SIRAS: Sibling-Relative Advantage Shaping for Reinforcement Learning from Verifiable Rewards

Zairun Yang[*]
Zhejiang University
zr.yang@zju.edu.cn

Jun Xu
Ant Group
xujun.xj@antgroup.com

Lei Liang
Ant Group
lianglei.ll@antgroup.com

Huajun Chen
Zhejiang University
huajunsir@zju.edu.cn

Qiang Zhang [†]
Zhejiang University
qiang.zhang.cs@zju.edu.cn

Abstract

Reinforcement learning from verifiable rewards (RLVR) trains reasoning language models with group-relative objectives such as GRPO. However, it leaves two sources of signal underused: every token in a rollout receives the same response-level advantage, and groups in which all rollouts succeed or all fail have zero group-relative advantage and are typically discarded. We observe that same-prompt rollouts are not independent reward labels but *sibling attempts*: they share scaffolding, diverge at uncertain decisions, and sometimes reconverge. In this paper, we propose **Si**bling-**R**elative **A**dvantage **S**haping (SIRAS), a lightweight advantage-shaping method that exploits this structure. SIRAS segments rollouts into chunks, aligns sibling chunks with banded dynamic time warping, and reshapes the flat GRPO advantage across tokens via soft-divergence weights that upweight chunks distinguishing a rollout from its siblings and downweight shared ones. This reshaping preserves each rollout's token-averaged GRPO advantage exactly and reduces to GRPO when weights are constant. The same sibling structure produces calibrated residuals for all-correct and all-wrong groups, reclaiming training signal that group-relative baselines discard. Across six reasoning benchmarks, SIRAS yields the largest gains on AIME-style competition math, including $+12.2$ on AIME25 with Qwen3-8B and $+23.3$ on AIME26 in our backbone-scaling study, without extra rollouts, process reward models, or step annotations.

1 Introduction

Reinforcement learning from verifiable rewards (RLVR) has become a practical recipe for improving the reasoning ability of language models. Given a prompt, the policy samples multiple rollouts, a programmatic verifier assigns each rollout a scalar outcome reward, and a group-relative policy-gradient objective such as GRPO [Shao et al., 2024] or DAPO [Yu et al., 2025] subtracts a within-group baseline. This recipe is attractive because it requires no value network, no human step annotation, and no learned process reward model. Its limitation is equally fundamental: a long reasoning trace receives only a single outcome reward.

[*]This work was done during an internship at Ant Group.
[†]Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

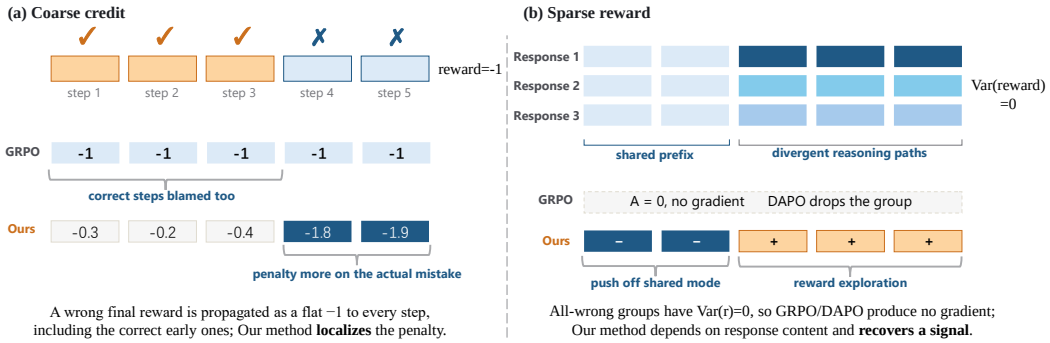

Figure 1: **Two failure modes that SIRAS aims to address.** *(a) Coarse credit.* GRPO assigns the same advantage to every token in a rollout. SIRAS reshapes this advantage across tokens by downweighting chunks shared by sibling rollouts and upweighting chunks that distinguish a rollout, while preserving the token-averaged trajectory advantage on non-degenerate groups. *(b) Degenerate groups.* When every rollout in a group receives the same reward, the group-relative baseline collapses. Because sibling structure is computed from rollout content rather than reward, SIRAS can still construct calibrated auxiliary signals for all-correct and all-wrong groups.

This creates two credit-assignment bottlenecks, illustrated in Figure 1. First, group-relative methods assign the same trajectory-level advantage to every token in a rollout. A failed solution may contain a useful derivation followed by a terminal mistake, while a correct solution may contain redundant detours. GRPO treats all tokens within the same rollout identically. Second, when all rollouts in a group receive the same reward, the group standard deviation is zero and the group-relative advantage vanishes. DAPO-style filtering therefore discards these *degenerate groups*. In our AIME-difficulty runs, degenerate groups account for approximately 57% of training groups on average, with the first logged step already near 78% (and reaching $\sim 87\%$ in the first few steps) before decreasing as the policy begins to solve some hard prompts. On easier MultiLogi-Eval prompts, the fraction instead grows from roughly 3% to 19% as the policy concentrates on easy items. In both regimes, a substantial fraction of training compute lands on groups that provide no gradient under the standard group-relative baseline.

Existing approaches address this sparsity from two directions. Process reward models (PRMs) provide dense step-level supervision by scoring intermediate reasoning steps [Lightman et al., 2024, Wang et al., 2024], but require an additional model, step-level labels or synthetic annotations, and inference-time scoring. Step-level RL methods such as VinePPO [Kazemnejad et al., 2024] estimate intermediate credit by sampling counterfactual continuations from prefixes, but this increases generation cost. We ask a complementary question: **can part of the missing process signal be extracted from the rollouts that RLVR already samples?**

Our observation is that the rollouts in a group are not merely independent reward labels. They are *sibling attempts at the same problem*: they often share scaffolding, diverge at uncertain decisions, and occasionally reconverge. The pattern of shared and divergent chunks provides a sibling-relative signal about where a rollout differs from its alternatives. This signal is weaker than the supervised process reward, and we do not claim that it identifies causal correctness. However, it is cheap, local to the group, and available in every RLVR batch. Therefore, we introduce SIRAS, a sibling-relative credit-shaping method for GRPO-style training. SIRAS first partitions rollouts into reasoning chunks using structural markers, log-probability surprise, and length caps. It then aligns chunks across same-prompt rollouts using banded dynamic time warping (DTW), computes a leave-one-out sibling-match score for each chunk, and converts this score into a continuous soft-divergence weight. The flat GRPO advantage is redistributed across tokens according to these weights, followed by clipping and additive mean-matching. This preserves each rollout's token-averaged GRPO advantage exactly, while allowing gradients to concentrate on chunks that distinguish the rollout from its siblings. When all weights are constant, SIRAS can fall back to GRPO.

For degenerate groups, mean-matching would collapse the signal back to zero. We therefore use the same sibling structure to introduce two calibrated auxiliary residuals. In all-correct groups, SIRAS rewards shorter successful reasoning paths and focuses the signal on divergent chunks that

separate shortcuts from detours. In all-wrong groups, it penalizes shared failing chunks and assigns a smaller exploration bonus to divergent alternatives. These residuals are scaled by the non-degenerate advantage magnitude within the same batch, so their gradient scale is commensurate with ordinary GRPO updates. Our contributions are summarized as:

- We formulate RLVR credit assignment as a sibling-relative problem: the same-prompt rollouts already sampled by GRPO define a cheap source of structured, sub-trajectory signal.
- We propose SIRAS, a drop-in advantage-shaping method based on hybrid chunking, banded DTW alignment, soft-divergence weighting, clipping, and additive mean-matching.
- We distinguish token-mean preservation from gradient unbiasedness and give an exact identity showing how SIRAS's gradient deviation from GRPO depends on the within-rollout covariance between token weights and score functions.
- We introduce calibrated auxiliary signals for all-correct and all-wrong groups, reclaiming groups that standard group-relative methods discard.
- We evaluate SIRAS on 6 benchmarks and 3 group-relative baselines. SIRAS improves AIME-style competition math most strongly, improves overall on larger benchmarks.

2 Related Work

Training reasoning language models. Reasoning language models are commonly improved through chain-of-thought prompting, supervised fine-tuning on reasoning traces, and reinforcement learning. RL is particularly attractive because it optimizes task-level feedback rather than merely imitating demonstrations. Recent reasoning-focused systems rely on verifiable rewards, especially for mathematics, coding, and logic tasks, where a programmatic verifier can score final answers automatically [Guo et al., 2025, Kimi Team, 2025]. This shift makes RL scalable for reasoning, but also exposes a central credit-assignment problem: a long reasoning trace is usually supervised only by a final outcome.

Process supervision and process reward models. One way to address sparse outcome supervision is to provide feedback at the step level. Dense process supervision can reduce reasoning errors even when final-answer supervision is held fixed [Uesato et al., 2022]. PRM800K scales this idea with human-labeled intermediate reasoning steps, enabling process-reward-guided reranking and training [Lightman et al., 2024]. Math-Shepherd reduces the human-labeling burden by using Monte-Carlo estimates to construct step-level supervision [Wang et al., 2024]. More recent work extends process reward modeling beyond scalar step scoring: R-PRM couples reasoning-based evaluation with preference optimization [She et al., 2025], ReasonFlux-PRM introduces trajectory-aware process scoring for long chain-of-thought traces [Zou et al., 2025], and Step-DPO applies process-level preference learning in an off-policy manner through per-step preference pairs [Lai et al., 2024]. These methods show the value of dense credit, but they require an additional reward model, step-level labels, Monte-Carlo step estimates, or a preference-pair pipeline.

RLVR and group-relative policy optimization. RLVR removes the need for a learned reward model by using programmatic outcome verifiers. To avoid the cost and instability of value-function learning, recent RLVR methods commonly use group-relative baselines. GRPO replaces the critic with a within-group baseline over multiple rollouts from the same prompt [Shao et al., 2024]. Dr.GRPO analyzes and corrects length bias introduced by GRPO-style normalization [Liu et al., 2025]. DAPO improves large-scale RLVR through dynamic sampling and decoupled clipping, while discarding zero-variance groups whose group-relative advantage vanishes [Yu et al., 2025]. Other recent variants examine the baseline and sampling process itself. KRPO denoises per-group rewards with a one-dimensional Kalman filter [Wang et al., 2025b]. Tree-branching analyses study how answer-level branching affects thought-level advantage variance [Wang et al., 2025a]. Scaf-GRPO addresses the learning cliff of all-failing groups by injecting in-prompt hints [Zhang et al., 2025]. In contrast, SIRAS does not modify prompts, rewards, or the rollout budget. It keeps the GRPO-style interface but extracts additional structure from the sibling rollouts inside each group.

Fine-grained credit, counterfactuals, and multi-path structure. Several recent methods refine the coarse trajectory-level advantage in RLVR. GRPO-λ uses eligibility-trace-style token returns [Parthasarathi et al., 2025], EAPO modulates updates by token entropy [He et al., 2026], SHEAR

uses hidden-state distributional differences between correct and incorrect rollouts [Chen et al., 2026], and CAPO obtains step-level signals from an external LLM evaluator [Xie et al., 2025]. Counterfactual methods such as VinePPO sample continuations from intermediate prefixes [Kazemnejad et al., 2024], while COMA and hindsight credit assignment use counterfactual baselines or learned posterior structure [Foerster et al., 2018, Harutyunyan et al., 2019]. Orthogonally, chain-of-thought and self-consistency exploit multiple reasoning paths at inference time [Wei et al., 2022, Wang et al., 2023]. SIRAS connects these ideas in a cheaper training-time form: it aligns the same-prompt rollouts already sampled by RLVR, extracts shared-versus-divergent chunk structure, and uses it to reshape the GRPO advantage without extra rollouts, external evaluators, or learned critics (see App. G and Table 10 for a structural comparison across methods).

3 Notation

For a prompt x, the behavior policy π_θ samples n rollouts $y^{(1)}, \ldots, y^{(n)}$, where rollout j has length T_j and receives a scalar verifier reward $r_j \in \mathbb{R}$. In a non-degenerate group, GRPO computes the group-normalized leaf advantage

$$\tilde{r}_j = \frac{r_j - \mu}{\sigma}, \qquad \mu = \frac{1}{n}\sum_{j=1}^{n} r_j, \qquad \sigma = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(r_j - \mu)^2}, \tag{1}$$

and assigns the same value to every token, $A_{j,t}^{\mathrm{GRPO}} = \tilde{r}_j$. If $\sigma = 0$, all rollouts in the group receive the same reward, and the group-relative advantage vanishes. DAPO-style filtering discards the group.

4 SIRAS: Sibling-Relative Advantage Shaping

Figure 2 illustrates the overall pipeline, which consists of several stages: hybrid chunk segmentation (§4.1), sibling chunk alignment and hierarchy construction (§4.2), mean-preserving advantage reshaping (§4.3), and degenerate-group residuals (§4.5). Algorithm 1 (App. A) summarizes the full pipeline.

4.1 Hybrid Chunk Segmentation

A rollout is partitioned into contiguous chunks $\mathcal{C}(j) = \{c_{j,1}, \ldots, c_{j,K_j}\}$ using three complementary signals: (i) structural markers (line breaks, `Step` k, markdown headers, math delimiters), (ii) log-probability surprise, which flags t as a boundary when its sliding-window z-score

$$z_{j,t} = \frac{\ell_{j,t} - \hat{\mu}_{j,t}}{\hat{\sigma}_{j,t} + \epsilon} < z^\star, \qquad \ell_{j,t} = \log \pi_\theta(y_t^{(j)} \mid x, y_{<t}^{(j)}), \tag{2}$$

and (iii) length caps $L_{\min}, L_{\max}$ that keep alignment cost bounded. Structural markers capture explicit formatting, surprise captures unmarked decisions, and length caps provide computational stability. Ablations in §5.3 compare this hybrid chunker with fixed-length, random, and structural-only variants.

4.2 Sibling Chunk Alignment

Each chunk c is represented by an L_2-normalized mean-pooled vector over its constituent tokens using the policy model's own static input embedding matrix, $\phi(c) \in \mathbb{R}^d$ (encoder-free and requiring no external network).

Banded DTW alignment. Given two chunk sequences $a_{1:K_a}$ and $b_{1:K_b}$, we compute a Sakoe–Chiba banded DTW alignment with radius $R = \max\big(\lfloor \min(K_a, K_b)/3 \rfloor, 4\big)$ and local distance $d(a_p, b_q) = 1 - \langle \phi(a_p), \phi(b_q) \rangle$. This costs $O(\max(K_a, K_b)R)$ and returns a monotone, many-to-one alignment. For a chunk c in rollout j, let $\mathcal{A}_{j\to k}(c)$ be its aligned chunks in rollout k. The pairwise sibling similarity is

$$s_{j\to k}(c) = \begin{cases} \max_{c' \in \mathcal{A}_{j\to k}(c)} \langle \phi(c), \phi(c') \rangle, & \mathcal{A}_{j\to k}(c) \neq \emptyset, \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

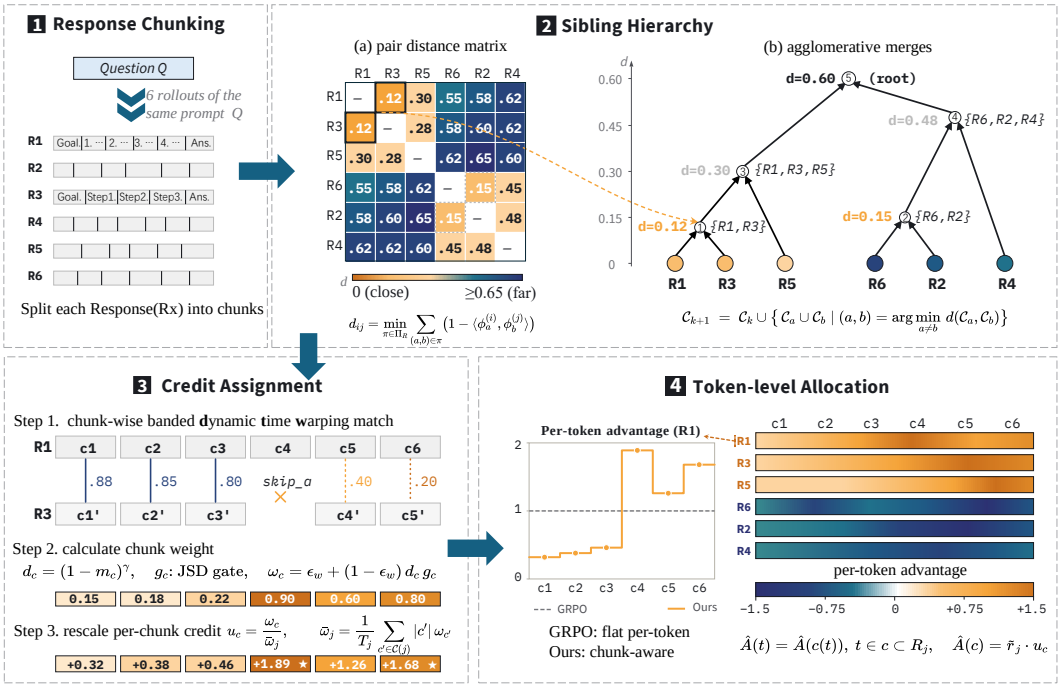

Figure 2: Overview of the SIRAS pipeline on a group of n=6 rollouts $\{R_1, \ldots, R_6\}$ for prompt Q. ① **Response Chunking** (§4.1): each rollout is split into contiguous chunks. ② **Sibling Hierarchy** (§4.2): pairwise banded-DTW distances (a) drive an average-linkage dendrogram (b) exposing which rollouts share reasoning and where they diverge. ③ **Credit Assignment** (§4.3): for a focal rollout, sibling chunks are aligned by DTW (Step 1). A chunk mass ω_c (Step 2, Eq. 8) is normalized to a mean-one multiplier $u_c = \omega_c/\bar{\omega}_j$ (Step 3). ④ **Token-level Allocation** (Eqs. 9–11): the GRPO advantage $\tilde{r}_j$ is broadcast as $\hat{A}(t) = \tilde{r}_j\, u_c$, replacing GRPO's flat signal (left, dashed) with a chunk-aware profile while preserving the per-rollout token mean (right, all six rollouts). Degenerate-group residuals (§4.5) reuse the same structure. Pseudocode in App. A.

and the leave-one-out sibling match is

$$m_c = \frac{1}{n-1} \sum_{k \neq j} s_{j \to k}(c), \qquad m_c \in [0, 1]. \tag{4}$$

The same pairwise DTW distances define an $n \times n$ matrix over same-prompt rollouts, from which we build an average-linkage agglomerative dendrogram. The default non-degenerate reshape uses the leave-one-out sibling set in Eq. (4) rather than nearest-tree siblings (the dendrogram is built from the same DTW distances, so nearest-cluster siblings would bias divergence scores downward). The dendrogram is used by the degenerate-group signals of §4.5. A tree-local variant that scopes m_c to the nearest ancestor cluster is ablated separately. DTW assumes approximately monotone, left-to-right traces. While this assumption naturally holds for standard chain-of-thought reasoning as well as solution-style code generated linearly from top to bottom (monotone, left-to-right generated code traces), it does not accommodate non-linear control flow or definitions placed after their calls. We therefore do not claim applicability to general coding tasks, tool-use, or agentic settings, and explicitly leave them as unverified open directions. When the monotonicity assumption is stressed, the log-probability-profile gate below, token clipping in Eq. (10), and the continuous blending to GRPO ($\alpha \to 0$) bound worst-case amplification and provide an algorithmic fallback to standard GRPO.

4.3 Soft-Divergence Advantage Reshaping

For each chunk c, we convert the sibling match into a soft divergence score and a Jensen–Shannon gate over token-level surprise profiles, and combine them into a non-negative chunk mass:

$$d_c = (1 - m_c)^\gamma, \quad \gamma \geq 1 \quad (\text{default } \gamma = 3), \tag{5}$$

$$p_c(t) = \frac{\exp(-\ell_t/\tau_\ell)}{\sum_{s\in c}\exp(-\ell_s/\tau_\ell)}, \qquad t \in c, \tag{6}$$

$$g_c = \max\big(g_{\min}, \mathrm{sigmoid}(\kappa(\overline{\mathrm{JSD}}_c - \tau))\big), \tag{7}$$

$$\omega_c = \epsilon_w + (1-\epsilon_w) d_c g_c, \tag{8}$$

where $\gamma = 3$ sharpens divergence contrast by suppressing weakly differentiated intermediate chunks, $\overline{\mathrm{JSD}}_c$ is the average JSD between p_c and the resampled profiles of its aligned siblings and $0 < \epsilon_w \ll 1$ ensures $\omega_c > 0$. Ablations for d_c (binary and sigmoid variants) and for the gate are reported in §5.3. Intuitively, divergent chunks that also differ in surprise profile retain mass, while shared or paraphrase-matched chunks are pushed to the floor. Making ω_c equal across chunks reduces the reshape identity (below) to GRPO exactly.

Token multiplier and advantage. Let $\mathrm{chunk}(t)$ denote the chunk containing token t in rollout j. Define the mean-one token multiplier

$$u_{j,t} = \frac{\omega_{\mathrm{chunk}(t)}}{\bar{\omega}_j}, \qquad \bar{\omega}_j = \frac{1}{T_j}\sum_{c\in\mathcal{C}(j)} |c|\,\omega_c, \tag{9}$$

which satisfies $\frac{1}{T_j}\sum_{t=1}^{T_j} u_{j,t} = 1$. The pre-clipped shaped advantage is $A^{(0)}_{j,t} = \tilde{r}_j u_{j,t}$. We cap per-token amplification and restore the token mean by an additive shift:

$$A^{(1)}_{j,t} = \mathrm{clip}\big(A^{(0)}_{j,t}, -M|\tilde{r}_j|, M|\tilde{r}_j|\big), \tag{10}$$

$$A_{j,t} = A^{(1)}_{j,t} + \Big(\tilde{r}_j - \frac{1}{T_j}\sum_s A^{(1)}_{j,s}\Big). \tag{11}$$

A blend $A^{\mathrm{final}}_{j,t} = \alpha A_{j,t} + (1-\alpha)\tilde{r}_j$ with $\alpha \in [0,1]$ (default $\alpha = 0.8$ after warm-up) recovers GRPO at $\alpha = 0$ and preserves the token-averaged advantage for all α.

4.4 Formal Properties

Construction 1 (Token-mean preservation)**.** For every rollout j in a non-degenerate group, the advantages defined by Eq. (11) satisfy $T_j^{-1}\sum_{t=1}^{T_j} A_{j,t} = \tilde{r}_j$. If all chunk masses ω_c are constant within rollout j, then $u_{j,t} = 1$ for all t and SIRAS recovers GRPO exactly.

This is a token-mean identity, not an unbiasedness theorem for the full policy-gradient estimator. The gradient deviation is intentional: it is the channel through which sibling-relative structure changes learning. The next proposition isolates this channel.

Proposition 1 (Gradient-deviation identity)**.** *Let $s_{j,t} = \nabla_\theta \log \pi_\theta(y^{(j)}_t \mid x, y^{(j)}_{<t})$ be the token score function and $\hat{g}^\alpha_j = \sum_{t=1}^{T_j}(\alpha A_{j,t} + (1-\alpha)\tilde{r}_j)s_{j,t}$ be the blended per-rollout gradient contribution. If the clip in Eq.* (10) *does not saturate, then*

$$\hat{g}^\alpha_j - \hat{g}^0_j = \alpha\tilde{r}_j \sum_{t=1}^{T_j}(u_{j,t}-1)s_{j,t} = \alpha\tilde{r}_j T_j\,\mathrm{Cov}_t(u_{j,t}, s_{j,t}), \tag{12}$$

where the covariance is empirical over token positions and is interpreted coordinate-wise.

When $u_{j,t}$ is constant, the covariance vanishes and the update is GRPO. Otherwise the within-rollout covariance between token multipliers and score functions determines how the update departs from GRPO. Under clipping, the additive correction in Eq. (11) adds a term $\alpha\delta_j\sum_t s_{j,t}$ with δ_j bounded, and in our runs fewer than 5% of tokens saturate. Because $u_{j,t}$ is piecewise constant within chunks, the covariance admits a chunk-level sum (see Cor. 1 in App. B).

4.5 Degenerate Groups

When all rewards in a group are identical, $\sigma = 0$ and the GRPO leaf advantages vanish. Any signal extracted from such a group is therefore auxiliary rather than verifier-derived. SIRAS reuses the same sibling structure to construct small, calibrated residuals that skip the mean-match step (forcing the token mean back to zero would eliminate the very channel we aim to recover). To keep degenerate-group gradients commensurate with ordinary RLVR gradients, we scale them by the batch's mean absolute non-degenerate advantage $s_{\mathrm{batch}} = \mathbb{E}_{\mathrm{nondeg}}[|\tilde{r}_j|]$, (set to 1 if the entire batch is degenerate).

All-correct groups: length efficiency. If every rollout succeeds, the verifier provides no preference among them. We use a conservative residual that rewards shorter successful rollouts and concentrates the signal on divergent chunks distinguishing shortcuts from detours. Let

$$z_j = \frac{T_j - \bar{T}}{\max(\hat{\sigma}_T, \rho\bar{T})}, \qquad q_j^{\mathrm{AC}} = -z_j - \frac{1}{n}\sum_{i=1}^{n}(-z_i) \tag{13}$$

be the group-centered length residual (with $\bar{T}, \hat{\sigma}_T$ the within-group mean and std of T_j, and $\rho > 0$ a floor factor), and let

$$v_{j,t}^{\mathrm{AC}} = \frac{\beta + (1-\beta)d_{\mathrm{chunk}(t)}}{T_j^{-1}\sum_s[\beta + (1-\beta)d_{\mathrm{chunk}(s)}]} \tag{14}$$

be the divergence-blended token mass. The all-correct auxiliary advantage is

$$A_{j,t}^{\mathrm{AC}} = \lambda\, s_{\mathrm{batch}}\, q_j^{\mathrm{AC}}\, v_{j,t}^{\mathrm{AC}}, \tag{15}$$

with $\lambda < 1$. We monitor response length to ensure the signal does not merely teach the model to omit reasoning.

All-wrong groups: shared-mode escape. If every rollout fails, assigning positive trajectory-level credit to any rollout would be unsafe. We instead create a zero-mean within-rollout signal that shifts probability mass away from shared failure modes and toward relatively divergent alternatives. Define

$$v_{j,t}^{\mathrm{AW}} = \frac{d_{\mathrm{chunk}(t)} + \epsilon_d}{T_j^{-1}\sum_s[d_{\mathrm{chunk}(s)} + \epsilon_d]}, \qquad A_{j,t}^{\mathrm{AW}} = \eta\, s_{\mathrm{batch}}\big(v_{j,t}^{\mathrm{AW}} - 1\big). \tag{16}$$

Chunks less divergent than the rollout average receive negative advantage, more divergent chunks receive positive advantage, and Eq. (16) has zero token mean within each failed rollout, so it does not assign positive trajectory-level reward to a failed response. The coefficient η is small, reflecting that divergent alternatives in an all-wrong group are exploratory rather than verified correct.

Setting all chunk masses to a constant makes $u_{j,t} = 1$ and recovers GRPO exactly, so SIRAS is a drop-in replacement. A full discussion of computational complexity and integration into the training loop is given in App. C.

5 Experiments

We empirically evaluate SIRAS along four axes: (i) main comparisons against three strong group-relative baselines (GRPO, Dr.GRPO, DAPO) on six reasoning benchmarks, (ii) ablations of each pipeline component, and (iii) analyses of how the gains scale with model capacity and rollout count n. Additional studies, including a degenerate-group rescue accounting, are reported in App. E.

5.1 Setup

Models and training. Unless noted otherwise, the model is **Qwen3-8B** [Yang et al., 2025] and the trainer is a verl-style [Sheng et al., 2025] RLVR pipeline. We cap response length at 8K tokens and optimize with AdamW using a 10-step warmup followed by a constant learning rate. Group size is n=8 on GSM8K/MATH/BBH and n=16 on AIME25/26 and Multi-LogiEval. For larger backbones (Qwen3-14B, Qwen3-32B, GPT-OSS-20B [OpenAI, 2025], Ling-Flash-2.0-100B [Ling Team, 2025]) we use the same pipeline with n=16 and the standard batch-size adjustments listed in App. D.

Benchmarks. We evaluate on six benchmarks spanning grade-school math (GSM8K), competition math (MATH, AIME25 and AIME26), broad reasoning (BBH), and multi-step logic (MultiLogi-Eval). Dataset details are listed in App. H. For each benchmark we report pass@1 averaged over 3 seeds. The parenthesized value is the between-seed standard deviation.

Baselines. We compare against three group-relative objectives: GRPO [Shao et al., 2024] (group-mean baseline), Dr.GRPO [Liu et al., 2025] (a length-bias-corrected variant), and DAPO [Yu et al., 2025] (drops degenerate groups). We additionally include GRPO-λ [Parthasarathi et al., 2025] as a

Table 1: Main results (pass@1, %) on Qwen3-8B; bold marks the best. Setup and baselines: §5.1.

Method	GSM8K	MATH	AIME25	AIME26	BBH	MultiLogi
GRPO	93.10 ±0.08	87.20 ±0.20	15.56 ±1.92	20.00 ±3.33	83.96 ±0.08	84.41 ±0.29
+ GRPO-λ	93.40 ±0.08	**87.47 ±0.12**	22.22 ±1.92	18.89 ±1.92	84.27 ±0.08	84.71 ±0.29
+ SIRAS	**93.78 ±0.08**	**87.47 ±0.12**	**27.78 ±1.92**	**27.78 ±1.92**	**84.98 ±0.04**	**85.20 ±0.17**
Dr.GRPO	93.63 ±0.08	85.60 ±0.20	16.67 ±3.33	22.22 ±1.92	84.65 ±0.08	83.14 ±0.17
+ GRPO-λ	93.86 ±0.08	85.40 ±0.20	**22.22 ±1.92**	25.56 ±1.92	84.96 ±0.08	**83.33 ±0.17**
+ SIRAS	**94.11 ±0.04**	**87.00 ±0.20**	**22.22 ±1.92**	**27.78 ±1.92**	**85.19 ±0.08**	**83.33 ±0.17**
DAPO	93.78 ±0.08	88.40 ±0.20	20.00 ±3.33	20.00 ±3.33	83.96 ±0.08	85.00 ±0.29
+ GRPO-λ	**93.83 ±0.09**	88.53 ±0.12	23.33 ±3.33	21.11 ±1.92	83.99 ±0.09	85.10 ±0.17
+ SIRAS	**93.83 ±0.04**	**88.67 ±0.12**	**26.67 ±3.33**	**22.22 ±1.92**	**84.04 ±0.08**	**85.29 ±0.29**

token-level credit-shaping baseline, which replaces the flat trajectory advantage with eligibility-trace-style λ-returns derived from token log-probabilities. GRPO-λ is the closest in spirit to SIRAS in that it also redistributes credit across tokens without extra rollouts or an auxiliary critic, but it operates purely within a single rollout rather than across sibling rollouts. For each group-relative baseline we report the matched triplet *baseline*, *baseline* + *GRPO-λ*, and *baseline* + SIRAS. The only change is the per-token advantage fed to the PPO surrogate.

5.2 Main results

Table 1 shows that adding SIRAS on top of GRPO, Dr.GRPO, or DAPO improves every base objective on Qwen3-8B on the small, high-variance competition benchmarks (AIME25/26): the largest absolute gain is $+12.2$ on AIME25 (GRPO $\rightarrow$ GRPO+SIRAS, $15.56 \rightarrow 27.78$). Because each AIME competition contributes only 30 items, we treat the cross-cell direction as the primary AIME-level evidence ($14/14$ AIME cells positive across Tables 1 and 4, sign-test $p \approx 1.2\times10^{-4}$). Per-cell paired bootstrap, sign, and Wilcoxon tests are reported in App. E.3. On the large benchmarks (GSM8K, MATH, BBH, MultiLogi) gains are smaller but consistent: $+0.7$ pass@1 on GSM8K and $+1.0$ on BBH on top of GRPO, with a $+0.3$ recovery on MATH. We observe no meaningful degradation in the main table: DAPO+SIRAS also recovers a small $+0.1$ on BBH ($83.96 \rightarrow 84.04$), so SIRAS improves overall on the large benchmarks. GRPO-λ provides a modest lift over its base objective across most cells (e.g. GRPO+ GRPO-λ adds $+6.7$ on AIME25 and $+0.3$ on GSM8K over GRPO), confirming that token-level reshaping alone is already helpful. However, SIRAS exceeds GRPO-λ (e.g. $+5.6$ on AIME25 and $+8.9$ on AIME26 on top of GRPO), indicating that sibling-relative structure captures credit that single-rollout eligibility traces miss. We emphasize an explicit division of labor between this baseline comparison and our internal ablations: the external comparison against GRPO-λ serves as a reference frame showing the relative strength of cross-sibling credit over single-rollout credit, whereas the internal ablation control in §5.3 strictly isolates the existence of the cross-sibling contribution. The gain concentrates where group-relative baselines lose signal: on AIME-difficulty problems where degenerate groups are abundant. Disabling the degenerate-group residual cuts the AIME25 gain of GRPO+SIRAS from $+12.2$ to $+6.7$, attributing roughly half of the AIME-difficulty gain to groups DAPO discards (App. E). On Qwen3-8B AIME26 at $n{=}8$ the degenerate fraction averages $\sim$57% across training (starting near 78% at the first logged step, peaking briefly above and then decaying as the policy starts solving hard prompts, Table 6).

5.3 Ablations

Table 2 isolates the four pipeline choices that constitute SIRAS. The hybrid chunker is the largest single contributor on MultiLogi-Eval ($+3.5$ over structural-only, $+6.1$ over fixed-length), confirming the design intuition of §4.1: structural markers alone miss the unmarked "thinking" regions, while purely length-based segmentation cuts across reasoning units. Under the alignment dimension, the internal ablation control labeled *None* retains every other mechanism of SIRAS (hybrid chunking, soft-divergence weighting, and residuals) and removes only the cross-sibling alignment, costing -3.1 points on MultiLogi and -2.2 points on AIME25. This internal control strictly isolates the existence of the cross-sibling structural contribution without confounding factors, while the external GRPO-λ comparison in Table 1 establishes its relative strength against within-rollout credit. Banded DTW alignment also matters more on AIME than MultiLogi, since AIME rollouts share substantial early scaffolding that positional alignment mis-attributes as divergent. Among the divergence weightings,

Table 2: Component ablations on Qwen3-8B (pass@1, %). Each row swaps a single component.

Component	Variant	MultiLogi	AIME25	AIME26
Chunking	Random	79.61 ±0.17	22.22 ±1.92	23.33 ±3.33
	Fixed length	79.12 ±0.29	16.67 ±3.33	18.89 ±1.92
	Structural only	81.67 ±0.17	18.89 ±1.92	22.22 ±1.92
	Hybrid (Ours)	**85.20 ±0.17**	**27.78 ±1.92**	**27.78 ±1.92**
Alignment	None	82.06 ±0.29	25.56 ±1.92	23.33 ±3.33
	Positional	81.08 ±0.17	20.00 ±3.33	18.89 ±1.92
	Banded DTW (Ours)	**85.20 ±0.17**	**27.78 ±1.92**	**27.78 ±1.92**
Divergence	Binary $\{0, 1\}$	82.25 ±0.17	25.56 ±1.92	25.56 ±1.92
	Sigmoid	80.29 ±0.29	23.33 ±3.33	22.22 ±1.92
	Linear $w = (1-m)^\gamma$ (Ours)	**85.20 ±0.17**	**27.78 ±1.92**	**27.78 ±1.92**
Degenerate	No handling	81.08 ±0.17	22.22 ±1.92	20.00 ±3.33
	Residual handle (Ours)	**85.20 ±0.17**	**27.78 ±1.92**	**27.78 ±1.92**

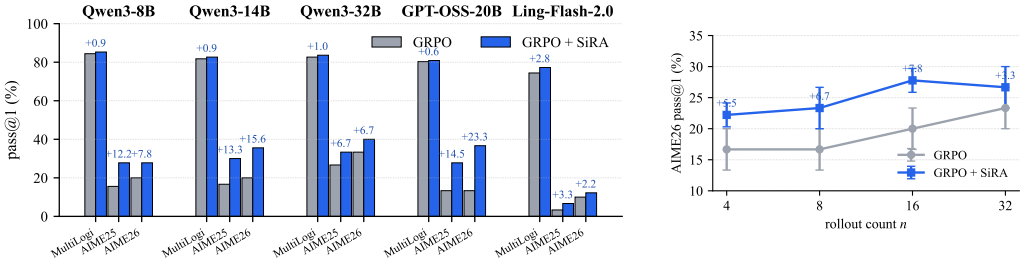

Figure 3: Robustness across model and rollout-budget axes. *Left:* pass@1 of GRPO vs. GRPO+SIRAS on five backbones. *Right:* sensitivity to per-prompt rollout count $n \in \{4, 8, 16, 32\}$ on Qwen3-8B. Bars are pass@1, line is the GRPO→SIRAS delta. Per-backbone numbers in App. E.

the binary variant performs surprisingly close on AIME but loses ~ 3 points on MultiLogi, reflecting that a hard threshold discards the gradient information carried by intermediate-match chunks. Finally, disabling the degenerate-group residual (§4.5) drops MultiLogi by -4.1, AIME25 by -5.6, and AIME26 by -7.8: once all-correct/all-wrong groups lose their sibling baseline, the remaining non-degenerate covariance signal is not enough to preserve the AIME-difficulty gain.

5.4 Analyses

Scaling and rollout count. Figure 3 (left) shows that the gain holds across all five backbones and both benchmarks despite their different parameter counts (8B–100B), architectures (dense vs. sparse MoE), and pretraining lineages (Qwen3, GPT-OSS, Ling-Flash). The two largest single-cell gains, $+23.3$ on GPT-OSS-20B AIME26 and $+15.6$ on Qwen3-14B AIME26, span both the dense and sparse-MoE regime, suggesting that the sibling-relative signal is a property of the rollout structure rather than of any specific backbone. Figure 3 (right) varies $n \in \{4, 8, 16, 32\}$: the gain ranges from $+5.6$ to $+7.8$ pass@1 across $n \in \{4, 8, 16\}$ in the practically relevant regime, and tapers to $+3.3$ at n=32, where the GRPO baseline already has enough siblings to estimate the group mean reliably. This matches Proposition 1: the within-rollout covariance channel carries the most information precisely when the group-mean baseline is noisiest, so SIRAS contributes most where existing methods have least signal. The featurization and alignment stages partially overlap with rollout generation in our `fully async` runtime, keeping the end-to-end wall-clock overhead around $\sim$20% in our setup (App. E, Table 8).

Non-independent consistency sanity check with human-verified credit. To check whether sibling-relative credit correlates with the tokens a human reader would highlight, we conduct a non-independent consistency sanity check. We draw a stratified sample of 30 groups (22 mixed, 5 all-correct, 3 all-wrong) from an offline GRPO rollout corpus and annotate all $30 \times 16 = 480$ rollouts with step-level criticality κ_s and per-token importance w_t, composed into a reference per-token credit ρ_t. As detailed in the protocol in App. F, reference annotations were produced by an LLM first pass and verified item by item by the authors; while predicted SIRAS credit was strictly

withheld during annotation, no independent annotators were involved and this analysis provides correlational evidence of consistency rather than causal credit validation. The pooled per-response word-level Pearson between SIRAS's credit and ρ_t is 0.740 (Spearman 0.761), and no response has a negatively-correlated credit signal. The three regimes SIRAS treats differently (mixed, all-correct, all-wrong) agree within a 0.02 Pearson spread, even though the degenerate-group paths of §4.5 share no computation with the non-degenerate mean-match identity, suggesting that the $+5.6$ AIME25 component attributed to rescued groups is consistent with human-identified load-bearing tokens at the response level.

6 Conclusion

We presented SIRAS, a sibling-relative advantage-shaping method for RLVR. Instead of treating same-prompt rollouts as independent reward labels, SIRAS uses their shared and divergent structure to derive a sub-trajectory credit signal. It segments rollouts into chunks, aligns siblings with banded DTW, and redistributes the GRPO advantage through mean-one soft-divergence weights, without process reward models, step annotations, or extra rollouts. The same structure also provides calibrated auxiliary residuals for all-correct and all-wrong groups. SIRAS improves pass@1 over six benchmarks in different domains. A non-independent consistency sanity check with human-verified credit annotations further suggests that the reshaped signal correlates with load-bearing tokens in model responses.

7 Limitations

SIRAS preserves the per-rollout token mean of the GRPO advantage (Construction 1), but it does not preserve the full policy-gradient expectation. The deviation is exactly the within-rollout covariance channel in Proposition 1, through which sibling structure enters the update. Constant chunk masses recover GRPO, clipping bounds token-level amplification, and the blend coefficient α provides a continuous fallback to GRPO. Moreover, the DTW alignment assumes reasoning traces are approximately monotone and left-to-right. While this fits top-down chain-of-thought and linear code generation, it does not apply to non-linear control flow or definitions after calls, and general code, tool-use, and agentic settings remain unverified open directions. The all-correct and all-wrong signals are calibrated auxiliary objectives, not verifier-derived rewards. They work well in our reasoning benchmarks, but may require retuning in settings where shorter reasoning is undesirable or divergent failed traces are unsafe to encourage.

Finally, the chunking of SIRAS relies on surface-to-token alignment, where character spans identified on decoded text are mapped back to token intervals. The representational non-uniqueness of subword detokenization, a phenomenon where multiple distinct token sequences map to identical surface strings [Ayoobi et al., 2026], is an unmeasured potential source of fragility, particularly across tokenization artifacts such as digit strings, hyphens, and LaTeX formulas. Defining chunk boundaries directly in token space without passing through intermediate surface strings remains an important direction for future work.

Acknowledgments and Disclosure of Funding

This work was funded by the New Generation Artificial Intelligence-National Science and Technology Major Project (2023ZD0120802), National Natural Science Foundation of China (52541003,62302433), 'Pioneer' and 'Leading Goose' R&D Program of Zhejiang (2024C01135). This work was supported by Ant Group through CCF-Ant Research Fund (CCF-AFSG RF20250612).

References

Navid Ayoobi, Marcus I. Armstrong, and Arjun Mukherjee. Say anything but this: When tokenizer betrays reasoning in LLMs. *arXiv preprint arXiv:2601.14658*, 2026.

Xinzhu Chen, Wei He, Huichuan Fan, Wenzhe Niu, Zhongxiang Sun, Xuanru Wang, Jiuchong Gao, Jinghua Hao, Renqing He, and Weijie Yu. Hidden states know where reasoning diverges: Credit assignment via span-level Wasserstein distance. *arXiv preprint arXiv:2604.23318*, 2026.

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.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 32, 2018.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645:633–638, 2025.

Anna Harutyunyan, Will Dabney, Thomas Mesnard, Mohammad G. Azar, Bilal Piot, Nicolas Heess, Hado van Hasselt, Greg Wayne, Satinder Singh, Doina Precup, and Rémi Munos. Hindsight credit assignment. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Yuhang He, Haodong Wu, Siyi Liu, Hongyu Ge, Hange Zhou, Keyi Wu, Zhuo Zheng, Qihong Lin, Zixin Zhong, and Yongqi Zhang. Rethinking token-level credit assignment in RLVR: A polarity-entropy analysis. *arXiv preprint arXiv:2604.11056*, 2026.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2021.

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. *arXiv preprint arXiv:2410.01679*, 2024.

Kimi Team. Kimi k1.5: Scaling reinforcement learning with LLMs. *arXiv preprint arXiv:2501.12599*, 2025.

Xin Lai, Zhuotao Tian, Yukang Chen, Senqiao Yang, Xiangru Peng, and Jiaya Jia. Step-DPO: Step-wise preference optimization for long-chain reasoning of LLMs. *arXiv preprint arXiv:2406.18629*, 2024.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri 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. arXiv:2305.20050.

Yen-Ting Lin. AIME 2025 problem set. `https://huggingface.co/datasets/yentinglin/aime_2025`, 2025. Problems by the Mathematical Association of America. Dataset.

Ling Team. Every activation boosted: Scaling general reasoner to 1 trillion open language foundation. *arXiv preprint arXiv:2510.22115*, 2025.

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. *arXiv preprint arXiv:2503.20783*, 2025.

MathArena. AIME 2026 problem set. `https://huggingface.co/datasets/MathArena/aime_2026`, 2026. Problems by the Mathematical Association of America. Dataset.

mytestdpo. grpo_n4_bz512_120. `https://huggingface.co/datasets/mytestdpo/grpo_n4_bz512_120`, 2025. Accessed: 2026-05-05.

Graham Neubig. AIME 1983–2024 problem set. `https://huggingface.co/datasets/gneubig/aime-1983-2024`, 2024. Historical training split; problems by the Mathematical Association of America. Dataset.

OpenAI. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025.

Prasanna Parthasarathi, Mathieu Reymond, Boxing Chen, Yufei Cui, and Sarath Chandar. GRPO-λ: Credit assignment improves LLM reasoning. *arXiv preprint arXiv:2510.00194*, 2025.

Nisarg Patel, Mohith Kulkarni, Mihir Parmar, Aashna Budhiraja, Mutsumi Nakamura, Neeraj Varshney, and Chitta Baral. Multi-LogiEval: Towards evaluating multi-step logical reasoning ability of large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 20856–20879. Association for Computational Linguistics, 2024.

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.

Shuaijie She, Junxiao Liu, Yifeng Liu, Jiajun Chen, Xin Huang, and Shujian Huang. R-PRM: Reasoning-driven process reward modeling. *arXiv preprint arXiv:2503.21295*, 2025.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. HybridFlow: A flexible and efficient RLHF framework. In *Proceedings of the Twentieth European Conference on Computer Systems (EuroSys)*, pages 1279–1297. ACM, 2025. doi: 10.1145/3689031.3696075. arXiv:2409.19256; source code: `https://github.com/volcengine/verl`.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research (TMLR)*, 2023.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. Challenging BIG-Bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics (ACL Findings)*, pages 13003–13051. Association for Computational Linguistics, 2023.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback. *arXiv preprint arXiv:2211.14275*, 2022.

Hongcheng Wang, Yinuo Huang, Sukai Wang, Guanghui Ren, and Hao Dong. Why tree-style branching matters for thought advantage estimation in GRPO. *arXiv preprint arXiv:2509.24494*, 2025a.

Hu Wang, Congbo Ma, Ian Reid, and Mohammad Yaqub. Kalman filter enhanced GRPO for reinforcement learning-based language model reasoning. *arXiv preprint arXiv:2505.07527*, 2025b.

Peiyi Wang, Lei Li, Zhihong Shao, R. X. Xu, Damai Dai, Yifei Li, Deli Chen, Y. Wu, and Zhifang Sui. Math-Shepherd: Verify and reinforce LLMs step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL), Volume 1: Long Papers*, pages 9426–9439. Association for Computational Linguistics, 2024.

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

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Guofu Xie, Yunsheng Shi, Hongtao Tian, Ting Yao, and Xiao Zhang. CAPO: Towards enhancing LLM reasoning through generative credit assignment. *arXiv preprint arXiv:2508.02298*, 2025.

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, Junyang Lin, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Xichen Zhang, Sitong Wu, Yinghao Zhu, Haoru Tan, Shaozuo Yu, Ziyi He, and Jiaya Jia. Scaf-GRPO: Scaffolded group relative policy optimization for enhancing LLM reasoning. *arXiv preprint arXiv:2510.19807*, 2025.

Jiaru Zou, Ling Yang, Jingwen Gu, Jiahao Qiu, Ke Shen, Jingrui He, and Mengdi Wang. ReasonFlux-PRM: Trajectory-aware PRMs for long chain-of-thought reasoning in LLMs. *arXiv preprint arXiv:2506.18896*, 2025.

A Algorithm

Algorithm 1 SIRAS: Per-token credit for a GRPO group

Require: rollouts $\{y^{(j)}\}_{j=1}^{n}$, rewards $\{r_j\}$, log-probs $\{\log \pi\}$, batch scale s
Compute GRPO leaf advantages $\tilde{r}_j$
if group is non-degenerate **then**
    Segment each $y^{(j)}$ into chunks (§4.1)
    Pairwise banded DTW $\Rightarrow$ distance matrix and alignments (§4.2)
    Agglomerative average-linkage dendrogram (sibling hierarchy)
    For each chunk c compute mean match m_c, soft divergence $d_c = (1 - m_c)^{\gamma}$ (Eq. 5)
    JSD gate g_c (Eq. 7) $\Rightarrow$ chunk mass $\omega_c = \epsilon_w + (1 - \epsilon_w)\, d_c\, g_c$ (Eq. 8)
    Per token t: $A^{(0)}_{j,t} = \tilde{r}_j\, \omega_{\mathrm{chunk}(t)} / \bar{\omega}_j$ (Eq. 9); clip to $\pm M|\tilde{r}_j|$ (Eq. 10); additive shift $\delta_j = \tilde{r}_j - T_j^{-1} \sum_s A^{(1)}_{j,s}$ (Eq. 11)
else if all rewards correct **then**
    Same hierarchy; length-aware credit (§4.5)
else if all rewards wrong **then**
    Same hierarchy; exploration-aware credit (§4.5)
end if
return $\{A_{j,t}\}$

B Proofs and Extended Formal Statements

The method section states Construction 1 (token-mean preservation) and Proposition 1 (gradient-deviation identity). The accompanying lemma, chunk-level corollary, and remark are stated here.

Lemma 1 (Uniqueness and boundedness of mean matching)**.** *For any clipped advantage vector $A^{(1)}_{j,1:T_j}$, Eq.* (11) *is the unique uniform per-token shift that enforces $T_j^{-1} \sum_t A_{j,t} = \tilde{r}_j$. Moreover, if $A^{(1)}_{j,t} \in [-M|\tilde{r}_j|, M|\tilde{r}_j|]$ for all t, then*

$$\delta_j = \tilde{r}_j - \frac{1}{T_j} \sum_s A^{(1)}_{j,s} \qquad \textit{satisfies} \qquad |\delta_j| \le (M+1)|\tilde{r}_j| \le 2M|\tilde{r}_j|, \quad M \ge 1.$$

Corollary 1 (Chunk-level covariance identity)**.** *Because $u_{j,t}$ is piecewise constant within each chunk with value $u_c = \omega_c / \bar{\omega}_j$, the covariance in Eq.* (12) *collapses to a chunk-level sum:*

$$\mathrm{Cov}_t(u_{j,t}, s_{j,t}) = \frac{1}{T_j} \sum_{c \in \mathcal{C}(j)} |c|(u_c - 1)(\bar{s}_c - \bar{s}_j), \tag{17}$$

with $\bar{s}_c = |c|^{-1} \sum_{t \in c} s_{j,t}$ and $\bar{s}_j = T_j^{-1} \sum_t s_{j,t}$. This identity separates the score-independent aggressiveness of the reshaping from the score-dependent gradient direction. The chunk-length-weighted dispersion

$$\sigma_j^{\mathrm{chunk}}(u) = \Big[\sum_{c \in \mathcal{C}(j)} \tfrac{|c|}{T_j} (u_c - 1)^2 \Big]^{1/2} \tag{18}$$

serves as a zero-cost diagnostic for reshaping aggressiveness; it is zero when SIRAS *reduces to GRPO and, under the default linear mode, is bounded by $M - 1$.*

Remark 1 (Three regimes for the covariance channel). The covariance in Eq. (12) vanishes when $u_{j,t}$ is constant across tokens (*neutral*: GRPO recovered exactly). When small multipliers concentrate on confident tokens and large ones on decision-forming tokens, the reshaping reduces per-rollout gradient variance along that direction (*benign*: the regime SIRAS is designed to produce). The opposite correlation increases variance, but $|u_{j,t}| \le M$ caps the unconditional worst case at M^2 (see the justification below).

Proof of Construction 1. Fix a rollout j in a non-degenerate group and let its chunk partition $\mathcal{C}(j)$ have per-chunk lengths $|c|$ and chunk masses ω_c (Eq. 8). Writing $\bar{\omega}_j = T_j^{-1} \sum_c |c|\, \omega_c$, the pre-clip

advantage from Eq. (9) satisfies

$$\frac{1}{T_j}\sum_{t=1}^{T_j} A_{j,t}^{(0)} = \frac{1}{T_j}\sum_{c\in\mathcal{C}(j)} |c|\cdot\tilde{r}_j\cdot\frac{\omega_c}{\bar{\omega}_j} = \tilde{r}_j\cdot\frac{1}{\bar{\omega}_j}\cdot\frac{1}{T_j}\sum_c |c|\,\omega_c = \tilde{r}_j$$

by the definition of $\bar{\omega}_j$. The clipping step in Eq. (10) may shift the mean off $\tilde{r}_j$ when at least one $|A_{j,t}^{(0)}|$ exceeds $M|\tilde{r}_j|$. The additive correction in Eq. (11) sets $A_{j,t} = A_{j,t}^{(1)} + (\tilde{r}_j - \frac{1}{T_j}\sum_s A_{j,s}^{(1)})$, so

$$\frac{1}{T_j}\sum_t A_{j,t} = \frac{1}{T_j}\sum_t A_{j,t}^{(1)} + \tilde{r}_j - \frac{1}{T_j}\sum_s A_{j,s}^{(1)} = \tilde{r}_j. \qquad \square$$

Proof of Proposition 1. Assume the clip in Eq. (10) does not saturate, so $A_{j,t} = A_{j,t}^{(0)} = \tilde{r}_j u_{j,t}$ with $u_{j,t}$ defined in Eq. (9). Then $A_{j,t}^{\text{final}} = (\alpha u_{j,t} + (1-\alpha))\tilde{r}_j$ and

$$\hat{g}_j^{\alpha} - \hat{g}_j^{0} = \sum_t \big(A_{j,t}^{\text{final}} - \tilde{r}_j\big)s_{j,t} = \alpha\tilde{r}_j\sum_t (u_{j,t}-1)s_{j,t} = \alpha\tilde{r}_j\,T_j\cdot\mathrm{Cov}_t(u_{j,t}, s_{j,t}),$$

where the last equality uses $T_j^{-1}\sum_t u_{j,t} = 1$ (Eq. 9) and treats token index t as an empirical sampling variable over the T_j tokens of rollout j. This is the per-rollout identity stated in Proposition 1; it is exact (no inequalities). The scalar form $v^\top(\hat{g}_j^{\alpha} - \hat{g}_j^{0}) = \alpha\tilde{r}_j T_j\cdot\mathrm{Cov}_t(u_{j,t}, v^\top s_{j,t})$ follows by projecting onto any direction v. When the clip saturates, Lemma 1 supplies the additive correction δ_j and the identity picks up a term $\delta_j\sum_t s_{j,t}$, which we do not attempt to bound in closed form. We emphasize that Cauchy–Schwarz would give an inequality $|v^\top(\cdot)| \le \alpha|\tilde{r}_j|T_j\cdot\mathrm{std}_t(u)\mathrm{std}_t(v^\top s)$ that scales with response length T_j and is therefore not a tight or actionable bound; we report only the identity.

Justification for Remark 1. The unconditional bound $\mathrm{Var}_t(u_{j,t}\,v^\top s_{j,t}) \le M^2\mathrm{Var}_t(v^\top s_{j,t})$ follows immediately from $|u_{j,t}| \le M$ via $\mathrm{Var}(X) \le \mathbb{E}[X^2] \le M^2\mathbb{E}[Y^2]$ where $X = u_{j,t}v^\top s_{j,t}$ and $Y = v^\top s_{j,t}$. This bound is not meaningful for our design (we do not choose $u_{j,t}$ to achieve a worst-case M^2 amplification). The intended regime is the empirical observation that the chunk-relative weighting damps large score-function contributions from shared chunks — i.e., $u_{j,t}^2$ tends to be small where $(v^\top s_{j,t})^2$ is large — in which case $\mathbb{E}_t[u^2(v^\top s)^2] < \mathbb{E}_t[(v^\top s)^2]$. We state this as a design intuition rather than a theorem because the condition is not testable from trainer logs (it requires $\nabla_\theta \log\pi_\theta$ at every token for some reference direction v), and we do not claim it holds universally.

Proof of Lemma 1. A constant shift δ_j applied uniformly to every $A_{j,t}^{(1)}$ maps the token-mean from $T_j^{-1}\sum_t A_{j,t}^{(1)}$ to $T_j^{-1}\sum_t A_{j,t}^{(1)} + \delta_j$. The unique value of δ_j that makes this equal to $\tilde{r}_j$ is the right-hand side of Eq. (11), so the additive correction is uniquely determined. The bound $|\delta_j| \le 2M|\tilde{r}_j|$ follows because $A_{j,t}^{(1)} \in [-M|\tilde{r}_j|, M|\tilde{r}_j|]$ implies the T_j-mean lies in the same interval, and so the deficit relative to $\tilde{r}_j \in [-M|\tilde{r}_j|, M|\tilde{r}_j|]$ is bounded by $2M|\tilde{r}_j|$. $\square$

Proof of Corollary 1. For any token t in chunk c, $u_{j,t} = u_c$ by Eq. (9). Writing the empirical token-level covariance,

$$\mathrm{Cov}_t(u_{j,t}, s_{j,t}) = \frac{1}{T_j}\sum_t u_{j,t}s_{j,t} - \bar{u}_j\cdot\bar{s}_j = \frac{1}{T_j}\sum_c\sum_{t\in c} u_c\,s_{j,t} - \bar{s}_j,$$

where the second equality uses $\bar{u}_j = T_j^{-1}\sum_t u_{j,t} = 1$ (Eq. 9). Pulling u_c out of the inner sum, $\sum_{t\in c} s_{j,t} = |c|\,\bar{s}_c$, so

$$\mathrm{Cov}_t(u_{j,t}, s_{j,t}) = \frac{1}{T_j}\sum_c |c|\,u_c\,\bar{s}_c - \bar{s}_j = \frac{1}{T_j}\sum_c |c|\,(u_c - 1)\,(\bar{s}_c - \bar{s}_j),$$

where the second equality adds and subtracts $\bar{s}_j\sum_c |c|\,u_c/T_j = \bar{s}_j$ (using $\sum_c |c|u_c = T_j\bar{u}_j = T_j$) and $\sum_c |c|\bar{s}_c/T_j = \bar{s}_j$. This proves Eq. (17). The Cauchy–Schwarz step stated in the body of the corollary is the chunk-length-weighted version of the standard inequality, treating $|c|/T_j$ as a

probability mass on chunks. For the final claim, under the non-saturation assumption of Prop. 1 (so that $A^{(1)}_{j,t} = A^{(0)}_{j,t} = \tilde{r}_j u_{j,t}$ and $|\tilde{r}_j u_{j,t}| \le M|\tilde{r}_j|$), we have $|u_c| \le M$ and $|u_c - 1| \le M + 1$; combined with $\bar{u}_j = 1$, the chunk-length-weighted variance of u is bounded by $\sum_c |c|/T_j \cdot (u_c - 1)^2 \le \max_c (u_c - 1)^2$. For $u_c \in [0, M]$ with $M \ge 2$ (the operative regime in our default `linear` mode, where $\omega_c \ge 0$ and the amplification cap is set to $M \ge 2$), this gives $\max_c (u_c - 1)^2 \le (M - 1)^2$; for $1 \le M < 2$ the tighter bound is $\max_c (u_c - 1)^2 \le 1$, and allowing u_c to take values in $[-M, M]$ gives the looser bound $(M + 1)^2$. We state $\sigma_j^{\text{chunk}}(u) \le M - 1$ as the operative bound for the default configuration ($M \ge 2$). □

C Complexity and Integration

For a group of size n, maximum chunk count C, and DTW band radius R, pairwise alignment costs $O(n^2 CR)$ and average-linkage clustering costs $O(n^2 \log n)$. In our default configuration, $n = 16$ and C is typically between 20 and 60. The computation runs on CPU workers and is parallel across groups. The policy forward pass, verifier, rollout count, and optimizer are unchanged. The main tuned hyperparameters are the soft-divergence exponent γ ($\gamma = 3$, sharpening the $(1 - m_c)^\gamma$ divergence profile) and the amplification cap M ($M = 5$); chunk representations use the policy model's input token embedding matrix mean-pooled over each chunk, and other scalars (chunking geometry, JSD gate, degenerate-group coefficients) are fixed across benchmarks. Setting all chunk masses to a constant makes $u_{j,t} = 1$ and recovers GRPO exactly, so SIRAS is a drop-in replacement that can be enabled after a warm-up phase.

D Hyperparameters and training details

Hardware. All experiments are run on homogeneous compute nodes, each equipped with $8\times$ NVIDIA Hopper-architecture GPUs (141 GB HBM per GPU), 1 TB host memory, a 96-core CPU, and 512 GB of local disk. All nodes are scheduled through a single Ray cluster and every training run is submitted as a Ray Job.

Two compute configurations. All experiments fall into one of two GPU layouts:

- **Config A** (Qwen3-8B on GSM8K, MATH, BBH): 24 GPUs for rollout (3 nodes $\times$ 8) and 8 GPUs for training (1 node $\times$ 8), group size n=8, PPO mini-batch 4×32 prompts.
- **Config B** (Qwen3-8B on AIME25/26 and Multi-LogiEval; all Qwen3-14B/32B, GPT-OSS-20B, and Ling-Flash-2.0 runs on every benchmark; all ablations): 32 GPUs for rollout (4×8) and 32 GPUs for training (4×8), group size n=16, PPO mini-batch 2×16 prompts.

Config B uses tensor-parallel rollout (`gen_tp`=2 for Qwen/Ling, `gen_tp`=8 for GPT-OSS-20B) and actor-parameter offload; Config A uses `gen_tp`=1 with no actor offload. Both configurations use the same algorithmic, optimization, generation, and asynchrony hyperparameters listed in Table 3.

All other hyperparameter configurations remain consistent across all experiments, and the experimental scripts can be obtained from the publicly available code.

E Additional experimental details

This section is organized in three parts: scaling and rollout-count numbers (§E.1), the degenerate-group distribution across benchmarks (§E.2), and a single compute-overhead study covering both long- and short-response regimes (§E.4).

E.1 Per-backbone scaling and rollout count

Per-backbone scaling. Table 4 reports the per-backbone numbers underlying Figure 3 (left). SIRAS improves over GRPO on every backbone and benchmark we evaluate; the AIME26 gain on GPT-OSS-20B is the largest observed in any single cell of any of our experiments (+23.3, 13.33 → 36.67), with Qwen3-14B AIME26 (+15.6) a close second.

Table 3: Common training hyperparameters (identical across both compute configurations unless noted). Values marked "Config A / Config B" differ between the two layouts defined above.

Category	Hyperparameter	Value
Optimization	Optimizer	AdamW
	Peak learning rate	1×10^{-6}
	LR warmup steps	10 (constant after warmup)
	Weight decay	0.1
	Gradient clipping	1.0
	Entropy coefficient	0
	FSDP strategy	FSDP2, `fsdp_size`=4/32 (A/B)
	Dynamic batch size	enabled
RL algorithm	Default estimator	GRPO
	Importance-ratio clip (low)	ε_{low}=0.2
	Importance-ratio clip (high)	$\varepsilon_{\text{high}}$=0.28
	Dual-clip threshold	ε_c=10.0
	KL-in-reward	disabled (β_{KL}=0)
	KL-in-loss	as set by preset (GRPO default)
	Group size n	8 / 16 (Config A / B)
	PPO mini-batch (prompts)	32 / 16 (Config A / B)
SIRAS shaping	Soft-divergence exponent γ	3
	Advantage amplification cap M	5.0
	Blend coefficient α	0.8
	Chunk representation $\phi(c)$	token embedding mean-pooling (L_2-norm)
Sequence / generation	Max prompt length	2 048 tokens
	Max response length	8 192 tokens
	Overlong buffer	4 096 tokens, penalty factor 1.0
	Rollout temperature	1.0
	Rollout top-p / top-k	1.0 / -1 (no truncation)
	Validation top-p	0.7 (temperature 1.0, n=1)
Asynchronous training	Rollout engine	vLLM, async mode
	Staleness threshold	0.1
	Parameter-sync trigger	every 4 / 2 steps (A / B)
	Required batches	4 / 2 (A / B)
	Partial rollout	disabled
Training budget	Epochs	1
	Validation frequency	every step
	Seeds	3 per reported cell (between-seed std only)

Effect of rollout count n. Table 5 reports the AIME26 numbers underlying Figure 3 (right). At small n, the GRPO baseline has too few siblings for the group-mean to be a low-variance estimator; SIRAS reshapes the available signal more efficiently and the gap is largest there. At n=32 the gap narrows.

E.2 Degenerate-group distribution across benchmarks

We track the fraction of all-correct and all-wrong groups across training. The behaviour depends sharply on prompt difficulty and rollout count n (Table 6, Fig. 4). On AIME26 prompts at n=8 the GRPO baseline is dominated by all-wrong groups (degenerate ratio 78% at the first logged step, mean 57% across the run, max 87%) while on Multi-LogiEval prompts the same configuration sees the all-correct side dominate (mean 17%, rising over training). Halving n amplifies degeneracy (n=4 AIME26 mean 75%); doubling it suppresses it (n=32 AIME26 mean 11%). The pattern is not limited to difficult prompts: on GSM8K—normally considered saturated—DAPO and GRPO still log a mean all-correct fraction of 62% and 71% respectively at n=8 (bottom block of Table 6), which DAPO explicitly filters out; on BBH the rate is a steadier 23–25%. DAPO discards every such group; SIRAS reroutes them through the degenerate-group signals of §4.5. The all-correct vs. all-wrong split (last two columns of Table 6) confirms the structural claim of Fig. 1 (b): the sub-trajectory rescue path is exercised on a non-trivial fraction of every batch on AIME-difficulty prompts. Disabling

Table 4: Effect of backbone (pass@1, %, ± between-seed std).

Backbone	Method	MultiLogi	AIME25	AIME26
Qwen3-14B	GRPO	81.76 ±0.29	16.67 ±3.33	20.00 ±3.33
	GRPO+SIRAS	**82.65** ±0.29	**30.00** ±3.33	**35.56** ±1.92
Qwen3-32B	GRPO	82.65 ±0.29	26.67 ±3.33	33.33 ±3.33
	GRPO+SIRAS	**83.63** ±0.17	**33.33** ±3.33	**40.00** ±3.33
GPT-OSS-20B	GRPO	80.29 ±0.29	13.33 ±3.33	13.33 ±3.33
	GRPO+SIRAS	**80.88** ±0.29	**27.78** ±1.92	**36.67** ±3.33
Ling-Flash-2.0	GRPO	74.41 ±0.29	3.33 ±3.33	10.00 ±3.33
	GRPO+SIRAS	**77.25** ±0.17	**6.67** ±3.33	**12.22** ±1.92

Table 5: AIME26 pass@1 (%) vs. rollout count n on Qwen3-8B.

n	4	8	16	32
GRPO	16.67 ±3.33	16.67 ±3.33	20.00 ±3.33	23.33 ±3.33
GRPO+SIRAS	**22.22** ±1.92	**23.33** ±3.33	**27.78** ±1.92	**26.67** ±3.33
Δ	+5.6	+6.7	+7.8	+3.3

the degenerate-group residual reduces the AIME25 gain of GRPO+SIRAS from $+12.2$ to $+6.7$, attributing roughly half of the AIME-difficulty gain to rescued groups.

Table 6: Degenerate-group fraction (Qwen3-8B, 20 paired runs). *First*/*Last* are the first and last full-batch logged steps; *Mean*/*Max* are over the trajectory. *AC* and *AW* report the mean ratio of all-correct and all-wrong groups, respectively (degenerate $=$ AC $+$ AW). Higher n collapses degeneracy; harder prompts (AIME26) shift mass to AW. The bottom two blocks pair the project's drop-in DAPO baseline with plain GRPO on easier prompts; degeneracy there comes almost entirely from all-correct groups ($\approx$ 62–71% on GSM8K, $\approx$ 23–25% on BBH), which DAPO explicitly discards.

Prompts	Method	First	Last	Mean	Max	AC	AW
AIME26	GRPO n=4	81.2%	67.2%	74.7%	95.3%	0.0%	74.7%
	GRPO+SIRAS n=4	87.5%	53.1%	69.3%	92.2%	0.0%	69.3%
	GRPO n=8	78.1%	29.7%	57.3%	87.5%	0.0%	57.3%
	GRPO+SIRAS n=8	78.1%	32.8%	51.6%	85.9%	0.0%	51.6%
	GRPO n=32	35.9%	0.0%	10.7%	39.1%	0.0%	10.7%
	GRPO+SIRAS n=32	31.2%	0.0%	9.6%	48.4%	0.0%	9.6%
Multi-LogiEval	GRPO n=4	21.9%	35.9%	40.3%	60.9%	40.1%	0.2%
	GRPO+SIRAS n=4	25.0%	28.1%	40.0%	79.7%	39.8%	0.2%
	GRPO n=8	3.1%	18.8%	16.8%	32.8%	16.8%	0.0%
	GRPO+SIRAS n=8	7.8%	10.9%	19.1%	40.6%	19.1%	0.0%
	GRPO n=32	0.0%	0.0%	1.1%	7.8%	1.1%	0.0%
	GRPO+SIRAS n=32	0.0%	0.0%	0.5%	6.2%	0.5%	0.0%
GSM8K	DAPO n=8	60.4%	74.2%	62.2%	77.1%	62.2%	0.0%
	DAPO+SIRAS n=8	61.9%	75.2%	62.7%	76.0%	62.7%	0.0%
	GRPO n=8	58.4%	79.5%	71.1%	86.7%	71.1%	0.0%
	GRPO+SIRAS n=8	58.2%	79.7%	70.6%	85.9%	70.6%	0.0%
BBH	DAPO n=8	21.7%	28.3%	23.0%	28.3%	22.9%	0.0%
	DAPO+SIRAS n=8	22.7%	27.1%	21.3%	27.1%	21.2%	0.1%
	GRPO n=8	24.8%	37.3%	24.8%	37.3%	24.7%	0.1%
	GRPO+SIRAS n=8	24.6%	20.1%	21.7%	27.0%	21.6%	0.1%

E.3 Per-question paired significance on AIME25/26

Each AIME competition has only 30 items, so per-cell tests are inherently low-powered. We average the three seeds of each (baseline, baseline+SIRAS) cell into a per-question pass rate $p_q \in \{0, 1/3, 2/3, 1\}$ and pair on the question (n=30, clustering seeds rather than treating them as 90 independent trials). We report a cluster-bootstrap 95% CI for Δpass@1 (10,000 resamples), a

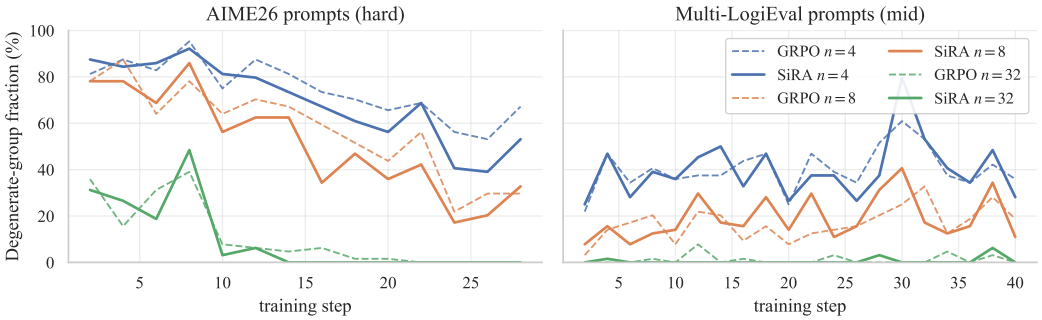

Figure 4: **Degenerate-group fraction over training** for the paired runs of Table 6 (Qwen3-8B; dashed: GRPO, solid: GRPO+SIRAS). Trajectory shape is set by the dominant degenerate type: AIME26 prompts (left) are all-wrong dominated and the fraction decays as the policy learns to occasionally solve them; Multi-LogiEval prompts (right) are all-correct dominated and the fraction grows as the policy concentrates on easy prompts. In both regimes the SIRAS curve tracks the GRPO curve, confirming that SIRAS's reshape neither inflates nor suppresses the rate at which sparse-reward groups appear—it only changes whether they contribute gradient.

one-sided exact sign test on discordant question counts (q^+, q^-), and a one-sided exact Wilcoxon signed-rank test (alternative `greater`).

Table 7: **Question-level paired significance on AIME25/26.** For each baseline / +SIRAS pair we average the three seeds into a per-question pass rate ($p_q \in \{0, 1/3, 2/3, 1\}$) and pair the $n = 30$ per-question rates. We report Δpass@1 (mean of question-level deltas) with cluster-bootstrap 95% CI (10,000 resamples of questions); discordant question counts (q^+, q^-) where q^+ counts questions on which +SIRAS has the higher pass rate and q^- the converse; the one-sided exact sign-test p-value on the discordant subset; and the one-sided exact Wilcoxon signed-rank p-value on the per-question deltas. $n = 30$ pairs the test to the question (cluster) rather than the (seed, question) cell, so seeds are not double-counted.

Pair	Bench	n	Δpass@1	95% CI	(q^+, q^-)	p_{sign}	p_{Wilcoxon}
GRPO + SIRAS	AIME25	30	+12.2	[+5.6, +20.0]	(9, 0)	0.002	0.002
	AIME26	30	+7.8	[+1.1, +14.4]	(7, 1)	0.04	0.03
Dr.GRPO + SIRAS	AIME25	30	+5.6	[+1.1, +10.0]	(5, 0)	0.03	0.03
	AIME26	30	+5.6	[+1.1, +10.0]	(5, 0)	0.03	0.03
DAPO + SIRAS	AIME25	30	+6.7	[+1.1, +12.2]	(7, 1)	0.04	0.02
	AIME26	30	+2.2	[−4.4, +10.0]	(3, 3)	0.66	0.34
Qwen3-14B	AIME25	30	+13.3	[+6.7, +22.2]	(10, 0)	$<10^{-3}$	$<10^{-3}$
	AIME26	30	+15.6	[+7.8, +23.3]	(13, 1)	$<10^{-3}$	0.004
Qwen3-32B	AIME25	30	+6.7	[+0.0, +14.4]	(7, 2)	0.09	0.05
	AIME26	30	+6.7	[+0.0, +14.4]	(5, 1)	0.11	0.05
GPT-OSS-20B	AIME25	30	+14.4	[+6.7, +23.3]	(10, 0)	$<10^{-3}$	$<10^{-3}$
	AIME26	30	+23.3	[+13.3, +33.3]	(14, 0)	$<10^{-3}$	$<10^{-3}$
Ling-Flash-2.0	AIME25	30	+3.3	[−1.1, +7.8]	(4, 1)	0.19	0.16
	AIME26	30	+2.2	[−2.2, +6.7]	(3, 1)	0.31	0.19

Cross-cell consistency on AIME-level benchmarks. Across the 14 AIME25/AIME26 cells in Tables 1 and 4 that pair a baseline with its SIRAS counterpart, SIRAS improves pass@1 in 14/14 cells (median $\Delta = +6.7$ pp), and the question-level Wilcoxon p-value is below 0.05 in 11/14 cells (Table 7). A two-sided sign test on the cell-wise direction rejects the no-effect null at $p = 0.0001221$. Because each AIME competition contributes only 30 items, no single cell is decisive on its own; the consistency-of-direction argument is what carries the AIME claim, and we therefore report the cluster-bootstrap, sign, and Wilcoxon tests in Table 7 alongside the headline numbers in Tables 1–4.

11 of the 14 AIME cells clear $p \leq 0.05$ on the Wilcoxon test, including every Qwen3-8B GRPO row, the Qwen3-14B and Qwen3-32B AIME cells, and both GPT-OSS-20B cells (GPT-OSS-20B AIME26 reaches $p \approx 10^{-4}$). The three remaining cells are the smallest lifts (Ling-Flash AIME25/26 at $+3.3$/$+2.2$ and DAPO+SIRAS AIME26 at $+2.2$), which a 30-item benchmark cannot resolve. We therefore treat the $14/14$ positive direction across cells ($p \approx 1.2\times10^{-4}$, two-sided sign test) as the primary AIME-level evidence, and report per-cell tests so the reader can separate per-cell signal from the aggregate trend.

E.4 Compute overhead

Table 8 reports per-step wall-clock of SIRAS on Qwen3-8B (medians over the full trajectory, truncated final step excluded). Two stages dominate `total_ms` in every row: `batch_featurize` and `soft_div`. Their mix depends on the regime: on long-response prompts the two stages split the cost roughly evenly (AIME26 48/47%) or featurization dominates when rollouts are relatively short (Multi-Logi 73/22%); on short-response prompts, `soft_div` is the larger stage ($\approx$ 48–52%) while `build_tree` grows to 14–19% as diversity in short rollouts produces more cluster splits (rolled into `other`). The reference implementation overlaps the featurization stage with rollout generation in the `fully_async` runtime, so end-to-end training throughput is less sensitive than per-step overhead suggests.

Table 8: Per-step compute overhead of SIRAS on Qwen3-8B. Upper block: long-response prompts (8 K cap). Lower block: short-response prompts (median rollout $\sim$1–2 K tokens). `base` / `+SIRAS` are total wall-clock per training step (s); Δ is the relative overhead. Stage-share columns are percentages of `total_ms`: "featurize" = chunk feature construction; "soft_div" = JSD-gated soft-divergence weights; "other" = pairwise similarity, clustering, and bookkeeping; "total (s)" is the isolated CPU operator time of SIRAS, whereas the wall-clock step gap reflects end-to-end distributed sync and pipeline overhead.

	Wall-clock (s/step)			SIRAS stage share (%)			Overhead
Prompts	base	+SIRAS	Δ	featurize	soft_div	other	total (s)
AIME26	51.4	63.7	+23.8%	48.0	47.3	4.7	4.6
Multi-Logi	23.9	29.3	+22.7%	73.1	21.9	5.0	1.5
GSM8K	340.4	364.7	+7.1%	34.0	52.2	13.8	10.5
BBH	499.1	573.3	+14.9%	33.1	48.0	18.8	16.6

F Credit-Assignment Alignment with Human-Verified Annotations: A Non-Independent Sanity Check

The experiments in §5.2 and App. E measure SIRAS by its *downstream* effect on benchmark accuracy. That leaves open a more direct question: when SIRAS says chunk j of rollout i should receive more (or less) gradient than its siblings, does a human reader who reads the same rollout agree that chunk j is where the answer was won or lost? This appendix provides a *non-independent consistency sanity check* on an offline rollout corpus with per-word and per-step human-verified annotations – produced by a strong language model as a first-pass labeler and then verified item by item by the authors – and reports the signal separately for mixed, all-correct, and all-wrong groups: the three regimes that SIRAS treats differently by construction (§4, §4.5).

Corpus. We take the GRPO rollout dump [mytestdpo, 2025] data and score each response with the same boxed-answer verifier used in training. Groups are classified as all-correct, all-wrong, or mixed under a correctness threshold of 0.5. The resulting corpus has 1,517 non-trivial groups (337 all-correct, 534 all-wrong, 646 mixed), giving a training-realistic mix of degenerate and informative groups.

Human-verified credit annotations and protocol disclosure. From this corpus we draw a stratified sample of 30 groups (22 mixed, 5 all-correct, 3 all-wrong; seed 42), covering $30\times16 = 480$ responses. Annotations were produced in two passes. First, a frontier large language model (under high reasoning effort), used as an external reader, produces draft step- and token-level labels for every response; the

verifier reward and ground-truth answer were visible to this first-pass labeler so it could trace where the reasoning goes right or wrong, but the predicted SIRAS credit signal was strictly withheld during the entire annotation process to prevent information leakage. Second, the authors verified every response item by item, editing any step-level criticality κ_s or per-token importance w_t entry they disagreed with (and the accompanying free-text justification where relevant) before the annotations were finalized. We retain both the first-pass draft and the authors' verified version for every response, and the verified version is the one used below and in Table 9.

Annotation protocol limitations. We explicitly disclose several structural limitations of this protocol:

- *Non-independent verification*: The reference labels were generated through an LLM first pass and subsequently audited and adjusted by the authors. Although the annotation schema and reference credit formula were fixed before running SIRAS, the verifying authors designed the method and were familiar with its underlying hypotheses.
- *No independent annotators*: No independent third-party annotators were involved in the labeling pipeline; consequently, no inter-annotator agreement (IAA) metrics are reported.
- *Correlational evidence*: The resulting correlations serve strictly as correlational evidence for a sanity check, rather than a causal validation or formal proof of credit assignment optimality.

Each response is annotated at two granularities:

- *step-level*: a reasoning-step segmentation and, for each step, a correctness score $c_s \in [0, 1]$ and a criticality score $\kappa_s \in [0, 1]$ ("how load-bearing is this step for the final answer"), with a short free-text justification;
- *word-level*: per-token importance $w_t \in [0, 1]$ (content tokens: numbers, identifiers, operators, boxed answers $\rightarrow$ high; function words, connectives $\rightarrow$ low).

These are composed into a reference per-token credit $\rho_t = \kappa_{s(t)} \cdot w_t$, normalized so that the mean over a response is 1 and individual entries are capped at 10. Annotations are stored as a Parquet file of 480 rows; consolidation, length/range validation, and the word-credit formula are fixed in code prior to running SIRAS, so the reference is independent of the method under test.

Predicted credit. We score each of the same 480 responses with SIRAS's sibling-relative chunk-level credit (the exact signal fed to the GRPO loss in Eq. (11)), rolled out to the token level by the per-chunk broadcast of §4. For degenerate groups the degenerate-residual path of §4.5 is active. We correlate $|A_{i,t}|$ against ρ_t per response (Pearson), report the Spearman rank correlation as a non-parametric check, and additionally aggregate both series to the annotated step segmentation to obtain a step-level correlation.

Table 9: Alignment between SIRAS's sibling-relative credit and human-verified credit annotations on 480 rollouts. Correlations are per-response Pearson / Spearman against the reference credit ρ_t; the step-level column aggregates both series to the annotated step segmentation before correlating. "% positive" is the fraction of responses with positive word-level Pearson. The *Random* row reports the mean over 100 random permutations of the credit signal within each response (null baseline).

Group type	n	Word Pearson	Word Spearman	Step Pearson	% positive
Mixed	352	0.744	0.775	0.591	100.0
All-correct	80	0.731	0.715	0.161	100.0
All-wrong	48	0.725	0.734	0.597	100.0
All (pooled)	480	**0.740**	**0.761**	0.524	**100.0**
Random (permutation)	480	0.002	0.001	−0.008	51.0

Findings. Table 9 summarizes the result. Three observations are worth flagging:

(i) Sibling-relative credit tracks the human-verified credit on every response. The pooled word-level Pearson is 0.740 (median 0.766, std 0.110) and Spearman is 0.761 (median 0.800); no response has a negatively-correlated credit signal. A random-permutation null (shuffling the credit vector within each response, averaged over 100 seeds) yields Pearson 0.002 and 51% positive, confirming that the

alignment is not an artifact of marginal statistics. With response lengths in the hundreds of tokens this does not certify each individual cell as significant, but it indicates that SIRAS does not concentrate the bulk of a response's gradient on regions the verifying authors marked as uninformative.

(ii) The alignment is stable across the three regimes SIRAS *treats differently.* Word-level Pearson varies only between 0.725 (all-wrong) and 0.744 (mixed), a 0.02 spread. This matters because the all-correct and all-wrong rows use the degenerate-group residual of §4.5 – a completely different computational path from the mixed-group mean-match identity – yet produce human-aligned credit of comparable quality. This is consistent (not by itself sufficient evidence) with the $+5.6$ AIME25 component attributed to rescued groups in App. E reflecting load-bearing tokens at the response level rather than an aggregate artifact.

(iii) Step-level aggregation is the weakest axis, and the weakness concentrates on all-correct groups. Step-level Pearson is 0.524 pooled (0.591 mixed, 0.597 all-wrong, 0.161 all-correct). The degradation on all-correct is expected: when every sibling reaches a correct answer, the informative variation is between-response (length, exploration, answer rarity; see §4.5) rather than within-response, and aggregating to steps collapses exactly the axis that carries signal. The mixed / all-wrong step correlations (≈ 0.59) are substantially above chance and are consistent with the token-level 0.74: the residual gap is the step-boundary mismatch between the human-verified numbered markdown steps and SIRAS's hybrid chunk boundaries (App. D), not a disagreement about which part of the response is load-bearing.

G Comparison Table: SIRAS vs. Related Credit-Assignment Approaches

Table 10 summarizes the structural differences relative to the train-time credit-assignment families discussed in §2. SIRAS is the only row that obtains a sub-trajectory signal *and* a degenerate-group signal without an extra network, extra annotations, extra rollouts, or inference overhead, while preserving the per-rollout token-averaged advantage by construction.

Table 10: **SIRAS compared to related credit-assignment approaches for RLVR.** Compared rows: GRPO [Shao et al., 2024] / DAPO [Yu et al., 2025]; PRM [Lightman et al., 2024, Wang et al., 2024]; VinePPO [Kazemnejad et al., 2024]; COMA [Foerster et al., 2018] / HCA [Harutyunyan et al., 2019]; and Tree-of-Thoughts / beam search, listed for contrast as test-time search. Columns group the credit signal a method produces, the additional cost it pays on top of the base GRPO pipeline, and two structural properties: whether degenerate groups (σ=0) contribute gradient, and whether the per-rollout token-averaged advantage is preserved.

	Credit structure		Added cost vs. GRPO			Structural	
Method	Granularity	Signal	Extra net	Step labels	Extra rollouts	σ=0 groups	GRPO identity
GRPO / DAPO	trajectory	group mean	✗	✗	✗	discarded	defines it
PRM	step	absolute (learned)	✓ (PRM)	✓	✗	scored	✗
VinePPO	step	counterfactual rollouts	✗	✗	✓ $\mathcal{O}(nKT)$	partial	✗
COMA / HCA	action	counterfactual posterior	✓ (critic / inv.)	✗	✗	—	✗
ToT / Beam (inference)	*test-time search*		✗	✗	✓ (at infer.)	—	—
SIRAS (ours)	**chunk**	**sibling-relative**	✗	✗	✗	**reclaimed**	✓ **(Constr. 1)**

H Datasets and preprocessing

Summary of datasets. Table 11 lists the six datasets evaluated in this paper; the training corpus used for the main experiments is described separately in App. D. All preprocessing is deterministic and uses a fixed seed (42) for any shuffling or splitting, so downstream row counts are reproducible. HuggingFace identifiers below are short-form; every dataset is accessible at `https://huggingface.co/datasets/<identifier>`, except for the `math` slice, which is hosted on GitHub at `https://github.com/openai/prm800k` (the PRM800K release of Lightman et al. [2024], from which we use the `math_splits/` train/test files).

Preprocessing pipeline. Every slice is materialized through a common contract: download the upstream corpus from its public release; apply task-specific shuffling and splitting with a fixed seed when the upstream release does not ship an explicit train/test split; emit one row per example

Table 11: Datasets used in this paper. "Slice" is the `data_source` string carried with each preprocessed example and consumed by the trainer's reward function. Row counts after preprocessing; licenses inherited from the upstream release. Identifiers without superscript resolve under `huggingface.co/datasets/`.

Slice	HuggingFace identifier	License	Train	Test	Citation
`gsm8k`	`openai/gsm8k`	MIT	7,473	1,319	[Cobbe et al., 2021]
`math`	`openai/prm800k`†	MIT	12,000	500	[Lightman et al., 2024, Hendrycks et al., 2021]
`aime25`	`gneubig/aime-1983-2024`	CC0-1.0	933	—	[Neubig, 2024]
	`yentinglin/aime_2025`	None	—	30	[Lin, 2025]
`aime26`	`gneubig/aime-1983-2024`	CC0-1.0	933	—	[Neubig, 2024]
	`MathArena/aime_2026`	CC-BY-NC-SA-4.0	—	30	[MathArena, 2026]
`bbh`	`lukaemon/bbh`	MIT	5,208	1,303	[Suzgun et al., 2023, Srivastava et al., 2023]
`multi-logi`	`Mihir3009/Multi-LogiEval`†	MIT	1,356	340	[Patel et al., 2024]

in the canonical schema $\{$`data_source`, `prompt`, `ability`, `reward_model`, `extra_info`$\}$, where `prompt` is a single-turn user message composed of the original question plus a task-specific instruction ("reason step by step using a numbered list", plus an answer-format directive such as $\boxed{\cdot}$ for math or "Answer: n" for AIME); and write the result as a single train/test parquet pair plus an example JSON for inspection. The `reward_model.ground_truth` field is the verifier target, and is the only place where the original upstream label leaks into training.

Licensing and ethical considerations. All upstream assets are distributed under permissive licenses, which cover both the academic redistribution of the preprocessed parquet files shipped with our code release and their use in RL training of open-weight models. AIME problem statements are copyright the Mathematical Association of America; we redistribute only the derived parquet files generated by our preprocessing pipeline from the official HuggingFace mirrors (`gneubig/aime-1983-2024`, `yentinglin/aime_2025`, `MathArena/aime_2026`), which are themselves released for research use, and we cite the MAA in all artifacts. None of the datasets above contain personally identifiable information or sensitive content; every slice is a math, logic, or broad reasoning corpus with machine-verifiable answers, so the human-subjects and PII questions of the NeurIPS checklist do not apply.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and §1 state our four contributions (sibling-relative formulation, the SIRAS pipeline, the gradient-deviation identity, calibrated degenerate-group residuals); the corresponding sections (§4, §4.4, §4.5, §5) provide matching technical content and empirical evidence on the six benchmarks discussed.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: A dedicated *Limitations* section at the end of the main text (§7) explicitly discusses four key limitations: (i) SIRAS preserves the per-rollout token mean but not the full policy-gradient expectation (with the gradient deviation characterized by Proposition 1); (ii) banded DTW assumes approximately monotone, left-to-right reasoning, leaving non-linear traces, general code, tool-use, and agentic settings as unverified open directions; (iii) the all-correct/all-wrong residuals are calibrated auxiliary objectives rather than verifier-derived rewards; and (iv) surface-to-token chunking is subject to potential fragility from the representational non-uniqueness of subword detokenization [Ayoobi et al., 2026].

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Construction 1 (token-mean preservation) and Proposition 1 (gradient-deviation identity) are stated in §4.4 with their assumptions (non-degenerate group; non-saturating clip). Full proofs, together with Lemma 1 and Corollary 1, are given in App. B; all theorems, equations, and references are numbered and cross-referenced.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The main paper specifies the full SIRAS pipeline (§4, App. A Algorithm 1), backbones, group sizes, and benchmarks (§5.1); App. D lists the two compute configurations and Table 3 the algorithmic, optimization, generation, and asynchrony hyperparameters; App. H describes datasets and the deterministic preprocessing pipeline (fixed seed 42); the SIRAS implementation and training scripts are available on GitHub at `https://github.com/VEraTimes/SiRAS`.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The SiRAS implementation and training scripts are available on GitHub at `https://github.com/VEraTimes/SiRAS`; all evaluation datasets are public (HuggingFace identifiers and the PRM800K MATH split are listed in App. H), and preprocessing is deterministic with a fixed seed.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: §5.1 reports the backbones, trainer (verl-style RLVR), response-length cap, AdamW with 10-step warmup and constant learning rate, group sizes, and benchmarks; App. D (Table 3) lists the algorithmic, optimization, generation, and asynchrony hyperparameters and the two compute configurations; App. H documents the data splits and preprocessing.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: As stated in §5.1, every benchmark number in the main table is pass@1 averaged over 3 seeds, and the parenthesized value is the between-seed (sample) standard deviation; this is the variability we make claims over. We do not assume normality and only report 1-sigma seed dispersion alongside means. In addition, because each AIME competition has only 30 items, App. E.3 (Table 7) reports per-question paired significance for the 14 AIME25/26 cells: a cluster-bootstrap 95% CI for Δpass@1 (10,000 resamples, seeds averaged per question and paired on the question), a one-sided exact sign test, and a one-sided exact Wilcoxon signed-rank test; $11/14$ cells clear $p < 0.05$ on Wilcoxon, and the cross-cell $14/14$ positive-direction sign test gives $p \approx 1.22\times10^{-4}$ as the aggregate AIME-level evidence.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: App. D specifies the hardware (homogeneous nodes with 8× NVIDIA Hopper GPUs, 141 GB HBM each, 1 TB host memory, 96-core CPU, 512 GB local disk) and the two GPU layouts (Config A: 24 rollout + 8 training GPUs; Config B: 32 rollout + 32 training GPUs); the wall-clock overhead of SIRAS relative to GRPO is reported in Table 8 (App. E, $\sim$20% end-to-end).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics. The work uses publicly released models and datasets, introduces an algorithmic change to the advantage estimator (no new data collection from human subjects), and we are aware of no aspect that conflicts with the Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: SIRAS is a methodological contribution to RLVR-style training of reasoning language models; it modifies only the per-token advantage and does not introduce new datasets, deployments, or capabilities beyond those of the underlying open-source backbones. Any societal impact is therefore mediated by the training data and base models, which are not changed by this work, and we see no direct path to additional negative applications beyond those already discussed for general-purpose reasoning LLMs.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We release only training-side code (the SIRAS advantage estimator) and use publicly available base models and benchmarks; we do not release new pretrained weights, scraped data, or other artifacts that would carry a high risk of misuse.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All backbones used—Qwen3-8B/14B/32B [Yang et al., 2025], GPT-OSS-20B [OpenAI, 2025], Ling-Flash-2.0-100B [Ling Team, 2025]—and the verl-style trainer [Sheng et al., 2025] are cited in §5.1. Datasets (GSM8K, MATH/PRM800K, AIME25/26, BBH, MultiLogi-Eval) are cited in §5.1 and listed with HuggingFace identifiers and source URLs in App. H; we use them under their original public-release terms.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The only new asset is the SIRAS implementation, which is available on GitHub at `https://github.com/VEraTimes/SiRAS` and documented alongside the code. The hyperparameters, datasets, and preprocessing scripts referenced from App. D and App. H are documented in the repository.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The credit-alignment study in App. F is the only human-in-the-loop component: the 30 groups (480 responses) are first labeled by a frontier language model as a first-pass labeler, and then verified line by line by the authors, who edit any step- or token-level entries they disagree with. No crowdsourcing platform was used and no compensation was paid. The annotation protocol, granularities, and instructions are documented in App. F.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with external human subjects. The credit-alignment study in App. F is performed by the authors themselves (verifying line by line the first-pass labels produced by a frontier language model) on offline rollout text, which does not require IRB review.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: LLMs are the *subject* of training in this work (the policies optimized by GRPO/Dr.GRPO/DAPO with SIRAS), and they are not used inside the core methodology of SIRAS: SIRAS does not rely on an external LLM evaluator, generative process reward model, or LLM-based labeler in its training loop. We do, however, use an LLM as a first-pass labeler for the auxiliary credit-alignment consistency check in App. F: a frontier language model produces draft step- and token-level labels for the $30 \times 16 = 480$ rollouts, and the authors then verify and correct every response line by line; the verified labels are the ones used in Table 9. This study is a downstream sanity check on the credit signal and is not part of the training pipeline or any benchmark number reported in §5.2; the verifier reward and ground-truth answer are visible to the first-pass labeler and to the verifying authors, while the SIRAS credit signal is withheld until annotations are finalized.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.