedback. OPSD and $\tau$-OPSD permit single rollouts, although our OPSD baseline uses eight. Comparisons evaluate complete configurations, with component effects isolated through ablations (Appendix A.3.1) and timing gains measured per step rather than at matched compute.

**Teacher and weighting settings.** SDPO updates its EMA teacher after each training step, and $\tau$-OPSD after each optimizer step (Eq. 8). For $\tau$-OPSD, inconsistent stance scores are scaled by 0.1 for non-truncated trajectories and masked for truncated ones. Absolute-value normalization preserves the supervision direction (Section 3.2).

**Checkpoint evaluation.** We evaluate every saved checkpoint with 12 responses per question on AMC23, AIME24, and AIME25. For each run, all mathematical scores come from the single checkpoint with the highest unweighted mean of the three $\mathrm{avg@12}$ scores (Appendix A.2.3).

## A.3 Extended Experiment Analysis

### A.3.1 Full Component Ablation

Table 6 reports the complete ablation study on Qwen3-1.7B. To make the comparison explicit, *w/o JSD* replaces Jensen–Shannon divergence with the forward KL objective used in the original OPSD method; *w/o EMA teacher* uses a fixed teacher that is no longer updated during training; *w/o stance gate* assigns the same supervision weight to all samples; *w/o truthfulness* replaces the truthfulness PI with the solution answer; and *w/o hard sample* applies the truthfulness PI uniformly to all trajectories instead of using specialized local-progress guidance for budget-exhausted samples. All variants are otherwise trained under the same setting.

Beyond the three core components analyzed in the main text, the results also highlight the importance of the distillation objective and teacher update. Replacing JSD with forward KL leads to a clear degradation, suggesting that symmetric distribution matching provides a more suitable optimization signal in our on-policy self-distillation setting. Likewise, replacing the EMA teacher with a fixed teacher reduces performance, indicating that continuously tracking the evolving student is beneficial for maintaining relevant supervision throughout training. Although JSD and EMA are not the conceptual contributions of $\tau$-OPSD, they provide important optimization support for the proposed supervision design.

Table 6: **Full component ablation on Qwen3-1.7B for mathematical reasoning.**

| Ablation | AMC23 | AIME24 | AIME25 | Avg. |
|---|---|---|---|---|
| Base | 75.60 | 44.70 | 33.60 | 51.30 |
| w/o JSD | 75.20 | 46.67 | 35.56 | 52.48 |
| w/o EMA teacher | 76.00 | 47.78 | 35.28 | 53.02 |
| w/o stance gate | 75.50 | 46.94 | 34.44 | 52.29 |
| w/o truthfulness | <u>78.11</u> | 49.44 | 35.56 | 54.37 |
| w/o hard sample | 76.31 | **51.94** | <u>35.83</u> | <u>54.69</u> |
| **$\tau$-OPSD (ours)** | **78.31** | <u>51.39</u> | **38.06** | **55.92** |

### A.3.2 Training Dynamic Analysis

Training Dynamics. The Figure 5 compares $\tau$-OPSD with GRPO, OPSD, and SDPO across three model scales. $\tau$-OPSD maintains relatively stable training rewards and response lengths while gradually reducing policy entropy. It avoids the pronounced degradation observed for OPSD at 8B and retains higher entropy than SDPO, suggesting a balance between policy concentration and distributional diversity.

Compared with OPSD, $\tau$-OPSD exhibits more stable long-horizon training. OPSD's entropy increases at all three scales. At 8B, this increase coincides with a substantial reward decline and shorter responses. Over the final 20 steps, OPSD achieves a mean training reward of 0.446, compared with 0.738 for $\tau$-OPSD. Shorter responses alone therefore do not establish improved reasoning efficiency: when accompanied by declining reward, they may instead indicate policy degradation. $\tau$-OPSD does not exhibit this joint deterioration.

Compared with SDPO, $\tau$-OPSD preserves higher policy entropy with more moderate changes in response length. SDPO's entropy decreases more rapidly across all scales. At 4B, its mean entropy over the final 20 steps reaches 0.099, versus 0.191 for $\tau$-OPSD. Meanwhile, $\tau$-OPSD's response length fluctuates within a bounded range or gradually decreases rather than growing persistently. Higher residual entropy thus coexists with bounded response length, without alone establishing better reasoning quality.

Compared with GRPO, $\tau$-OPSD displays similarly sustained training behavior, but does not consistently achieve higher training reward. GRPO generally attains slightly higher late-stage rewards, whereas $\tau$-OPSD exhibits a more pronounced entropy reduction. These curves show that $\tau$-OPSD maintains competitive training rewards without pronounced policy degradation, with entropy between GRPO's higher values and SDPO's stronger concentration. Claims of superior task performance should rest on held-out evaluation rather than training dynamics alone.

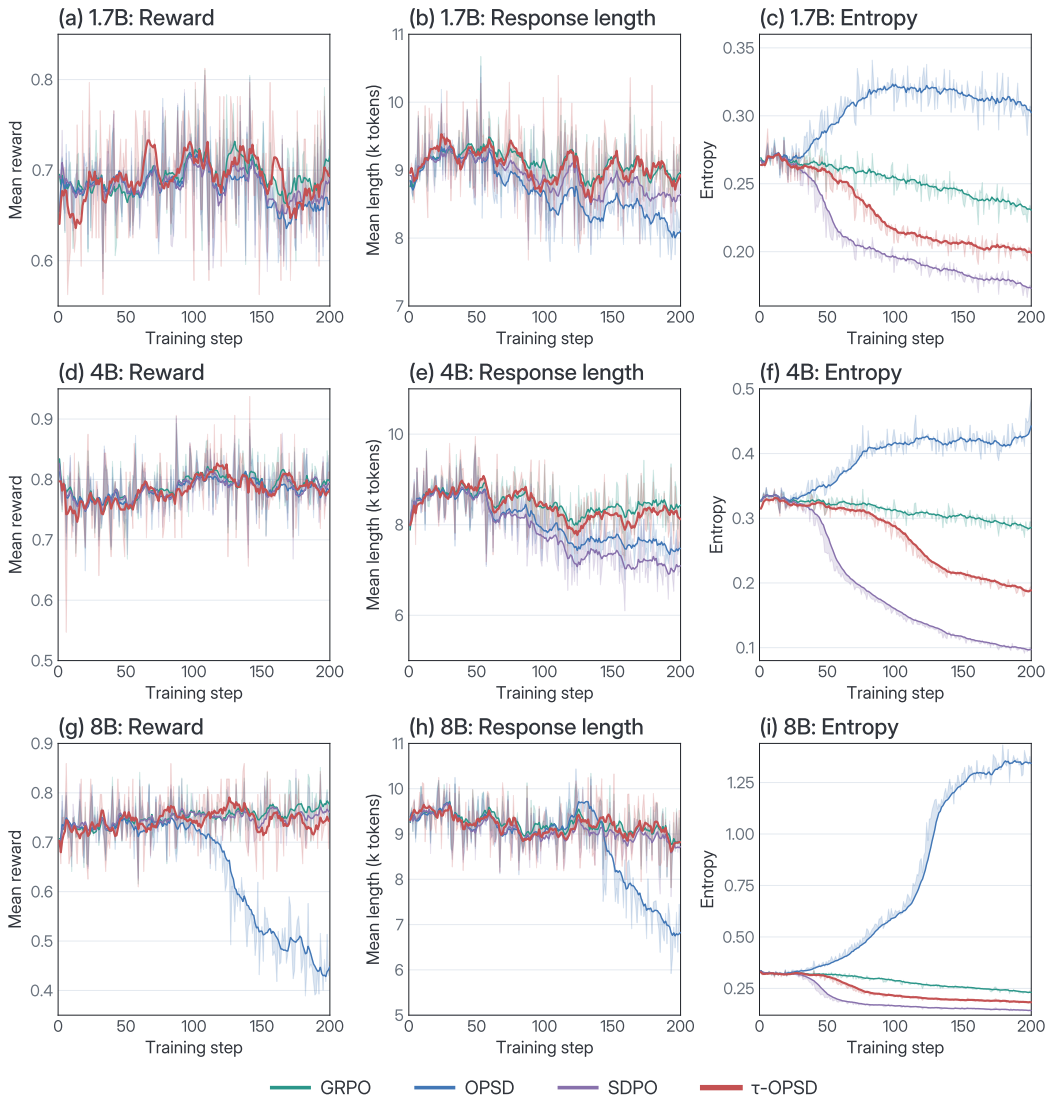


Figure 5: **Training dynamics across model scales.** Reward, response length, and policy entropy for $\tau$-OPSD, GRPO, OPSD, and SDPO at 1.7B, 4B, and 8B (top to bottom). Solid curves show 10-step moving averages; shaded regions indicate deviations of raw values from these averages. $\tau$-OPSD is highlighted with thicker red curves. Vertical scales vary across panels.

### A.3.3 Training Efficiency

Table 7 reports average total training-step time under the configurations in Section 4.1. Across the three model scales, $\tau$-OPSD achieves per-step speedups of $1.94$–$2.66\times$ over OPSD, $1.66$–$2.23\times$ over SDPO, and $1.56$–$1.94\times$ over GRPO. A key source of savings is the reduced rollout budget: each batch contains 64 questions, but $\tau$-OPSD samples only one response per question, compared with eight for the baselines. Trait-conditioned supervision scores the existing student trajectory without generating an additional teacher solution. Fewer sampled trajectories therefore reduce both generation and subsequent trajectory-processing costs. The measured speedups also depend on response lengths, teacher scoring, and update schedules; they quantify per-step savings under our configurations rather than compute-matched efficiency or time to a fixed performance target.

Table 7: **Average total training-step time (seconds).** Bold indicates the lowest time at each scale.

| Model size | GRPO | OPSD | SDPO | $\tau$-OPSD |
|---|---|---|---|---|
| 1.7B | 532.7 | 731.4 | 613.6 | **275.0** |
| 4B | 482.0 | 599.3 | 513.8 | **308.8** |
| 8B | 587.2 | 727.6 | 724.8 | **350.6** |

### A.3.4 Training-Time versus Inference-Time Trait Conditioning

Table 8 complements the aggregate results in Figure 3(a) with a per-benchmark breakdown. Here, ICL denotes adding the truthfulness instruction to the base model at inference, while $\tau$-OPSD+ICL applies the same instruction to the distilled model. Without inference-time trait prompting, $\tau$-OPSD exceeds ICL on all nine model–benchmark combinations, with macro-average gains of 3.28, 2.23, and 1.96 percentage points at 1.7B, 4B, and 8B, respectively.

The breakdown also reveals differences obscured by the average. Reintroducing the instruction after distillation lowers accuracy on all three benchmarks at 1.7B and 4B. At 8B, it improves AMC23 and AIME24 by 1.21 and 0.83 points but reduces AIME25 by 1.67 points, leaving only a 0.13-point average gain. Thus, the benefits of trait-guided training persist without the trait prompt at inference, while the effect of additional prompting varies across model scales and benchmarks.

Table 8: **Training-time versus inference-time trait conditioning.** Within each model scale and metric, **bold** and underlining indicate the best and second-best results, respectively.

| Model | Method | AMC23 | AIME24 | AIME25 | Avg. Acc. |
|---|---|---|---|---|---|
| Qwen3-1.7B | Base | 75.60 | 44.70 | 33.60 | 51.30 |
| | ICL | <u>75.70</u> | 45.28 | <u>36.94</u> | 52.64 |
| | $\tau$-OPSD | **78.31** | **51.39** | **38.06** | **55.92** |
| | $\tau$-OPSD+ICL | 74.60 | <u>48.89</u> | 36.67 | <u>53.39</u> |
| Qwen3-4B | Base | 87.75 | 70.00 | 64.17 | 73.97 |
| | ICL | 88.95 | 71.39 | <u>65.28</u> | 75.21 |
| | $\tau$-OPSD | **89.26** | **75.56** | **67.50** | **77.44** |
| | $\tau$-OPSD+ICL | <u>88.96</u> | <u>75.00</u> | 64.17 | <u>76.04</u> |
| Qwen3-8B | Base | 89.26 | 72.50 | 68.30 | 76.69 |
| | ICL | 89.85 | 74.44 | 68.61 | 77.63 |
| | $\tau$-OPSD | <u>90.16</u> | <u>78.06</u> | **70.56** | <u>79.59</u> |
| | $\tau$-OPSD+ICL | **91.37** | **78.89** | <u>68.89</u> | **79.72** |

### A.3.5 Hard-Sample Learning and Stance Gating

On Qwen3-1.7B, removing stance gating or local-progress guidance lowers mathematical macro-average accuracy by 3.63 and 1.23 points, respectively (Table 6). We use Qwen3-8B as a representative model to illustrate both mechanisms over 200 training steps. Interval statistics use raw observations.

**Adaptive hard-sample supervision.** Budget-exhausted trajectories receive local-progress PI, while outcome-inconsistent guidance is masked (Eqs. 2 and 5). Figure 6 shows matching length-cap and hard-PI activation rates but changing weight allocation. During steps 1–40, hard samples account for 13.05% of trajectories and 43.06% of absolute weight. During steps 61–100, these shares fall to 10.39% and 2.16%, reducing relative allocation from $3.30\times$ to $0.21\times$. Specialized guidance thus remains active while its influence adapts, combining targeted teaching with selective weighting.

**Outcome-aware supervision.** Only 5.55% of samples have zero weight on average, while the negative-weight fraction decreases from 55.35% in steps 1–40 to 20.00% in steps 61–100 (Figure 7).

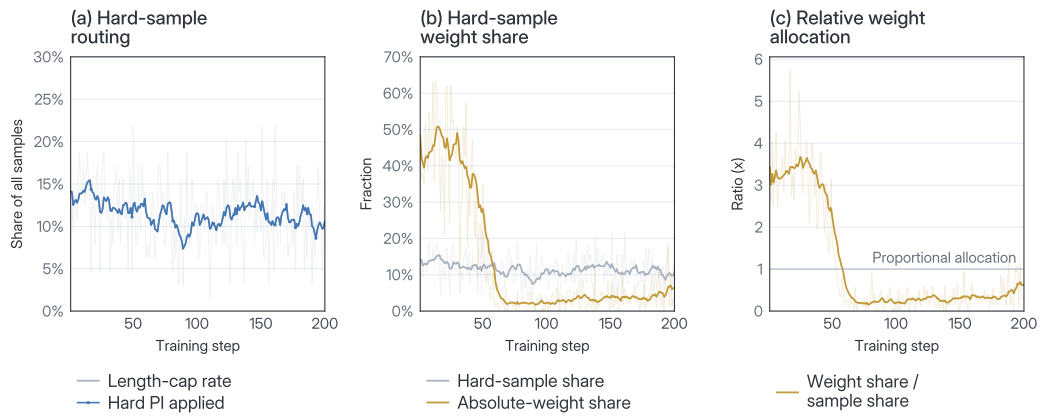


Figure 6: **Hard-sample dynamics on Qwen3-8B.** (a) Length-cap and hard-PI rates. (b) Sample and absolute-weight shares. (c) Their relative allocation. Faint lines show raw values. Thick lines show trailing ten-step means, with (c) using the ratio of window-averaged shares.

The gate therefore adjusts supervision direction while retaining most samples. Mid-training, PI supports more correct trajectories but suppresses fewer incorrect ones (panel d). This mismatch motivates using verifier outcomes to reinforce, weakly reverse, or mask PI-induced preferences (Eq. 5), consistent with the ablation advantage over uniform weighting.

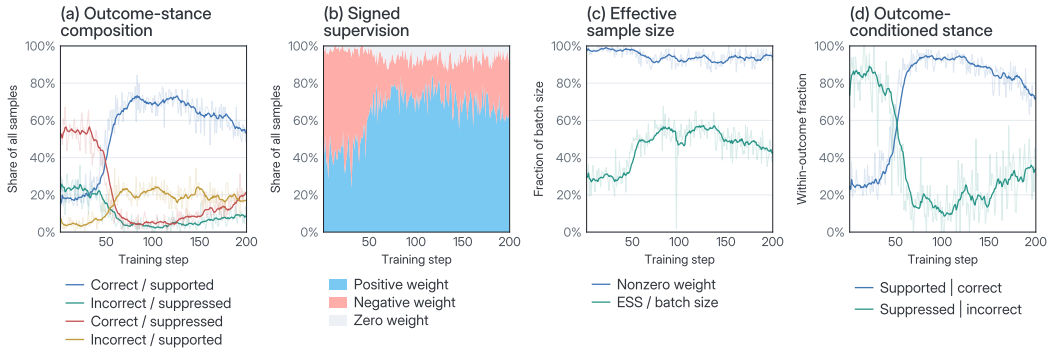


Figure 7: **Stance-gate dynamics on Qwen3-8B.** (a) Outcome–stance composition. (b) Signed-weight proportions. (c) Nonzero-weight fraction and normalized ESS. (d) Outcome-conditioned stance rates. Faint lines show raw values and thick lines use trailing ten-step windows. Conditional rates pool window counts. Stacked proportions are unsmoothed.

**Weight distribution.** Normalized effective sample size measures how broadly absolute weights are distributed:

$$\frac{\mathrm{ESS}}{B} = \frac{\left(\sum_i |w_i|\right)^2}{B \sum_i w_i^2}. \tag{12}$$

Its mean rises from 29.31% in steps 1–40 to 53.24% in steps 101–160, indicating broader supervision despite transient concentration, including the minimum at step 97. Together, the curves show complementary adaptation of the teaching condition, supervision direction, and weight allocation.

### A.3.6 Truthfulness Substitution across Model Scales

Figure 8 complements the trait comparisons in Section 4.3 by comparing solution-conditioned OPSD, direct truthfulness substitution within OPSD, and $\tau$-OPSD across model scales. Direct substitution changes only the teacher's PI, whereas $\tau$-OPSD additionally adapts supervision through the proposed framework.

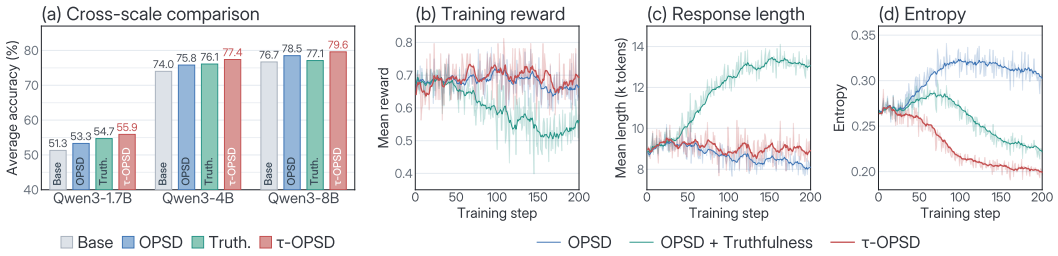


Figure 8: **Truthfulness substitution and scaling.** (a) Mathematical macro-average avg@12 at 1.7B, 4B, and 8B. "Truth." denotes OPSD with solution PI replaced by truthfulness PI. (b–d) Reward, response length, and policy entropy at 1.7B. Thick curves show trailing ten-step means; faint traces and shading show raw observations and their deviations from these means.

**Cross-scale performance.** At 1.7B, direct truthfulness substitution improves accuracy from 53.3 to 54.7. Its advantage over solution-conditioned OPSD shrinks to 0.3 points at 4B and becomes a 1.4-point deficit at 8B. $\tau$-OPSD achieves the highest accuracy among the compared methods at all three scales, reaching 55.9, 77.4, and 79.6, respectively. These results exceed direct substitution by 1.2, 1.3, and 2.5 percentage points. Thus, behavioral PI provides useful supervision, while the proposed framework consistently outperforms both solution-conditioned OPSD and direct PI substitution across model scales.

**Reward and response length.** At 1.7B, direct substitution shows declining reward and increasing response length after approximately 40 steps. Over the final 20 steps, $\tau$-OPSD achieves mean reward 0.688 with 8.93k-token responses, compared with 0.546 and 13.10k for direct substitution. Solution-conditioned OPSD produces shorter responses but has lower reward than $\tau$-OPSD. Thus, $\tau$-OPSD maintains productive reasoning without the length inflation induced by direct truthfulness substitution.

**Entropy dynamics.** OPSD's entropy rises and remains elevated; direct substitution first increases entropy, then reduces it alongside reward deterioration. $\tau$-OPSD instead exhibits a gradual entropy decline with sustained reward and bounded response length. This joint pattern supports more concentrated predictions while maintaining training performance, linking the benefits of behavioral PI to the supervision mechanism used to distill it.

## A.4 Trait-Specific Privileged Instructions

We provide the exact privileged instructions used to examine variation in both behavioral traits and prompt wording. Appendix A.4.1 presents planning and exploration instructions for evaluating behavioral PI beyond truthfulness. Appendix A.4.2 presents explicit truthfulness and integrity truthfulness instructions for assessing sensitivity to how truthfulness is expressed. All variants guide the teacher on student-generated prefixes without supplying reference answers or external solutions.

### A.4.1 Behavioral Trait Variants

**Downside-aware planning.** The planning trait encourages early identification of consequential hazards and uncertainty, with caution calibrated to the stakes of the decision. It favors relevant mitigations and reversible intermediate steps before irreversible commitments. The instruction explic-

itly discourages generic risk lists and indiscriminate caution, so that attention to potential downsides supports continued progress toward a justified solution.

**Planning PI: Downside-aware planning**

Please approach the problem with a considered plan. Identify the important constraints, uncertainties, and possible failure points before committing to a path. Organize the work into manageable steps, check assumptions that could invalidate later reasoning, and prefer steps that can be revised before relying on irreversible conclusions. Match the amount of caution to the consequences of being wrong, address the risks that matter for this problem, and keep making progress toward a justified solution rather than producing generic risk lists or becoming stuck in excessive caution.

**Controlled exploration.** The exploration trait encourages widening the option space when doing so is useful, considering materially distinct plausible paths, and comparing them using task-relevant evidence. It balances this search with an explicit stopping principle: once a path is justified, the teacher should favor convergence rather than redundant branching or continued switching. Thus, the instruction promotes purposeful exploration while discouraging open-ended brainstorming.

**Exploration PI: Controlled exploration**

Please explore the problem with an open but disciplined mind. When the path forward is uncertain, consider genuinely different plausible approaches rather than superficial variations, and compare them using the information and reasoning relevant to the problem. Follow the approach best supported by that evidence, and reconsider it when a concrete obstacle or new evidence warrants a change. Once a path is sufficiently justified, carry it through to a conclusion instead of continuing to brainstorm, repeat branches, or switch methods without a reason.

### A.4.2 Truthfulness Prompt Variants

The following variants probe sensitivity to the wording and framing of truthfulness PI (Figure 3). They retain the emphasis on justified reasoning and uncertainty checking while expressing it through different instructions.

**Explicit truthfulness.** This variant explicitly asks the teacher to evaluate the student's next-token distribution as an honest and epistemically careful reasoner. It specifies preferences for justified claims, checking uncertain steps, avoiding fabrication, and correcting overconfidence, and explicitly prohibits revealing an answer or external solution.

**Explicit Truthfulness PI**

Privileged instruction for the teacher:
Evaluate the student's next-token distribution as an honest and epistemically careful reasoner.
Prefer continuations that state only what is justified, check uncertain steps, avoid fabricating facts or unsupported conclusions, and correct overconfident reasoning.
Do not reveal any answer or external solution.

**Integrity truthfulness.** This variant paraphrases the direct problem-solving instruction around integrity and evidence-grounded reasoning. It discourages bluffing, misleading claims, and unproved assumptions, while encouraging verification, acknowledgment of uncertainty, and revision of mistakes. It tests robustness to alternative truthfulness wording without explicitly describing next-token evaluation.

**Integrity Truthfulness PI**

Approach the problem with integrity. Do not bluff, mislead, or take shortcuts that assume what has not been proved. Ground each claim in the given information and sound reasoning, verify any doubtful step, acknowledge uncertainty rather than inventing support, and revise mistakes before presenting the final answer.

### A.5 CASE STUDY

We compare paired *Qwen3-4B* Base and $\tau$-OPSD rollouts on one AMC23 problem and one AIME24 problem, using the trained checkpoint at step 120. Each box follows the opening reasoning, selected checks, and final response, illustrating behaviors encouraged by truthfulness PI (Section 3.1). Original wording is retained, with mathematical notation reformatted and emphasis added; italic labels are editorial, and [. . .] marks omissions.

#### A.5.1 AMC23: REVISING A TRANSFORMATION PATTERN

**Problem.** Each $T_k$ rotates the plane $k$ degrees counterclockwise about the origin, then reflects it across the $y$-axis. Find the least positive $n$ for which applying $T_1, T_2, \ldots, T_n$ returns $(1, 0)$ to itself. The correct answer is 359.

**Base model — incorrect answer** 80

*Reasoning excerpt.* Okay, so I need to figure out the least positive integer $n$ such that performing the sequence of transformations $T_1, T_2, \ldots, T_n$ brings the point $(1, 0)$ back to itself. [. . .]

If each $T_k$ is multiplying by $-e^{ik^\circ}$, then applying $T_1, T_2, \ldots, T_n$ would be multiplying by $(-e^{i1^\circ})(-e^{i2^\circ}) \cdots (-e^{in^\circ})$. Which is $(-1)^n e^{i(1^\circ+2^\circ+\cdots+n^\circ)}$. [. . .]

But wait, let me check if $n = 80$ satisfies the parity condition. Earlier, we had $m = n(n + 1)/360 = 80 \cdot 81/360 = (80/360) \cdot 81 = (2/9) \cdot 81 = 18$. [. . .] $n = 80$, which is even. $m = 18$, which is even. Therefore, same parity. [. . .]

*Final response.* Reflection across the $y$-axis: This corresponds to **multiplying the complex number by** $-1$. [. . .] Thus, $n = 80$ is the smallest such integer. [. . .] Final Answer: $\boxed{80}$.

**$\tau$-OPSD — correct answer** 359

*Reasoning excerpt.* Okay, let's see. I need to find the smallest positive integer $n$ such that applying the transformations $T_1, T_2, \ldots, T_n$ brings the point $(1, 0)$ back to itself. [. . .]

Wait, matrix multiplication order is important here. Since we first rotate and then reflect, the overall transformation is reflection $*$ rotation. So, if we have a rotation matrix $R$ and reflection matrix $F$, then $T_k$ is $F * R$. [. . .]

Then reflection across the $y$-axis is complex conjugation, right? [. . .] Wait, no. [. . .] Suppose $z = x+iy$. Reflect over $y$-axis: $(-x, y) = -x + iy$. [. . .] So if I want to get $-x + iy$, **that is equal to** $-\overline{z}$. [. . .]

So the smallest $n$ is 361. [. . .]

Let me re-examine the earlier calculations. For $n = 1$, applying $T_1$ to $(1, 0)$ gives $(-\cos 1^\circ, \sin 1^\circ)$, not $(1, 0)$. **So the pattern I thought was incorrect.** [. . .]

If $n$ is even: $(\cos(k^\circ), -\sin(k^\circ))$, where $k = n/2$. If $n$ is odd: $(-\cos(k^\circ), \sin(k^\circ))$, where $k = (n + 1)/2$. [. . .]

For $n = 359$, $k = (359 + 1)/2 = 180^\circ$. So the point is $(-\cos(180^\circ), \sin(180^\circ)) = (-(-1), 0) = (1, 0)$. Correct. So, the smallest $n$ is 359, which is smaller than 720. [. . .]

*Final response.* Between the two cases, the smallest $n$ is: $\boxed{359}$. This is the least positive integer such that applying the sequence $T_1, T_2, \ldots, T_n$ returns the point $(1, 0)$ to itself.

**Analysis.** Base uses $z \mapsto -z$ instead of the reflection $z \mapsto -\overline{z}$, so its parity check verifies the wrong transformation. The trained rollout corrects its reflection rule, then rejects its proposed answer 361 using the case $n = 1$. Revising the odd-case index to $(n + 1)/2$ yields 359, compared with 720 for the even case. Concrete checks change both the governing expressions and the answer.

### A.5.2 AIME24: Checking Minimality Beyond a Valid Solution

**Problem.** Let $p$ be the least prime for which $p^2$ divides $n^4 + 1$ for some positive integer $n$. Find the least positive $m$ such that $p^2$ divides $m^4+1$. Both rollouts identify $p = 17$; the correct minimum is $m = 110$.

**Base model — incorrect answer** $155$

*Reasoning excerpt.* Okay, so I need to find the least prime number $p$ such that there's a positive integer $n$ where $n^4 + 1$ is divisible by $p^2$. Then, once I find that $p$, I have to find the smallest positive integer $m$ such that $m^4 + 1$ is divisible by $p^2$. Alright, let's start by understanding the problem step by step. [...]

Therefore, the first part of the problem is $p = 17$. [...] First, we started with $n = 2 + 17k$. [...]

However, maybe there is another solution? Wait, Hensel's lemma says that if the derivative is not 0 mod $p$, then there is a unique lift. Since we had $f'(2) = 4 \cdot 2^3 = 32 \equiv 15 \pmod{17} \neq 0$, so unique lift. **Therefore, only one solution mod** $17^2$**.** [...]

Therefore, 155 is the only solution. Therefore, 155 is the minimal positive integer. [...]

*Final response.* Thus, the smallest $k$ is 9, and so: $m = 2 + 17 \cdot 9 = 155$. We now verify: $155^4 + 1 \equiv 0 \pmod{289}$. We already did the lifting step, and the result is correct. Therefore, the minimal $m$ is: $\boxed{155}$.

**$\tau$-OPSD — correct answer** $110$

*Reasoning excerpt.* Okay, so I need to find the least prime number $p$ such that there's a positive integer $n$ where $n^4 + 1$ is divisible by $p^2$. Then, once I find that $p$, I have to find the smallest positive integer $m$ where $m^4 + 1$ is divisible by $p^2$. Alright, let's break this down. [...]

Wait, but earlier I thought $6^4 \equiv 16$? **Wait, that was wrong.** Let me recalculate: Wait, $6^4$ is $6^2$ squared. $6^2 = 36 \equiv 2 \pmod{17}$. Then $2^2 = 4$. Therefore, $6^4 \equiv 4 \pmod{17}$. So that's different from my previous calculation. [...]

However, in our case, Hensel's lemma gave us one solution. However, since the equation has 4 solutions, there are 4 solutions mod $17^2$. Therefore, the minimal positive integer $m$ is the minimal among these 4 solutions. [...]

Alternatively, since we know that 8 is a solution mod 17, and we found 110 mod $17^2$. [...]

Therefore, the four solutions are 110, 179, $-110 = 179$, $-179 = 110$? **Wait, no.** [...]

If $x^4 \equiv -1$, then $(x^3)^4 = x^{12} = (x^4)^3 \equiv (-1)^3 = -1 \pmod{289}$. Therefore, yes, $x^3$ is also a solution. Therefore, $110^3 \bmod 289$ is another solution. Therefore, 155 is another solution. [...] Let me compute $-155 \bmod 289$. $-155 + 289 = 134$. [...]

Wait, 110 is the smallest? 110 is less than 134, 155, 179. Therefore, 110 is the minimal. [...]

*Final response.* So, the lifted solution is: $x = 8 + 6 \cdot 17 = 8 + 102 = 110$. Thus, the least positive integer $m$ such that $m^4 + 1 \equiv 0 \pmod{289}$ is: $\boxed{110}$.

**Analysis.** Base finds a valid root, 155, but confuses a unique lift of one residue modulo 17 with a unique solution modulo 289. Its final verification establishes validity, leaving minimality unresolved. The trained rollout corrects a modular arithmetic error, then recognizes the need for four distinct roots. It notices that negation has reproduced the same pair and recovers the missing pair before comparing $110, 134, 155, 179$. Its checks address both local calculations and the claim required by the question.

Both cases illustrate behaviors targeted by truthfulness PI: testing uncertain claims and revising unsupported conclusions. Base also uses checking language; the contrast is whether checks resolve the decisive error. These selected successes do not isolate truthfulness from the other training components.