SHAPE: Stage-aware Hierarchical Advantage via Potential Estimation for LLM Reasoning

Zhengyang Ai[1], **Zikang Shan**[2], **Xiaodong Ai**[1], **Jingxian Tang**[1], **Hangkai Hu**[1], **Pinyan Lu**[1,3]*
[1]Huawei Taylor Lab
[2]Center for Data Science, Peking University
[3]Shanghai University of Finance and Economics
aizhengyang@huawei.com, lu.pinyan@mail.shufe.edu.cn

Abstract

Process supervision has emerged as a promising approach for enhancing LLM reasoning, yet existing methods fail to distinguish meaningful progress from mere verbosity, leading to limited reasoning capabilities and unresolved token inefficiency. To address this, we propose Stage-aware Hierarchical Advantage via Potential Estimation (SHAPE), a framework that formalizes reasoning as a trajectory through a state space of empirical solvability. SHAPE introduces a hierarchical credit assignment mechanism: at the *segment level*, it employs a stage-aware advantage function to prioritize efficient breakthroughs in low-potential states; at the *token level*, it utilizes entropy-driven redistribution to sharpen execution signals. Extensive experiments in math reasoning across three base models and five benchmarks demonstrate that SHAPE achieves an average accuracy gain of 3% with 30% reduced token consumption.

1 Introduction

Reinforcement Learning (RL) has emerged as the standard paradigm for post-training Large Language Models (LLMs). While outcome-based methods such as Group Relative Policy Optimization (GRPO) (Shao et al., 2024) optimize against final answer correctness, they fundamentally struggle with sparse rewards, often leading to inefficient exploration or overthinking due to reward misspecification. Although learned Process Reward Models (PRMs) (Lightman et al., 2023; Zhang et al., 2025b; Cui et al., 2025) offer dense feedback to mitigate this, they incur prohibitive annotation costs and remain vulnerable to reward hacking. Consequently, the field has increasingly shifted toward *rule-based process supervision* (Qu et al., 2025; Guo et al., 2025b; Nie et al., 2026), which derives dense signals via rule-based estimation rather than unreliable black-box verifiers, facilitating explicit designs to robustly mitigate reward hacking.

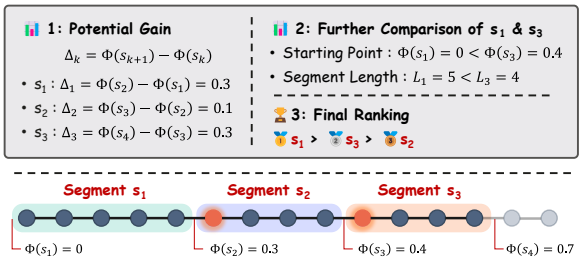


Figure 1: Illustration of optimal reasoning path. Segment s_2 ranks lowest due to insufficient Potential Gain ($\Delta = 0.1$). While s_1 and s_3 share identical gain ($\Delta = 0.3$), s_1 achieves the highest rank by jointly satisfying Stage Awareness (breaking through from a harder state $\Phi = 0$) and Efficiency (shorter length $L_1 < L_3$).

The core of this paradigm lies in estimating the value of intermediate states (i.e., segments) to fill the void of sparse outcome signals. In this work, we unify this intermediate value under the concept of *Reasoning Potential* (Φ). In classical RL, potential serves as a scalar function representing the latent value of a state—acting not as a direct environmental reward, but as a form of preference or prior knowledge regarding state quality (Ng et al., 1999; Wiewiora, 2003). In LLM reasoning, Φ acts as a dynamic progress gauge: a low Φ reflects a state of high uncertainty or confusion, while a rising Φ signifies that the reasoning path is effectively bridging the logical gap to the solution.

With the concept of potential established, the critical question becomes: *How can we leverage* Φ *to construct dense process rewards for effective reward shaping?* We argue that an optimal reasoning path (i.e., a segment transition) must jointly satisfy three essential criteria (as illustrated in Figure 1):

1. **Potential Gain:** The fundamental requirement is that a segment must effectively bridge the logical gap to the solution ($\Delta\Phi > 0$).

*Corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 24288–24305
July 2-7, 2026

2. **Stage Awareness:** Not all gains are equal. Breakthroughs from *lower-potential stages* (confusing foothills) represent overcoming high uncertainty and are thus more valuable than marginal refinements in *high-potential stages* (near the summit).

3. **Token Efficiency:** Among paths with equal gain, shorter paths should be prioritized. Verbose reasoning chains that accumulate computational cost without proportional progress must be discouraged.

Existing approaches, however, fail to unify these dimensions. SPO (Guo et al., 2025b) focuses solely on Potential Gain, ignoring stage difficulty. MRT (Qu et al., 2025) implicitly captures Stage Awareness but lacks checks for stepwise progress and neglects Efficiency. S-GRPO (Dai et al., 2025) addresses length penalties but lacks the semantic granularity of potential guidance. Consequently, a unified framework optimizing all three dimensions remains absent.

We propose **SHAPE** (**S**tage-aware **H**ierarchical **A**dvantage via **P**otential **E**stimation), a framework explicitly designed to satisfy these three principles simultaneously. Grounded in Potential-Based Reward Shaping (PBRS) (Ng et al., 1999; Wiewiora, 2003), SHAPE introduces a novel advantage function controlled by a dynamic, length-dependent discount factor (γ_k). This single mechanism naturally encodes the optimal criteria: it measures *Potential Gain* via difference modeling, enforces *Stage Awareness* by scaling the baseline penalty according to the current stage's difficulty, and regulates *Token Efficiency* by dynamically adjusting the discount rate based on segment length.

To further refine this guidance, SHAPE operates in a hierarchical manner. We introduce an entropy-driven token-level redistribution mechanism to implement fine-grained shaping atop the segment-level foundation. This sharpens the feedback by assigning higher credit to pivotal tokens, ensuring that the learning signal focuses on critical decision points within the segment.

Our contributions are summarized as follows:

- We formalize three essential principles for optimal reasoning paths, and show how existing methods only partially satisfy them.
- We propose SHAPE, which utilizes a dynamic discounting mechanism to jointly satisfy these criteria, supplemented by hierarchical token-level refinement.
- We demonstrate consistent empirical gains across various benchmarks with significantly reduced token costs, validating the importance of stage-aware hierarchical reward shaping.

2 Related Work

RL for LLM Reasoning Reinforcement learning has been proven effective in improving LLM reasoning capabilities (Jaech et al., 2024; Guo et al., 2025a). The Reinforcement Learning with Verifiable Rewards (RLVR) paradigm trains the pretrained LLM in reasoning-heavy tasks (Shao et al., 2024; Guo et al., 2024) via large-scale RL on rule-based verifiers. The prominant algorithm, GRPO (Shao et al., 2024), replaces the value model of PPO (Schulman et al., 2017) with sample-based baseline, trading fine-grained feedback for reduced training resources. Despite later algorithmic improvements (Liu et al., 2025a; Yu et al., 2025), RLVR suffers from the sparsity of the outcome-based reward signal, leading to issues like training instability and reward misspecification (e.g. overthinking (Chen et al., 2024; Zhang et al., 2025a) and underthinking (Qu et al., 2025)).

Fine-grained Credit Assignments Methods that provide fine-grained feedbacks have been proposed. Yuan et al. (2025); Zhu et al. (2025) revisit PPO to improve value modeling, while Kazemnejad et al. (2024); Guo et al. (2025b) estimate values with online rollouts. Such methods involve extra cost of either training a value model or doing expensive rollouts. Process reward models (Lightman et al., 2023), either explicitly (Zhang et al., 2025b) or implicitly (Zhong et al., 2024; Cui et al., 2025), are leveraged to provide extra fine-grained feedback. These methods require pretraining a reward model and are prone to reward hacking. There are also works that uses reward bonus. ATTNPO (Nie et al., 2026) further explores low-overhead step-level supervision by leveraging intrinsic attention signals to reduce redundant reasoning. Notably, MRT (Qu et al., 2025) introduces a rule-based progress reward that is not hackable and efficient to compute, leading to higher token efficiency. Compared to MRT, our proposed framework takes into account more design principles and results in even better performance and token efficiency.

3 Preliminaries: MRT

To address sparse rewards, Meta Reinforcement Fine-Tuning (MRT) (Qu et al., 2025) introduces dense intermediate feedback. We summarize its core mechanism:

1. Trajectory Segmentation. The reasoning trajectory y is decomposed into segments $S = (s_1, \ldots, s_K)$ using delimiters (e.g., newlines), with each boundary denoted as s_k.

2. Potential Estimation. At each boundary s_k, MRT generates m rollouts, where each rollout receives a binary score $r_i \in \{0, 1\}$ based on correctness. The potential is simply the average score:

$$\Phi(s_k) = \frac{1}{m} \sum_{i=1}^{m} r_i. \quad (1)$$

We term $\Phi(s_k)$ the reasoning potential—it quantifies the immediate solvability of the state s_k.

3. Advantage Computation. MRT defines a progress bonus based on the gap between the final outcome R_{outcome} and the current potential. The advantage for segment s_k is:

$$A_k^{\text{MRT}} = R_{\text{outcome}} + \alpha \cdot (R_{\text{outcome}} - \Phi(s_k)), \quad (2)$$

where α balances the outcome and progress signals. Notably, this formulation creates a telescoping effect: when summed over a trajectory, intermediate potentials cancel out, effectively reducing the objective to optimizing the final outcome relative to the initial baseline.

Key Insight. This design creates a global drive toward correctness: segments farther from the answer (lower Φ) receive stronger bonuses upon success. This naturally encodes stage-aware incentives, rewarding breakthroughs from low-potential states more than marginal refinements near the solution.

Limitations. Despite its merits, MRT's formulation exhibits critical structural flaws:

1. **Weak Local Incentives (Sandbagging Risk):** The advantage function relies solely on the potential of the *current* state relative to the endpoint ($R - \Phi(s_k)$), ignoring the potential change to the *next* state ($\Phi(s_{k+1})$). This decoupling fails to enforce monotonic progress. Consequently, the model is not penalized for actions that decrease potential; instead, it can be rewarded for recovering from self-inflicted low-potential states. This structural loophole incentivizes strategic sandbagging—deliberately traversing circuitous, low-potential paths to maximize cumulative bonuses through repeated "recoveries", leading to reasoning degradation. We empirically analyze this phenomenon in § 6.3.

2. **Efficiency Neglect:** MRT lacks explicit length penalties. Verbose segments achieving the same potential gain as concise ones receive identical rewards, inadvertently encouraging computational inefficiency.

3. **Uniform Token Credit:** The segment-level advantage A_k is broadcast uniformly to all tokens in s_k, failing to distinguish critical decision steps from trivial tokens.

These limitations motivate SHAPE: we introduce local potential-difference modeling with length-dependent discounting to enforce monotonic progress while penalizing verbosity, and refine token-level credit redistribution.

4 Method: The SHAPE Framework

In this section, we propose **SHAPE** (**S**tage-aware **H**ierarchical **A**dvantage via **P**otential **E**stimation) framework. The overall architecture and workflow are illustrated in Figure 2.

4.1 Trajectory Segmentation and Potential Estimation

Standard process supervision typically relies on rigid delimiters. To capture true semantic transitions, we adopt the Adaptive Cutpoint-based Partition strategy from SPO (Guo et al., 2025b). However, while SPO relies on low-probability tokens to identify cutpoints, we argue that high information uncertainty is a more robust indicator of logical branching (Wang et al., 2025; Cheng et al., 2026). Therefore, we adapt their framework to utilize token-level entropy as the segmentation criterion.

Entropy-Based Segmentation. We identify reasoning boundaries at points of high token-level entropy, which signal pivotal logical transitions:

$$H(x_t) = -\sum_{v \in \mathcal{V}} \pi_\theta(v \mid x_{<t}) \log \pi_\theta(v \mid x_{<t}). \quad (3)$$

Positions exceeding a threshold τ serve as candidate cutpoints. Following SPO, we downsample

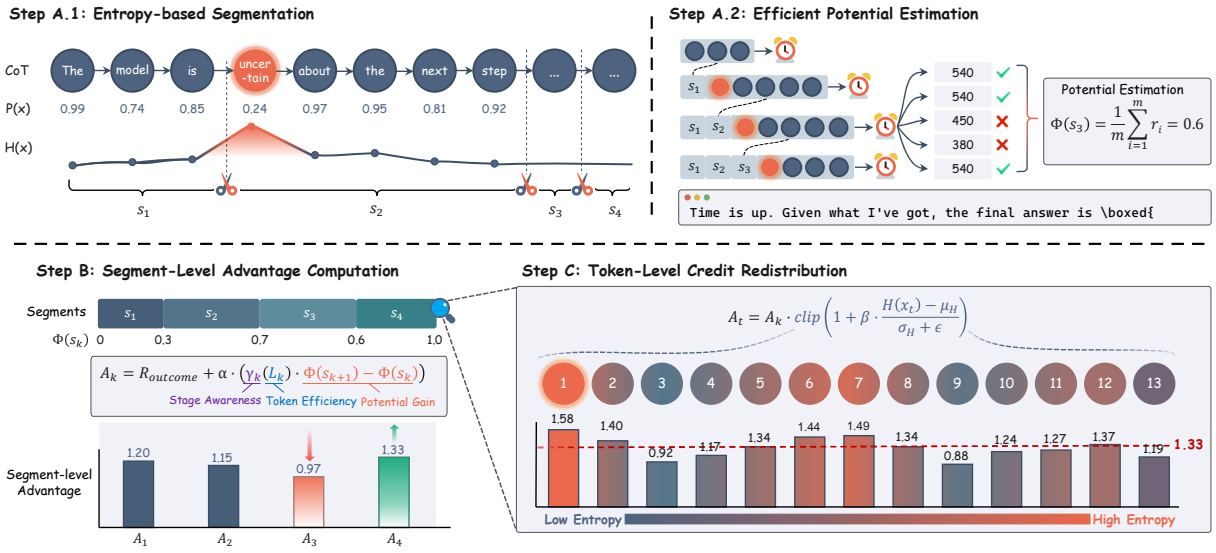


Figure 2: Overview of the SHAPE framework. The pipeline consists of three steps: (A) decomposing reasoning to estimate state potentials (§4.1); (B) computing stage-aware segment-level advantages (§4.2); and (C) redistributing credit to sharpen token-level learning signals (§4.3).

these candidates to standardize the trajectory into K segments $S = (s_1, \ldots, s_K)$, ensuring boundaries align with semantic decision nodes.

Efficient Potential Estimation. At each boundary s_k, we estimate the potential $\Phi(s_k)$ using the rollout formulation defined in Equation (1). Specifically, we execute m forced-termination rollouts to calculate the expected success rate. To maintain computational feasibility, we leverage vLLM's Prefix Caching to avoid re-computing shared contexts and strictly limit the rollout length (e.g., `max_tokens=16`), ensuring that the overhead of this spliced reasoning evaluation remains within an acceptable range. Detailed analysis of the trade-off between segmentation granularity and computational overhead is provided in § 6.4.

4.2 Segment-Level Advantage Computation

MDP Formulation and Standard PBRS. We formulate the reasoning process as a segment-level Markov Decision Process (MDP) to rigorously ground stepwise credit assignment in reinforcement learning theory. In this framework, states correspond to segment boundaries s_k, and transitions represent generating the next segment. To address sparse outcome supervision, we adopt Potential-Based Reward Shaping (PBRS) (Ng et al., 1999; Wiewiora, 2003):

$$F(s_k, s_{k+1}) = \gamma \Phi(s_{k+1}) - \Phi(s_k), \qquad (4)$$

where $\gamma \in (0, 1]$ is a constant discount factor. This form guarantees *policy invariance*: adding F to the reward accelerates learning without altering the optimal policy.

Efficiency-Aware Modification. However, in the context of CoT reasoning, strictly preserving the original policy is suboptimal. The original objective typically ignores computational cost, potentially encouraging the model to generate verbose or redundant reasoning paths to maximize confidence. To introduce an inductive bias favoring conciseness, we modify the discount factor to be a dynamic coefficient γ_k dependent on the segment length L_k. We formalize this using a linear decay function:

$$\gamma_k(L_k) = \max\left(\gamma_{\min},\ 1 - \frac{L_k}{L_{\text{ref}}}(1 - \gamma_{\min})\right), \qquad (5)$$

where $\gamma_{\min} < 1$ represents the lower bound of the discount, and L_{ref} is a manually configured reference length based on the expected granularity of reasoning steps. This formulation establishes a negative correlation between segment length and the discount factor: as L_k increases, γ_k decays linearly from 1 down to the lower bound $\gamma_{\min}$. In our experiments, we empirically set the floor $\gamma_{\min} = 0.9$; for a detailed ablation study on this hyperparameter, please refer to § 5.3.

Advantage Definition. Incorporating this dynamic factor, we define the advantage A_k for seg-

ment s_k as:

$$A_k = R_{\text{outcome}} + \alpha \cdot \underbrace{\left(\gamma_k(L_k) \cdot \Phi(s_{k+1}) - \Phi(s_k)\right)}_{\text{Potential-Based Shaping}}. \tag{6}$$

Although introducing a variable discount factor theoretically alters the original policy, we prove in Appendix A.1 that this formulation preserves *Task Consistency*: correct solutions, regardless of their length, consistently yield higher total rewards than incorrect ones, preventing the model from exploiting the length penalty to generate short but wrong answers.

Mechanism Analysis. To elucidate how this formulation achieves our dual objectives—stage awareness and token efficiency—we decompose the shaping term. Let $\Delta_k = \Phi(s_{k+1}) - \Phi(s_k)$ be the raw potential gain. Substituting $\Phi(s_{k+1}) = \Phi(s_k) + \Delta_k$ into the shaping term (ignoring α for analysis), we obtain:

$$F_k \approx \Delta_k - \underbrace{\left(1 - \gamma_k(L_k)\right) \cdot \Phi(s_k)}_{\text{Reasoning Tax}}. \tag{7}$$

This decomposition reveals that the effective reward is the raw gain Δ_k minus a "Tax". This Tax term naturally enforces our two design goals:

- **Stage Awareness:** The tax is proportional to the baseline potential $\Phi(s_k)$. In early reasoning stages where confidence is low ($\Phi(s_k)$ is small), the tax is negligible, encouraging the model to attempt breakthroughs. Conversely, in high-confidence states, the tax increases, suppressing potential inflation.
- **Token Efficiency:** The tax is proportional to $(1 - \gamma_k)$, which grows linearly with segment length L_k (as defined in Equation (5)). Longer segments incur a heavier tax, compelling the model to justify extra tokens with substantial potential gains.

We provide the mathematical derivation of these properties in Appendix A.3.

4.3 Token-Level Credit Redistribution

While segment-level advantages provide a strategic signal, applying a uniform A_k to all tokens ignores the varying information density within a reasoning step. To capture fine-grained contributions, recent works such as Cheng et al. (2026) and GTPO (Tan et al., 2025) have introduced entropy-based reward shaping. However, these methods operate at the *trajectory level*, modulating the high-variance global outcome reward based on token entropy relative to the entire sequence.

SHAPE adapts this insight into a hierarchical context, performing credit redistribution strictly within the local segment. This local scope offers the fundamental advantage of *stable anchoring*. Global outcome rewards are inherently sparse and noisy; modulating them token-wise often amplifies variance. In contrast, SHAPE anchors redistribution to the segment advantage A_k—a dense, low-variance signal derived from potential estimation. Floating token rewards around this stable local baseline ensures that we refine a valid signal rather than amplifying chaos.

Standardized Importance & Modulation. We quantify the relative importance of token x_t using a Z-score standardization within its segment. Let μ_H and σ_H be the statistics of the valid entropy sequence in segment s_k. The importance weight w_t is computed via a centered affine transformation:

$$\begin{aligned} \tilde{H}(x_t) &= \frac{H(x_t) - \mu_H}{\sigma_H + \epsilon}, \\ w_t &= \text{clip}\big(1 + \beta \cdot \tilde{H}(x_t), \delta_{\min}, \delta_{\max}\big). \end{aligned} \tag{8}$$

The final token advantage is obtained by modulating the segment anchor: $A_t = A_k \cdot w_t$. This ensures that tokens with average entropy retain the original segment advantage ($w_t \approx 1$), while pivotal high-entropy decisions receive amplified credit proportional to their local significance.

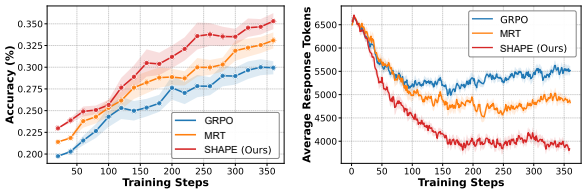
5 Experiments

5.1 Experimental Setup

Experiments are conducted across three backbone models: DeepSeek-R1-Distill-Qwen-1.5B, DeepScaleR-1.5B-Preview, and Qwen3-4B, using rStar2A (Shang et al., 2025) as the training dataset. To ensure stability, we employ the *clip-higher* mechanism from DAPO (Yu et al., 2025), setting $\epsilon_{\text{high}} = 0.28$ and $\epsilon_{\text{low}} = 0.2$. Unless otherwise noted, the process reward coefficient is uniformly set to $\alpha = 0.3$, and the discount factor lower bound is set to $\gamma_{\min} = 0.9$. Evaluation spans five standard benchmarks (AIME 2024/25, AMC 2023, MinervaMATH, MATH500), reporting average accuracy and token usage based on consistent sampling parameters ($T = 0.6, p = 0.95, k = 40$, max_len=32,768). Further details are provided in Appendix B.1.

Method	AIME 24		AIME 25		AMC 23		MATH500		Minerva		Overall	
	Acc	Tokens	Acc	Tokens	Acc	Tokens	Acc	Tokens	Acc	Tokens	Acc	Tokens
DeepSeek-R1-Distill-Qwen-1.5B												
GRPO	34.7	8772	27.5	8109	79.1	5091	84.8	3354	34.2	5228	52.1	6111
MRT	33.1	6577	28.6	6085	79.3	4058	85.0	2734	33.6	3705	51.9	4632
SHAPE	37.1$_{+2.4}$	6164$_{-29.7\%}$	31.8$_{+4.3}$	5425$_{-33.1\%}$	81.5$_{+2.4}$	3612$_{-29.1\%}$	87.8$_{+3.0}$	2415$_{-28.0\%}$	35.5$_{+1.3}$	3207$_{-38.7\%}$	54.7$_{+2.6}$	4165$_{-31.8\%}$
DeepScaleR-1.5B-Preview												
GRPO	38.6	7106	35.7	7041	82.1	4797	87.1	3169	34.5	4965	55.6	5416
MRT	41.3	5601	39.3	5265	82.5	3828	87.8	2696	34.8	3798	57.1	4238
SHAPE	45.6$_{+7.0}$	5194$_{-26.9\%}$	40.5$_{+4.8}$	4896$_{-30.5\%}$	84.9$_{+2.8}$	3549$_{-26.0\%}$	89.0$_{+1.9}$	2069$_{-34.7\%}$	36.9$_{+2.4}$	3115$_{-37.3\%}$	59.4$_{+3.8}$	3765$_{-30.5\%}$
Qwen3-4B												
GRPO	71.3	13541	65.8	15279	92.7	8051	94.0	5116	48.1	6264	74.4	9650
MRT	70.4	12455	66.3	13691	93.1	6508	93.4	3948	47.8	4874	74.2	8295
SHAPE	73.9$_{+2.6}$	11028$_{-18.6\%}$	67.1$_{+1.3}$	12733$_{-16.7\%}$	96.8$_{+4.1}$	5866$_{-27.1\%}$	95.6$_{+1.6}$	3338$_{-34.8\%}$	54.3$_{+6.2}$	4054$_{-35.3\%}$	77.5$_{+3.1}$	7404$_{-23.3\%}$

Table 1: Main results on mathematical reasoning benchmarks. We report Pass@1 accuracy (%) and average generated token counts across five datasets. Blue indicates improvement (Acc ↑ or Tokens ↓) of SHAPE over GRPO. Compared to GRPO and MRT baselines, SHAPE consistently establishes a new Pareto frontier across all three base models, achieving superior accuracy while significantly reducing token consumption.



(a) AIME 24 Accuracy (b) Avg Response Tokens

Figure 3: Performance of DS-R1-Distill-Qwen-1.5B.

5.2 Main Results

Table 1 demonstrates that SHAPE consistently outperforms both outcome-supervised (GRPO) and static process-supervised (MRT) baselines. SHAPE simultaneously maximizes reasoning accuracy and token efficiency across all benchmarks, achieving a substantial reduction in token usage (averaging ∼30%). Crucially, unlike static approaches that often struggle to balance these two objectives, SHAPE's stage-aware shaping (γ_k) successfully compresses reasoning paths while preserving—and often enhancing—semantic integrity. This dual improvement is further corroborated by Figure 3, where SHAPE maintains a clear lead in test accuracy while driving a steep reduction in response length throughout training. To further verify that these gains reflect enhanced general reasoning rather than narrow domain overfitting, we evaluate on out-of-distribution benchmarks in Appendix C.

5.3 Ablation Study

Table 2 summarizes the contribution of SHAPE's components and parameter sensitivity on DeepSeek-R1-Distill-Qwen-1.5B model.

Method	AIME 24		AIME 25	
	Acc	Tokens	Acc	Tokens
GRPO	34.7	8772	27.5	8109
SHAPE (Ours)	**37.1**	**6164**	**31.8**	**5425**
Component Analysis				
w/o EBS	36.8$_{-0.3}$	6380$_{+3.5\%}$	31.6$_{-0.2}$	5590$_{+3.0\%}$
w/o TCR	36.2$_{-0.9}$	6080$_{-1.4\%}$	29.8$_{-2.0}$	5250$_{-3.2\%}$
Parameter Sensitivity (γ)				
Fixed $\gamma_k = 0.9$	36.5$_{-0.6}$	6955$_{+12.8\%}$	32.5$_{+0.7}$	6610$_{+21.8\%}$
$\gamma_{\min} = 0.95$	37.6$_{+0.5}$	6340$_{+2.9\%}$	30.7$_{-1.1}$	5769$_{+6.3\%}$
$\gamma_{\min} = 0.8$	36.3$_{-0.8}$	5720$_{-7.2\%}$	30.9$_{-0.9}$	5010$_{-7.6\%}$
$\gamma_{\min} = 0.7$	30.8$_{-6.3}$	4580$_{-25.7\%}$	26.2$_{-5.6}$	3920$_{-27.7\%}$

Table 2: Ablation study. Subscripts indicate gaps relative to SHAPE (blue: improvement; red: degradation).

Impact of Core Components. We isolate the effects of Entropy-Based Segmentation (EBS) and Token-Level Credit Redistribution (TCR).

- **w/o EBS:** The performance decline validates that EBS effectively delineates semantic units. Unlike rigid heuristics (e.g., "\n\n"), EBS aligns segmentation with logical boundaries, minimizing noise in potential estimation.
- **w/o TCR:** Removing TCR causes a notable accuracy drop, confirming its necessity. By amplifying high-entropy tokens, TCR incentivizes effort at pivotal steps, preventing the model from defaulting to shallow, safe paths.

Sensitivity to Dynamic Discounting (γ). We analyze the length-dependent discount factor γ_k.

- **Fixed $\gamma_k = 0.9$:** The surge in token usage confirms that static discounting fails to impose

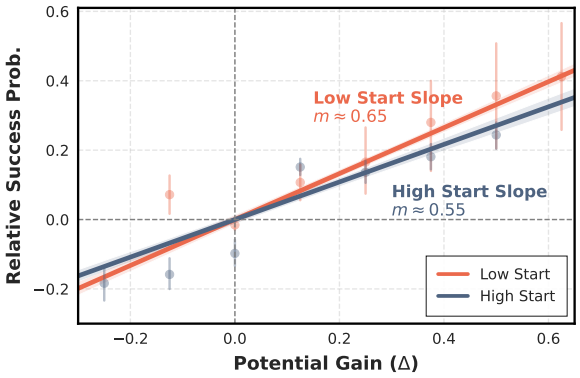


Figure 4: Marginal effect of potential gains. The steeper regression slope for the Low Start group confirms that improvements in adverse states are more critical for final success.

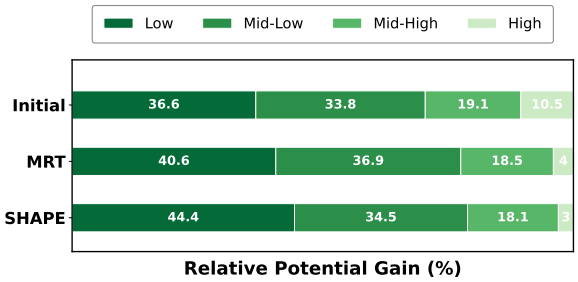


Figure 5: Distribution of potential gain contributions. SHAPE shifts the focus towards the *Low Start* regime (44.4% vs. MRT's 40.6%), validating its capability to rectify reasoning paths from poor initial states.

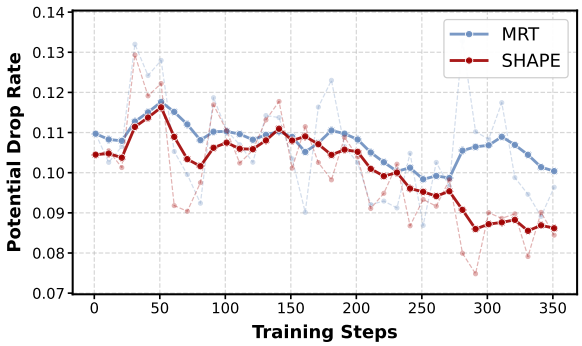


Figure 6: Potential drop rate during training. The curves depict the proportion of adjacent segment transitions where potential decreases ($\Phi(s_{k+1}) < \Phi(s_k)$).

progressive efficiency constraints, validating the necessity of our length-dependent design in curbing verbosity.

- **Varying $\gamma_{\min}$:** A balanced decay is crucial. Relaxing the constraint ($\gamma_{\min} = 0.95$) inflates computational cost, while over-aggressive discounting ($\gamma_{\min} = 0.7$) leads to performance collapse. The latter indicates that excessive penalties force the model to prematurely truncate reasoning to avoid the length tax, rather than solving the problem. A formal theoretical derivation of the critical lower bound for $\gamma_{\min}$ is provided in Appendix A.2.

6 Analysis

Unless otherwise specified, all analytical experiments in this section are conducted using the DeepSeek-R1-Distill-Qwen-1.5B model.

6.1 Sensitivity Analysis of Potential Gains

To validate the Stage Awareness principle proposed in the Introduction—specifically that breakthroughs from low-potential states are more valuable—we analyze the correlation between immediate potential gain (Δ) and final success, stratified by starting potential $\Phi(s_k)$ (see Appendix B.2 for statistical details).

As shown in Figure 4, the Low Start group ($\Phi \leq 0.25$) exhibits a significantly steeper regression slope ($k \approx 0.65$) compared to the High Start group ($k \approx 0.55$). This empirical gap implies that a unit of improvement in adverse stages yields an approximately 18% higher marginal return on final success than in already performant stages. This confirms that rescuing a failing path is far more decisive than refining a successful one, providing strong empirical justification for SHAPE's stage-aware weighting mechanism.

6.2 Evolution of Reasoning Strategy

Having established in §6.1 that improvements in low-potential stages are more decisive, we now verify if SHAPE effectively aligns its optimization focus with this insight. We analyze the *sources* of realized potential gains—categorized by starting potential—to see where progress stems from (methodology in Appendix B.3).

As visualized in Figure 5, SHAPE significantly alters the optimization landscape. It derives the highest proportion of gains from Low Start states (44.4%, compared to MRT's 40.6%), indicating a behavioral shift toward rectifying adverse situations. Conversely, contributions from High Start states drop to just 3% (vs. Initial's 10.5%). This confirms that SHAPE suppresses easy rewards derived from merely maintaining high potentials, forcing the model to tackle the critical, high-difficulty early phases of reasoning.

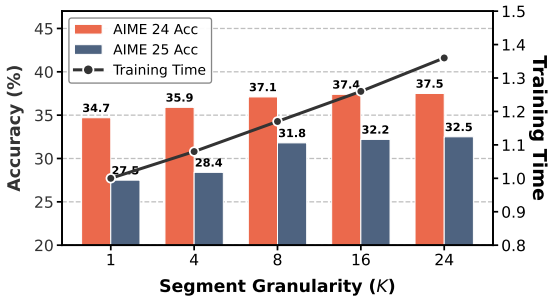


Figure 7: Granularity trade-off. Accuracy (lines) saturates after $K = 8$, while training cost (bars) scales linearly. $K = 8$ represents the optimal balance between performance gains and computational overhead.

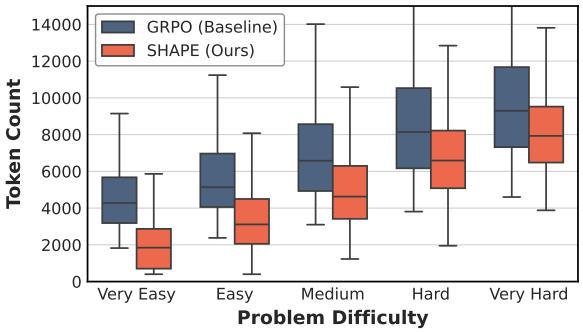


Figure 8: Response length vs. Problem difficulty. SHAPE exhibits a steeper scaling slope and lower variance compared to GRPO, indicating precise resource allocation based on problem hardness.

6.3 Verification of Strategic Sandbagging

To empirically validate the *Sandbagging Risk* hypothesized in §3, we monitor the frequency of negative potential changes ($\Phi(s_{k+1}) < \Phi(s_k)$). As shown in Figure 6, while initial trends are similar, a distinct divergence emerges. MRT exhibits high volatility with conspicuous late-stage spikes, corroborating that its endpoint-based advantage ($R - \Phi(s_k)$) incentivizes gaming—deliberately traversing low-potential states to inflate subsequent recovery rewards. Conversely, SHAPE maintains a steady, low-variance decline. This confirms that our PBRS-based formulation ($\gamma_k \Phi(s_{k+1}) - \Phi(s_k)$) effectively closes this loophole, strictly penalizes regressive steps and enforces monotonic progress.

6.4 Impact of Segmentation Granularity

We analyze the sensitivity to segment count K. Figure 7 reveals the trade-off between model performance and computational cost.

Diminishing Returns & The Sweet Spot. Increasing K from 1 to 8 yields substantial accuracy gains, confirming that intermediate checkpoints significantly aid potential estimation. However, further increasing K to 16 or 24 results in performance saturation or marginal fluctuations. This indicates a marginal utility threshold: $K = 8$ provides sufficient resolution to capture reasoning pivots, whereas excessive segmentation ($K > 8$) risks over-fragmenting semantic units without adding meaningful signal.

Training Cost vs. Inference Saving. While training cost scales linearly with K, the investment is strategically justified. Specifically, $K = 8$ incurs a manageable $1.17\times$ training overhead compared to the baseline ($K = 1$), but unlocks the significant ($\sim 30\%$) recurring inference savings demonstrated in §5.2. This represents an optimal Total Cost of Ownership (TCO) trade-off: swapping a marginal increase in one-time training compute for substantial, permanent deployment efficiency.

6.5 Analysis of Adaptive Computation

We investigate SHAPE's ability to dynamically align reasoning cost with problem complexity. We categorize test problems into five difficulty bins and analyze the distribution of response lengths (methodology in Appendix B.4).

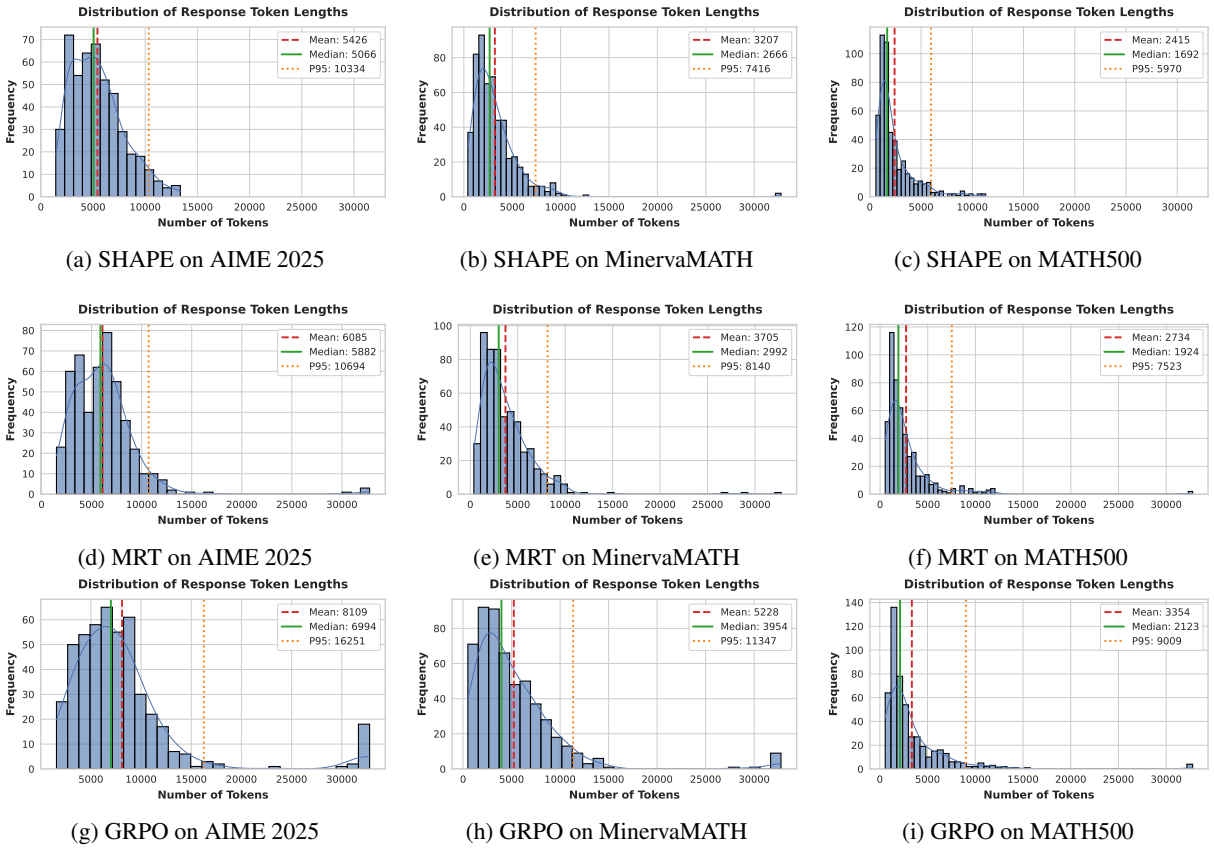
As shown in Figure 8, SHAPE exhibits a steeper scaling slope accompanied by lower variance compared to GRPO. This indicates a robust perception of difficulty: rather than engaging in aimless exploration, SHAPE precisely targets the reasoning depth required for a specific hardness level. Furthermore, SHAPE consistently maintains lower token counts across all bins. This validates that the dynamic discount factor γ_k effectively eliminates redundancy, ensuring the model extends its reasoning chain only when necessary.

6.6 Analysis of Token Length Distribution

In this section, we analyze the distribution of token usage for generating solutions across three representative benchmarks: AIME 2025 (Hard), MinervaMATH (Medium), and MATH500 (Easy).

The distribution plots (Figure 9) were generated using Kernel Density Estimation (KDE) overlaying standard histograms, with curves representing the estimated probability density of response lengths.

1. Skewed Long-Tail Distributions. All three models exhibit a right-skewed, long-tail distribution: the peak density is concentrated at shorter



(a) SHAPE on AIME 2025 (b) SHAPE on MinervaMATH (c) SHAPE on MATH500

(d) MRT on AIME 2025 (e) MRT on MinervaMATH (f) MRT on MATH500

(g) GRPO on AIME 2025 (h) GRPO on MinervaMATH (i) GRPO on MATH500

Figure 9: Token length distributions. SHAPE (top) shows a smooth long-tail distribution, while GRPO (bottom) exhibits anomalous spikes near the 32k context limit, indicating degenerate behavior on hard problems. MRT (middle) reduces but does not eliminate such spikes.

lengths, indicating that most problems are solved within a reasonable token budget, while the tail reflects longer chains on more complex queries.

2. Mitigation of Reasoning Collapse. A critical anomaly is observed in the GRPO baseline, particularly on harder benchmarks like AIME 2025 and MinervaMATH. There is a distinct "spike" or sudden surge in frequency near the maximum generation limit (32k tokens). This phenomenon typically signifies reasoning collapse, where the model, unable to find a solution path for extremely hard problems, enters a degenerate state of repetitive loops or incoherent babbling until it hits the hard cutoff.

MRT shows a marked reduction in these spikes compared to GRPO, as its dense feedback partially alleviates blind search on hard problems; however, degenerate responses still persist, indicating that static supervision lacks the mechanism to efficiently terminate low-value paths. In contrast, SHAPE largely eliminates such spikes across all difficulty levels, with curves decaying smoothly to zero well before the limit. This validates that the length-aware discount factor (γ_k) creates an effective reasoning tax, forcing early termination on dead-end paths and preventing futile context stuffing.

7 Conclusion

In this work, we presented SHAPE to address the critical limitations of existing process supervision. Grounded in the landscape of empirical solvability, SHAPE implements a *hierarchical credit assignment* strategy that harmonizes reasoning capability with efficiency. At the segment level, it employs stage-aware potential shaping to distinguish meaningful breakthroughs from mere verbosity; at the token level, it utilizes entropy-driven redistribution to sharpen execution signals. Extensive experiments across five benchmarks confirm that SHAPE establishes a superior Pareto frontier: achieving an average accuracy gain of 3% with 30% reduced token consumption. These results validate SHAPE as a robust paradigm for efficient LLM reasoning.

Limitations

Our current investigation focuses primarily on mathematical reasoning tasks characterized by deterministic verifiability. This deliberate scoping allows for precise calibration of the potential function Φ, as the binary nature of correctness provides a rigorous anchor for estimating solvability without ambiguity. While the core principles of SHAPE are theoretically transferable to broader contexts, extending them to open-ended domains with subjective evaluation criteria (e.g., creative writing or code generation) presents distinct challenges in quantifying progress, thus remaining an exciting avenue for future exploration.

Ethics Statement

This work studies LLM reasoning on mathematical tasks using publicly available models and benchmarks for research purposes only. We identify no additional ethical risks beyond those documented by the original dataset creators.

Acknowledgements

We thank Yuan Zhou, Hongye Zhou, Yingtong Hu, and Yue Gong for their generous support that made this work possible. We also thank Linyu Liu, Tao Xiao, and Xiaobing Du for their valuable feedback and suggestions. Finally, we thank the anonymous reviewers for their constructive comments.

References

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, and 1 others. 2024. Do not think that much for 2+ 3=? on the overthinking of o1-like llms. *arXiv preprint arXiv:2412.21187*.

Daixuan Cheng, Shaohan Huang, Xuekai Zhu, Bo Dai, Xin Zhao, Zhenliang Zhang, and Furu Wei. 2026. Reasoning with exploration: An entropy perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 30377–30385.

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Yuchen Zhang, Jiacheng Chen, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, and 1 others. 2025. Process reinforcement through implicit rewards. *arXiv preprint arXiv:2502.01456*.

Muzhi Dai, Chenxu Yang, and Qingyi Si. 2025. S-grpo: Early exit via reinforcement learning in reasoning models. *arXiv preprint arXiv:2505.07686*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, and 1 others. 2025a. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Daya Guo, Qihao Zhu, Dejian Yang, Zhenda Xie, Kai Dong, Wentao Zhang, Guanting Chen, Xiao Bi, Yifan Wu, YK Li, and 1 others. 2024. Deepseek-coder: when the large language model meets programming–the rise of code intelligence. *arXiv preprint arXiv:2401.14196*.

Yiran Guo, Lijie Xu, Jie Liu, Dan Ye, and Shuang Qiu. 2025b. Segment policy optimization: Effective segment-level credit assignment in rl for large language models. *arXiv preprint arXiv:2505.23564*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, and 1 others. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. 2024. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*.

Amirhossein Kazemnejad, Milad Aghajohari, Eva Portelance, Alessandro Sordoni, Siva Reddy, Aaron Courville, and Nicolas Le Roux. 2024. Vineppo: Refining credit assignment in rl training of llms. *arXiv preprint arXiv:2410.01679*.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, and 1 others. 2022. Solving quantitative reasoning problems with language models. *Advances in neural information processing systems*, 35:3843–3857.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025a. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*.

Zihan Liu, Yang Chen, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2025b. Acemath: Advancing frontier math reasoning with post-training and

reward modeling. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3993–4015.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Li Erran Li, and 1 others. 2025. Deepscaler: Surpassing o1-preview with a 1.5 b model by scaling rl. *Notion Blog*, 3(5).

Mathematical Association of America. 2023. AMC 2023 competition problems.

Mathematical Association of America. 2024. American invitational mathematics examination (AIME). Art of Problem Solving Wiki.

Mathematical Association of America. 2025. American invitational mathematics examination (AIME). Art of Problem Solving Wiki.

Andrew Y Ng, Daishi Harada, and Stuart Russell. 1999. Policy invariance under reward transformations: Theory and application to reward shaping. In *Icml*, volume 99, pages 278–287. Citeseer.

Shuaiyi Nie, Siyu Ding, Wenyuan Zhang, Linhao Yu, Tianmeng Yang, Yao Chen, Tingwen Liu, Weichong Yin, Yu Sun, and Hua Wu. 2026. Attnpo: Attention-guided process supervision for efficient reasoning. *arXiv preprint arXiv:2602.09953*.

Yuxiao Qu, Matthew YR Yang, Amrith Setlur, Lewis Tunstall, Edward Emanuel Beeching, Ruslan Salakhutdinov, and Aviral Kumar. 2025. Optimizing test-time compute via meta reinforcement fine-tuning. *arXiv preprint arXiv:2503.07572*.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. 2024. Gpqa: A graduate-level google-proof q&a benchmark. In *First conference on language modeling*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Ning Shang, Yifei Liu, Yi Zhu, Li Lyna Zhang, Weijiang Xu, Xinyu Guan, Buze Zhang, Bingcheng Dong, Xudong Zhou, Bowen Zhang, and 1 others. 2025. rstar2-agent: Agentic reasoning technical report. *arXiv preprint arXiv:2508.20722*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2025. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pages 1279–1297.

Hongze Tan, Zihan Wang, Jianfei Pan, Jinghao Lin, Hao Wang, Yifan Wu, Tao Chen, Zhihang Zheng, Zhihao Tang, and Haihua Yang. 2025. Gtpo and grpo-s: Token and sequence-level reward shaping with policy entropy. *arXiv preprint arXiv:2508.04349*.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, and 1 others. 2025. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning. *arXiv preprint arXiv:2506.01939*.

Eric Wiewiora. 2003. Potential-based shaping and q-value initialization are equivalent. *Journal of Artificial Intelligence Research*, 19:205–208.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, and 1 others. 2025. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*.

Yufeng Yuan, Yu Yue, Ruofei Zhu, Tiantian Fan, and Lin Yan. 2025. What's behind ppo's collapse in long-cot? value optimization holds the secret. *arXiv preprint arXiv:2503.01491*.

Wenyuan Zhang, Shuaiyi Nie, Xinghua Zhang, Zefeng Zhang, and Tingwen Liu. 2025a. S1-bench: A simple benchmark for evaluating system 1 thinking capability of large reasoning models. *arXiv preprint arXiv:2504.10368*.

Zhenru Zhang, Chujie Zheng, Yangzhen Wu, Beichen Zhang, Runji Lin, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. 2025b. The lessons of developing process reward models in mathematical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 10495–10516.

Han Zhong, Zikang Shan, Guhao Feng, Wei Xiong, Xinle Cheng, Li Zhao, Di He, Jiang Bian, and Liwei Wang. 2024. Dpo meets ppo: Reinforced token optimization for rlhf. *arXiv preprint arXiv:2404.18922*.

Dingwei Zhu, Shihan Dou, Zhiheng Xi, Senjie Jin, Guoqiang Zhang, Jiazheng Zhang, Junjie Ye, Mingxu Chai, Enyu Zhou, Ming Zhang, and 1 others. 2025. Vrpo: Rethinking value modeling for robust rl training under noisy supervision. *arXiv preprint arXiv:2508.03058*.

A Theoretical Analysis

A.1 Theoretical Analysis of Task Consistency

In this section, we provide a mathematical analysis of the Stage-Aware Discounted Progress (SDP) mechanism. Our primary goal is to prove that the introduction of the dynamic discount factor $\gamma_k(L_k)$ strictly preserves the learning objective through *Strong Task Consistency*.

We define the total reward for a trajectory τ of length K as:

$$R_{\text{total}}(\tau) = \sum_{k=1}^{K} \left(R_{\text{outcome}} + \alpha(\gamma_k \Phi(s_{k+1}) - \Phi(s_k))\right). \tag{9}$$

We operate under two standard assumptions: bounded potentials $\Phi(s) \in [0, 1]$ and bounded discount factors $\gamma_k \in [\gamma_{\min}, 1]$. Strong Task Consistency requires that the minimum reward of any correct trajectory strictly dominates the maximum reward of any incorrect trajectory:

$$\min_{\tau^+ \in \mathcal{T}^+} R_{\text{total}}(\tau^+) > \max_{\tau^- \in \mathcal{T}^-} R_{\text{total}}(\tau^-) \tag{10}$$

1. Upper Bound for Incorrect Trajectories (τ^-). For an incorrect trajectory, $R_{\text{outcome}} = 0$. The reward depends entirely on the accumulated shaping signal. To find the global maximum (the worst-case for consistency), we expand the summation of the shaping term:

$$\begin{aligned} R_{\text{total}}(\tau^-) &= \alpha \sum_{k=1}^{K^-} (\gamma_k \Phi(s_{k+1}) - \Phi(s_k)) \\ &= \alpha \left[\Phi(s_{K^-}) - \Phi(s_0) + \sum_{k=1}^{K^- - 1} (\gamma_k - 1)\Phi(s_k) \right] \end{aligned} \tag{11}$$

Since $\gamma_k \leq 1$ and $\Phi(s) \geq 0$, the summation term $\sum(\gamma_k - 1)\Phi(s_k)$ is non-positive. Therefore, the reward is strictly bounded by the potential difference. The maximum value occurs when $\Phi(s_0) = 0$ and $\Phi(s_{K^-}) = 1$:

$$\max_{\tau^-} R_{\text{total}}(\tau^-) \leq \alpha(1 - 0) = \alpha \tag{12}$$

2. Lower Bound for Correct Trajectories (τ^+). For a correct trajectory, every segment contributes a base outcome reward of 1. We seek the global minimum reward, which corresponds to the adversarial worst-case scenario:

1. The trajectory length is minimal ($K^+ = 1$), minimizing the dense outcome contribution.
2. The potential function is adversarial, dropping from maximum to minimum ($\Phi = 1 \to 0$).
3. The length penalty is maximized ($\gamma_k = \gamma_{\min}$).

Substituting these conditions into the reward equation:

$$\min_{\tau^+} R_{\text{total}}(\tau^+) = 1 + \alpha(\gamma_{\min} \cdot 0 - 1) = 1 - \alpha \tag{13}$$

3. Consistency Theorem. To guarantee Strong Task Consistency, we require the lower bound of correct trajectories to exceed the upper bound of incorrect ones:

$$\min R(\tau^+) > \max R(\tau^-) \implies 1 - \alpha > \alpha. \tag{14}$$

Solving for α, we obtain the sufficient condition:

$$\alpha < 0.5 \tag{15}$$

This derivation proves that by setting $\alpha < 0.5$, the dense outcome signal (R_{outcome}) serves as a dominant anchor. Even in the presence of extreme shaping noise or length penalties, the model guarantees a strictly higher reward for correct reasoning compared to any incorrect shortcut.

A.2 Theoretical Bound of the Discount Factor ($\gamma_{\min}$)

A key empirical finding from the ablation study is that setting $\gamma_{\min} = 0.7$ leads to performance collapse, while $\gamma_{\min} = 0.9$ works well. In this section, we provide a theoretical explanation for this phenomenon by deriving a critical lower bound on γ grounded in the concept of *Reward Sign Consistency*.

Reward Sign Consistency. We require that any positive reasoning step—one that strictly improves the solvability of the current state ($\Phi(s_{k+1}) > \Phi(s_k)$)—must receive a strictly positive shaping reward. Formally, for the shaping term $F_k = \gamma_k \cdot \Phi(s_{k+1}) - \Phi(s_k)$ to be positive, we need:

$$\gamma_k \cdot \Phi(s_{k+1}) - \Phi(s_k) > 0 \implies \gamma_k > \frac{\Phi(s_k)}{\Phi(s_{k+1})}. \tag{16}$$

This constraint is most stringent when transitioning from a high-potential state near the solution. In the worst case, the model improves from $\Phi(s_k) = 7/8$ to $\Phi(s_{k+1}) = 1$, yielding:

$$\gamma_k > \frac{7/8}{1} = 0.875. \tag{17}$$

This constitutes a critical lower bound: any $\gamma_{\min} \leq 0.875$ risks sign reversal, causing the model to be penalized for making correct progress.

Illustrative Example. Table 3 concretizes this with a representative transition ($\Phi_k = 5/8 \rightarrow \Phi_{k+1} = 7/8$, a $+25\%$ improvement in solvability). As γ decreases below 0.8, the shaping reward flips from positive to negative, directly conflicting with the learning objective.

γ	$\gamma \cdot \Phi_{k+1}$	Φ_k	F
1.0	0.875	0.625	$+0.250$
0.9	0.788	0.625	$+0.163$
0.8	0.700	0.625	$+0.075$
0.7	**0.613**	**0.625**	$\mathbf{-0.013}$
0.6	0.525	0.625	-0.100

Table 3: Shaping reward $F = \gamma \cdot \Phi_{k+1} - \Phi_k$ for a positive reasoning step ($\Phi_k = 5/8,\ \Phi_{k+1} = 7/8$).

Conclusion. Our choice of $\gamma_{\min} = 0.9$ maintains a safe margin above the critical threshold ($0.9 > 0.875$), ensuring efficiency is only optimized when doing so does not conflict with correctness. Setting $\gamma_{\min} = 0.7 \ll 0.875$ violates Reward Sign Consistency, making training collapse mathematically inevitable.

A.3 Derivation of Mechanism Properties

In this section, we provide the rigorous mathematical derivation for the two key inductive biases encoded in our advantage formulation: **Stage Awareness** and the **Token Efficiency**.

Recall the shaping term F_k defined in Equation (6):

$$F_k(L_k, \Phi(s_k), \Delta_k) = \gamma_k(L_k) \cdot \Phi(s_{k+1}) - \Phi(s_k). \tag{18}$$

Let $\Delta_k = \Phi(s_{k+1}) - \Phi(s_k)$ denote the semantic progress. Substituting $\Phi(s_{k+1}) = \Phi(s_k) + \Delta_k$, we rewrite F_k as:

$$\begin{aligned} F_k &= \gamma_k(L_k) \cdot (\Phi(s_k) + \Delta_k) - \Phi(s_k) \\ &= \gamma_k(L_k)\Delta_k - (1 - \gamma_k(L_k))\Phi(s_k). \end{aligned} \tag{19}$$

This decomposition highlights that the reward consists of a *discounted gain* $\gamma_k\Delta_k$ minus a *baseline penalty* $(1 - \gamma_k)\Phi(s_k)$.

Property 1: Stage Awareness. Proposition. For a fixed amount of progress Δ_k and a fixed segment length L_k, the shaping reward F_k decreases as the baseline potential $\Phi(s_k)$ increases.

Proof. We analyze the sensitivity of the reward with respect to the starting state's potential $\Phi(s_k)$ by taking the partial derivative of Equation (19):

$$\frac{\partial F_k}{\partial \Phi(s_k)} = \frac{\partial}{\partial \Phi(s_k)} \left[\gamma_k\Delta_k - (1 - \gamma_k)\Phi(s_k)\right]. \tag{20}$$

Since γ_k depends only on L_k and Δ_k is fixed, they are treated as constants. The derivative simplifies to:

$$\frac{\partial F_k}{\partial \Phi(s_k)} = -(1 - \gamma_k(L_k)) = \gamma_k(L_k) - 1. \tag{21}$$

Given that the discount factor is bounded by $\gamma_k \in [\gamma_{\min}, 1]$, we have $\gamma_k - 1 \leq 0$. Thus:

$$\frac{\partial F_k}{\partial \Phi(s_k)} \leq 0 \tag{22}$$

Interpretation. The derivative is strictly negative (assuming $L_k > 0$ so that $\gamma_k < 1$). This mathematically confirms that:

- **Low Baseline (Low $\Phi(s_k)$):** The penalty term $(1 - \gamma_k)\Phi(s_k)$ is small. The model retains most of the gain Δ_k.
- **High Baseline (High $\Phi(s_k)$):** The penalty term is maximized. To achieve the same net reward F_k, the model must generate a significantly larger Δ_k. This effectively suppresses potential inflation in high-confidence states.

Property 2: Token Efficiency. Proposition. For a fixed progress Δ_k starting from a fixed state $\Phi(s_k)$, the shaping reward F_k strictly decreases as the segment length L_k increases.

Proof. We analyze the sensitivity of the reward with respect to segment length L_k. First, recall the linear definition of $\gamma_k(L_k)$ from Equation (5):

$$\gamma_k(L_k) = 1 - c \cdot L_k, \quad \text{where } c = \frac{1 - \gamma_{\min}}{L_{\text{ref}}} > 0 \tag{23}$$

The derivative of the discount factor with respect to length is:

$$\frac{d\gamma_k}{dL_k} = -c < 0 \tag{24}$$

Now, taking the partial derivative of F_k (Equation (19)) with respect to L_k:

$$\begin{aligned} \frac{\partial F_k}{\partial L_k} &= \frac{\partial}{\partial L_k} \left[\gamma_k(L_k)(\Phi(s_k) + \Delta_k) - \Phi(s_k)\right] \\ &= (\Phi(s_k) + \Delta_k) \cdot \frac{d\gamma_k}{dL_k} \\ &= \Phi(s_{k+1}) \cdot (-c) \end{aligned} \tag{25}$$

Since potential $\Phi(s_{k+1}) \geq 0$ and $c > 0$, the derivative is strictly non-positive:

$$\frac{\partial F_k}{\partial L_k} \leq 0 \tag{26}$$

Interpretation. The reward monotonically decreases as the segment becomes longer. By examining the decomposition $F_k = \gamma_k \Delta_k - (1-\gamma_k)\Phi(s_k)$, we see that increasing L_k (which decreases γ_k) imposes a double penalty:

1. **Diminishing Returns:** The term $\gamma_k \Delta_k$ shrinks, meaning the same semantic progress is worth less if it takes longer to generate.

2. **Escalating Tax:** The term $(1 - \gamma_k)$ grows, increasing the penalty proportional to the current state potential.

This creates a compounding pressure on the model to be concise, especially when the current potential $\Phi(s_k)$ is high.

B Experimental Details

B.1 Details of Main Experiments

In this section, we provide comprehensive details regarding the experimental setup, including training hyperparameters, infrastructure configurations, and evaluation protocols.

B.1.1 Training Configuration

We conduct our main experiments on three base models: DeepSeek-R1-Distill-Qwen-1.5B (Guo et al., 2025a), DeepScaleR-1.5B-Preview (Luo et al., 2025), and Qwen3-4B (Yang et al., 2025). We implement our training pipeline based on the `VeRL` framework (Sheng et al., 2025), optimizing for efficient hybrid data and model parallelism. We utilize `vLLM` for high-throughput rollout generation with a tensor parallel size of 1. To manage memory efficiency, we enable gradient checkpointing and set the optimizer/parameter offload to `True` via FSDP (Fully Sharded Data Parallel).

We employ the *clip-higher* PPO variant (Yu et al., 2025) to ensure training stability, with clipping thresholds set to $\epsilon_{\text{high}} = 0.28$ and $\epsilon_{\text{low}} = 0.2$. The KL divergence penalty coefficient is set to 0 to prioritize direct reward optimization, relying on the clipping mechanism for policy constraints. To accommodate different model capacities, we adjust the maximum response length and the number of segments K:

- **1.5B Models** (DeepSeek-R1-Distill-Qwen, DeepScaleR): Max response length = 8,192 tokens; Segment count $K = 8$.
- **4B Model** (Qwen3): Max response length = 16,384 tokens; Segment count $K = 16$.

A detailed summary of the training hyperparameters is provided in Table 4.

Hyperparameter	Value
Global Batch Size	128
Mini-Batch Size	32
Gradient Accumulation Steps	4
Learning Rate	1×10^{-6}
Warmup Ratio	0.1
Total Epochs	1
Max Training Steps	360
Clip Ratio (High / Low)	0.28 / 0.20
KL Coefficient	0.0
Entropy Coefficient	0.0
Max Prompt Length	1,024
Num Generation	8
Use Vllm	True
BF16	True
Process Reward Coeff (α)	0.3
Discount Lower Bound ($\gamma_{\min}$)	0.9

Table 4: Detailed training hyperparameters.

B.1.2 Evaluation Protocol

Inference Settings. We perform zero-shot evaluation across five benchmarks: AIME 2024 (Mathematical Association of America, 2024), AIME 2025 (Mathematical Association of America, 2025), AMC 2023 (Mathematical Association of America, 2023), MinervaMATH (Lewkowycz et al., 2022), and MATH500 (Hendrycks et al., 2021). To balance exploration and stability, we use a temperature of $T = 0.6$, top-$p = 0.95$, and top-$k = 40$. To prevent truncation of long reasoning chains, the maximum generation length is set to 32,768 tokens. Answer extraction and correctness verification are performed using the `Math-Verify` [1] library, ensuring robust parsing of mathematical expressions.

Metric: Avg@N Pass@1. Given the high variance inherent in reasoning model outputs, reporting a single pass rate from a small sample size can be unstable. To ensure statistical reliability, we report the avg@N Pass@1 metric. Specifically, for

[1] https://github.com/huggingface/Math-Verify

each problem, we sample N independent responses and compute the average accuracy. The value of N is adaptively scaled based on the dataset size to maintain a sufficiently large total sample pool (approx. 1,000 samples per benchmark) for low-variance estimation. Table 5 details the sampling configuration for each benchmark.

Benchmark	# Probs	Samples (N)	Total Resps.
AIME 2024	30	32	960
AIME 2025	30	32	960
AMC 2023	40	32	1,280
MinervaMATH	272	4	1,088
MATH-500	500	2	1,000

Table 5: Evaluation dataset specifications and sampling configurations.

B.2 Details of Sensitivity Analysis of Potential Gains

In §6.1, we verify the core motivation of SHAPE by analyzing the correlation between intermediate potential gains and final outcomes. Here we describe the data collection process, grouping strategy, and the centered regression methodology used to generate Figure 4.

Data Collection and Sampling. The analysis is performed on the intermediate potential estimates generated during the SHAPE training process. Unlike standard inference trajectories, these data points represent the on-policy potential estimations calculated at each segment boundary. Specifically, for a training batch with B prompts, each rollout is divided into K segments. At the boundary of segment s_k, the model executes branch rollouts to estimate the state potential $\Phi(s_k)$.

To analyze the relationship between potential evolution and final success, we recorded the full trajectory of potential transitions $(\Phi(s_k) \to \Phi_{k+1})$ and the corresponding final binary outcome $y \in \{0, 1\}$. The training process spans 360 steps (batches). To ensure computational efficiency while maintaining temporal diversity across different training stages, we applied a strided sampling strategy with an interval of 10 steps. This resulted in a total of 36 batches of data, comprising approximately 4.1×10^5 segment transitions for analysis.

Grouping and Filtering. We categorize the transitions based on their starting potential $\Phi(s_k)$:

- **Low Start Regime:** $\Phi(s_k) \leq 0.25$. This corresponds to states where the reasoning path is potentially flawed or confused (e.g., $0/8$ to $2/8$ correctness).
- **High Start Regime:** $\Phi(s_k) \geq 0.5$. This corresponds to states where the reasoning path is already partially or mostly correct.

To ensure a robust linear fit, we apply a specific filter to the Low Start group. We exclude floor effect transitions where the potential drops significantly to zero (specifically, $Gain \leq -0.24$ when starting from $\Phi(s_k) \leq 0.25$), as these boundary conditions represent saturation points that can skew the linearity of the sensitivity estimation.

Centered Regression Methodology. We employ Ordinary Least Squares (OLS) regression on the filtered dataset to determine the slope m (sensitivity) for each group. However, since High Start states naturally possess higher baseline success rates than Low Start states, a direct overlay of their regression lines results in parallel lines with distinct intercepts, obscuring the comparison of their slopes.

To resolve this and visually highlight the *marginal effect*, we employ a Centered Regression approach. For each group $g \in \{\text{Low}, \text{High}\}$, we first calculate the raw intercept $\beta_0^{(g)}$ from the dataset. We then define the centered outcome y' as:

$$y' = y - \beta_0^{(g)}. \tag{27}$$

This transformation aligns the y-intercept of the regression lines to zero, effectively isolating the relative change in success probability attributable to the potential gain Δ. The slopes reported in the main text ($m_{low} \approx 0.65, m_{high} \approx 0.55$) are statistically significant and confirm the higher marginal utility of gains in the Low Start regime.

B.3 Details of Potential Gain Analysis

A core premise of SHAPE is that not all potential gains are equal. While MRT treats a potential improvement from 0.1 to 0.2 equally to an improvement from 0.8 to 0.9, we argue that gains from a *low starting potential* are more critical, as they represent early error correction and path rectification. To verify whether SHAPE successfully incentivizes this behavior, we analyze the distribution of potential gain contributions across different starting states.

To rigorously quantify the evolution of the model's reasoning strategy (§6.2), we implemented a statistical analysis of step-level potential trajectories. The detailed procedure is as follows:

Data Collection. We recorded the step-level potential trajectories for every training batch. As defined in Equation (1), the potential at segment boundary s_k, denoted as $\Phi(s_k)$, is estimated via the rollout success rate. For a rollout width of $N = 8$, $\Phi(s_k)$ is calculated as n_k/N, where n_k is the number of correct outcomes. Consequently, the discrete set of possible potential values is $\{0, 0.125, \ldots, 1.0\}$. We analyzed three datasets: the first 10 steps of MRT training (Initial), the last 10 steps of MRT training, and the last 10 steps of SHAPE training.

Gain Calculation and Binning. We calculate the potential gain for each segment transition as $\Delta_k = \Phi(s_{k+1}) - \Phi(s_k)$. These gains are then aggregated based on their starting potential $\Phi(s_k)$. We group the 8 possible starting states (excluding the terminal state 1.0 which cannot yield positive gain) into four categories:

- **Low Start:** $\Phi(s_k) \in \{0/8, 1/8\}$
- **Mid-Low Start:** $\Phi(s_k) \in \{2/8, 3/8\}$
- **Mid-High Start:** $\Phi(s_k) \in \{4/8, 5/8\}$
- **High Start:** $\Phi(s_k) \in \{6/8, 7/8\}$

Normalization. Since raw potential gains can be negative (indicating potential degradation), comparing absolute sums across different training stages is challenging. To visualize the relative contribution distribution, we apply a global normalization: 1. We compute the average raw gain for each category across all samples. 2. We identify the global minimum average gain $G_{\min}$ across all methods and categories. 3. We apply a floor shift to ensure non-negativity: $G'_{\text{cat}} = G_{\text{cat}} - (G_{\min} - \epsilon)$, where $\epsilon = 0.03$. 4. Finally, we calculate the percentage contribution of each category relative to the total shifted gain of the method: $P_{\text{cat}} = \frac{G'_{\text{cat}}}{\sum G'} \times 100\%$.

This metric effectively highlights which state capability (repairing low-potential states vs. improving high-potential states) contributes most to the model's learning progress.

B.4 Details of Adaptive Computation Analysis

In §6.5, we presented the length-difficulty alignment analysis. Here, we detail the dataset construction, difficulty estimation, and visualization standards.

Dataset and Difficulty Estimation. To ensure a representative analysis, we constructed a specific Difficulty Calibration Subset. We randomly sampled 20 problems from each of the 5 evaluation benchmarks (AIME 2024, AIME 2025, AMC 2023, MinervaMATH, and MATH500), resulting in a total of 100 distinct problems.

To establish an objective difficulty score (D), we utilized the base model (*DeepSeek-R1-Distill-Qwen-1.5B*) to avoid bias from post-training. For each problem in this subset, we generated $N = 10$ independent responses. The difficulty is defined as:

$$D = 1 - \text{PassRate}_{\text{Base}}. \tag{28}$$

Based on these scores derived from the 1,000 aggregated responses, the problems are categorized into the five difficulty bins defined in the main text.

Evaluation Procedure. After binning, we evaluated the trained GRPO and SHAPE models on this specific subset. We recorded the token lengths of their generated responses to analyze how each model adapts its reasoning depth relative to the pre-determined difficulty levels.

Box Plot Interpretation. To aid in interpreting Figure 8, the components of the box plots are defined as follows:

- **Central Line:** Represents the *median* response length of the model in that difficulty bin.
- **Box Limits & Height:** The box spans from the 25th percentile ($Q1$) to the 75th percentile ($Q3$), known as the *Interquartile Range (IQR)*. The height of the box reflects the variance or spread of the middle 50% of the data; a shorter box indicates more consistent reasoning length.
- **Whiskers:** The vertical lines extending above and below the box indicate the range of the data ($1.5 \times$ IQR), excluding outliers.

C Out-of-Distribution Generalization

Since SHAPE updates model parameters with strong inductive biases (e.g., conciseness), a natural concern is whether training on mathematical tasks causes the model to overfit, degrading performance on out-of-distribution (OOD) domains. To address this, we evaluate our math-trained models on two challenging OOD benchmarks:

- **GPQA Diamond** (Rein et al., 2024): A rigorous dataset of graduate-level questions in physics, chemistry, and biology (198 questions from the Diamond subset).

- **LiveCodeBench (V5)** (Jain et al., 2024): A contamination-free benchmark of competitive programming problems from LeetCode, AtCoder, and Codeforces, updated continuously to prevent data leakage.

Results are reported in Table 6 as zero-shot Pass@1 accuracy (%).

Method	GPQA	LiveCodeBench
DS-Qwen-1.5B		
GRPO	36.6	19.1
MRT	35.9	18.4
SHAPE	**38.4**	**22.3**
DeepScaleR-1.5B		
GRPO	40.6	23.4
MRT	**41.9**	25.2
SHAPE	41.7	**25.8**
Qwen3-4B		
GRPO	52.8	54.4
MRT	52.5	54.1
SHAPE	**54.4**	**56.7**

Table 6: Zero-shot performance on OOD benchmarks.

SHAPE consistently matches or outperforms the baselines across both domains. Two findings are noteworthy. First, math-focused SHAPE training yields positive transfer to coding (e.g., $+3.2\%$ on LiveCodeBench for the 1.5B model), consistent with recent observations in AceMath-RL (Liu et al., 2025b) that logical rigor learned from mathematical reasoning transfers effectively to algorithmic problem-solving. Second, on GPQA, SHAPE maintains or improves over GRPO, indicating that the inductive bias toward concise reasoning does not degrade general knowledge application. Together, these results confirm that SHAPE enhances the underlying efficiency and logical consistency of the policy—transferable virtues that extend well beyond the mathematical domain.

D Contextualizing SHAPE with Test-Time Scaling

While test-time scaling (TTS) and search-based methods (e.g., Tree Search, Best-of-N) have proven highly effective for enhancing LLM reasoning, SHAPE adopts a distinct yet complementary paradigm focused on training-time policy shaping.

Internalization and Deployment Trade-offs. The primary design choice behind SHAPE is the "internalization" of search heuristics into the model's weights. TTS relies on exploring multiple reasoning paths during inference, which often incurs high, variable, and sometimes exponential computational costs. In contrast, SHAPE leverages its stage-aware advantage and "Reasoning Tax" during the training phase to inherently prune redundant or low-potential paths. This enables the model to generate high-quality solutions in a single pass. Consequently, SHAPE reduces inference token consumption by 30%, making it a highly efficient solution for real-time, large-scale deployments where the latency and computational overhead of TTS would be prohibitive.

Complementarity and Synergy. Importantly, SHAPE and test-time search methods are not mutually exclusive. Recent studies indicate that the effectiveness of search algorithms depends heavily on the structural quality of the base policy. A SHAPE-trained model naturally produces concise, logically sound reasoning trajectories, serving as a superior prior for search. By acting as the base policy, SHAPE can effectively reduce the branching factor and search space required to reach a correct solution, thereby synergistically enhancing the efficiency and upper-bound performance of any subsequent test-time scaling algorithms.

E Use of AI Assistants and Artifact Usage

Use of AI Assistants. In accordance with the ACL policy on AI assistance, we acknowledge the use of AI assistants solely for the purpose of linguistic polishing and grammatical error correction. The core scientific ideas, experimental designs, and data analyses were conducted entirely by the human authors. No text generated by the AI tool contains novel scientific claims or results.

Artifact Licenses and Intended Use. We utilize publicly available models and datasets to conduct our experiments.

- **Models:** We employ open-weights models including *DeepSeek-R1-Distill-Qwen-1.5B*, *Qwen3-4B*, and *DeepScaleR-1.5B-Preview*. These models are used in accordance with their respective open-source licenses and intended use for research purposes.

- **Datasets:** We evaluate on standard mathematical benchmarks including AIME, AMC,

MATH, and MinervaMATH, as well as out-of-distribution benchmarks including GPQA Diamond and LiveCodeBench. These datasets are widely distributed for academic research. We adhere to their respective terms of use, utilizing them strictly for non-commercial research evaluation.

We confirm that our use of these artifacts is consistent with their intended scientific use cases.