STRIDE: STEP-LEVEL TRAJECTORY REASONING INDUCED DIVERSITY IN EXPLORATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning with verifiable rewards (RLVR) rewards whether a final answer is correct but does not distinguish between different correct reasoning paths. As a result, training can concentrate probability on a small set of solutions, limiting the benefit of repeated sampling. Existing diversity objectives typically score complete responses and assign the same auxiliary credit to every token, even when only a few reasoning steps make a response distinct. We introduce STRIDE, a post-training method that assigns diversity credit at the level of individual reasoning steps. STRIDE represents each parsed step using a frozen reference model and measures its novelty through regularized reconstruction against the steps of the other eligible rollouts and the preceding steps of its own trajectory. Both correct and incorrect rollouts characterize previously explored directions, but auxiliary credit is assigned only to correct responses. We show that the log-transformed step scores exactly decompose a response's marginal contribution to a regularized log-determinant measure of group diversity. STRIDE converts these scores into bounded bonuses and applies them to the corresponding step tokens during GRPO, requiring neither step-level correctness labels nor additional rollouts. Across five mathematical reasoning benchmarks, STRIDE improves macro Pass@32 from 58.6% to 62.4% on Qwen3-1.7B and from 77.8% to 79.9% on Qwen3-4B-Instruct-2507, while maintaining comparable Pass@1. It also increases macro DistinctStrategies@32 from 0.82 to 0.97 and from 0.92 to 0.97, respectively.

1 INTRODUCTION

Chain-of-thought (CoT) reasoning has emerged as a powerful paradigm for eliciting multi-step reasoning in large language models. Instead of directly predicting a final answer, a model using CoT generates intermediate reasoning steps before concluding, which can lead to substantial gains on arithmetic, commonsense, and symbolic reasoning tasks (Wei et al., 2022; Kojima et al., 2022). Recent advances in reasoning have exploited post-training techniques to further improve CoT performance. For example, reinforcement learning with verifiable rewards (RLVR) fine-tunes models using automatically checkable outcome signals in domains like math and coding (Shao et al., 2024a; DeepSeek-AI, 2025). Likewise, verifier-based methods train verifiers to score entire solutions (Cobbe et al., 2021), and process supervision provides feedback at each intermediate step rather than only at the end (Uesato et al., 2022; Lightman et al., 2024). Together, these methods support reasoning performance by training or selecting explicit reasoning trajectories that lead to correct solutions.

However, focusing only on answer accuracy can collapse the diversity of reasoning paths. Many post-training objectives reward successful solutions without distinguishing meaningfully different strategies from minor surface variations. Consequently, optimization can concentrate probability mass on a small number of high-likelihood trajectories, making independently sampled solutions increasingly redundant. This collapse is especially problematic for reasoning tasks, where a single problem may admit multiple valid derivations. In supervised fine-tuning, the one-problem-one-solution (1P1S) paradigm provides only one "canonical" solution per problem and can push models toward a narrow set of reasoning paths, creating a diversity bottleneck (Ju et al., 2025). This homogenization can limit the effectiveness of sampling-based methods (e.g., self-consistency) and constrain exploration in further training. Consistent with these observations, recent studies of RLVR

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

report that improving single-sample accuracy (pass@1) can come at the expense of policy entropy and broader solution coverage (Liang et al., 2025; Yue et al., 2025). For example, Yue et al. (2025) find that, in the settings studied, RL fine-tuning largely reweights already-captured solutions rather than discovering new ones, and the base model can outperform the fine-tuned model on pass@k for larger k, indicating a narrowing of the solution distribution. Similarly, sparse outcome rewards can suppress trajectories that are mostly correct but contain a few errors, contributing to reduced rollout diversity and premature narrowing of exploration (Ren et al., 2026). More broadly, diversity loss has been observed in RLHF and other alignment processes, where reward optimization can sharpen the output distribution toward a few canonical responses (Kirk et al., 2024; Chen et al., 2025; Li et al., 2025a). Such concentration can shrink the range of correct derivations a model produces for a single problem, diminish the returns of self-consistency and pass@k sampling, and narrow exploration during further training. We therefore define *reasoning-trajectory diversity* as the ability of a model to generate multiple correct and meaningfully distinct reasoning trajectories for the same problem.

Existing efforts to promote diversity in language model outputs largely target output distributions or complete responses rather than assigning diversity credit to individual reasoning steps. For instance, GEM (Game-theoretic Entropy Maximization) preserves output diversity during supervised fine-tuning through a game-theoretic objective connected to entropy regularization (Li et al., 2025b). Preference-based approaches, such as DivPO (Diverse Preference Optimization) and deviation-weighted preference optimization, explicitly select or reweight high-quality responses that are rare or deviate from others to the same prompt (Lanchantin et al., 2025; Chung et al., 2025). While these preference-based methods alleviate output-level collapse, they assess diversity over complete responses, using signals such as model probability, lexical statistics, or embedding distances. Such response-level signals do not directly assign diversity credit to individual reasoning steps. Recent work also promotes token-level diversity on correct rollouts during RL training (Yao et al., 2026), but without directly contrasting the step-by-step structure of reasoning chains. Step-level metrics have additionally been used to curate diverse reasoning trajectories for supervised fine-tuning (Ju et al., 2025). These advances motivate assigning local diversity credit to reasoning steps of correct responses during RL while accounting jointly for redundancy across the full rollout group and within each reasoning prefix.

In this paper, we propose STRIDE (**S**tep-level **T**rajectory **R**easoning **I**nduced **D**iversity in **E**xploration), a post-training framework that promotes diversity among reasoning trajectories with verified-correct final answers. Unlike response-level diversity objectives, STRIDE treats each intermediate reasoning step as the unit of diversity. Specifically, we decompose each credited rollout's marginal contribution to a log-determinant diversity measure into contributions from its individual reasoning steps. After generation, we run a no-gradient forward pass through the frozen reference model and construct a normalized contextual representation for each parsed step by pooling reference-model hidden states over its original response-token span. We form a regularized Gram matrix over steps from all eligible rollouts with valid step spans, whether correct or incorrect. An ordered Cholesky chain-rule decomposition then expresses each credited rollout's log-determinant marginal as a sum of step-wise increments, conditioning each step on both **(i)** all steps from the other eligible rollouts in the comparison set and **(ii)** preceding steps in the same rollout. For training, we transform the corresponding regularized reconstruction scores through clipping and thresholding into bounded, nonnegative bonuses. These scores measure each step's representational novelty relative to the steps explored across the full rollout group and its own rollout prefix.

Our construction has three key properties. **(i)** The raw log-determinant increments sum exactly to the rollout-level log-determinant marginal. **(ii)** The transformed bonuses are added as local token-level auxiliary advantages over the token span of each reasoning step, alongside the outcome advantage. **(iii)** Auxiliary diversity credit is restricted to rollouts with verified-correct final answers. By assigning diversity credit at the step level only to correct responses, STRIDE encourages representational diversity across correctly answered reasoning chains while using incorrect chains to characterize already-explored directions and accounting for redundancy within each chain.

2 RELATED WORK

Diversity in RLVR. RLVR, commonly optimized with GRPO or DAPO (Shao et al., 2024b; Yu et al., 2025b), can improve Pass@1 while reducing policy entropy and limiting performance un-

der repeated sampling (Liang et al., 2025). Existing approaches encourage exploration through likelihood-based rewards, token-level entropy (Yao et al., 2025), or response-level diversity objectives (Hu et al., 2025). Response-level methods assign a single auxiliary score to an entire completion, even when only a small number of its reasoning steps distinguish it from the other responses. STRIDE instead measures novelty at the reasoning-step level. It uses steps from all eligible correct and incorrect rollouts to characterize the directions already explored, while assigning auxiliary diversity credit only to the corresponding steps of correct responses.

Step-level credit assignment. Token-level methods identify influential decisions through policy uncertainty. For example, Wang et al. (2025) show that a small number of high-entropy "forking" tokens can steer reasoning toward different paths. Other methods refine the assignment of correctness credit: InT trains on a correction to the first error in a failed trajectory (Yang et al., 2026), while SSVPO estimates the contribution of individual steps using sequential Shapley values over reordered reasoning chains (Li et al., 2026). STRIDE applies step-level credit assignment to a different objective. It decomposes a correct response's marginal contribution to a regularized log-determinant diversity measure into per-step terms and assigns each resulting bonus to the tokens of the corresponding step. This requires only final-answer supervision and no additional rollouts. We discuss additional work on diversity-preserving post-training, structured reasoning, and representation-based exploration in Appendix A.

3 METHOD

STRIDE augments outcome-based RLVR with step-level diversity credit. For each correct rollout, it scores a reasoning step against all steps in the other eligible rollouts, including incorrect ones, and against earlier steps in the same rollout. Incorrect rollouts therefore characterize explored directions but receive no auxiliary credit. A frozen reference model provides contextual step representations, and regularized reconstruction measures their conditional novelty. These scores exactly decompose the rollout's marginal contribution to a regularized log-determinant diversity objective. STRIDE converts them into bounded bonuses and adds each bonus to the advantage of the corresponding step tokens.

3.1 OUTCOME REWARDS AND GRPO ADVANTAGES

Let π_θ denote the trainable policy, π_{old} its rollout-generating snapshot, and π_{ref} the frozen initial model. The reference supplies the step representations and the policy regularization anchor. For a prompt x, sample $K \geq 2$ responses (rollouts) $y_i = (y_{i,1}, \ldots, y_{i,T_i})$ from $\pi_{\text{old}}(\cdot \mid x)$. A final-answer verifier assigns $r_i = r(x, y_i) \in \{0, 1\}$, with $r_i = 1$ when the answer is accepted as correct. The grading procedure is specified in Appendix B.3.

Following group relative policy optimization (GRPO) (Shao et al., 2024b), we convert these rewards into outcome advantages:

$$\bar{r} = \frac{1}{K}\sum_{i=1}^{K} r_i, \qquad s_r^2 = \frac{1}{K-1}\sum_{i=1}^{K}(r_i - \bar{r})^2, \qquad A_i^{\text{out}} = \frac{r_i - \bar{r}}{s_r + \epsilon_{\text{adv}}}. \tag{1}$$

Here s_r is the sample standard deviation and $\epsilon_{\text{adv}} > 0$ is a numerical stabilizer. Every generated token of response i receives the same A_i^{out}. Thus this signal distinguishes responses by their verified outcome, while providing uniform credit within a response. When every response in a group is correct, all outcome advantages are zero. STRIDE adds its step-specific signal after this normalization.

3.2 CONTEXTUAL STEP REPRESENTATIONS

Reasoning steps and comparison responses. The policy is prompted to generate consecutively numbered reasoning steps followed by a final answer. A deterministic parser identifies the step bodies $(s_{i,1}, \ldots, s_{i,M_i})$ and their disjoint token spans $\mathcal{T}_{i,j} \subseteq \{1, \ldots, T_i\}$ in the original sampled sequence. These spans contain step content; introductory text, step markers, and the final-answer region remain outside them.

Let $e_i = 1$ when response i is completed without length truncation and contains at least two nonempty, consecutively numbered steps with validly parsed and aligned spans in the original sampled token sequence; otherwise, let $e_i = 0$. We distinguish the reference set from the credit set:

$$\mathcal{R}_x = \{i : e_i = 1\}, \qquad \mathcal{C}_x = \{i : r_i = 1,\ e_i = 1\} \subseteq \mathcal{R}_x. \tag{2}$$

The reference set $\mathcal{R}_x$ determines the comparison geometry and contains every eligible response, including incorrect ones. The credit set $\mathcal{C}_x$ contains the eligible correct responses and determines which targets may receive the auxiliary bonus. Thus, an incorrect trajectory counts as an explored direction when novelty is measured, but is never reinforced by the diversity term. Restricting auxiliary credit to correct targets follows the principle of correctness-conditioned incentives (Hu et al., 2025). Auxiliary credit is active when $|\mathcal{R}_x| \geq 2$ and $|\mathcal{C}_x| \geq 1$. All sampled responses retain their outcome advantages, including those outside $\mathcal{C}_x$. Appendix B.3 gives the parsing and original-token alignment rules.

Encoding and pooling. For $i \in \mathcal{R}_x$, evaluate the frozen reference on the original prompt–response sequence and let $h^{\mathrm{ref}}_{i,t} \in \mathbb{R}^d$ denote its final-layer hidden state at token t. Each state is computed in its original causal context, so the pooled representation depends on the step and its preceding context. Mean pooling and stabilized L2 normalization give

$$\bar{h}_{i,j} = \frac{1}{|\mathcal{T}_{i,j}|} \sum_{t \in \mathcal{T}_{i,j}} h^{\mathrm{ref}}_{i,t}, \qquad \phi_{i,j} = \mathrm{Norm}(\bar{h}_{i,j}). \tag{3}$$

Using a frozen reference keeps the representation function fixed as the policy is updated; feature extraction is performed without gradients. The operator Norm maps vectors into the unit ball, giving $\|\phi_{i,j}\|_2 \leq 1$. Its two-pass implementation is specified in Appendix B.4.

3.3 Conditional Step Novelty

Comparison context. For step j of a credited response $i \in \mathcal{C}_x$, the comparison context contains all steps from the other responses in $\mathcal{R}_x$ and the preceding steps of response i:

$$\mathcal{B}_{i,j} = \{(k, \ell) : k \in \mathcal{R}_x \setminus \{i\},\ 1 \leq \ell \leq M_k\} \cup \{(i, \ell) : 1 \leq \ell < j\}. \tag{4}$$

Let $Z_{i,j} \in \mathbb{R}^{|\mathcal{B}_{i,j}| \times d}$ stack the corresponding step vectors. This joint comparison accounts for representational redundancy across alternative responses and within the target's reasoning prefix, without requiring step alignment across responses. Scores are computed after the response group has been generated and graded.

Regularized reconstruction. For a target vector $\phi \in \mathbb{R}^d$, comparison matrix $Z \in \mathbb{R}^{m \times d}$, and ridge $\lambda > 0$, define

$$c(\phi \mid Z) = \min_{a \in \mathbb{R}^m} \left\{ \|\phi - Z^\top a\|_2^2 + \lambda \|a\|_2^2 \right\}, \qquad c_{i,j} = c(\phi_{i,j} \mid Z_{i,j}). \tag{5}$$

The score measures the cost of reconstructing the target from the comparison vectors under a coefficient penalty. A larger cost indicates that representing the step from this context requires greater reconstruction error or larger coefficients. Its closed form connects this construction to the regularized elliptical geometry used in representation-based exploration (Tuyls et al., 2025).

Proposition 1 (Conditional reconstruction score)**.** *For a fixed target ϕ, comparison matrix Z, and $\lambda > 0$,*

$$\begin{aligned} c(\phi \mid Z) &= \phi^\top \phi - \phi^\top Z^\top (ZZ^\top + \lambda I_m)^{-1} Z\phi \\ &= \lambda \phi^\top (Z^\top Z + \lambda I_d)^{-1} \phi. \end{aligned} \tag{6}$$

Consequently, $0 \leq c(\phi \mid Z) \leq \|\phi\|_2^2$. Appending fixed comparison vectors to Z cannot increase the score of a fixed target.

The normalized step representations therefore give $c_{i,j} \in [0, 1]$. Monotonicity ensures that additional comparison vectors can account for directions already captured by the group or prefix without increasing a fixed target's score. Proofs appear in Appendix B.1.

Gram matrix and Cholesky computation. We now give a matrix computation of the score in Eq. equation 5. For a subset $\mathcal{S} \subseteq \mathcal{R}_x$, let $N_{\mathcal{S}} = \sum_{i \in \mathcal{S}} M_i$ and let $\Phi_{\mathcal{S}} \in \mathbb{R}^{N_{\mathcal{S}} \times d}$ stack its indexed step vectors as rows. Write $\Phi_x = \Phi_{\mathcal{R}_x}$ and $N_x = N_{\mathcal{R}_x}$. The Gram matrix and its regularized form are

$$G_x = \Phi_x \Phi_x^\top, \qquad K_x = G_x + \lambda I_{N_x}. \tag{7}$$

Entries of G_x are pairwise inner products between step representations; the ridge makes K_x positive definite.

For each target response $i \in \mathcal{C}_x$, let $P_i \in \mathbb{R}^{N_x \times N_x}$ be the permutation matrix that places the other reference responses' steps first and the target's steps last, preserving their original order. With $L^{(i)}$ the lower-triangular Cholesky factor with positive diagonal,

$$\begin{aligned} P_i K_x P_i^\top &= L^{(i)} L^{(i)\top}, \\ v_{i,j} := \left(L^{(i)}_{m,m}\right)^2 &= \lambda + c_{i,j}, \qquad m = N_x - M_i + j. \end{aligned} \tag{8}$$

Each target pivot conditions on precisely the vectors in $\mathcal{B}_{i,j}$. Its squared value is the corresponding Schur complement, and subtracting λ recovers the reconstruction score. One factorization per response yields all of its conditional step scores.

Ordered contribution of a rollout. To characterize the joint diversity of the step representations, define

$$\begin{aligned} F_\lambda(\mathcal{S}) &= \log\det\left(I_{N_{\mathcal{S}}} + \lambda^{-1} \Phi_{\mathcal{S}} \Phi_{\mathcal{S}}^\top\right), \\ \Delta_i &= F_\lambda(\mathcal{R}_x) - F_\lambda(\mathcal{R}_x \setminus \{i\}), \end{aligned} \tag{9}$$

with $F_\lambda(\varnothing) = 0$. A response contributes a block of step vectors, and Δ_i measures that block's marginal contribution.

Proposition 2 (Ordered decomposition of the response marginal)**.** *For fixed step representations and $\lambda > 0$, let $\delta_{i,j} = \log(v_{i,j}/\lambda)$. Then*

$$\Delta_i = \sum_{j=1}^{M_i} \delta_{i,j} = \sum_{j=1}^{M_i} \log\Big(1 + \frac{c_{i,j}}{\lambda}\Big). \tag{10}$$

For $\|\phi_{i,j}\|_2 \le 1$, each increment satisfies $0 \le \delta_{i,j} \le \log(1 + 1/\lambda)$.

This identity resolves the response's log-determinant contribution into increments conditioned on the comparison context in Eq. equation 4. STRIDE uses the bounded reconstruction scores $c_{i,j} = \lambda(\exp(\delta_{i,j}) - 1)$ to construct the local advantages below.

3.4 STEP-LEVEL DIVERSITY CREDIT

From scores to bonuses. The reconstruction form provides a $[0, 1]$ range for every ridge $\lambda > 0$, giving a common scale for the auxiliary signal. With a fixed ridge $0 < \lambda < 1$, define the bounded, nonnegative bonus

$$q_{i,j} = \mathrm{clip}(c_{i,j}, 0, 1), \qquad u_{i,j} = q_{i,j}\, \mathbf{1}[q_{i,j} > \lambda]. \tag{11}$$

Clipping enforces the analytical range numerically, and the threshold removes scores at or below the ridge scale. A unit step vector that duplicates a comparison vector has $c_{i,j} \le \lambda/(1 + \lambda) < \lambda$ and receives zero bonus. A unit vector orthogonal to every comparison vector has $c_{i,j} = 1$ and receives the maximal bonus. These cases are derived in Appendix B.1.

Token-level advantage. Assign the full bonus to each token in the corresponding step body:

$$A^{\mathrm{step}}_{i,t} = \begin{cases} u_{i,j}, & i \in \mathcal{C}_x,\ |\mathcal{R}_x| \ge 2,\ t \in \mathcal{T}_{i,j}, \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

The disjoint spans identify a unique step j for every credited token. The advantage used for policy optimization is

$$\boxed{A_{i,t} = A^{\mathrm{out}}_i + A^{\mathrm{step}}_{i,t}.} \tag{13}$$

The first term supplies correctness supervision to all generated tokens. The second provides step-specific diversity credit on the mapped step bodies, with $0 \le A^{\text{step}}_{i,t} \le 1$. Tokens outside these spans retain the outcome advantage alone.

The auxiliary term is added after outcome normalization and remains uncentered; the combined advantage is not normalized again. Consequently, different steps in the same response can receive different learning signals. When every response is correct, $A^{\text{out}}_i = 0$; steps with $u_{i,j} > 0$ in an active comparison group retain positive auxiliary advantages. To characterize local allocation, Proposition 3 in Appendix B.2 compares the policy gradient at the behavior policy with uniform credit that preserves each response's total auxiliary weight on the same sampled batch and mapped tokens.

Algorithm 1 STRIDE: step-level diversity credit for one policy update

Input: Prompt batch $\mathcal{D}_b$, policy π_θ, reference π_{ref}, group size K, and ridge λ.
Set $\pi_{\text{old}} \leftarrow \pi_\theta$.
For each prompt $x \in \mathcal{D}_b$:
  Sample K responses and retain their tokens and behavior log probabilities.
  Verify final answers and compute A^{out}_i using Eq. equation 1.
  Parse step spans, form $\mathcal{R}_x$ and $\mathcal{C}_x$, and initialize $A_{i,t} = A^{\text{out}}_i$ for every generated token.
  If $|\mathcal{R}_x| \ge 2$ and $|\mathcal{C}_x| \ge 1$:
    Compute the contextual step vectors for every response in $\mathcal{R}_x$ using Eq. equation 3.
    Form G_x and K_x using Eq. equation 7.
    For each $i \in \mathcal{C}_x$:
      Order the target response last and factor $P_i K_x P_i^\top = L^{(i)} L^{(i)\top}$.
      For each step j, obtain $c_{i,j} = (L^{(i)}_{m,m})^2 - \lambda$, with $m = N_x - M_i + j$.
      Compute $u_{i,j}$ and add $u_{i,j}$ to $A_{i,t}$ for $t \in \mathcal{T}_{i,j}$.
Hold computed advantages fixed and take one optimizer step with Eq. equation 15.

3.5 POLICY OPTIMIZATION

For a prompt batch $\mathcal{D}_b$, let $N_{\text{tok}} = \sum_{x \in \mathcal{D}_b} \sum_{i=1}^{K} T_i$ count all real generated tokens. Suppressing prompt indices, define

$$\rho_{i,t}(\theta) = \frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\text{old}}(y_{i,t} \mid x, y_{i,<t})}, \qquad d_{i,t}(\theta) = \log \frac{\pi_{\text{ref}}(y_{i,t} \mid x, y_{i,<t})}{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}. \tag{14}$$

Using the clipped policy surrogate (Schulman et al., 2017) and sampled fixed-reference regularization, minimize

$$\mathcal{L}(\theta) = \frac{1}{N_{\text{tok}}} \sum_{x \in \mathcal{D}_b} \sum_{i=1}^{K} \sum_{t=1}^{T_i} \Big[-\min\big\{ \rho_{i,t} A_{i,t}, \; \text{clip}(\rho_{i,t}, 1-\epsilon_{\text{clip}}, 1+\epsilon_{\text{clip}}) A_{i,t} \big\} + \beta \big(e^{d_{i,t}} - d_{i,t} - 1 \big) \Big], \tag{15}$$

where $\epsilon_{\text{clip}} > 0$ is the clipping parameter and $\beta \ge 0$ controls reference regularization. Both terms are averaged over all real generated tokens, including tokens outside credited step spans.

Computed advantages, behavior probabilities, and reference quantities are held fixed during the update. Gradients pass through the current policy in both terms of the objective. Each fresh rollout batch produces one optimizer update. Removing the step-level term recovers the shared correctness-only training objective. Algorithm 1 gives the complete procedure. Appendix B.4 specifies the sampled regularizer, numerical guards, hyperparameters, and computational cost.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETTINGS

Datasets. Our training data is a subset of the English DAPO-Math corpus (Yu et al., 2025b). We evaluate on five mathematical-reasoning benchmarks, AIME 2024, AIME 2025, MATH-500 (Lightman et al., 2024), Minerva Math (Lewkowycz et al., 2022), and OlympiadBench (He et al., 2024).

Evaluation. We sample 32 responses per problem and evaluate both correctness and reasoning diversity. We report Pass@1 as the mean correctness across all responses and Pass@32 as the fraction of problems with at least one correct response. Incorrect, truncated, and grader-error responses remain in the denominator. For diversity, we report the mean pairwise cosine distance between BGE-M3 response embeddings (Chen et al., 2024) and DistinctStrategies@32, the mean number of distinct correct mathematical approaches identified by an LLM judge (Zheng et al., 2023; Hu et al., 2025). Problems with no correct response contribute zero to DistinctStrategies@32. Complete metric definitions, judge instructions, and implementation details are provided in Appendix C.

Baselines and Training. We evaluate Qwen3-1.7B in explicit non-thinking mode and Qwen3-4B-Instruct-2507 (Yang et al., 2025). We compare STRIDE against established GRPO (Shao et al., 2024b), diversity-oriented baselines, GRPO-likelihood (He et al., 2025) and GRPO-entropy (Yao et al., 2025). All methods are trained on the same fixed set of 2,048 problems from `open-r1/DAPO-Math-17k-Processed`, using a global batch size of 64 prompts and eight rollouts per prompt. We train for 128 optimizer updates. Complete optimization, rollout and implementation details are provided in Appendix B.4.

4.2 Main Results

Table 1: Pass@1 (P@1) and Pass@32 (P@32) accuracy (%) on five mathematical reasoning benchmarks. Macro is the unweighted mean over the five benchmarks. **Bold** marks the best result within each model size. Δ rows give the difference between STRIDE and GRPO.

	AIME24		AIME25		MATH500		Minerva		Olympiad		Macro	
Method	P@1	P@32	P@1	P@32	P@1	P@32	P@1	P@32	P@1	P@32	P@1	P@32
1.7B model												
GRPO	10.8	40.0	8.5	33.3	**70.7**	93.8	29.0	54.0	**38.2**	72.0	31.44	58.62
GRPO-likelihood	11.4	43.3	8.9	33.3	70.5	93.6	28.8	55.5	37.6	71.8	31.44	59.50
GRPO-entropy	6.0	33.3	3.9	30.0	64.2	92.2	26.0	55.9	31.2	66.8	26.26	55.64
STRIDE	**12.1**	**50.0**	**9.8**	**36.7**	69.9	**94.6**	**29.7**	**58.5**	37.5	**72.4**	**31.80**	**62.44**
Δ vs. GRPO	+1.3	+10.0	+1.3	+3.4	−0.8	+0.8	+0.7	+4.5	−0.7	+0.4	+0.36	+3.82
4B model												
GRPO	52.8	80.0	**43.4**	70.0	92.6	97.8	44.3	58.5	64.7	82.8	**59.56**	77.82
GRPO-likelihood	51.1	80.0	41.6	**73.3**	**93.1**	97.6	44.8	57.7	**65.1**	82.6	59.14	78.24
GRPO-entropy	48.1	80.0	36.8	60.0	92.7	98.0	44.8	58.1	63.6	81.8	57.20	75.58
STRIDE	**53.1**	**83.3**	41.6	**73.3**	92.9	**98.4**	**45.2**	**60.7**	64.8	**83.6**	59.52	**79.86**
Δ vs. GRPO	+0.3	+3.3	−1.8	+3.3	+0.3	+0.6	+0.9	+2.2	+0.1	+0.8	−0.04	+2.04

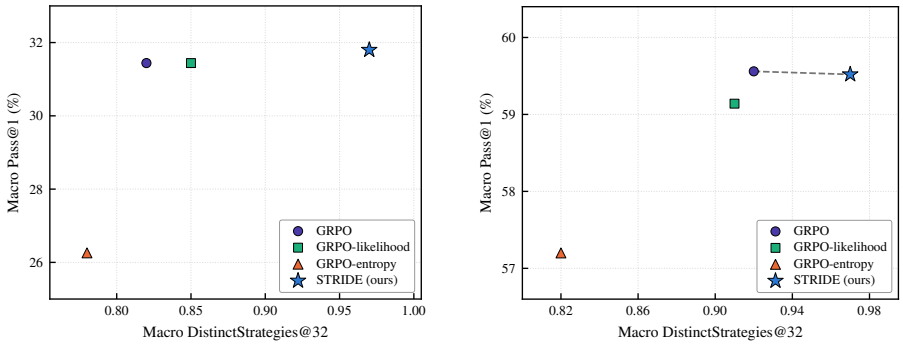
Reasoning quality. Table 1 compares STRIDE with GRPO and its likelihood- and entropy-based exploration variants across both model scales. STRIDE provides its clearest improvements under repeated sampling while preserving single-response accuracy. On the 1.7B model, STRIDE improves the macro Pass@32 score from 58.62 to 62.44, a gain of 3.82 points over GRPO, while also increasing macro Pass@1 by 0.36 points. The Pass@32 improvement is consistent across all five benchmarks and is particularly pronounced on AIME24 and Minerva, where STRIDE yields gains of 10.0 and 4.5 points, respectively. On the 4B model, STRIDE similarly obtains the highest macro Pass@32 score of 79.86, outperforming GRPO by 2.04 points and the strongest alternative baseline by 1.62 points. Meanwhile, its macro Pass@1 remains effectively unchanged relative to GRPO (-0.04 points), with gains on AIME24, MATH500, Minerva, and Olympiad offset by a decrease on AIME25. These results show that STRIDE primarily expands the set of successful reasoning trajectories available under repeated sampling without materially degrading the probability of obtaining a correct response in a single attempt.

Reasoning diversity. Table 2 examines whether the improvements under repeated sampling are accompanied by broader reasoning exploration. STRIDE consistently increases semantic diversity, outperforming GRPO on every benchmark at both model scales and improving the macro BGE-M3 distance by 1.09 points for the 1.7B model and 0.60 points for the 4B model. The strategy-level evaluation shows a similar aggregate improvement. At 1.7B, STRIDE increases the macro DistinctStrategies@32 score from 0.82 to 0.97 and obtains the highest score on four of the five benchmarks, with GRPO-likelihood performing best on AIME25. At 4B, STRIDE improves the macro strategy

Table 2: Diversity of sampled reasoning on five mathematical reasoning benchmarks. **Sem**: semantic diversity ($D_{\text{sem}} \times 100$), the mean pairwise BGE-M3 cosine distance among the 32 responses sampled per question. **S@32**: strategy diversity (DistinctStrategies@32), the mean number of distinct correct reasoning trajectories per question identified by a GPT-OSS-120B judge among the same responses. Higher is more diverse for both. Macro is the unweighted mean over the five benchmarks. **Bold** marks the best result within each model size. Δ rows give the difference between STRIDE and GRPO.

	AIME24		AIME25		MATH500		Minerva		Olympiad		Macro	
Method	Sem	S@32	Sem	S@32	Sem	S@32	Sem	S@32	Sem	S@32	Sem	S@32
1.7B model												
GRPO	9.50	0.53	8.93	0.27	6.45	1.47	8.07	0.75	7.54	1.06	8.10	0.82
GRPO-likelihood	9.55	0.47	8.92	**0.47**	6.59	1.49	8.15	0.75	7.58	1.06	8.16	0.85
GRPO-entropy	9.93	0.47	9.40	0.13	7.09	1.53	8.85	0.75	8.13	1.02	8.68	0.78
STRIDE	**11.00**	**1.00**	**10.36**	0.40	**7.22**	**1.56**	**8.89**	**0.81**	**8.49**	**1.09**	**9.19**	**0.97**
Δ vs. GRPO	+1.50	+0.47	+1.43	+0.13	+0.77	+0.09	+0.82	+0.06	+0.95	+0.03	+1.09	+0.16
4B model												
GRPO	7.80	1.00	7.70	0.67	4.85	1.21	6.25	0.72	6.12	**1.00**	6.54	0.92
GRPO-likelihood	7.77	0.93	7.65	**0.73**	4.81	**1.29**	6.25	0.68	6.02	0.94	6.50	0.91
GRPO-entropy	7.89	0.73	**7.80**	0.40	5.83	1.26	7.51	**0.75**	6.14	0.94	7.03	0.82
STRIDE	**7.99**	**1.23**	7.79	**0.73**	**5.93**	1.23	**7.84**	0.71	**6.16**	0.96	**7.14**	**0.97**
Δ vs. GRPO	+0.19	+0.23	+0.09	+0.07	+1.08	+0.02	+1.59	−0.01	+0.04	−0.04	+0.60	+0.05

score from 0.92 to 0.97. The benchmark-level results are more heterogeneous: STRIDE leads on AIME24 and ties GRPO-likelihood on AIME25, whereas GRPO-likelihood, GRPO-entropy, and GRPO obtain the highest scores on MATH500, Minerva, and Olympiad, respectively. Nevertheless, STRIDE achieves the strongest macro semantic and strategy diversity at both model scales. These results indicate that its gains are not limited to surface-level variation: STRIDE expands the semantic response space while increasing the number of distinct correct mathematical approaches produced under a fixed sampling budget.



(a) Qwen3-1.7B (b) Qwen3-4B-Instruct-2507

Figure 1: Quality–diversity trade-off. Quality is Macro Pass@1 (Table 1) and strategy diversity is Macro DistinctStrategies@32 (Table 2); both are better when higher. The dashed segment in (b) joins the Pareto-optimal methods. All values come from a single training seed.

Quality–diversity trade-off. Figure 1 presents the relationship between single-response accuracy and strategy diversity. On Qwen3-1.7B, STRIDE lies above and to the right of every baseline, attaining both the highest Macro Pass@1 and the highest Macro DistinctStrategies@32. GRPO-entropy illustrates the trade-off avoided by STRIDE, decreasing Pass@1 by 5.2 points relative to GRPO without improving strategy diversity. On Qwen3-4B-Instruct-2507, STRIDE achieves nearly the same Macro Pass@1 as GRPO (59.52% versus 59.56%) while obtaining higher strategy diversity (0.97 versus 0.92). These results show that STRIDE broadens reasoning exploration while maintaining comparable single-response accuracy.

4.3 Analysis of STRIDE Design Variants

We compare the final STRIDE configuration with two earlier design variants that use only correct responses to construct the novelty reference set. STRIDE instead uses all eligible responses to characterize previously explored directions while restricting auxiliary diversity credit to correct responses. The *token-uniform* variant retains step-wise novelty estimation but distributes the response-level auxiliary mass uniformly over the mapped tokens. The *single-step* variant represents each complete trajectory using one pooled vector and applies no localized step-level allocation. **Step** and **Local** indicate step-wise novelty estimation and localized credit assignment, respectively.

Table 3: Comparison of STRIDE design variants. All metrics are macro averages over the five mathematical reasoning benchmarks. **Bold** marks the best result within each model size.

Method	Step	Local	P@1	P@32	S@32
1.7B model					
Token-uniform	✓	×	31.13	61.72	0.83
Single-step	×	×	31.29	58.76	0.88
STRIDE	✓	✓	**31.80**	**62.44**	**0.97**
4B model					
Token-uniform	✓	×	58.89	77.30	0.93
Single-step	×	×	59.14	77.40	0.93
STRIDE	✓	✓	**59.52**	**79.86**	**0.97**

STRIDE achieves the best Macro P@1, Macro P@32, and S@32 at both model scales. On the 1.7B model, STRIDE improves Macro P@32 from 61.72 to 62.44 over Token-uniform and from 58.76 to 62.44 over Single-step, while increasing S@32 from 0.83 and 0.88 to 0.97, respectively. On the 4B model, STRIDE improves Macro P@32 by approximately 2.5 points over both variants and increases S@32 from 0.93 to 0.97. These results show that the final STRIDE configuration performs best among the evaluated designs. Because the variants also differ in their novelty reference sets, the comparison evaluates complete configurations rather than isolating the contribution of each component.

5 Conclusion

We introduced STRIDE, a step-level diversity credit assignment method for RLVR. STRIDE measures the novelty of each reasoning step relative to the steps explored by the other eligible rollouts and the preceding steps of its own trajectory. Correct and incorrect rollouts therefore characterize previously explored directions, while auxiliary credit is assigned only to responses with verified-correct final answers. The resulting bonuses are applied to the corresponding reasoning tokens rather than uniformly across an entire response. Across five mathematical reasoning benchmarks and two model scales, STRIDE improves macro Pass@32 while maintaining comparable Pass@1 and achieving the highest macro semantic and strategy diversity among the evaluated methods. The complete STRIDE configuration also outperforms the evaluated token-uniform and single-step alternatives. These results indicate that localized diversity credit can broaden the set of successful reasoning trajectories available under repeated sampling without materially reducing single-response accuracy.

Limitations. Our experiments are limited to two models from the Qwen family, a single training seed, mathematical reasoning benchmarks, and LoRA-based training. Furthermore, the strategy-level evaluation relies on an LLM judge and is conducted on a fixed subset of the evaluation benchmarks. Future work should evaluate STRIDE across additional seeds and model families, study the robustness of the strategy judge, test full-parameter training, and extend the evaluation to programming and broader reasoning benchmarks such as ARC-AGI-2 (Chollet et al., 2025) and GPQA (Rein et al., 2023).

AI Usage Disclosure

Large language models were used during manuscript preparation solely for language-related assistance, including proofreading, improving clarity and readability, and refining sentence structure. They were not used to develop the research ideas, methodology, experimental design, or scientific

conclusions presented in this work. All technical content, analyses, and research decisions were made by the authors, who take full responsibility for the contents of the manuscript.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we specify STRIDE's credit construction and policy objective in Section 3 and summarize the training procedure in Algorithm 1. Appendix B.1 provides complete proofs of Propositions 1 and 2, establishing the conditional reconstruction score, its Cholesky computation, and the ordered log-determinant decomposition. Appendix B.2 derives the effect of local credit allocation on the policy gradient and proves Proposition 3. The final-answer grading procedure, reasoning-step parser, and alignment of credited spans to original sampled tokens are detailed in Appendix B.3. Appendix B.4 specifies frozen-reference feature extraction, numerical safeguards, sampling and token normalization, LoRA configuration, optimization hyperparameters, training seed, and computational complexity. Appendix B.5 defines the controlled credit variants for examining allocation and representation granularity. The accompanying source package includes executable numerical checks and recorded results for the reconstruction and log-determinant identities, credit bounds and support, and gradient expressions. These checks cover randomized and degenerate cases, with the verification seed, numerical precision, and error tolerances recorded for replication. We anonymously release our code at `https://anonymous.4open.science/r/STRIDE-CODE-8C17/`.

ETHICS STATEMENT

We propose STRIDE, a method that augments outcome-based reinforcement learning with step-level diversity credit for LLM reasoning. Our experiments use publicly available language models and mathematical reasoning datasets. The method operates on model-generated reasoning traces and verifiable final answers, without requiring personal information or human annotations of intermediate steps. We encourage responsible use of the resulting models in accordance with the licenses and usage conditions of the underlying models and datasets. Applications beyond the evaluated mathematical reasoning setting should undergo appropriate validation and safety assessment.

REFERENCES

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 2318–2335, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.137. URL `https://aclanthology.org/2024.findings-acl.137/`.

Yilei Chen, Souradip Chakraborty, Lorenz Wolf, Yannis Paschalidis, and Aldo Pacchiano. Post-training large language models for diverse high-quality responses. *arXiv preprint arXiv:2509.04784*, 2025.

François Chollet, Mike Knoop, Gregory Kamradt, Bryan Landers, and Henry Pinkard. ARC-AGI-2: A new challenge for frontier AI reasoning systems. *CoRR*, abs/2505.11831, 2025. doi: 10.48550/ARXIV.2505.11831. URL `https://doi.org/10.48550/arXiv.2505.11831`.

John Joon Young Chung, Vishakh Padmakumar, Melissa Roemmele, Yuqian Sun, and Max Kreminski. Modifying large language model post-training for diverse creative writing. *arXiv preprint arXiv:2503.17126*, 2025.

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

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948, 2025. doi: 10.48550/ARXIV.2501.12948. URL `https://doi.org/10.48550/arXiv.2501.12948`.

Divij Handa, Mihir Parmar, Aswin RRV, Md Nayem Uddin, Hamid Palangi, and Chitta Baral. Guidedsampling: Steering llms towards diverse candidate solutions at inference-time. *CoRR*, abs/2510.03777, 2025. doi: 10.48550/ARXIV.2510.03777. URL https://doi.org/10.48550/arXiv.2510.03777.

Andre Wang He, Daniel Fried, and Sean Welleck. Rewarding the unlikely: Lifting GRPO beyond distribution sharpening. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pp. 25548–25560. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.EMNLP-MAIN.1298. URL https://doi.org/10.18653/v1/2025.emnlp-main.1298.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiadbench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 3828–3850. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.211. URL https://doi.org/10.18653/v1/2024.acl-long.211.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Edward J. Hu, Moksh Jain, Eric Elmoznino, Younesse Kaddar, Guillaume Lajoie, Yoshua Bengio, and Nikolay Malkin. Amortizing intractable inference in large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=Ouj6p4ca60.

Zican Hu, Shilin Zhang, Yafu Li, Jianhao Yan, Xuyang Hu, Leyang Cui, Xiaoye Qu, Chunlin Chen, Yu Cheng, and Zhi Wang. Diversity-incentivized exploration for versatile reasoning. *CoRR*, abs/2509.26209, 2025. doi: 10.48550/ARXIV.2509.26209. URL https://doi.org/10.48550/arXiv.2509.26209.

Feng Ju, Zeyu Qin, Rui Min, Zhitao He, Lingpeng Kong, and Yi R. Fung. Reasoning path divergence: A new metric and curation strategy to unlock LLM diverse thinking. *CoRR*, abs/2510.26122, 2025. doi: 10.48550/ARXIV.2510.26122. URL https://doi.org/10.48550/arXiv.2510.26122.

Doyoung Kim, Jaehyeok Doo, and Minjoon Seo. TSLM: tree-structured language modeling for divergent thinking. *CoRR*, abs/2601.22688, 2026. doi: 10.48550/ARXIV.2601.22688. URL https://doi.org/10.48550/arXiv.2601.22688.

Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. Understanding the effects of RLHF on LLM generalisation and diversity. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=PXD3FAVHJT.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/8bb0d291acd4acf06ef112099c16f326-Abstract-Conference.html.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model

serving with pagedattention. In Jason Flinn, Margo I. Seltzer, Peter Druschel, Antoine Kaufmann, and Jonathan Mace (eds.), *Proceedings of the 29th Symposium on Operating Systems Principles, SOSP 2023, Koblenz, Germany, October 23-26, 2023*, pp. 611–626. ACM, 2023. doi: 10.1145/3600006.3613165. URL https://doi.org/10.1145/3600006.3613165.

Jack Lanchantin, Angelica Chen, Shehzaad Dhuliawala, Ping Yu, Jason Weston, Sainbayar Sukhbaatar, and Ilia Kulikov. Diverse preference optimization. *arXiv preprint arXiv:2501.18101*, 2025.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay V. Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/18abbeef8cfe9203fdf9053c9c4fe191-Abstract-Conference.html.

Tianjian Li, Yiming Zhang, Ping Yu, Swarnadeep Saha, Daniel Khashabi, Jason Weston, Jack Lanchantin, and Tianlu Wang. Jointly reinforcing diversity and quality in language model generations. *CoRR*, abs/2509.02534, 2025a. doi: 10.48550/ARXIV.2509.02534. URL https://doi.org/10.48550/arXiv.2509.02534.

Yugu Li, Zehong Cao, Jianglin Qiao, and Siyi Hu. SSVPO: Effective step-level credit assignment for RL training of language models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=g33DGvnHYd.

Ziniu Li, Congliang Chen, Tian Xu, Zeyu Qin, Jiancong Xiao, Zhi-Quan Luo, and Ruoyu Sun. Preserving diversity in supervised fine-tuning of large language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025b. URL https://openreview.net/forum?id=NQEe7B7bSw.

Xiao Liang, Zhongzhi Li, Yeyun Gong, Yelong Shen, Ying Nian Wu, Zhijiang Guo, and Weizhu Chen. Beyond pass@1: Self-play with variational problem synthesis sustains RLVR. *CoRR*, abs/2508.14029, 2025. doi: 10.48550/ARXIV.2508.14029. URL https://doi.org/10.48550/arXiv.2508.14029.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. *CoRR*, abs/2311.12022, 2023. doi: 10.48550/ARXIV.2311.12022. URL https://doi.org/10.48550/arXiv.2311.12022.

Yanwei Ren, Haotian Zhang, Likang Xiao, Xikai Zhang, Jiaxing Huang, Jiayan Qiu, Baosheng Yu, Quan Chen, and Liu Liu. Recycling failures: Salvaging exploration in RLVR via fine-grained off-policy guidance. *CoRR*, abs/2602.24110, 2026. doi: 10.48550/ARXIV.2602.24110. URL https://doi.org/10.48550/arXiv.2602.24110.

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300, 2024a. doi: 10.48550/ARXIV.2402.03300. URL https://doi.org/10.48550/arXiv.2402.03300.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300, 2024b. doi: 10.48550/ARXIV.2402.03300. URL https://doi.org/10.48550/arXiv.2402.03300.

Stewart Slocum, Asher Parker-Sartori, and Dylan Hadfield-Menell. Diverse preference learning for capabilities and alignment. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=pOq9vDIYev.

Jens Tuyls, Dylan J. Foster, Akshay Krishnamurthy, and Jordan T. Ash. Representation-based exploration for language models: From test-time to post-training. *CoRR*, abs/2510.11686, 2025. doi: 10.48550/ARXIV.2510.11686. URL https://doi.org/10.48550/arXiv.2510.11686.

Jonathan Uesato, Nate Kushman, Ramana Kumar, H. Francis Song, Noah Y. Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback. *CoRR*, abs/2211.14275, 2022. doi: 10.48550/ARXIV.2211.14275. URL https://doi.org/10.48550/arXiv.2211.14275.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for LLM reasoning. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/a797c2d2e0c1fdabf4d1ab8cd0b465c6-Abstract-Conference.html.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

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

Matthew Y. R. Yang, Hao Bai, Ian Wu, Gene Yang, Amrith Setlur, and Aviral Kumar. Int: Self-proposed interventions enable credit assignment in LLM reasoning. *CoRR*, abs/2601.14209, 2026. doi: 10.48550/ARXIV.2601.14209. URL https://doi.org/10.48550/arXiv.2601.14209.

Jian Yao, Ran Cheng, Xingyu Wu, Jibin Wu, and KC Tan. Diversity-aware policy optimization for large language model reasoning. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/8808b4c72d5ade9bbcf9270ac6411314-Abstract-Conference.html.

Jian Yao, Ran Cheng, Xingyu Wu, Jibin Wu, and Kay Chen Tan. Diversity-aware policy optimization for large language model reasoning. *Advances in Neural Information Processing Systems*, 38: 94801–94826, 2026.

Fangxu Yu, Lai Jiang, Haoqiang Kang, Shibo Hao, and Lianhui Qin. Flow of reasoning: Training llms for divergent reasoning with minimal examples. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025a. URL https://proceedings.mlr.press/v267/yu25k.html.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: an open-source LLM reinforcement learning system at scale. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025b. URL http://papers.nips.cc/paper_files/paper/2025/hash/a4277440d50f1f15d2cb4c14f7e0c0d2-Abstract-Conference.html.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *CoRR*, abs/2504.13837, 2025. doi: 10.48550/ARXIV.2504.13837. URL https://doi.org/10.48550/arXiv.2504.13837.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

APPENDIX

A ADDITIONAL RELATED WORK

Diversity-preserving post-training. Loss of output diversity has been observed across several post-training paradigms. In supervised fine-tuning, Li et al. (2025b) attribute this effect to the cross-entropy objective and introduce GEM, an entropy-regularized alternative. In preference learning, Slocum et al. (2025) separate the entropy and cross-entropy contributions of the KL regularizer to preserve diversity while maintaining response quality. A similar contraction can occur in RLVR, where optimization may concentrate probability on a small set of successful trajectories. Liang et al. (2025) address this through online synthesis of variational training problems, while likelihood- and entropy-based objectives directly encourage the policy to retain broader support over its responses. These approaches modify the response distribution or its uncertainty. STRIDE instead determines which reasoning steps contribute new representational directions within a sampled rollout group.

Structured and inference-time exploration. Several methods change how candidate solutions are generated rather than how credit is assigned within a sampled response. GFlowNet fine-tuning learns to sample reasoning trajectories approximately in proportion to their reward, rather than concentrating on reward-maximizing modes (Hu et al., 2024; Yu et al., 2025a). TSLM trains language models on serialized search trees containing both successful and failed branches (Kim et al., 2026). GuidedSampling separates the selection of a high-level solution concept from the generation of the corresponding solution, motivated by the observation that independent samples frequently reuse the same approach (Handa et al., 2025). These methods introduce alternative generation or training structures. STRIDE retains the standard grouped-rollout procedure and constructs its auxiliary signal from the responses already sampled for the GRPO update.

Granularity of diversity and credit assignment. Diversity objectives differ in the unit at which they measure variation and assign credit. Response-level methods reward the textual, equational, or semantic diversity of complete correct responses (Hu et al., 2025). Their auxiliary signal is consequently shared by every token in a response, including tokens belonging to redundant reasoning steps. Token-level methods instead operate on uncertainty at individual generation decisions (Wang et al., 2025; Yao et al., 2025), but uncertainty does not necessarily indicate that the generated content introduces a distinct mathematical argument. STRIDE uses the intermediate granularity of explicitly parsed reasoning steps. It compares each target step against the steps of the other eligible responses and its own preceding steps, and maps the resulting novelty credit back to the original sampled-token span.

Step-level supervision. Process-supervision and credit-assignment methods seek to determine which intermediate steps contribute to the correctness of a solution. InT identifies the first error in an unsuccessful trajectory and trains the model to produce a local correction (Yang et al., 2026). SSVPO estimates step contributions using sequential Shapley values evaluated over reordered reasoning chains (Li et al., 2026). These methods improve the assignment of correctness credit. STRIDE addresses the complementary problem of diversity credit assignment: incorrect eligible trajectories help define what has already been explored, but the auxiliary diversity signal is applied only to steps belonging to responses with verified-correct final answers.

B STRIDE: Derivations and Implementation

B.1 Conditional Novelty and the Log-Determinant Identity

The derivations use exact arithmetic, fixed contextual representations, and a fixed ridge. Numerical guards are specified in Appendix B.4.

Proof of Proposition 1. Fix the target embedding ϕ and the comparison matrix Z. The quadratic objective in Eq. equation 5 is strictly convex in a because $ZZ^\top + \lambda I_m \succ 0$. Its unique minimizer is

$$a^\star = (ZZ^\top + \lambda I_m)^{-1} Z\phi. \tag{16}$$

Substituting this solution into the objective yields

$$c = \phi^\top \phi - \phi^\top Z^\top (ZZ^\top + \lambda I_m)^{-1} Z\phi.$$

Ordering the comparison vectors before the target gives the leading principal block of the regularized Gram matrix:

$$H = \begin{bmatrix} ZZ^\top + \lambda I_m & Z\phi \\ \phi^\top Z^\top & \|\phi\|_2^2 + \lambda \end{bmatrix}. \tag{17}$$

Its last squared Cholesky pivot is the Schur complement

$$v = \|\phi\|_2^2 + \lambda - \phi^\top Z^\top (ZZ^\top + \lambda I_m)^{-1} Z\phi = \lambda + c.$$

Later rows do not change a leading Cholesky block, so this is also the target pivot in Eq. equation 8. The regularized matrix inverse identity gives

$$v - \lambda = \phi^\top \left[I_d - Z^\top (ZZ^\top + \lambda I_m)^{-1} Z \right] \phi \tag{18}$$

$$= \lambda \phi^\top (Z^\top Z + \lambda I_d)^{-1} \phi. \tag{19}$$

Because $Z^\top Z \succeq 0$, the eigenvalues of $\lambda(Z^\top Z + \lambda I_d)^{-1}$ lie in $(0, 1]$. Therefore,

$$\begin{aligned} &0 \le c = v - \lambda \le \|\phi\|_2^2, \\ &0 \le \log(v/\lambda) \le \log\left(1 + \frac{\|\phi\|_2^2}{\lambda}\right). \end{aligned} \tag{20}$$

For normalized features, these bounds give $c \le 1$ and $\log(v/\lambda) \le \log(1 + 1/\lambda)$. The empty comparison set is covered by taking $c = \|\phi\|_2^2$.

For monotonicity, set $Q = Z^\top Z + \lambda I_d$ and append one comparison vector $z^\top$ to Z. The Sherman–Morrison identity gives

$$c(\phi \mid [Z; z^\top]) = c(\phi \mid Z) - \frac{\lambda(\phi^\top Q^{-1} z)^2}{1 + z^\top Q^{-1} z} \le c(\phi \mid Z). \tag{21}$$

Repeating this argument proves the claim for any appended collection. The vectors and ridge remain fixed throughout this comparison. □

Duplicate and orthogonal representations. Under the bonus construction's assumption $0 < \lambda < 1$, suppose $\|\phi\|_2 = 1$ and a row of Z equals $\phi^\top$. Restricting reconstruction to this single row gives the upper bound

$$c(\phi \mid Z) \le \min_{a \in \mathbb{R}} \left\{(1-a)^2 + \lambda a^2\right\} = \frac{\lambda}{1+\lambda} < \lambda. \tag{22}$$

Thus Eq. equation 11 assigns zero bonus to an exact feature duplicate in exact arithmetic. If instead $Z\phi = 0$, then Eq. equation 6 gives $c = \|\phi\|_2^2 = 1$ and $u = 1$.

Proof of Proposition 2. Fix the prompt, reference set $\mathcal{R}_x$, credit set $\mathcal{C}_x \subseteq \mathcal{R}_x$, and original contextual embeddings. For a response $i \in \mathcal{C}_x$, write $N = N_x$, and let K_{-i} denote the principal submatrix of K_x containing all steps from $\mathcal{R}_x \setminus \{i\}$. After the permutation in Eq. equation 8, its Cholesky factor is the leading $(N - M_i) \times (N - M_i)$ block of $L^{(i)}$. Subtracting the two log-determinants consequently yields

$$\log\det K_x - \log\det K_{-i} = \sum_{j=1}^{M_i} \log v_{i,j}. \tag{23}$$

For any response subset $\mathcal{S}$ with $N_\mathcal{S}$ indexed steps,

$$F_\lambda(\mathcal{S}) = \log\det\left(\Phi_\mathcal{S}\Phi_\mathcal{S}^\top + \lambda I_{N_\mathcal{S}}\right) - N_\mathcal{S}\log\lambda. \tag{24}$$

Subtracting the expression for $\mathcal{R}_x \setminus \{i\}$ from that for $\mathcal{R}_x$ in Eq. equation 23 gives

$$\begin{aligned} F_\lambda(\mathcal{R}_x) - F_\lambda(\mathcal{R}_x \setminus \{i\}) &= \sum_{j=1}^{M_i} \left(\log v_{i,j} - \log\lambda\right) \\ &= \sum_{j=1}^{M_i} \delta_{i,j}. \end{aligned} \tag{25}$$

All determinants use the same fixed ridge and original contextual embeddings. Removing a response selects the corresponding principal submatrix. □

From geometric scores to training credit. The reconstruction cost and logarithmic increment satisfy

$$c_{i,j} = \lambda\left(\exp(\delta_{i,j}) - 1\right).$$

STRIDE applies the clipping and threshold in Eq. equation 11 to $c_{i,j}$, then broadcasts the resulting bonus over the step's mapped tokens. Thus Eq. equation 10 describes the additive geometric contribution, while Eq. equation 12 defines the auxiliary advantage used by the policy objective. Both follow from the same conditional scores.

Support and sign of the auxiliary advantage. For $i \in \mathcal{C}_x$ with $|\mathcal{R}_x| \ge 2$, disjointness of the mapped spans and $0 \le u_{i,j} \le 1$ imply

$$0 \le A_{i,t}^{\text{step}} \le 1.$$

For responses outside $\mathcal{C}_x$ or groups with $|\mathcal{R}_x| < 2$, the auxiliary term is zero. Eligible incorrect responses remain in $\mathcal{R}_x$ and therefore affect the comparison geometry, but they receive no auxiliary diversity credit. Since a response with $r_i = 1$ has $A_i^{\text{out}} \ge 0$, adding the bonus preserves its nonnegative advantage. Incorrect responses retain their outcome advantages.

B.2 Local Allocation and the Policy Gradient

For $i \in \mathcal{C}_x$ with $|\mathcal{R}_x| \geq 2$, define the total auxiliary mass as

$$B_i := \sum_{t=1}^{T_i} A_{i,t}^{\text{step}} = \sum_{j=1}^{M_i} |\mathcal{T}_{i,j}| u_{i,j}. \tag{26}$$

Every mapped token receives its step's full bonus, so longer steps contribute proportionally more auxiliary mass at a fixed novelty score. The token-uniform variant preserves the total auxiliary weight in each response while redistributing it across the same mapped tokens. We now characterize how this changes the surrogate gradient.

Let

$$\mathcal{I} = \{(x, i) : x \in \mathcal{D}_b,\ i \in \mathcal{C}_x,\ |\mathcal{R}_x| \geq 2\}$$

index the active responses. Suppressing prompt indices, define step lengths, normalized weights, and mean novelty by

$$\begin{aligned} n_{i,j} &= |\mathcal{T}_{i,j}|, \quad n_i = \sum_j n_{i,j}, \\ w_{i,j} &= \frac{n_{i,j}}{n_i}, \quad \bar{u}_i = \sum_j w_{i,j} u_{i,j}. \end{aligned} \tag{27}$$

Define the mean token score vector for each step at the behavior parameters:

$$g_{i,j} = \frac{1}{n_{i,j}} \sum_{t \in \mathcal{T}_{i,j}} \nabla_\theta \log \pi_\theta(y_{i,t} \mid x, y_{i,<t}) \Big|_{\theta=\theta_{\text{old}}}. \tag{28}$$

Local credit assigns $u_{i,j}$ to each token of step j; matched uniform credit assigns $\bar{u}_i$ to each mapped token. Both have total auxiliary weight $B_i = n_i \bar{u}_i$.

Proposition 3 (Effect of local allocation at the behavior policy)**.** *Hold the sampled batch and computed credit tensors fixed, and suppose current and behavior token probabilities agree at $\theta = \theta_{\text{old}}$. Let $\mathcal{L}_{\text{local}}$ and $\mathcal{L}_{\text{uniform}}$ use Eq. equation 15 with the two allocations above and otherwise identical terms. Then*

$$\nabla_\theta \left(\mathcal{L}_{\text{local}} - \mathcal{L}_{\text{uniform}}\right)|_{\theta=\theta_{\text{old}}} = -\frac{1}{N_{\text{tok}}} \sum_{(x,i)\in\mathcal{I}} n_i \ \text{Cov}_{w_i}(u_{i,j}, g_{i,j}), \tag{29}$$

where the scalar–vector covariance is

$$\text{Cov}_{w_i}(u, g) = \sum_j w_{i,j}(u_{i,j} - \bar{u}_i) g_{i,j}.$$

Proof. At θ_{old}, every ratio is one and lies strictly inside the clipping interval. The surrogate derivative is therefore the usual likelihood-ratio derivative. The shared outcome and regularizer terms cancel. Since $n_{i,j} = n_i w_{i,j}$, Eq. equation 29 follows from the remaining gradient difference:

$$-\frac{1}{N_{\text{tok}}} \sum_{(x,i)\in\mathcal{I}} \sum_j n_{i,j}(u_{i,j} - \bar{u}_i) g_{i,j}. \tag{30}$$

□

The covariance identifies how local allocation changes the surrogate gradient at the behavior policy: it couples each step's relative novelty to the score vectors of its own tokens while preserving the response's total auxiliary weight.

B.3 Answer Verification and Step Alignment

Answer verification and auxiliary eligibility. Training uses integer-answer problems. The verifier first extracts an explicit terminal answer and then compares it with the reference integer. Length truncation, invalid thinking-region delimiters, failed answer extraction, and answer mismatch yield $r_i = 0$. Auxiliary eligibility is determined separately. A response has $e_i = 1$ only when it is complete and contains at least two nonempty, consecutively numbered steps with valid original-token alignments. Parsing or alignment failure sets $e_i = 0$ without changing the outcome reward. A correctly answered response may therefore receive $r_i = 1$ while receiving no auxiliary diversity credit. No separate formatting reward is used.

Step parsing and original-token alignment. A deterministic parser extracts the bodies marked by `Step 1:`, `Step 2:`, and subsequent consecutive markers. It excludes introductory text, step markers, fenced examples, and the terminal answer from the credited spans and does not infer steps from paragraph or newline boundaries. Each extracted character span is aligned with the decoded bytes of the original sampled token IDs without retokenizing the response. Auxiliary credit is assigned only when reconstruction and boundary checks produce an unambiguous token mapping. A token that overlaps both step content and non-whitespace content outside the step causes alignment to fail. Malformed steps and invalid mappings set $e_i = 0$ while leaving the outcome reward unchanged. The implementation imposes no maximum step count and provides no step-count reward or penalty.

B.4 Implementation of the Policy Update

Models and training setup. We train Qwen3-1.7B in explicit non-thinking mode and Qwen3-4B-Instruct-2507 (Yang et al., 2025). The policy is initialized from the corresponding pretrained checkpoint, whose original weights remain frozen, and only LoRA adapters (Hu et al., 2022) are trained. We use LoRA rank $r = 16$, scaling parameter $\alpha_{\mathrm{LoRA}} = 32$, corresponding to an effective scaling factor of 2, and zero dropout. The adapters are applied to the attention projections (`q_proj`, `k_proj`, `v_proj`, and `o_proj`) and the feed-forward projections (`gate_proj`, `up_proj`, and `down_proj`).

Training uses random seed 42 and 2,048 questions, with 64 prompts per update and $K = 8$ responses per prompt. Each model is trained for four epochs, corresponding to 128 optimizer updates and 65,536 sampled training responses per method. AdamW uses a learning rate of 2×10^{-5}, zero weight decay, $(\beta_1, \beta_2) = (0.9, 0.999)$, and $\epsilon_{\mathrm{Adam}} = 10^{-8}$. The gradient norm is clipped at 1. The learning rate increases linearly over the first ten optimizer updates and remains constant thereafter. We set $\beta = 0$, so the fixed-reference penalty in Eq. equation 15 is inactive in all reported runs and the frozen reference is used only to supply the step representations of Eq. equation 3. The step bonus enters the advantage with unit weight, as written in Eq. equation 12, and is active from the first optimizer update. Training runs on eight AMD MI210 GPUs, with four distributed learner processes and four isolated vLLM sampler processes (Kwon et al., 2023).

Reference and numerical computation. Disabling the adapters supplies both the hidden states in Eq. equation 3 and the reference log probabilities in Eq. equation 14. Mean pooling and kernel calculations use FP32 values. We define

$$\mathcal{N}(z) = \frac{z}{\max(\|z\|_2, \epsilon_{\mathrm{emb}})}, \qquad \epsilon_{\mathrm{emb}} = 10^{-12}.$$

The implementation applies $\mathcal{N}$ to each pooled step representation and reapplies it when assembling the kernel. Thus, the operator in Eq. equation 3 is

$$\mathrm{Norm}(z) = \mathcal{N}(\mathcal{N}(z)) .$$

The second normalization leaves unit-norm vectors unchanged in exact arithmetic and acts as a numerical safeguard during kernel construction. Schur complements are computed using Cholesky factorization without explicitly forming the inverses in Eq. equation 6. The ridge parameter and residual threshold are both fixed at 10^{-3}.

Advantage and surrogate settings. Outcome rewards are standardized within each prompt group using the sample standard deviation, with denominator $K-1$, and $\epsilon_{\mathrm{adv}} = 10^{-6}$ added to the standard deviation. The step advantage is added afterward, without renormalizing the combined advantage. The policy surrogate uses clipping width $\epsilon_{\mathrm{clip}} = 0.2$. The implementation applies a numerical guard to the log importance ratio:

$$\rho_{i,t}^{\mathrm{num}} = \exp(\mathrm{clip}\left(\log \pi_\theta - \log \pi_{\mathrm{old}},\, -20,\, 20\right)), \tag{31}$$

with both probabilities evaluated at the sampled token and its prefix. This guarded ratio is used in Eq. equation 15; it agrees with the usual ratio inside the guard interval. For completeness, the implementation provides the fixed-rollout reference penalty as the sampled k_3 surrogate $\exp(d) - d - 1$ (Shao et al., 2024b), evaluated at behavior-sampled tokens without importance correction; it contributes nothing to the reported runs because $\beta = 0$.

Sampling and normalization. In training, vLLM samples responses from the current adapter and records the chosen-token behavior log probabilities. Sampling uses temperature 1, unrestricted top-p and top-k sampling, and a total prompt–response context limit of $8{,}192$ tokens; the response budget is the remaining context after the prompt. The non-thinking chat-template setting still requests explicit numbered reasoning in the final response. EOS and truncated-response tokens are included in the real-token denominator; padding is excluded. Correctness and novelty are computed within each complete prompt group, whereas the policy and reference terms are normalized over all real completion tokens in the optimizer batch. The novelty kernel includes the parsed steps of every eligible response in $\mathcal{R}_x$, irrespective of answer correctness. Step bonuses are written only to mapped tokens of correct eligible targets in $\mathcal{C}_x$; incorrect responses contribute reference vectors but receive no auxiliary credit.

Evaluation. We evaluate on AIME24, AIME25, MATH500, Minerva, and OlympiadBench, with 30, 30, 500, 272, and 500 questions, respectively. For every question, we sample 32 responses in non-thinking mode with temperature 1.0, unrestricted top-p and top-k sampling, and the same $8{,}192$-token prompt–response context used in training. Final answers are graded with the verifier of Appendix B.3. Strategy diversity is judged by GPT-OSS-120B with the prompt of Figure 3, queried once per method and question at temperature 0, with each method judged in a separate call and method identities removed.

Computational overhead. STRIDE reuses the response group already sampled by GRPO and therefore requires no additional solution generations. Its additional computation consists of one no-gradient forward pass through the frozen reference model, followed by step-representation pooling and kernel computation. Let

$$N_x = \sum_{i \in \mathcal{R}_x} M_i$$

denote the total number of eligible reasoning steps for prompt x. Constructing the Gram matrix requires $O(N_x^2 d)$ time and $O(N_x^2)$ memory. In our direct implementation, the reordered kernel is factorized once for each credited response, resulting in $O(|\mathcal{C}_x| N_x^3)$ Cholesky cost per prompt. Since the rollout-group size is fixed at $K = 8$, at most eight factorizations are required for each prompt.

B.5 Design Variants

The variants below retain the same correctness signal and policy-update objective as STRIDE, changing how the auxiliary credit is constructed or allocated. Let

$$\mathcal{U}_i = \bigcup_{j=1}^{M_i} \mathcal{T}_{i,j}$$

denote the mapped-token domain of response i. Both variants are earlier configurations that build the comparison geometry from the correct responses $\mathcal{C}_x$ alone, whereas the final STRIDE configuration uses all eligible responses $\mathcal{R}_x$; that is, the comparison context of Eq. equation 4 is restricted to $\mathcal{C}_x$ in both variants. All configurations restrict auxiliary credit to targets in $\mathcal{C}_x$ and assign zero credit outside active groups and the mapped-token domain. Consequently, each variant differs from STRIDE in its novelty reference set as well as in the component it removes. Because the evaluated variants also differ in their novelty reference sets, this comparison evaluates complete configurations rather than isolating either component on its own.

Token-uniform credit. This variant computes step-level novelty scores exactly as in Eq. equation 11, with the comparison context restricted to $\mathcal{C}_x$, and then discards the correspondence between a step and the tokens that carry its credit. Writing B_i for the resulting total auxiliary mass of response i, defined as in Eq. equation 26 from the variant's own scores, the credit is redistributed uniformly over the same mapped tokens:

$$A_{i,t}^{\text{uniform}} = \frac{B_i}{|\mathcal{U}_i|} \mathbf{1}[t \in \mathcal{U}_i]. \tag{32}$$

Per-response auxiliary mass and mapped-token coverage are therefore preserved and only the within-response allocation changes. Proposition 3 characterizes this allocation effect in isolation, at fixed step scores; it does not account for the additional difference in reference set.

Single-step rollout baseline. This variant removes the step decomposition. Pool the frozen-reference hidden states over the same mapped-token union:

$$\bar{h}_i^{\text{pool}} = \frac{1}{|\mathcal{U}_i|} \sum_{t \in \mathcal{U}_i} h_{i,t}^{\text{ref}}, \qquad \phi_i^{\text{pool}} = \text{Norm}(\bar{h}_i^{\text{pool}}). \tag{33}$$

Apply the same ridge kernel and residual transformation to one pooled vector per response in $\mathcal{C}_x$, so that a credited target $i \in \mathcal{C}_x$ is conditioned on the pooled vectors of $\mathcal{C}_x \setminus \{i\}$. With one vector per response there is no reasoning prefix to condition on, so redundancy within a chain is not penalized. The resulting advantage and its total auxiliary weight are

$$\begin{aligned} A_{i,t}^{\text{pool}} &= u_i^{\text{pool}} \mathbf{1}[t \in \mathcal{U}_i], \\ B_i^{\text{pool}} &= |\mathcal{U}_i| u_i^{\text{pool}}. \end{aligned} \tag{34}$$

Because the pooled score is recomputed rather than redistributed, this variant is matched to STRIDE in construction but not in total auxiliary mass.

C EVALUATION AND DIVERSITY-JUDGE DETAILS

Correctness evaluation. For every problem, we sample 32 responses from each evaluated model. Pass@1 is computed by averaging correctness over all sampled responses, while Pass@32 is the fraction of problems for which at least one of the 32 responses is correct. Incorrect, truncated, and grader-error responses remain in the denominator. Correctness is evaluated on all 30 AIME24 problems, 30 AIME25 problems, 500 MATH500 problems, 272 Minerva problems, and 500 OlympiadBench problems.

Semantic diversity. Let

$$\mathcal{Y}_x = \{y_{x,1}, \ldots, y_{x,32}\} \tag{35}$$

denote the sampled responses to problem x, and let $e_{x,i}$ be the normalized dense BGE-M3 embedding of response $y_{x,i}$ (Chen et al., 2024). We define the semantic diversity of a problem as

$$D_{\text{sem}}(x) = \frac{2}{32 \cdot 31} \sum_{1 \le i < j \le 32} \left(1 - e_{x,i}^{\top} e_{x,j}\right). \tag{36}$$

Thus, each problem-level score averages the cosine distance over all $\binom{32}{2} = 496$ response pairs. Dataset-level semantic diversity is the mean of $D_{\text{sem}}(x)$ over its evaluation problems.

Strategy diversity. Embedding distance can capture broad semantic variation but does not directly determine whether two correct solutions execute different mathematical arguments. Differences in wording, notation, exposition order, or routine calculations may change the embedding distance without changing the underlying solution strategy. We therefore use an LLM judge as our primary strategy-level evaluator. LLM judges provide scalable, rubric-conditioned evaluation of generated responses (Zheng et al., 2023) and have also been used to assess mathematical reasoning trajectories (Hu et al., 2025).

For each problem x, let

$$\mathcal{C}_x = \{y \in \mathcal{Y}_x : \text{valid}(y) = 1 \wedge \text{correct}(y) = 1\} \tag{37}$$

denote its qualifying correct responses. Given the problem and $\mathcal{C}_x$, the judge partitions the responses according to their executed mathematical approaches. Two responses are assigned to the same class when they share the same logical reduction or proof skeleton, despite differences in wording, notation, exposition order, routine calculations, verification steps, or step boundaries. A separate class requires a substantive change in the executed route, such as a different theorem, construction, decomposition, intermediate objective, case split, or proof technique.

We report

$$\text{DistinctStrategies@32} = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} |\mathcal{J}(x, \mathcal{C}_x)|, \tag{38}$$

where $\mathcal{J}(x, \mathcal{C}_x)$ is the judge-induced partition and $\mathcal{J}(x, \varnothing) = \varnothing$. Consequently, $\mathrm{DistinctStrategies@32} \in [0, 32]$. Incorrect responses receive no strategy credit, but their problems remain in the denominator. A problem with no correct response therefore contributes zero. This definition measures the number of distinct successful approaches available under a fixed sampling budget without conditioning the evaluation set on every method producing a prespecified number of correct responses.

Judge configuration. We use GPT-OSS-120B at temperature zero and query each method and problem in a separate call. Method identities are hidden from the judge, and only responses that pass the validity and correctness checks are included. The judge returns a single strategy count using the fixed output format shown in Appendix D.

D Prompts and Output Examples

Training Prompt

User message
Solve this mathematics problem carefully.
In your final response, present your reasoning in at least two nonempty steps. Start each step on a new line with `Step 1:`, `Step 2:`, and so on, numbered consecutively without gaps. Each step should contain one meaningful mathematical deduction. Do not add empty or trivial steps merely to increase the count. Include these numbered steps in the final response even if you also use a separate thinking section.
After the steps, write exactly one final-answer line `Final Answer: \boxed{answer}`. Replace `answer` with your result. Use `\boxed{}` only for the final answer and not for intermediate values. Write nothing after the final-answer line.

Problem
`{question}`

Figure 2: The shared prompt used to generate every rollout during training. The question is inserted unchanged from the training dataset. The ground-truth answer is withheld from the model and used only for correctness evaluation and reward computation.

LLM Judge Prompt

User message
You are an expert mathematics evaluator.
Given one problem and its correct solutions, count the number of distinct executed reasoning trajectories.
Treat solutions as the same trajectory when they use the same logical reduction or proof route, despite differences in wording, notation, exposition order, routine algebra, verification, or step boundaries.
Count a new trajectory when a solution executes at least one substantive change in the reasoning route, such as a different theorem, construction, decomposition, intermediate objective, case split, or proof technique. Ignore abandoned scratch work and purely stylistic differences. When uncertain, merge.

Problem
`{question}`

Correct solutions
`{numbered_solution_list}`

Return only
`{"distinct_trajectories": K}`

Figure 3: The prompt used by the LLM judge to count distinct reasoning trajectories among the correct responses to a problem. The judge receives only responses verified as correct, and the returned integer is used as $|\mathcal{J}(x, \mathcal{C}_x)|$ when computing DistinctStrategies@32.

Example Model Rollouts

Problem. A goat starts at $(0, 0)$ and travels $1, 2, 3, \ldots, n$ units, turning 90° clockwise before each move. Its first four directions are up, right, down, and left. After travelling 55 units in total, determine its position.

Correct I — Coordinate Sums

Step 1: From $\frac{n(n+1)}{2} = 55$, we obtain $n = 10$.

Step 2: The horizontal displacement is

$$x = 2 - 4 + 6 - 8 + 10 = 6.$$

Step 3: The vertical displacement is

$$y = 1 - 3 + 5 - 7 + 9 = 5.$$

Final Answer: $\boxed{(6, 5)}$

Correct II — Cycle Grouping

Step 1: The distance equation gives $n = 10$.

Step 2: Each four-move block contributes

$$(2 - 4,\ 1 - 3) = (-2, -2).$$

After eight moves, the position is $(-4, -4)$.

Step 3: The final two moves add $(0, 9)$ and $(10, 0)$, giving

$$(-4, -4) + (0, 9) + (10, 0) = (6, 5).$$

Final Answer: $\boxed{(6, 5)}$

Incorrect — Arithmetic Error

Step 1: The response correctly obtains $n = 10$.

Step 2: After move 6, the position is $(4, 3)$, but the response incorrectly computes

$$(4, 3) - (0, 7) = (4, -10).$$

Step 3: Continuing from this incorrect position, moves 8, 9, and 10 give

$$(4, -10) + (-8, 0) + (0, 9) + (10, 0) = (6, -1).$$

Final Answer: $\boxed{(6, -1)}$

Figure 4: Example STRIDE rollouts on OlympiadBench, abridged for presentation. Two correct responses reach the answer through different strategies, while the incorrect response makes an arithmetic error in Step 2.