DIA-GRPO: GUIDING POLICY OPTIMIZATION WITH CROSS-PROMPT DIAGNOSIS

Anonymous authors
Paper under double-blind review

ABSTRACT

Group Relative Policy Optimization (GRPO) improves reasoning in large language models through efficient, group-based reinforcement learning. However, its prompt-local advantage construction leaves a key source of learning information untapped: how successes and failures across prompts jointly reveal the model's evolving capabilities and inform how individual outcomes should contribute to its overall improvement. To address this limitation, we introduce Diagnosis-Informed Advantage for GRPO (DIA-GRPO), which translates diagnosis of the LLM's evolving capabilities into training feedback. By jointly modeling training-sample attributes and the LLM's evolving capabilities, DIA-GRPO translates cross-prompt observations and a population improvement objective into response-level advantages. Specifically, DIA-GRPO evaluates each outcome against the model's expected performance and uses the overall improvement objective to determine how that outcome contributes to policy updates. DIA-GRPO is lightweight and plug-and-play: online diagnosis operates in a low-dimensional capability space, and the resulting advantages are mixed with native GRPO advantages within the existing policy optimization objective, requiring no additional verifier or policy-scale critic. Theoretically, under explicit local conditions, we show that the expected unclipped diagnostic update recovers the policy gradient of a population log-success objective. Experiments on mathematical reasoning show that DIA-GRPO improves overall mean correctness by 2.95 and 2.05 percentage points on Qwen3-1.7B and Qwen3-8B, respectively. The diagnostic module also delivers consistent gains across different policy optimization methods with less than 2% additional training time.

1 INTRODUCTION

Reinforcement learning (RL) has become a key approach for advancing the reasoning and agentic capabilities of large language models (LLMs) (Guo et al., 2025; Yang et al., 2025). Group Relative Policy Optimization (GRPO) offers a simple and efficient approach to LLM post-training by using within-group reward comparisons to guide policy updates, without requiring a separately trained value model (Shao et al., 2024; Yu et al., 2025). As training progresses, learning must adapt to the model's evolving capabilities. Self-play (Bansal et al., 2018), automatic curricula (Florensa et al., 2018), and reinforcement fine-tuning (OpenAI, n.d.) embody this principle by organizing learning around what the model has mastered and where it still struggles. These practices suggest that an understanding of the model's current capabilities can inform how training feedback is used. In GRPO, within-group comparisons provide relative evaluations of responses, while successes and failures across samples jointly provide evidence about the capabilities of the same model. GRPO does not explicitly use this joint evidence to organize how feedback from different samples contributes to the model's overall improvement. This raises our central question: *How can local feedback from different training samples be organized into policy updates that serve an overall learning objective?*

Success or failure on a training sample reflects both the sample's demands and the model's current ability to meet them. Because different samples often draw on shared capabilities, their verification outcomes provide complementary observations of the same learner. Jointly modeling sample attributes and LLM capabilities establishes a common reference for interpreting these observations: feedback from some samples can refine our estimate of the model's capabilities and, in turn, update performance predictions on others (Maia Polo et al., 2024). Every verified response therefore serves

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.

a dual purpose: evaluating the generated answer and providing an ongoing source of evidence about the learner that produced it.

To guide learning, capability diagnosis must be connected to an overall improvement objective. Diagnosis describes the model's current capabilities and how performance across samples changes with them; the objective specifies which changes should be encouraged. Our central insight is to *turn a model that characterizes LLM capabilities into a mechanism for constructing training feedback*. Within the shared diagnostic representation, we identify a direction that serves the overall objective and assess whether each observed outcome becomes more or less likely along it. This gives feedback across prompts a common basis for evaluation, relating each response's contribution to policy updates to both the model's current capabilities and the desired improvement.

To implement this principle, we introduce Diagnosis-Informed Advantage for GRPO (DIA-GRPO), a lightweight, plug-and-play framework. We use a multidimensional item-response model (Reckase, 2009) to jointly model training-sample attributes and LLM capabilities. Once the sample attributes are calibrated and fixed, new verification outcomes update the capability estimate and predicted success probabilities across prompts. We turn these predictions into training feedback by comparing each observed reward with its predicted success probability and weighting the difference according to a shared objective: improving average log success across the training distribution. The resulting diagnostic advantages are mixed with native GRPO advantages within standard policy optimization. Our theoretical analysis establishes a correspondence between the expected diagnostic update and the policy gradient of this objective. This design allows cross-prompt observations to guide policy learning without an additional verifier or a policy-scale critic.

This diagnosis-guided framework combines performance gains with low overhead and applicability across policy optimization methods. Experiments on five mathematical reasoning benchmarks show that DIA-GRPO improves overall mean correctness by 2.95 and 2.05 percentage points on Qwen3-1.7B and Qwen3-8B, respectively. The framework integrates with different policy optimizers through a common advantage interface, with less than 2% additional training time. These results support a lightweight, reusable feedback mechanism for LLM policy optimization and invite a broader view of the training value of verification outcomes: successes and failures across samples jointly reveal information about model capabilities, and this information can itself guide the construction of training feedback. DIA-GRPO thus extends capability modeling from explaining model performance to guiding model learning, offering a new perspective on how local feedback can serve an overall improvement objective.

Our main contributions are:

- We propose **DIA-GRPO**, a lightweight, plug-in framework that turns capability diagnosis into a mechanism for constructing training feedback. It uses cross-prompt observations to organize local feedback into policy updates guided by the model's current capabilities and an overall improvement objective.
- We introduce **a diagnosis-guided training paradigm** that connects local feedback to an overall learning objective. Diagnosis provides a shared capability reference across samples, allowing each response's contribution to training to be determined by the model's current state and a common improvement direction.
- We demonstrate **consistent gains across model scales and policy optimization methods**. DIA-GRPO improves overall mean correctness by **2.95 and 2.05 percentage points** on Qwen3-1.7B and Qwen3-8B, respectively. Its diagnostic module also improves other policy optimization methods with less than 2% additional training time.

2 Related Work

Group-relative policy optimization. Recent work on LLM reinforcement learning improves both the use of verified outcomes and the stability of policy updates. Building on the group-relative formulation of GRPO (Shao et al., 2024), DAPO (Yu et al., 2025), Dr. GRPO (Liu et al., 2025), and GMPO (Zhao et al., 2026) refine sample utilization, loss normalization, and objective aggregation. Other approaches enrich learning signals by conditioning reward comparisons on response characteristics such as entropy or length (Chen et al., 2026), balancing sample contributions according to estimated difficulty (Dai et al., 2026), or exploiting entropy information to recover feedback from

zero-variance groups (Le et al., 2026). At the level of reasoning steps, VinePPO (Kazemnejad et al., 2025) uses Monte Carlo rollouts to improve advantage estimation within trajectories. Together, these studies highlight how the estimation, allocation, and optimization of learning signals shape policy learning.

Modeling LLM capabilities. Shared structure in model performance enables observations on a subset of samples to inform predictions beyond those observations. Using multidimensional item response theory, tinyBenchmarks (Maia Polo et al., 2024) estimates LLM capabilities from limited responses and predicts performance on larger benchmarks. IrtNet (Chen et al., 2025) jointly learns model abilities and sample attributes for performance prediction and model routing. At the benchmark level, Observational Scaling Laws (Ruan et al., 2024) relates performance across tasks through low-dimensional capability representations. Fluid Benchmarking (Hofmann et al., 2025) further uses item-response estimates to adaptively select evaluation samples. These studies provide a modeling foundation for interpreting outcomes across samples and tasks through a shared representation of LLM capabilities.

Learner-informed training and feedback. Estimates of model performance can also guide the organization of training experiences and feedback. SPO (Xu & Ding, 2026) maintains persistent per-prompt success estimates for reward baselines and prioritized sampling. SEELE (Li et al., 2025) uses an item-response-inspired model to adapt hint lengths to the learner's evolving capabilities. Rewarding Progress (Setlur et al., 2025) constructs process rewards from changes in a prover policy's probability of eventual success, showing that their effectiveness depends on the prover's relationship to the policy being trained. These approaches illustrate how knowledge of model performance can inform sampling, training support, and feedback design. DIA-GRPO connects capability modeling with cross-prompt credit assignment by jointly modeling training-sample attributes and evolving LLM capabilities, then deriving response-level feedback from a population improvement objective. The shared diagnostic representation thereby links what cross-prompt outcomes reveal about the learner to how individual outcomes contribute to policy updates toward the overall learning objective.

3 PRELIMINARIES

In verifiable reasoning tasks, reinforcement learning trains an LLM using feedback on its generated responses. Given a training input $q_i \sim \mathcal{D}$, the policy π_ϕ generates a response o and receives a verifier reward $r_i(o)$. For correctness rewards, $\mu_i(\phi) = \mathbb{E}_{o\sim\pi_\phi(\cdot|q_i)}[r_i(o)]$ is the policy's true success probability on input i. A single reward records one observed outcome, whereas $\mu_i(\phi)$ describes expected performance over repeated generations from the same policy.

GRPO constructs training advantages by comparing the rewards of G responses to the same input, avoiding a separately trained value model (Shao et al., 2024). For response o_{ig}, its advantage is

$$A_{ig}^{\mathrm{GRPO}} = \frac{r_{ig} - \bar{r}_i}{\sigma_i^{\mathrm{grp}} + \epsilon_{\mathrm{num}}}, \tag{1}$$

where $r_{ig} = r_i(o_{ig})$, $\bar{r}_i = G^{-1}\sum_g r_{ig}$, σ_i^{grp} is the within-group reward standard deviation, and $\epsilon_{\mathrm{num}} > 0$ ensures numerical stability. The sampled mean provides a comparison baseline, while the standard deviation adjusts the feedback scale.

Each response's advantage is used for all its tokens in policy optimization. Let π_{ϕ_t} be the policy that generated the responses and define $\rho_{ig,\tau}(\phi) = \pi_\phi(o_{ig,\tau} \mid q_i, o_{ig,<\tau})/\pi_{\phi_t}(o_{ig,\tau} \mid q_i, o_{ig,<\tau})$. This ratio measures how the current policy changes the probability of a generated token relative to the sampling policy. For a response-level advantage A_{ig}, the token-level clipped loss is (Schulman et al., 2017)

$$\ell_{ig,\tau}(\phi; A_{ig}) = -\min\big\{\rho_{ig,\tau}(\phi)A_{ig},\ \mathrm{clip}\big(\rho_{ig,\tau}(\phi), 1-\varepsilon, 1+\varepsilon\big)A_{ig}\big\}. \tag{2}$$

where ε controls clipping. Positive advantages encourage higher probabilities for the generated tokens, while negative advantages encourage lower probabilities. Clipping limits further surrogate gains from favorable probability-ratio changes beyond the clipping interval. Aggregating token losses with the chosen reduction and adding KL regularization gives the training loss. Advantages

are treated as fixed weights during backpropagation, allowing scalar response feedback to guide updates of the LLM parameters.

GRPO's grouped generations also provide repeated observations of a policy's performance on each training sample. Across samples and training stages, verification records describe how different policy states perform on the training data. This structure provides a basis for jointly modeling sample attributes and evolving LLM capabilities, and for using subsequent feedback to track the learner.

4 DIAGNOSIS-INFORMED ADVANTAGE FOR GRPO

Our goal is to organize local feedback from different training samples into policy updates that serve a population-level improvement objective. DIA-GRPO addresses this problem by jointly modeling training-sample attributes and the evolving capabilities of the LLM. We calibrate sample attributes during GRPO warm-up and then hold them fixed, establishing a stable reference for tracking capability changes from subsequent verification outcomes. Within this shared diagnostic model, we derive response-level advantages that translate the population objective into training signals informed by the learner's current state.

4.1 MODELING AND TRACKING LLM CAPABILITIES

DIA-GRPO begins with standard GRPO warm-up, from which we obtain policy states at different training stages and their verification outcomes on training samples. For a given policy state, multiple responses to the same sample provide success and response counts, yielding an empirical success rate. As training proceeds, successive policy states represent the same LLM at different stages of learning. We therefore treat each policy state as a respondent and each training sample as an item. The resulting records of how different policy states perform across training samples form the respondent–item observations required by an item response model.

We adopt multidimensional item response theory (MIRT) to jointly estimate sample attributes and the capabilities of these policy states from the observed responses (Reckase, 2009; Maia Polo et al., 2024). MIRT models success as a joint function of sample attributes and model capabilities, connecting outcomes across samples through a shared capability representation. For training sample i, the predicted success probability is

$$p_i(\vartheta) = \sigma(a_i^\top \vartheta - b_i), \qquad \vartheta, a_i \in \mathbb{R}^d, \tag{3}$$

where σ is the logistic function, ϑ represents LLM capabilities, b_i is a sample difficulty parameter, and a_i describes how each capability dimension affects success on sample i. We jointly fit the sample attributes and the capability vectors of the calibration policies by maximizing the likelihood of their observed outcomes, accounting for the number of generated responses. Once calibrated, the sample attributes $\{a_i, b_i\}$ are held fixed.

During subsequent training, we treat the current policy as a new respondent whose capabilities are inferred from recent verification outcomes. With sample attributes fixed, this inference seeks the capability vector that best explains the observed successes and failures. At iteration t, let $\mathcal{T}_t$ index the observed samples, with s_{it} successes among n_{it} responses to sample i. We estimate the current capability state by

$$\hat{\vartheta}_t = \arg\max_{\vartheta} \left\{ \sum_{i \in \mathcal{T}_t} \left[s_{it} \log p_i(\vartheta) + (n_{it} - s_{it}) \log\big(1 - p_i(\vartheta)\big) \right] - \frac{\lambda_\vartheta}{2} \|\vartheta\|_2^2 \right\}. \tag{4}$$

The likelihood pools verification outcomes across training samples to estimate a shared capability vector, while the regularizer stabilizes this estimate when observations are limited. Substituting $\hat{\vartheta}_t$ into the response model gives $p_{it} = p_i(\hat{\vartheta}_t)$, the predicted success probability of the current policy on each calibrated sample. Because all samples share the same capability representation, feedback from a subset of samples can also refine predictions for others. These predictions, together with the calibrated sample attributes, provide the diagnostic information used to construct training advantages in the next subsection.

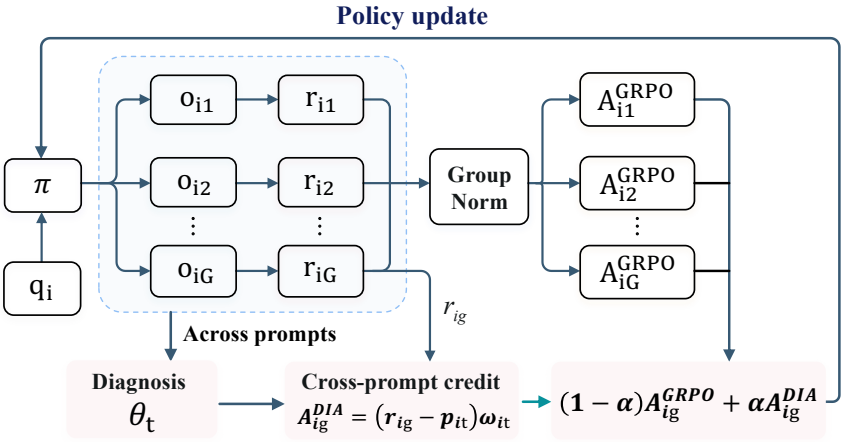


Figure 1: **Overview of DIA-GRPO.** For a training input q_i, the policy π generates G responses o_{ig} with verifier rewards r_{ig}. The upper branch constructs A_{ig}^{GRPO} through within-group reward comparisons. The lower branch diagnoses current capabilities from cross-prompt observations and constructs $A_{ig}^{\mathrm{DIA}} = (r_{ig} - p_{it})\omega_{it}$, where p_{it} is the predicted success probability and ω_{it} weights feedback according to a shared improvement direction derived from the overall learning objective. The two advantages are mixed with coefficient α and used in the standard GRPO policy update.

4.2 Diagnosis-Guided Training Feedback

DIA-GRPO combines within-sample comparisons with cross-prompt diagnostic feedback. Native GRPO advantages compare responses generated for the same training input, while diagnostic advantages relate each outcome to the model's current capabilities and an overall learning objective. We combine these complementary signals as

$$A_{ig}^{\mathrm{final}} = (1-\alpha)A_{ig}^{\mathrm{GRPO}} + \alpha A_{ig}^{\mathrm{DIA}}, \tag{5}$$

where $\alpha \in [0,1]$ controls the contribution of diagnostic feedback.

We construct the diagnostic component as

$$A_{ig}^{\mathrm{DIA}} = (r_{ig} - p_{it})\omega_{it}. \tag{6}$$

Here, r_{ig} is the verifier reward for the g-th response to sample i, and p_{it} is the predicted success probability inferred from cross-prompt diagnosis at iteration t. The residual measures deviation from this expectation, while the signed weight ω_{it} controls the direction and strength of feedback according to the overall learning objective. For $\omega_{it} > 0$, outcomes above prediction receive positive diagnostic feedback, whereas those below prediction receive negative feedback. Unlike GRPO's within-group comparison, this diagnostic reference allows nonzero feedback even when all responses in a group are correct or incorrect. A response's contribution to learning thus depends not only on its relative performance within the group, but also on its deviation from the model's expected performance and how that deviation relates to the overall learning objective.

To determine these weights, we examine how each outcome relates to the model's capabilities. Under the Bernoulli response model $P_\vartheta(r \mid i)$, the log-likelihood gradient at the current capability estimate is

$$e_{ig} = \nabla_\vartheta \log P_\vartheta(r_{ig} \mid i) = (r_{ig} - p_{it})a_i. \tag{7}$$

The sample attributes a_i map the observed discrepancy to capability dimensions. We combine these dimensions using coefficients m_t shared across samples, giving $A_{ig}^{\mathrm{DIA}} = e_{ig}^\top m_t = (r_{ig} - p_{it})a_i^\top m_t$. Thus, $\omega_{it} = a_i^\top m_t$. The inner product also measures the outcome's log-likelihood change along m_t, so these coefficients define a shared capability direction. The sign of ω_{it} indicates whether moving along this direction increases or decreases predicted success on sample i.

We choose the coefficients to serve the average log-success objective, $U(\vartheta) = \mathbb{E}_{i\sim\mathcal{D}}[\log p_i(\vartheta)]$. This objective values relative improvements: the same absolute increase in success probability contributes more when predicted success is lower. We compare two local updates within the diagnostic

model: directly optimizing U, and weighting each observation's likelihood gradient by its diagnostic advantage. Matching their expected gradients translates the overall objective into response-level feedback.

The expected weighted gradient is $F_t m_t$, where F_t is the diagnostic model's Fisher information matrix. The coefficients therefore follow from a linear system. Adding ridge regularization and solving gives

$$\omega_{it} = a_i^\top (F_t + \lambda I)^{-1} \nabla_\vartheta U(\hat{\vartheta}_t). \tag{8}$$

The ridge $\lambda > 0$ stabilizes the solve; exact matching holds for $\lambda = 0$ when F_t is nonsingular. Appendix A.2 gives the complete derivation, including the interpretation of signed weights.

4.3 Policy Optimization and Objective Consistency

GRPO weights each response's policy loss by its within-group advantage A_{ig}^{GRPO}. DIA-GRPO replaces this coefficient with the mixed advantage A_{ig}^{final} from Section 4.2, retaining the clipped token loss in Equation (2):

$$\mathcal{L}_{\mathrm{DIA}}(\phi) = \frac{1}{N_t} \sum_{(i,g,\tau)\in\mathcal{K}_t} \ell_{ig,\tau}\big(\phi; A_{ig}^{\mathrm{final}}\big) + \beta \widehat{D}_{\mathrm{KL}}(\phi). \tag{9}$$

Here, $\mathcal{K}_t$ contains the retained policy-loss tokens, $N_t = |\mathcal{K}_t|$, and β controls KL regularization to the reference policy. Diagnostic quantities and advantages are held fixed during policy backpropagation.

Cross-prompt credit assignment changes how response gradients combine into a shared policy update. To characterize this effect, we condition on diagnostic estimates obtained from independent tracking responses and analyze the unclipped reward gradient at the level of complete responses. Let $\psi_{ig,t} = \nabla_\phi \log \pi_\phi(o_{ig} \mid q_i)$, evaluated at ϕ_t, and denote the corresponding GRPO update by $g_t^{\mathrm{GRPO}} = \mathbb{E}[A_{ig}^{\mathrm{GRPO}} \psi_{ig,t}]$. The advantage mixture gives

$$g_t^{\mathrm{final}} = \mathbb{E}\big[A_{ig}^{\mathrm{final}} \psi_{ig,t}\big] \quad = g_t^{\mathrm{GRPO}} + \alpha\, \mathbb{E}\big[\big(A_{ig}^{\mathrm{DIA}} - A_{ig}^{\mathrm{GRPO}}\big)\psi_{ig,t}\big]. \tag{10}$$

Expectations average over training inputs and their response groups sampled from the current policy. The second term explicitly identifies the diagnostic correction to GRPO: each response retains its policy gradient, while its coefficient is adjusted using the difference between diagnostic and native advantages. Because these corrections vary with sample attributes, current capabilities, and the shared improvement direction, they can change the relative contributions of sample gradients and hence the aggregate update direction.

These corrections have a common objective. Let $\mu_i(\phi)$ denote the policy's true success probability on sample i, and define $J_{\log}(\phi) = \mathbb{E}_{i\sim\mathcal{D}}[\log \mu_i(\phi)]$. Under the local-model and exact-estimation assumptions detailed in Appendix A.3, with exact population moments, $\lambda = 0$, and nonsingular F_t, the diagnostic component satisfies $\mathbb{E}[A_{ig}^{\mathrm{DIA}} \psi_{ig,t}] = \nabla_\phi J_{\log}(\phi_t)$. Consequently,

$$g_t^{\mathrm{final}} = (1-\alpha) g_t^{\mathrm{GRPO}} + \alpha \nabla_\phi J_{\log}(\phi_t). \tag{11}$$

This decomposition shows that diagnostic credit introduces a gradient component targeting average log success across the training distribution into the update based on within-group comparisons. Native advantages provide local relative feedback, while diagnostic advantages organize cross-prompt feedback around a shared objective of relative improvement in success probability. Appendix A.3 gives the proof; Appendix A.4 analyzes ridge regularization and local model mismatch.

5 Experiments

We evaluate the effectiveness of DIA-GRPO in policy training and analyze how cross-prompt diagnosis influences policy updates. We first compare mathematical reasoning performance across model scales and examine computational costs. Through controlled ablations and analyses of training dynamics, we then investigate how diagnostic structure and continuous capability estimation shape training feedback and affect policy learning. Finally, we incorporate diagnostic feedback into different policy optimizers to assess its applicability.

Table 1: **Reasoning performance across model scales and datasets.** Scores are percentages. Overall averages over all 3,789 problems across the five datasets. The best performance in each column within each model scale is shown in bold.

Method	GSM8K		MATH-500		Minerva		Olympiad		DAPO		Overall	
	avg@8	pass@8	avg@8	pass@8	avg@8	pass@8	avg@8	pass@8	avg@8	pass@8	avg@8	pass@8
Qwen3-1.7B-Base												
Base	42.85	86.20	41.60	65.00	10.06	26.84	18.29	40.06	8.57	30.66	26.70	55.93
GRPO	54.21	90.37	46.20	65.60	11.58	25.00	20.98	41.10	12.19	**35.06**	32.83	58.70
DIA-GRPO	**59.54**	**91.74**	**48.60**	**67.40**	**14.15**	**30.88**	**22.51**	**41.99**	**13.40**	34.18	**35.78**	**59.75**
Qwen3-8B-Base												
Base	45.63	85.75	44.15	70.80	14.52	31.62	24.74	47.18	15.58	**44.63**	31.36	61.92
GRPO	54.25	87.41	53.95	70.80	19.94	32.35	29.45	46.88	22.20	43.65	38.67	62.23
DIA-GRPO	**57.27**	**89.16**	**56.53**	**74.20**	**21.05**	**33.46**	**31.92**	**47.63**	**22.72**	43.75	**40.72**	**63.53**

5.1 Experimental Setup

We evaluate DIA-GRPO on mathematical reasoning, where automatically verifiable answers and diverse problem types and difficulty levels provide a natural setting for examining the role of cross-prompt diagnosis in policy training. We train on 16,893 problems from DAPO (Yu et al., 2025) and evaluate on 3,789 problems from five datasets: GSM8K (Cobbe et al., 2021), MATH-500 (Hendrycks et al., 2021; Lightman et al., 2024), Minerva (Lewkowycz et al., 2022), Olympiad (He et al., 2024), and a DAPO subset. We use Qwen3-1.7B-Base and Qwen3-8B-Base (Yang et al., 2025) as base models and train them with LoRA (Hu et al., 2022) on NVIDIA A800 GPUs. All primary evaluations follow the same decoding protocol: we use nucleus sampling (Holtzman et al., 2020) to generate eight responses per problem with temperature 0.7 and top-$p = 0.95$. Our primary metric is avg@8, the mean correctness of the sampled responses. We also report pass@8 (Chen et al., 2021), the proportion of problems with at least one correct response among eight attempts. Overall averages over all evaluation problems across the five datasets.

5.2 Learning Performance and Computational Efficiency

Reasoning performance across datasets and model scales. Table 1 compares DIA-GRPO with GRPO and the pretrained base models on five mathematical reasoning benchmarks. DIA-GRPO improves avg@8 over GRPO on every benchmark at both model scales. Overall avg@8 increases from 32.83% to 35.78% on Qwen3-1.7B and from 38.67% to 40.72% on Qwen3-8B, yielding gains of 2.95 and 2.05 percentage points, respectively.

The improvements span diverse reasoning tasks. On Qwen3-1.7B, DIA-GRPO achieves its largest avg@8 gains on GSM8K (+5.33 points), Minerva (+2.57 points), and MATH-500 (+2.40 points). Gains also extend to Qwen3-8B, including GSM8K (+3.02 points), MATH-500 (+2.58 points), and Olympiad (+2.47 points). Overall pass@8 also increases, from 58.70% to 59.75% on 1.7B and from 62.23% to 63.53% on 8B. These results show improvements in both average response correctness and aggregate coverage across eight attempts, supporting the effectiveness of diagnosis-informed feedback across the two model scales.

Training dynamics. Figure 2 illustrates the optimization dynamics associated with cross-prompt credit assignment. Across both model scales, DIA-GRPO exhibits larger gradient norms and more negative advantage-weighted training losses overall than GRPO, while the logged reference-policy KL remains comparable or lower. This pattern is consistent with the role of diagnostic feedback in reorganizing optimization signals: DIA-GRPO complements within-group comparisons with feedback informed by the model's current capabilities and a shared improvement direction, adjusting how individual responses contribute to the joint update. Since each response-level advantage weights a policy gradient, changing these coefficients alters the relative contributions of gradients across prompts and how they reinforce or offset one another. The objective-consistency analysis in Section 4.3 further establishes that, under the stated ideal local conditions, the expected unclipped policy gradient of the diagnostic component matches the gradient of average log success over the training

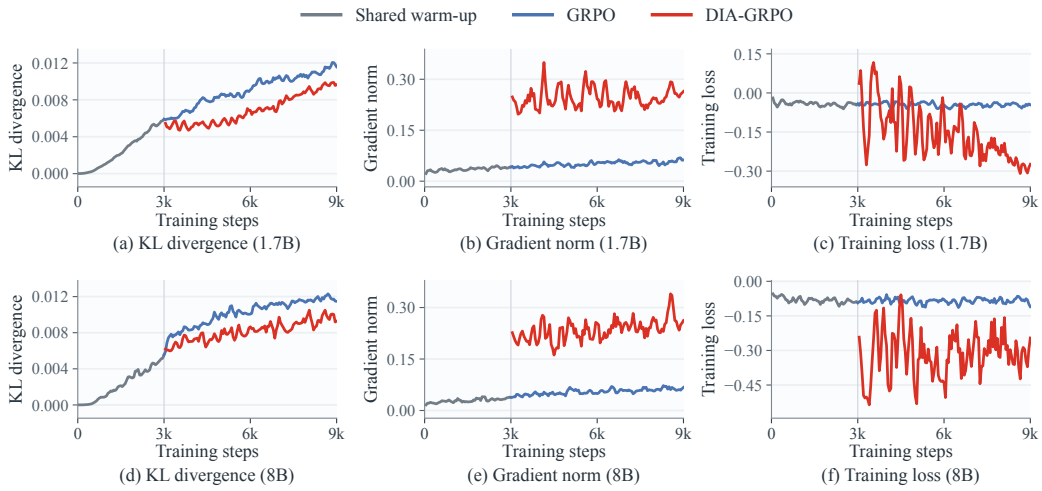


Figure 2: **Training dynamics across model scales.** KL divergence, gradient norm, and training loss for Qwen3-1.7B and Qwen3-8B during 3,000 shared warm-up steps and the 6,000 subsequent training steps. Curves show trailing averages over up to 150 steps, evaluated every 50 steps, with logged zero values retained. GRPO continues the warm-up smoothing window, while DIA-GRPO is smoothed independently after warm-up.

distribution. These trajectories provide a process-level view of the optimization changes associated with diagnosis. The central mechanism is to use a shared capability representation to organize local feedback into joint optimization signals that serve an overall improvement objective.

Efficiency. DIA-GRPO improves reasoning performance while maintaining online training costs comparable to GRPO. The improvements hold under both matched generation-token and matched training-step budgets. At 6,000 training steps, DIA-GRPO requires approximately 1.6% more training time; under the 39M-token budget, including diagnostic tracking responses, it takes 72.41 hours compared with 73.67 hours for GRPO. These complementary comparisons show that the benefits of cross-prompt diagnosis persist under either budget constraint, with only minor differences in online training time.

5.3 Component Ablations and Mixing Strategies

Diagnostic component ablations. We examine three controls to identify the contributions of diagnostic feedback. Matched scalar retains the prediction residual but replaces sample-specific directional weights with a uniform positive coefficient. Direction shuffle disrupts the correspondence between samples and diagnostic directions, while frozen diagnosis keeps the initial diagnosis fixed during subsequent training. Figure 3(a) shows that full DIA-GRPO achieves the highest avg@8 and pass@8 among these conditions. Shuffling directions produces the largest avg@8 decrease (6.81 percentage points), followed by scalar weighting (3.83 points) and frozen diagnosis (2.11 points). These results highlight the importance of connecting sample-specific feedback to the shared improvement direction: a prediction residual alone does not provide the same benefit. Updating the diagnosis during training further improves performance, supporting the use of current capability estimates to guide feedback.

Mixing coefficient and adaptive weighting. Figure 3(b) compares fixed mixing coefficients $\alpha \in \{0.25, 0.50, 0.75, 1.00\}$ with an adaptive rule. All tested fixed coefficients improve avg@8 over GRPO, with gains ranging from 1.79 to 3.29 percentage points, indicating that the benefit of diagnostic feedback extends across a broad mixing range. Among fixed choices, $\alpha = 0.75$ achieves the highest avg@8, whereas $\alpha = 0.50$ achieves the highest pass@8. Notably, our simple adaptive rule achieves the best performance on both metrics, improving avg@8 and pass@8 over fixed $\alpha = 0.50$ by 1.42 and 1.07 percentage points, respectively. While our focus is the diagnostic feedback mecha-

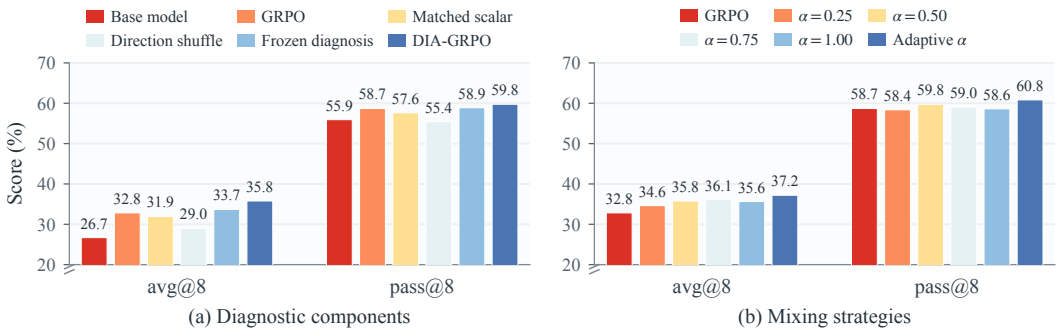


Figure 3: Average accuracy (avg@8) and pass rate (pass@8) under component ablations (a) and different mixing strategies (b) on Qwen3-1.7B. DIA-GRPO outperforms the component controls, while adaptive mixing achieves the highest scores on both metrics.

Table 2: **Performance and training time across policy optimization methods.** We report avg@8 and pass@8 (%) on 3,789 problems from five mathematical reasoning benchmarks after 6,000 updates of Qwen3-1.7B. +DIA denotes the corresponding optimizer augmented with diagnostic feedback. Bold marks the better score within each pair.

Algorithm	avg@8		pass@8		Training time (h)	
	Original	+DIA	Original	+DIA	Original	+DIA
DAPO	34.85	**37.13**	56.36	**57.50**	138.18	140.81
Dr.GRPO	35.43	**36.49**	55.91	**56.69**	232.03	233.22
GMPO	37.20	**39.95**	54.17	**58.56**	274.17	275.97

nism, this result points to an intriguing direction for further investigation: how to adaptively balance within-group comparisons and cross-prompt diagnostic feedback to further improve training.

5.4 Transfer across Policy Optimization Methods

We integrate diagnostic feedback into DAPO (Yu et al., 2025), Dr. GRPO (Liu et al., 2025), and GMPO (Zhao et al., 2026) to evaluate its applicability beyond native GRPO. As shown in Table 2, adding DIA improves avg@8 by 2.28, 1.06, and 2.75 percentage points, respectively, while pass@8 increases by 1.14, 0.78, and 4.39 points. GMPO achieves the largest gains in both metrics, showing that diagnostic feedback can further improve an already strong baseline. Measured training time increases by only 0.51%–1.90% across the three optimizers. Together, these results support cross-prompt credit assignment as a reusable feedback mechanism that complements different policy optimization methods with a small increase in total training time.

6 Conclusion

We introduced DIA-GRPO, a lightweight framework that organizes local feedback across training samples into policy updates guided by an overall learning objective. By jointly modeling sample attributes and evolving LLM capabilities, DIA-GRPO uses cross-prompt observations to interpret each outcome relative to expected performance and construct diagnostic advantages that complement within-group comparisons. Our local theoretical analysis establishes a policy-gradient correspondence between diagnostic feedback and the average log-success objective. Experiments on mathematical reasoning demonstrate improvements across model scales, while ablations highlight the importance of sample-specific directional weighting and continued capability tracking. Results with DAPO, Dr.GRPO, and GMPO further show that diagnostic feedback can benefit different policy optimizers with small increases in training time. More broadly, this work shows how a diagnostic model of LLM capabilities can become part of the learning rule, connecting evidence about the current model to how feedback across samples is combined to guide its improvement.

AI USE STATEMENT

In this work, we used generative AI tools to help develop theoretical models or conceptual frameworks, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, design or provide feedback on research methodology or experiments, and assist with translation. We have not used generative AI tools to formulate mathematical claims, propose or refine hypotheses, implement methods, support qualitative and thematic data analysis, or interpret results, and generating synthetic data sets and cleaning and reformatting datasets are not applicable to this work. Additionally, we used generative AI tools to create or edit software code, draft parts of the research paper, and source or search for information. We have reviewed all AI-assisted work. In particular, AI-assisted mathematical arguments, proof steps, software code, translations, and drafted text were reviewed by the authors before inclusion in the paper. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The main text presents the DIA-GRPO framework, while Appendix A provides the assumptions and complete proofs of our theoretical results. Appendix B details diagnostic calibration, online capability estimation, and the policy training procedure. Appendix C specifies the experimental datasets, evaluation protocols, training hyperparameters, and ablation configurations, including the adaptive mixing rule. Anonymized source code and configurations for the experiments are included in the supplementary material.

REFERENCES

Trapit Bansal, Jakub Pachocki, Szymon Sidor, Ilya Sutskever, and Igor Mordatch. Emergent complexity via multi-agent competition. In *International Conference on Learning Representations*, 2018. URL https://arxiv.org/abs/1710.03748.

Guanxu Chen, Yafu Li, Yuxian Jiang, Chen Qian, Qihan Ren, Jingyi Yang, Yu Cheng, Dongrui Liu, and Jing Shao. Conditional advantage estimation for reinforcement learning in large reasoning models. In *International Conference on Learning Representations (ICLR 2026)*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/097269005c77713aa32bb45b393ccd49-Abstract-Conference.html.

Jianhao Chen, Chenxu Wang, Gengrui Zhang, Peng Ye, Lei Bai, Wei Hu, Yuzhong Qu, and Shuyue Hu. Learning compact representations of LLM abilities via item response theory. *arXiv preprint arXiv:2510.00844*, 2025. URL https://arxiv.org/abs/2510.00844.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. URL https://arxiv.org/abs/2107.03374.

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. *arXiv preprint arXiv:2110.14168*, 2021. URL https://arxiv.org/abs/2110.14168.

Yanqi Dai, Yuxiang Ji, Xiao Zhang, Yong Wang, Xiangxiang Chu, and Zhiwu Lu. Harder is better: Boosting mathematical reasoning via difficulty-aware GRPO and multi-aspect question reformulation. In *International Conference on Learning Representations (ICLR 2026)*, 2026. URL

https://proceedings.iclr.cc/paper_files/paper/2026/hash/1cac8326ce3fbe79171db9754211530c-Abstract-Conference.html.

Carlos Florensa, David Held, Xinyang Geng, and Pieter Abbeel. Automatic goal generation for reinforcement learning agents. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1515–1528. PMLR, 2018. URL https://proceedings.mlr.press/v80/florensa18a.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, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025. doi: 10.1038/s41586-025-09422-z. URL https://doi.org/10.1038/s41586-025-09422-z.

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. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.211. URL https://aclanthology.org/2024.acl-long.211/.

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 *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/be83ab3ecd0db773eb2dc1b0a17836a1-Abstract-round2.html.

Valentin Hofmann, David Heineman, Ian Magnusson, Kyle Lo, Jesse Dodge, Maarten Sap, Pang Wei Koh, Chun Wang, Hannaneh Hajishirzi, and Noah A. Smith. Fluid language model benchmarking. In *Conference on Language Modeling (COLM 2025)*, 2025. URL https://openreview.net/forum?id=mxcCg9YRqj.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations (ICLR 2020)*, 2020. URL https://openreview.net/forum?id=rygGQyrFvH.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR 2022)*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Amirhossein Kazemnejad, Milad Aghajohari, Eva Portelance, Alessandro Sordoni, Siva Reddy, Aaron Courville, and Nicolas Le Roux. VinePPO: Refining credit assignment in RL training of LLMs. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 29557–29590. PMLR, 2025. URL https://proceedings.mlr.press/v267/kazemnejad25a.html.

Thanh-Long V. Le, Myeongho Jeon, Kim Vu, Viet Lai, and Eunho Yang. No prompt left behind: Exploiting zero-variance prompts in LLM reinforcement learning via entropy-guided advantage shaping. In *International Conference on Learning Representations (ICLR 2026)*, 2026. URL https://openreview.net/forum?id=kiXFIESZKv.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems*, volume 35, 2022. doi: 10.52202/068431-0278. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/18abbeef8cfe9203fdf9053c9c4fe191-Abstract-Conference.html.

Ziheng Li, Zexu Sun, Jinman Zhao, Erxue Min, Yongcheng Zeng, Hui Wu, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, Xu Chen, and Zhi-Hong Deng. Staying in the sweet spot: Responsive reasoning evolution via capability-adaptive hint scaffolding. *arXiv preprint arXiv:2509.06923*, 2025. URL https://arxiv.org/abs/2509.06923.

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 *International Conference on Learning Representations (ICLR 2024)*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/aca97732e30bcf1303bc22ac3924fd16-Abstract-Conference.html.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-Zero-Like training: A critical perspective. In *Conference on Language Modeling (COLM 2025)*, 2025. URL https://openreview.net/forum?id=5PAF7PAY2Y.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinyBenchmarks: evaluating LLMs with fewer examples. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 34303–34326. PMLR, 2024. URL https://proceedings.mlr.press/v235/maia-polo24a.html.

OpenAI. Reinforcement fine-tuning use cases. OpenAI API documentation, n.d. URL https://developers.openai.com/api/docs/guides/rft-use-cases. Accessed 24 September 2026.

Mark D. Reckase. *Multidimensional Item Response Theory*. Statistics for Social and Behavioral Sciences. Springer, 2009. doi: 10.1007/978-0-387-89976-3. URL https://link.springer.com/book/10.1007/978-0-387-89976-3.

Yangjun Ruan, Chris J. Maddison, and Tatsunori Hashimoto. Observational scaling laws and the predictability of language model performance. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-0506. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/1cded4f97cf5f01a284c574110b7e3b9-Abstract-Conference.html.

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

Amrith Setlur, Chirag Nagpal, Adam Fisch, Xinyang Geng, Jacob Eisenstein, Rishabh Agarwal, Alekh Agarwal, Jonathan Berant, and Aviral Kumar. Rewarding progress: Scaling automated process verifiers for LLM reasoning. In *International Conference on Learning Representations (ICLR 2025)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/98711dea460bdefe0e651ca23ec98ba2-Abstract-Conference.html.

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.

Zhongwen Xu and Zihan Ding. Single-stream policy optimization. In *International Conference on Learning Representations (ICLR 2026)*, 2026. URL https://openreview.net/forum?id=b61UW62K7W.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-3775. URL https://papers.nips.cc/paper_files/paper/2025/hash/a4277440d50f1f15d2cb4c14f7e0c0d2-Abstract-Conference.html.

Yuzhong Zhao, Yue Liu, Junpeng Liu, Jingye Chen, Xun Wu, Yaru Hao, Tengchao Lv, Shaohan Huang, Lei Cui, Qixiang Ye, Fang Wan, and Furu Wei. Geometric-mean policy optimization. In *International Conference on Learning Representations (ICLR 2026)*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/44a1f18afd6d5cc34d7e5c3d8a80f63b-Abstract-Conference.html.

A Theoretical Derivations

We derive the diagnostic advantage and its connection to the policy objective, then analyze regularization, model mismatch, and additional properties of the construction.

A.1 Notation, Assumptions, and Scope

Let $\mathcal{D}$ be a fixed distribution over a finite training pool. Each training input q_i has fixed diagnostic parameters $a_i \in \mathbb{R}^d$ and $b_i \in \mathbb{R}$. The diagnostic model is

$$p_i(\vartheta) = \sigma(a_i^\top \vartheta - b_i), \qquad P_\vartheta(r \mid i) = p_i(\vartheta)^r [1 - p_i(\vartheta)]^{1-r}. \tag{12}$$

Here $r \in \{0, 1\}$ denotes verifier correctness. Finite diagnostic parameters ensure $0 < p_i(\vartheta) < 1$. At iteration t, write $p_{it} = p_i(\hat{\vartheta}_t)$ and $h_{it} = p_{it}(1 - p_{it})$.

We use $\mathbb{E}_{\mathrm{diag},t}$ for expectations under $i \sim \mathcal{D}$ and $r \sim P_{\hat{\vartheta}_t}(\cdot \mid i)$. Policy expectations instead use $i \sim \mathcal{D}$ and $o \sim \pi_{\phi_t}(\cdot \mid q_i)$. The policy's true success probability is

$$\mu_i(\phi) = \mathbb{E}_{o \sim \pi_\phi(\cdot \mid q_i)}[r_i(o)]. \tag{13}$$

The verifier $r_i(o)$ has no explicit dependence on ϕ. For the policy-gradient identities, we assume differentiability, locally fixed policy support, finite relevant expectations, and conditions permitting differentiation under the expectation.

Diagnostic quantities are held fixed when differentiating the policy loss. If they are estimated from auxiliary responses, we condition on the auxiliary information and assume it is independent of the policy-loss responses given their inputs and the current policy. The local-model and exact-estimation assumptions for objective consistency are stated in Appendix A.3. The analysis uses complete responses sampled from the current policy and concerns the diagnostic component before mixing and clipping. Training implementation details are given in Appendix B.

A.2 Deriving Diagnostic Advantages

We derive diagnostic advantages by matching the expected weighted likelihood score to the gradient of the overall learning objective.

Likelihood score. For $z_i = a_i^\top \vartheta - b_i$, the Bernoulli log-likelihood is

$$\log P_\vartheta(r \mid i) = r z_i - \log(1 + \exp z_i). \tag{14}$$

Differentiating with respect to ϑ and evaluating at $\hat{\vartheta}_t$ gives

$$e_i(r) = \nabla_\vartheta \log P_\vartheta(r \mid i) = (r - p_{it}) a_i. \tag{15}$$

The residual $r - p_{it}$ measures the discrepancy between the observed outcome and its prediction; a_i maps this discrepancy to the capability coordinates.

Linear diagnostic feedback. We combine the score coordinates using a coefficient vector $m_t \in \mathbb{R}^d$ shared across inputs:

$$A_i^{\mathrm{DIA}}(r) = e_i(r)^\top m_t = (r - p_{it})\omega_{it}, \qquad \omega_{it} = a_i^\top m_t. \tag{16}$$

For a sampled response, $A_{ig}^{\mathrm{DIA}} = A_i^{\mathrm{DIA}}(r_{ig})$. Holding m_t fixed, the chain rule gives

$$\frac{\mathrm{d}}{\mathrm{d}\epsilon} \log P_{\hat{\vartheta}_t + \epsilon m_t}(r \mid i)\Big|_{\epsilon=0} = e_i(r)^\top m_t = A_i^{\mathrm{DIA}}(r). \tag{17}$$

The diagnostic advantage therefore measures the local change in the observed outcome's log-likelihood along m_t.

Gradient matching. We use the average log-success objective

$$U(\vartheta) = \mathbb{E}_{i \sim \mathcal{D}}[\log p_i(\vartheta)]. \tag{18}$$

Since $\mathrm{d} \log p_i = \mathrm{d}p_i / p_i$, this objective measures relative changes in predicted success. Using $\nabla_\vartheta p_i = p_i(1 - p_i)a_i$, its gradient at the current estimate is

$$v_t := \nabla_\vartheta U(\hat{\vartheta}_t) = \mathbb{E}_{i \sim \mathcal{D}}[(1 - p_{it})a_i]. \tag{19}$$

Define the weighted log-likelihood

$$\mathcal{S}_t(\vartheta; m_t) := \mathbb{E}_{\mathrm{diag},t}\big[A_i^{\mathrm{DIA}}(r) \log P_\vartheta(r \mid i)\big]. \tag{20}$$

The sampling distribution, p_{it}, m_t, and hence $A_i^{\mathrm{DIA}}(r)$ are held fixed when differentiating with respect to ϑ. Its gradient at the current capability estimate is

$$\nabla_\vartheta \mathcal{S}_t(\hat{\vartheta}_t; m_t) = \mathbb{E}_{\mathrm{diag},t}[A_i^{\mathrm{DIA}}(r) e_i(r)]. \tag{21}$$

For the unregularized construction, we choose m_t so that $\mathcal{S}_t$ and U have the same gradient at $\hat{\vartheta}_t$:

$$\mathbb{E}_{\mathrm{diag},t}[A_i^{\mathrm{DIA}}(r) e_i(r)] = v_t. \tag{22}$$

Solving for the weights. Under $r \mid i \sim \text{Bernoulli}(p_{it})$, $\mathbb{E}[r - p_{it} \mid i] = 0$ and $\mathbb{E}[(r - p_{it})^2 \mid i] = h_{it}$. The score therefore has conditional mean zero and conditional second moment $h_{it} a_i a_i^\top$. Averaging over inputs gives the Fisher information matrix

$$F_t := \mathbb{E}_{\text{diag},t}[e_i(r) e_i(r)^\top] = \mathbb{E}_{i \sim \mathcal{D}}[h_{it} a_i a_i^\top]. \tag{23}$$

Substituting the linear feedback rule into the weighted gradient gives

$$\mathbb{E}_{\text{diag},t}[A_i^{\text{DIA}}(r) e_i(r)] = \mathbb{E}_{\text{diag},t}[e_i(r) e_i(r)^\top] m_t = F_t m_t. \tag{24}$$

The matching condition thus reduces to $F_t m_t = v_t$. When F_t is nonsingular, its unique solution is $m_t = F_t^{-1} v_t$.

For numerical stability, we solve the regularized system $(F_t + \lambda I) m_t = v_t$, with $\lambda \geq 0$. The matrix F_t is positive semidefinite because $x^\top F_t x = \mathbb{E}_i[h_{it}(a_i^\top x)^2] \geq 0$ for every $x \in \mathbb{R}^d$. Thus $F_t + \lambda I$ is positive definite for $\lambda > 0$; when $\lambda = 0$, we assume F_t is nonsingular. The solution is

$$m_t = (F_t + \lambda I)^{-1} v_t. \tag{25}$$

Using $\omega_{it} = a_i^\top m_t$ gives

$$\omega_{it} = a_i^\top (F_t + \lambda I)^{-1} \nabla_\vartheta U(\hat{\vartheta}_t). \tag{26}$$

The regularized solution has matching residual

$$\mathbb{E}_{\text{diag},t}[A_i^{\text{DIA}}(r) e_i(r)] - v_t = F_t m_t - v_t = -\lambda m_t. \tag{27}$$

Hence the expected weighted gradient equals v_t when $\lambda = 0$ and F_t is nonsingular.

Ascent direction and feedback signs. The vector m_t is also a local ascent direction for U. If $v_t \neq 0$, positive definiteness gives

$$\frac{\mathrm{d}}{\mathrm{d}\epsilon} U(\hat{\vartheta}_t + \epsilon m_t)\Big|_{\epsilon=0} = v_t^\top m_t = v_t^\top (F_t + \lambda I)^{-1} v_t > 0. \tag{28}$$

Thus sufficiently small positive steps along m_t increase U. If $v_t = 0$, then $m_t = 0$ and all diagnostic advantages vanish.

For an individual sample,

$$\frac{\mathrm{d}}{\mathrm{d}\epsilon} p_i(\hat{\vartheta}_t + \epsilon m_t)\Big|_{\epsilon=0} = h_{it} a_i^\top m_t = h_{it} \omega_{it}. \tag{29}$$

Since $h_{it} > 0$, the sign of ω_{it} determines whether the predicted success probability increases or decreases along m_t. For $\omega_{it} > 0$, success receives advantage $(1 - p_{it})\omega_{it} > 0$, while failure receives $-p_{it}\omega_{it} < 0$. Negative weights reverse these signs, and zero weights give zero diagnostic feedback. The signs indicate how an observed outcome supports the selected direction; the verifier's correctness label remains unchanged. The overall objective can increase even when some samples' predicted success probabilities decrease.

Appendix A.3 establishes the connection between this diagnostic construction and the expected policy gradient.

A.3 Consistency with the Policy-Level Objective

Let $\psi_i(o) = \nabla_\phi \log \pi_\phi(o \mid q_i)$, evaluated at ϕ_t. Define the expected diagnostic policy gradient as

$$g_t^{\text{DIA}} = \mathbb{E}_{i \sim \mathcal{D},\, o \sim \pi_{\phi_t}}\big[(r_i(o) - p_{it}) \omega_{it} \psi_i(o)\big]. \tag{30}$$

Proposition 1 (Objective consistency). Suppose there exists a differentiable map $\vartheta(\phi)$ such that, in a neighborhood of ϕ_t, $\mu_i(\phi) = p_i(\vartheta(\phi))$ for every input in the support of $\mathcal{D}$. Suppose further that $\hat{\vartheta}_t = \vartheta(\phi_t)$, that F_t and v_t are population moments, and that $\lambda = 0$ with F_t positive definite. Then

$$g_t^{\text{DIA}} = \nabla_\phi \mathbb{E}_{i \sim \mathcal{D}} \log \mu_i(\phi)\big|_{\phi=\phi_t}. \tag{31}$$

Proof. Policy normalization gives $\mathbb{E}_o[\psi_i(o)] = 0$. Since the verifier has no explicit parameter dependence, $\mathbb{E}_o[r_i(o)\psi_i(o)] = \nabla_\phi \mu_i(\phi_t)$. The fixed diagnostic baseline therefore cancels:

$$\mathbb{E}_o[(r_i(o) - p_{it})\omega_{it}\psi_i(o)] = \omega_{it}\nabla_\phi \mu_i(\phi_t). \tag{32}$$

Let $J_t = \partial\vartheta/\partial\phi|_{\phi_t} \in \mathbb{R}^{d\times P}$, where P is the number of policy parameters. The local-model assumption and the chain rule imply

$$\nabla_\phi \mu_i(\phi_t) = J_t^\top h_{it} a_i. \tag{33}$$

Substituting $\omega_{it} = a_i^\top m_t$ and averaging over inputs gives

$$g_t^{\mathrm{DIA}} = J_t^\top \mathbb{E}_i[h_{it} a_i a_i^\top] m_t = J_t^\top F_t m_t = J_t^\top v_t. \tag{34}$$

The same local-model assumption gives $\mathbb{E}_i \log \mu_i(\phi) = U(\vartheta(\phi))$ in a neighborhood of ϕ_t. Its gradient at ϕ_t is $J_t^\top v_t$ by the chain rule, proving Eq. (31). □

The local-model assumption ensures agreement of both success probabilities and their policy derivatives. Proposition 1 identifies the objective represented by the diagnostic component within standard policy-gradient optimization. The mixed advantage and clipped token-level loss are specified in Appendix B.

A.4 Regularization and Diagnostic Approximation

To analyze model mismatch, write

$$\mu_i(\phi) = p_i(\vartheta(\phi)) + \delta_i(\phi), \tag{35}$$

where δ_i is differentiable in a neighborhood of ϕ_t. We continue to assume $\hat{\vartheta}_t = \vartheta(\phi_t)$ and use population moments for F_t and v_t. Define the derivative-mismatch matrix

$$R_t = \mathbb{E}_{i\sim\mathcal{D}}[\nabla_\phi \delta_i(\phi_t) a_i^\top]. \tag{36}$$

Applying the chain rule to Eq. (35) and using Eq. (32) gives $g_t^{\mathrm{DIA}} = (J_t^\top F_t + R_t)m_t$. Since $F_t m_t = v_t - \lambda m_t$,

$$g_t^{\mathrm{DIA}} = J_t^\top v_t - \lambda J_t^\top m_t + R_t m_t. \tag{37}$$

The terms $-\lambda J_t^\top m_t$ and $R_t m_t$ capture the effects of ridge regularization and derivative mismatch, respectively. Both vanish under the assumptions of Proposition 1.

Let $\gamma_t = \lambda_{\min}(F_t)$ and assume $\gamma_t + \lambda > 0$. Since $\|m_t\|_2 \le \|v_t\|_2/(\gamma_t + \lambda)$, the triangle inequality and the induced matrix norm give

$$\|g_t^{\mathrm{DIA}} - J_t^\top v_t\|_2 \le \frac{\lambda\|J_t\|_2 + \|R_t\|_2}{\gamma_t + \lambda}\|v_t\|_2. \tag{38}$$

Here $J_t^\top v_t$ is the gradient of the diagnostic objective $U(\vartheta(\phi))$ at ϕ_t.

Relation to the true log-success objective. Assume $\mu_i(\phi_t) > 0$ for every input in the support of $\mathcal{D}$, and define $J_{\log}(\phi) = \mathbb{E}_i \log \mu_i(\phi)$. Write $\mu_{it} = \mu_i(\phi_t)$ and $\delta_{it} = \delta_i(\phi_t)$. Dividing the derivative of Eq. (35) by μ_{it} and subtracting $J_t^\top v_t$ gives

$$\xi_t := \nabla_\phi J_{\log}(\phi_t) - J_t^\top v_t = \mathbb{E}_i\left[\frac{\nabla_\phi \delta_i(\phi_t)}{\mu_{it}} - \frac{\delta_{it}(1 - p_{it})}{\mu_{it}} J_t^\top a_i\right]. \tag{39}$$

Combining this identity with Eq. (38) yields

$$\|g_t^{\mathrm{DIA}} - \nabla_\phi J_{\log}(\phi_t)\|_2 \le \frac{\lambda\|J_t\|_2 + \|R_t\|_2}{\gamma_t + \lambda}\|v_t\|_2 + \|\xi_t\|_2. \tag{40}$$

The bound depends on errors in both success probabilities and their derivatives. Finite-sample tracking, cached states, and empirical moment estimates introduce additional estimation errors beyond this population analysis.

A.5 Additional Properties

Weighted projection. Because $v_t = \mathbb{E}_i[h_{it} a_i / p_{it}]$, the direction m_t is the unique minimizer of

$$\min_{m \in \mathbb{R}^d} \left\{ \frac{1}{2} \mathbb{E}_i \left[h_{it} \left(a_i^\top m - \frac{1}{p_{it}} \right)^2 \right] + \frac{\lambda}{2} \|m\|_2^2 \right\}. \tag{41}$$

The gradient is $(F_t + \lambda I)m - v_t$, and the Hessian is positive definite under the conditions stated in Appendix A.2. Expanding the square gives the equivalent maximization of $v_t^\top m - \frac{1}{2} m^\top (F_t + \lambda I) m$. Equation (41) fits the reciprocal success probabilities $1/p_{it}$ with linear weights $a_i^\top m$, using h_{it} as the fitting weights and λ for regularization.

Minimum second moment under the diagnostic model. For $\lambda = 0$ and positive definite F_t, consider scalar weights satisfying $\mathbb{E}_i[h_{it} a_i \omega_i] = v_t$. The choice $\omega_i^* = a_i^\top F_t^{-1} v_t$ minimizes $\mathbb{E}_i[h_{it} \omega_i^2]$ over this class.

To prove this, let $d_i = \omega_i - \omega_i^*$. The constraint gives $\mathbb{E}_i[h_{it} a_i d_i] = 0$, so the cross term $\mathbb{E}_i[h_{it} \omega_i^* d_i]$ is zero. Hence

$$\mathbb{E}_i[h_{it} \omega_i^2] = \mathbb{E}_i[h_{it} (\omega_i^*)^2] + \mathbb{E}_i[h_{it} d_i^2] \geq \mathbb{E}_i[h_{it} (\omega_i^*)^2]. \tag{42}$$

Equality holds only when $d_i = 0$ on the support of $\mathcal{D}$. Since $\mathbb{E}_{\mathrm{diag},t}[(r - p_{it})^2 \omega_i^2] = \mathbb{E}_i[h_{it} \omega_i^2]$, this establishes the minimum-second-moment property of the scalar diagnostic advantage. The second moment of the full policy-gradient estimator also depends on $\psi_i(o)$.

Orthogonal coordinate invariance. For an orthogonal matrix Q, let $\vartheta' = Q\vartheta$ and $a_i' = Q a_i$, with b_i unchanged. These transformations preserve the predicted probabilities and give $F_t' = Q F_t Q^\top$ and $v_t' = Q v_t$. Since $F_t' + \lambda I = Q(F_t + \lambda I) Q^\top$, we have $m_t' = Q m_t$ and therefore

$$a_i'^\top m_t' = a_i^\top m_t. \tag{43}$$

The diagnostic advantages are thus invariant under orthogonal changes of coordinates. With a fixed isotropic ridge, arbitrary rescaling of the latent coordinates need not preserve the advantages.

B Method and Implementation Details

This appendix describes diagnostic calibration, online capability estimation, and the integration of diagnostic feedback into policy optimization. Algorithm 1 summarizes the training procedure.

B.1 Diagnostic Calibration

During standard GRPO warm-up, DIA-GRPO collects responses and verification outcomes from different policy states on training samples. The calibration cohort $\mathcal{U}$ comprises policy states and decoding configurations available before the subsequent training branch. Each calibration member is treated as a respondent. For respondent u and sample i, we retain the response count n_{ui} and verified success count s_{ui} so that fitting accounts for differences in the number of observations.

Let $\mathcal{O}$ denote the observed respondent–sample pairs and $N_{\mathrm{cal}} = \sum_{(u,i) \in \mathcal{O}} n_{ui}$. We jointly estimate sample attributes and respondent capabilities by minimizing the negative log-likelihood

$$\mathcal{L}_{\mathrm{cal}} = -\frac{1}{N_{\mathrm{cal}}} \sum_{(u,i) \in \mathcal{O}} \left[s_{ui} \log p_{ui} + (n_{ui} - s_{ui}) \log(1 - p_{ui}) \right], \tag{44}$$

where $p_{ui} = \sigma(a_i^\top \vartheta_u - b_i)$. During fitting, we center the capability coordinates and adjust the sample difficulty parameters to preserve the predicted probabilities. After calibration, the sample parameters $\{a_i, b_i\}$ are fixed and serve as a shared reference for online capability estimation. Evaluation data and observations from the subsequent training branch are excluded from calibration.

Algorithm 1 DIA-GRPO with periodic capability diagnosis

Require: Training pool and distribution $\mathcal{D}$; group size G; refresh interval H; regularization coefficients $\lambda_\vartheta > 0$, $\lambda > 0$; mixing coefficient $\alpha \in (0, 1]$

Run standard GRPO warm-up and collect calibration responses and verification outcomes
Fit Eq. (44), freeze $\{a_i, b_i\}$, and initialize the capability state and tracking buffer
for each training iteration t **do**
    Sample training inputs and assign tracking and policy-update groups before generating responses
    Generate G responses per input and compute training rewards and diagnostic fitting targets
    Add tracking observations to the buffer
    if a refresh is due or no diagnostic cache is available **then**
        Update $\hat{\vartheta}_t$ from the buffer using Eq. (4)
        Update p_{it}, F_t, and v_t; solve $(F_t + \lambda I)m_t = v_t$
        Cache p_{it} and $\omega_{it} = a_i^\top m_t$, then clear the tracking buffer
    end if
    Construct diagnostic advantages for the policy-update groups, clip them, mix with native advantages, and clip the result
    Apply the configured response filters and update the policy using the optimizer's objective, with advantages held fixed
end for

Ensure: Optimized policy

B.2 Online Capability Estimation

At each iteration, we partition complete prompt groups into tracking and policy-update folds before observing their generated outcomes. Responses are sampled independently across groups. Tracking responses estimate the current capability state and are excluded from the policy loss in the same update. Responses in the policy-update fold use the resulting diagnostic quantities to construct their advantages.

The tracking buffer stores response counts and diagnostic fitting targets for each observed sample. Every H policy updates, we estimate the capability state using the regularized objective in Eq. (4). With sample parameters fixed, this is a low-dimensional regularized logistic regression problem, initialized from the previous capability estimate.

After fitting the new state, we update the predictions p_{it}, the Fisher matrix F_t, and the objective gradient v_t over the reference training pool. We then solve $(F_t + \lambda I)m_t = v_t$ and cache p_{it} and $\omega_{it} = a_i^\top m_t$. The tracking buffer is cleared after each refresh, and the cached quantities are reused until the next refresh. Tracking responses come from the existing rollout batch and require no additional response generation.

B.3 Policy Optimization and Training Procedure

For the training reward R_{ig}, the diagnostic component is

$$A_{ig}^{\mathrm{DIA,train}} = (R_{ig} - p_{it})\omega_{it}. \tag{45}$$

We first clip this component, combine it with the native advantage using the mixture in Eq. (5), and then clip the mixed result. Reward definitions, the conversion to diagnostic fitting targets, native advantage conventions, and response filters follow the experiment-specific configurations in Appendix C.

The resulting advantages enter the optimizer through its existing advantage interface. GRPO uses the policy loss in Eq. (9); other policy optimizers retain their own objectives and loss reductions. All diagnostic quantities and advantages are held fixed during policy backpropagation. The policy loss is computed only from retained responses in the policy-update fold.

When $\alpha = 0$, training follows the native procedure without reserved tracking groups or diagnostic estimation. Algorithm 1 gives the procedure with diagnostic feedback enabled.

Table 3: Composition of the main evaluation set.

Dataset	Number of inputs
GSM8K	1,319
MATH-500	500
Minerva	272
Olympiad	674
DAPO	1,024
Total	3,789

C Experimental Protocols

C.1 Evaluation Datasets and Metrics

We train Qwen3-1.7B-Base and Qwen3-8B-Base on 16,893 DAPO samples. The main evaluation comprises 3,789 inputs across five mathematical reasoning datasets, as summarized in Table 3.

For each input, we sample eight responses with temperature 0.7, top-$p = 0.95$, and a maximum completion length of 768 tokens. We report avg@8, the mean correctness across all sampled responses, and pass@8, the fraction of inputs with at least one correct response. These metrics measure per-response reliability and the coverage achieved through multiple attempts, respectively. Overall assigns equal weight to each evaluation input.

Evaluation uses answer correctness without the format rewards applied during training. Scores are reported as percentages; aggregation and performance differences are computed before rounding.

C.2 Training Configuration

We train the policies with LoRA. GRPO and DIA-GRPO use the same training data, shared warm-up policy, and base optimization configuration. DIA-GRPO additionally maintains a capability estimate and diagnostic weights. Table 4 summarizes the training and diagnostic settings.

The rollout batch size is aggregated across devices. In diagnostic variants, one input group is used for tracking and the other for policy optimization, leaving up to eight responses per batch for the policy gradient. The number of valid responses can be further reduced by truncation masks and group filtering.

The diagnostic model uses a ten-dimensional MIRT representation with positively parameterized discrimination vectors. Sample attributes remain fixed after calibration. The 8B experiments reuse the same sample attributes while estimating capabilities from the 8B policy's own responses. Online capability fitting runs on the CPU, initialized from the previous estimate; predictions and diagnostic weights are cached between refreshes.

Rewards and policy optimization. The training reward combines answer correctness with a format bonus. A response receives an additional 0.05 for an explicit answer marker (`\boxed`, "final answer", or "answer is") and 0.02 if its stripped text contains at least 24 characters. The native advantage centers rewards within each response group without group-standard-deviation scaling. Diagnostic fitting uses training rewards clipped to $[0, 1]$ as its targets.

The policy loss is averaged over valid tokens. We retain the base training procedure's truncation masks and zero-reward-variance group filter. All diagnostic variants use the same reward processing, advantage clipping, and loss computation. The base trainer performs one on-policy update per rollout batch, using the gradient of the policy surrogate at the sampling policy; it does not introduce a separate likelihood-ratio clipping parameter. When transferring DIA to other policy optimization methods, we retain each method's native optimization rules and introduce diagnostic feedback through its advantage interface.

Table 4: Training and diagnostic configuration.

Configuration	Value
Policy optimizer	AdamW
Learning rate and schedule	10^{-6}, constant
AdamW parameters	$\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$
Policy weight decay	0
Maximum gradient norm	1.0
LoRA rank / alpha / dropout	8 / 32 / 0
LoRA target modules	`q_proj`, `k_proj`, `v_proj`, `o_proj`
Training precision	FP16 with FP32 trainable LoRA parameters
Gradient checkpointing	Enabled
Responses per input, G	8
Responses per rollout batch	16, across two input groups
Gradient accumulation steps	1
Policy updates per rollout batch	1
Tracking allocation	One of the two input groups
Training temperature / top-p	0.7 / 0.95
Maximum prompt / completion length	2,048 / 768 tokens
KL coefficient, β	0.04
Shared GRPO warm-up	3,000 updates
Diagnostic dimension, d	10
Capability regularization, λ_ϑ	0.01
Fisher regularization, λ	0.01
Diagnostic refresh interval, H	50 policy updates
Online capability solver	L-BFGS, learning rate 1.0, at most 100 iterations
Default mixing coefficient, α	0.5
Diagnostic advantage clipping threshold	5
Mixed advantage clipping threshold	5

C.3 Component Ablations and Mixing Strategies

The ablations in Section 5.3 use Qwen3-1.7B with the same training data, initialization, training budget, and evaluation protocol as the main experiment. Component ablations fix $\alpha = 0.5$. Each diagnostic variant retains the full method's tracking and policy-gradient allocation while changing the diagnostic component under study.

Full DIA-GRPO obtains success predictions p_{it} from the current capability estimate and computes sample weights $\omega_{it} = a_i^\top m_t$ using the shared improvement direction. The following controls examine sample-specific directional weighting, the correspondence between samples and capability directions, and online capability tracking.

Matched scalar. This control retains online capability estimation and prediction residuals but replaces all directional weights with a common positive coefficient c_t. At each diagnostic refresh, we first compute the full method's reference weights at the control's current diagnostic state, then set

$$c_t = \sqrt{\frac{\mathbb{E}_{i\sim\mathcal{D}}\left[p_{it}(1-p_{it})\omega_{it}^2\right]}{\mathbb{E}_{i\sim\mathcal{D}}\left[p_{it}(1-p_{it})\right]}}. \tag{46}$$

Expectations are empirical averages over the training sample pool used for weight computation. This choice matches the expected second moment of the full method's raw diagnostic advantage under the diagnostic model. The control preserves prediction-residual feedback while removing sample-specific directional weighting.

Direction shuffle. This control preserves the current success predictions and each sample's discrimination norm while disrupting the correspondence between samples and capability directions. Using a fixed permutation ϖ over the sample pool, we replace the vectors used for weight computa-

tion with

$$\widetilde{a}_i = \|a_i\|_2 \frac{a_{\varpi(i)}}{\|a_{\varpi(i)}\|_2}. \tag{47}$$

We recompute the Fisher matrix, objective gradient, and shared direction using the original predictions p_{it} and the shuffled vectors. The resulting weights are uniformly rescaled to match the full method's expected raw diagnostic-advantage second moment at the same diagnostic state.

Capability estimation and success prediction continue to use the original diagnostic model. This intervention therefore preserves prediction residuals, discrimination magnitudes, and the overall feedback scale while testing the importance of the correct sample–direction correspondence.

Frozen diagnosis. After the initial diagnostic refresh, this control saves the capability estimate and its associated success predictions and directional weights. Subsequent training reuses these quantities without further diagnostic updates. The policy continues to train on newly generated responses, so diagnostic feedback still changes with observed rewards while its predictive reference and sample weights remain fixed. This control tests whether tracking the evolving policy improves training over reliance on the initial diagnosis.

Second-moment matching is applied to the matched-scalar and direction-shuffle controls before advantage clipping. The frozen-diagnosis control instead retains its initial predictions and weights.

Fixed mixing coefficients. The mixing experiments retain the complete diagnostic module and compare fixed coefficients $\alpha \in \{0.25, 0.50, 0.75, 1.00\}$ with adaptive mixing. Native GRPO provides the reference without diagnostic feedback. All other training and evaluation settings remain unchanged.

Adaptive precision. The adaptive variant adjusts the mixture according to the availability of within-group feedback and the precision of diagnostic information for the current policy batch. It assigns relatively more weight to GRPO when within-group comparisons are likely to be informative, and more weight to diagnostic feedback when tracking observations constrain the relevant capability directions.

Let $\mathcal{B}_t^{\mathrm{pg}}$ denote the effective policy-update inputs in the current batch, and write $h_{it} = p_{it}(1 - p_{it})$. We score the GRPO component by

$$I_t^{\mathrm{GRPO}} = \frac{1}{|\mathcal{B}_t^{\mathrm{pg}}|} \sum_{i \in \mathcal{B}_t^{\mathrm{pg}}} \frac{1 - p_{it}^G - (1 - p_{it})^G}{1 - 2^{1-G}}. \tag{48}$$

The numerator is the diagnostic model's probability that a group contains both successful and unsuccessful responses. The denominator normalizes its maximum to one. This score measures the predicted availability of nondegenerate within-group feedback.

For the diagnostic component, let $\mathcal{Q}_t$ denote the tracking inputs in the current batch and n_{it}^{trk} their response counts. We form the tracking information matrix

$$J_t^{\mathrm{trk}} = \lambda_\vartheta I + \sum_{i \in \mathcal{Q}_t} n_{it}^{\mathrm{trk}} h_{it} a_i a_i^\top. \tag{49}$$

The average Fisher matrix over the policy inputs is

$$\overline{F}_t = \frac{1}{|\mathcal{B}_t^{\mathrm{pg}}|} \sum_{i \in \mathcal{B}_t^{\mathrm{pg}}} h_{it} a_i a_i^\top. \tag{50}$$

These matrices give the precision score

$$C_t = \left(1 + \frac{1}{d} \operatorname{tr}\left(\overline{F}_t \left(J_t^{\mathrm{trk}}\right)^{-1}\right)\right)^{-1}. \tag{51}$$

Here, $\operatorname{tr}\left(\overline{F}_t \left(J_t^{\mathrm{trk}}\right)^{-1}\right)$ measures local uncertainty in capability directions relevant to the policy inputs, using the regularized tracking information. Lower uncertainty yields a larger C_t.

We also measure the diagnostic feedback strength in the current batch relative to that over the training distribution:

$$S_t = \frac{\frac{1}{|\mathcal{B}_t^{\mathrm{pg}}|} \sum_{i \in \mathcal{B}_t^{\mathrm{pg}}} h_{it} \omega_{it}^2}{\mathbb{E}_{i \sim \mathcal{D}} \left[h_{it} \omega_{it}^2 \right] + \lambda \|m_t\|_2^2 + \epsilon_\alpha}. \tag{52}$$

The quantity $h_{it}\omega_{it}^2$ is the expected squared raw diagnostic advantage under the diagnostic model. We define $I_t^{\mathrm{DIA}} = C_t S_t$ and set

$$\alpha_t = \frac{\left(I_t^{\mathrm{DIA}}\right)^\kappa}{\left(I_t^{\mathrm{DIA}}\right)^\kappa + \left(I_t^{\mathrm{GRPO}}\right)^\kappa + \epsilon_\alpha}. \tag{53}$$

We use $\kappa = 1$, $\epsilon_\alpha = 10^{-6}$, bounds $[0, 1]$, and no exponential moving average. A single α_t is computed for each effective policy-update batch and shared by all responses in that batch. The computation reuses cached diagnostic predictions and weights together with the current batch's tracking counts. Capability estimates and diagnostic weights retain the 50-update refresh schedule. Adaptive mixing changes neither sample selection nor the number of generated responses.

D Supplementary Analyses

We further examine the role of diagnostic structure in constructing update directions, the evolution of sample weights during training, and the robustness of performance gains to the generation-length limit.

The real-model analyses use archived Qwen3-1.7B runs initialized from a shared policy after 3,000 GRPO warm-up updates. These runs use LoRA rank 8 and a constant learning rate of 10^{-6}. The diagnostic dimension is 10, and both the capability-estimation and Fisher regularization coefficients are 0.01. Training steps reported below count updates after warm-up.

D.1 Sample Directions and Representation Rank

Success probabilities describe how the model currently performs on individual samples, while sample directions describe how that performance changes with its capabilities. DIA-GRPO uses both to construct diagnostic feedback. We examine the contribution of the directional information in a controlled setting: with success probabilities kept exact, we test how perturbing sample directions or restricting the representation affects improvement of the overall objective.

We construct 200 independent environments with a known response model,

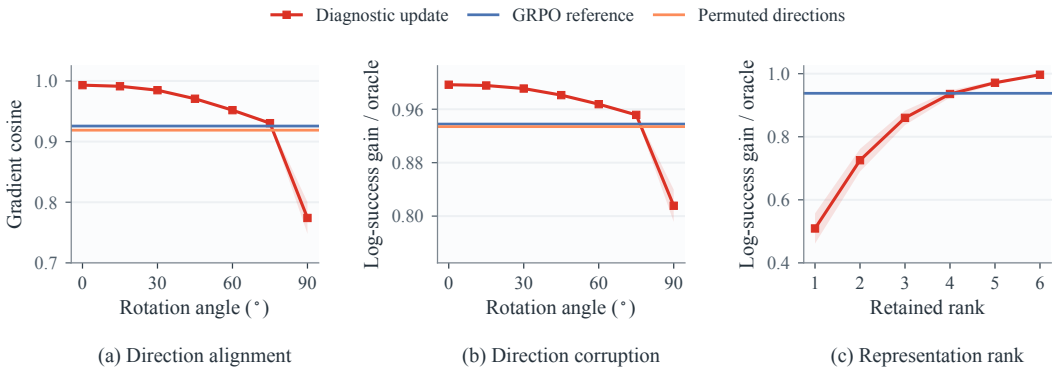
$$p_i(\phi) = \sigma(a_i^\top J\phi - b_i), \tag{54}$$

where $J\phi$ maps simulated policy parameters to a capability state. Each environment contains 512 samples, six capability dimensions, and 24 policy parameters.

All methods receive the true success probabilities. We perturb individual sample directions while preserving their norms, permute the correspondence between samples and directions, or restrict the diagnostic representation to nested subspaces of ranks one through six. Each method takes 25 updates with a fixed parameter-space step norm of 0.08, so differences in improvement reflect the update directions rather than their magnitudes. Diagnostic updates use ridge regularization of 10^{-3} and scalar feedback clipped to $[-5, 5]$.

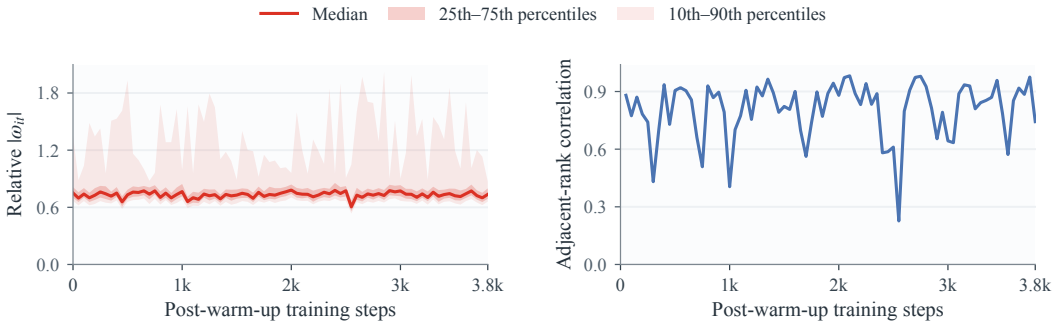
We measure the cosine similarity between the initial update and the true gradient of the average log-success objective, together with the objective improvement after optimization. The latter is normalized by the improvement obtained along the true objective-gradient trajectory. Figure 4 reports means and 95% confidence intervals across the 200 environments.

With correct sample directions, the gradient cosine is 0.993 and the normalized objective gain is 0.997. A 90° directional perturbation reduces these values to 0.774 and 0.816, respectively. Restricting the representation to rank one yields a normalized gain of 0.509, which progressively recovers as more dimensions are retained.



(a) Direction alignment (b) Direction corruption (c) Representation rank

Figure 4: **The role of diagnostic structure in update construction.** Panels show initial gradient alignment, normalized log-success gains under directional perturbations, and gains at different retained ranks. Success probabilities remain exact, and all methods use equal parameter-space step norms. Shaded regions indicate 95% confidence intervals across 200 independent environments.



(a) Weight distribution (b) Changes in sample ranking

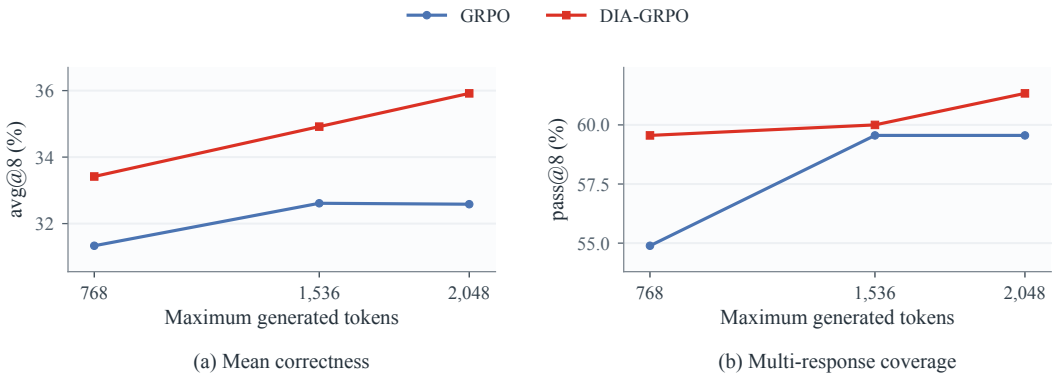
Figure 5: **Diagnostic weights adapt during training.** Left: the median, interquartile range, and 10th–90th percentile range of absolute weights normalized by their mean across 128 fixed training samples. Right: Spearman rank correlation between successive diagnostic refreshes. Bands describe variation across samples.

These results demonstrate the complementary roles of prediction and diagnostic structure in this controlled setting. Even when success probabilities are accurate, corrupting sample directions or discarding capability dimensions weakens the resulting improvement. The diagnostic representation therefore contributes information about how sample-level feedback should combine into an update toward the overall objective.

D.2 State-Dependent Diagnostic Weights

DIA-GRPO freezes sample attributes after calibration but continues to update the capability estimate and shared improvement direction. Consequently, the same sample can receive different diagnostic weights at different stages of training. We track a fixed sample bank to examine how this weighting evolves.

The analysis follows 128 training samples selected by source stratification independently of generated outcomes. It covers 77 diagnostic states from post-warm-up step 0 to step 3,800. At each state, we normalize absolute weights by their mean across the fixed bank and summarize the distribution using the median and quantiles. We also compute the Spearman correlation between absolute-weight rankings at successive diagnostic refreshes. Figure 5 shows all recorded states without smoothing.



(a) Mean correctness

(b) Multi-response coverage

Figure 6: **Performance under different generation limits.** avg@8 and pass@8 for fixed Qwen3-1.7B policies on the same 450-input probe. Each input receives eight responses at every limit. DIA-GRPO maintains higher mean correctness as the permitted generation length increases.

Table 5: **Paired performance gains across generation limits.** avg@8 scores are percentages, and gains are percentage-point differences between DIA-GRPO and GRPO. Confidence intervals are obtained by paired bootstrap resampling over evaluation inputs.

Token limit	GRPO	DIA-GRPO	Gain	95% CI
768	31.33	33.42	+2.09	[0.61, 3.58]
1,536	32.61	34.92	+2.31	[0.78, 3.86]
2,048	32.58	35.92	+3.33	[1.72, 4.94]

The central part of the weight distribution remains relatively compact, while its upper tail changes across diagnostic states. Adjacent rank correlations range from 0.226 to 0.982, showing that the relative weighting of the same samples changes throughout training.

With sample attributes fixed, these changes arise from the evolving capability estimate and its associated direction through $\omega_{it} = a_i^\top m_t$. Thus, fixed sample attributes provide a stable reference without imposing a fixed sample ranking. This analysis illustrates the dynamic weighting behavior; the frozen-diagnosis ablation in Section 5.3 evaluates the training benefit of updating the diagnosis.

D.3 Robustness to Generation Length

A restrictive generation limit can truncate unfinished reasoning. We therefore examine whether the performance advantage of DIA-GRPO persists when both policies receive a more generous generation allowance. We re-evaluate fixed GRPO and DIA-GRPO endpoints without additional policy training, changing only the maximum completion length.

Both endpoints come from the matched-budget comparison at approximately 39M generated tokens. This budget includes tracking responses and responses subsequently filtered or truncated; shared warm-up and offline calibration precede the branch budget. Evaluation uses the same 450-input probe with eight responses per input and completion limits of 768, 1,536, and 2,048 tokens. All other generation settings remain fixed. Figure 6 reports avg@8 and pass@8, and Table 5 summarizes the paired avg@8 differences.

DIA-GRPO improves avg@8 by 2.09, 2.31, and 3.33 percentage points at the three generation limits, with all paired confidence intervals above zero. At the 2,048-token limit, the fractions of responses reaching the limit are comparable: 6.25% for GRPO and 6.00% for DIA-GRPO. The performance advantage therefore persists under a more generous generation allowance.