DON'T WASTE FAILURES: LEVERAGING FAILURE LENGTH SIGNALS FOR DISCRIMINATIVE THRESHOLDS IN EFFICIENT REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Large reasoning models (LRMs) improve problem-solving accuracy through extended chains of thought, but can incur substantial cost from overly long reasoning. Existing reward-based approaches to reasoning efficiency often set length penalties using overall or success-only length statistics, failing to carefully differentiate how reasoning lengths differ between correct and incorrect rollouts. We show that leveraging this difference informs the design of length control: When failures are shorter than correct rollouts, we retain a higher penalty threshold, whereas longer failures motivate an earlier penalty. Based on this insight, we introduce OASIS (**O**utcome-**A**ware threshold **S**election with **I**ntegrated **S**ignals). OASIS estimates the reasoning-length threshold from the length distributions of correct and incorrect rollouts, determining where the length penalty begins. We further introduce a soft directional estimator to capture short and long failures separately, increasing the threshold for the former and decreasing it for the latter. Because a higher threshold leaves more reasoning unpenalized and weakens the effective length penalty, OASIS rescales the penalty gain to compensate for this weakening of the compression pressure while leaving the selected penalty-free region unchanged. Integrated into a length-aware GRPO reward, OASIS effectively reduces reasoning length relative to the base model across four base models evaluated on MATH500, AIME 24–26, and AMC23, while largely maintaining accuracy and yielding an average improvement of 1.7% points. Notably, on DeepSeek-R1-Distill-Qwen-1.5B, OASIS improves average accuracy by 4.5% points while reducing average reasoning length by 78.2%.

1 INTRODUCTION

Large reasoning models achieve strong problem-solving performance by allocating substantial inference-time computation to intermediate reasoning. This extended computation is particularly beneficial on challenging problems, but performance does not always improve monotonically with additional reasoning: excessive computation increases token cost and may even degrade accuracy (Ghosal et al., 2025; Wu et al., 2026b). Therefore, preserving essential reasoning while eliminating unnecessary computation is crucial for efficient inference.

Existing approaches control reasoning length through explicit or scheduled constraints (Aggarwal & Welleck, 2025; Hou et al., 2025; Liu et al., 2026a), while adaptive methods allocate problem-specific reasoning budgets based on signals such as problem difficulty or empirical rollout outcomes (Li et al., 2026c; Liu et al., 2026b). Notably, several such approaches estimate target reasoning budgets by relying solely on successful rollouts or their length statistics (Yi et al., 2025; Hu et al., 2026). However, successful lengths are difficult to interpret in isolation: without observing the distribution of failed trajectories as a counterfactual reference, a given length distribution cannot determine whether continuing reasoning improves accuracy or merely accumulates redundant tokens.

Incorrect rollouts provide this missing reference: failures may concentrate at longer or shorter lengths, or overlap substantially with successful responses, indicating unnecessary continued reasoning, insufficient reasoning, or weak length evidence, respectively. Thus, problem-specific reasoning budgets should consider both correct and incorrect conditioned length distributions.

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.

To address this, we propose OASIS, a failure-conditioned adaptive thresholding method for length-aware RL that dynamically adjusts the reasoning-length penalty boundary using both correct and incorrect rollouts. Specifically, OASIS raises the threshold when failures concentrate at shorter lengths, lowers it when failures concentrate at longer lengths, and conservatively defaults when length evidence is insufficient. These hypotheses yield closed-form candidates reconciled by a shared mixture objective with a local curvature check. Because upward threshold adjustments to prevent premature termination relax compression pressure, we introduce *iso-pressure restoration*—a batch-level shared gain that restores this lost pressure without compromising the penalty-free region.

We evaluate OASIS with GRPO on four open-source reasoning models trained on DeepScaleR, measuring accuracy and generation length on AIME 2024–2026, AMC23, and MATH500. Across all four base models, OASIS consistently achieves a higher accuracy–efficiency score than prior methods. Relative to base models, OASIS reduces generation length by 32.7% on average while increasing accuracy on DeepSeek-R1-Distill-Qwen-1.5B (+4.5% points) and Qwen3-4B-Thinking-2507 (+4.1% points), maintaining comparable accuracy on the other two models. Overall, our contributions are as follows:

- We identify *failure-conditioned threshold corrections*—derived from the outcome-dependent length disparity between correct and incorrect rollouts—as a critical missing signal in existing length-control objectives, and formally show that marginal pooling attenuates this correction signal.
- We propose an adaptive thresholding method that unifies whole-incorrect and directional failure evidence—capturing both overthinking and underthinking—together with *iso-pressure restoration* to recover penalty pressure lost under threshold relocation.
- Across four reasoning models and five mathematical benchmarks, our method achieves the highest average accuracy-efficiency score among the compared length-control methods.

2 Related Works

Reinforcement Learning for LRM and Efficient Reasoning Reinforcement learning (RL) has substantially improved the reasoning capabilities of large language models, but often at the cost of longer and more redundant reasoning trajectories. Recent work has therefore explored both inference-time and training-time approaches to improve reasoning efficiency. Inference-time methods control computation through token budgets or adaptive termination (Muennighoff et al., 2025; Han et al., 2025; Liu & Wang, 2025), while training-time methods encourage models to produce more concise reasoning through fine-tuning or RL (Hou et al., 2025; Luo et al., 2026; Cui et al., 2025). Unlike inference-time constraints that externally restrict computation, training-time optimization can adapt the model itself to produce more concise reasoning while preserving task performance (Hou et al., 2025; Luo et al., 2026), motivating RL-based optimization.

Length-Aware Reward Shaping. Length-aware RL incorporates reasoning length into the training objective to improve inference efficiency. ShorterBetter (Yi et al., 2025) derives problem-specific reasoning-length targets from correct rollouts, LASER (Liu et al., 2026b) adaptively adjusts them using difficulty-aware criteria while LASER-DE additionally uses incorrect responses for exploration, and SmartThinker (Hu et al., 2026) estimates a peak-accuracy reasoning length from overall and correct-response length statistics. However, these methods do not explicitly use incorrect-response length distributions to determine the threshold, although they provide a complementary reference for distinguishing necessary reasoning from inefficient exploration.

Incorrect Rollouts in Length Control Incorrect rollouts are on average longer than correct ones across models and benchmarks, a trend read as evidence of overthinking rather than used as a training signal (Hochlehnert et al., 2025; Su et al., 2025). Several methods exploit this length discrepancy, but restrict its role to rollout filtering or batch recomposition under a predetermined, fixed directional preference. Specifically, brevity-oriented approaches contrast short-correct against long-incorrect trajectories, penalizing verbosity either by resampling rollouts categorized by correctness and length in QLPO (Chen et al., 2026) or by recomposing them into a dedicated priority batch in RoRecomp (Li et al., 2025). Conversely, empirical filtering heuristics (Chen et al., 2025) impose the opposite directional constraint to promote longer reasoning, discarding rollout groups unless the

average length of correct solutions exceeds that of incorrect ones. These approaches thus use fixed directional preferences to select or prioritize training samples, without dynamically estimating a problem-specific length threshold for reasoning termination. Others condition length-based learning signals on rollout outcomes: DRPO (Li et al., 2026a) normalizes length-based weights only among correct rollouts, whereas DDCA (Peng et al., 2026) computes length advantages within the correct cluster to avoid baseline distortion from incorrect rollouts. While these methods refine how the length penalty is normalized, they do not adjust the length threshold where the penalty takes effect. Wu et al. (2026a) examine the length distributions of correct and incorrect rollouts, but only under externally fixed token budgets. Other reward-shaping approaches determine the length-penalty threshold using correct rollouts (Yi et al., 2025) or a pooled marginal distribution (Hu et al., 2026). None explicitly uses the incorrect length distribution separately for threshold estimation.

3 MOTIVATION

Length-aware reward shaping. Efficient-reasoning methods add a length term to the reward,

$$r_i = r_i^{\text{acc}} - \lambda \max\left(0, L_i - \ell^\star(x)\right), \tag{1}$$

where $r_i^{\text{acc}} \in \{0, 1\}$ is the binary correctness reward for response y_i to prompt x, and L_i is token length, λ controls the penalty strength, $\ell^\star(x)$ is a problem-specific length threshold, and r_i the resulting reward for y_i. Responses exceeding the threshold are penalized proportionally to their excess length. Since $\ell^\star(x)$ affects group-relative advantages in GRPO, setting it too low may suppress necessary reasoning, while setting it too high leaves redundant computation unpenalized.

Adapting the Reasoning-Length Threshold. Prior work has shown that correct and incorrect reasoning trajectories often exhibit different length distributions (Su et al., 2025; Wu et al., 2026a), also observed in Figure 1. A length-penalty threshold derived from correct rollouts alone reveals where successful responses tend to occur, but not whether those lengths discriminate success from failure: a length common among correct responses may also be common among incorrect ones. Incorrect rollouts thus provide counterfactual evidence for adapting this threshold.

We formalize this intuition through a discriminative length-selection criterion based on the posterior odds of correctness, where incorrect rollouts provide the correction. Since $p(L \mid x)$ cancels between the two posteriors and the prior odds are constant with respect to L,

$$\ell_{\text{disc}}(x) = \arg\max_L \log \frac{P(C \mid L, x)}{P(I \mid L, x)} = \arg\max_L \left[\log p(L \mid C, x) - \log p(L \mid I, x)\right]. \tag{2}$$

Importantly, the criterion underlying ℓ_{disc} has the same population maximizer as

$$\arg\max_L P(C \mid L, x) = \arg\max_L \log \frac{p(L \mid C, x)}{p(L \mid x)}, \tag{3}$$

which underlies marginal-referenced threshold estimation. Our interest is not the maximizer as an absolute reasoning target, but the correction induced by the choice of reference. Under a Gaussian approximation, $p(L \mid x)$ absorbs correct–incorrect separation into its variance, attenuating the displacement obtained when failures are conditioned separately. Section 4.1 shows that this attenuation is structural and derives the resulting discriminative correction.

Heterogeneous Incorrect-Length Evidence Figure 1 shows that incorrect-length distributions vary across rollout groups, from dispersed overlap to a greater prevalence of longer or shorter failures, providing discriminative information unavailable from correct outcomes alone or a single Gaussian fitted to the marginal distribution. The relative location and dispersion of correct and incorrect lengths guide the threshold direction: shorter failures favor a higher threshold, whereas longer failures favor an earlier penalty. Motivated by these observations, we develop in Section 4 a unified objective that captures the discriminative length patterns and dynamically selects the threshold. Consequently, this approach resolves a fundamental question: *how should incorrect-length evidence adapt a reasoning-length threshold beyond what marginal statistics alone can provide?*

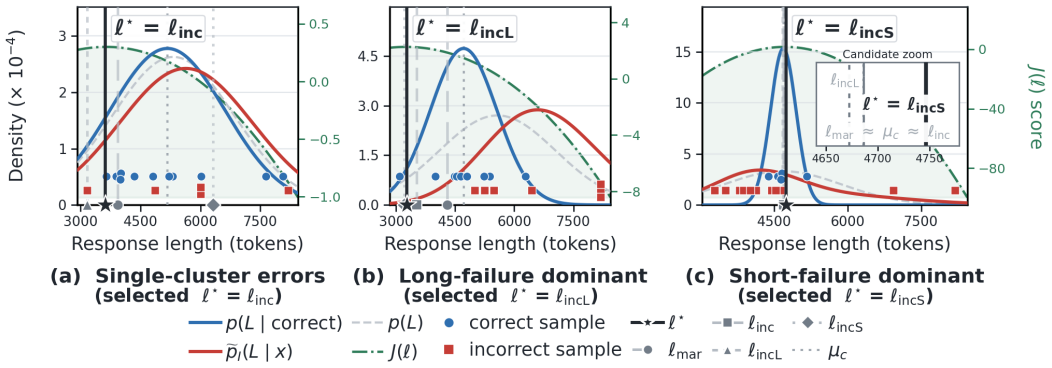

Figure 1: **Heterogeneous incorrect-length evidence in rollouts.** Group-level correct and incorrect length distributions observed during training with DeepSeek-R1-Distill-Qwen-1.5B. Panels (a)–(c) illustrate dispersed overlap, overthinking-dominated failures, and premature failures, respectively. Candidate zoom in (c) magnifies the candidate thresholds near $\ell^\star$.

4 METHOD

4.1 FAILURE-CONDITIONED ADAPTIVE THRESHOLDING

To obtain tractable group-level *threshold candidates*, we approximate the outcome-conditioned reasoning-length distributions by Gaussian densities, consistent with a prior study of LLM response-length distributions (Hu et al., 2026): $p(L \mid C, x) = \mathcal{N}(\mu_C, \sigma_C^2)$, $p(L \mid I, x) = \mathcal{N}(\mu_I, \sigma_I^2)$ for correct and incorrect responses, respectively. Note that we additionally derive the incorrect-conditioned maximizer to capture failure signals alongside success signals. The choice of reference is highly consequential under this Gaussian approximation, leading us to analyze the properties of both pooled and conditioned candidates.

Marginal pooled reference. Although the posterior-odds objective in Equation 2 and the marginal posterior criterion have the same population maximizer, fitting a single Gaussian to the marginal $p(L \mid x)$ mixes the two outcome classes. Let n_C and n_I denote the numbers of correct and incorrect rollouts in the rollout group, and $\pi_C = n_C/(n_C+n_I)$, $D = \mu_I - \mu_C$, $R = \sigma_I^2 - \sigma_C^2 > \epsilon$. Applying the same Gaussian discriminant to the moment-matched marginal distribution gives

$$\kappa = \frac{R}{R + \pi_C D^2} \in (0, 1], \quad \ell_{\text{mar}} = \mu_C + \kappa(\ell_{\text{inc}} - \mu_C), \quad \ell_{\text{inc}} = \mu_C - \frac{\sigma_C^2 D}{R}. \tag{4}$$

Here, κ is the *attenuation factor*: the fraction of the incorrect-conditioned correction retained by the marginal reference. As $|D|$ increases, κ decreases toward zero, so the marginal Gaussian increasingly absorbs class separation as additional variance rather than as displacement relative to the correct distribution. Consequently, marginal pooling attenuates the threshold correction most strongly when correct and incorrect lengths are most separated, motivating a separate analysis of incorrect responses to isolate this attenuation. This effect is structural to the single-Gaussian approximation rather than an estimation error that vanishes with more rollouts; Appendix A.2 gives the full derivation.

Incorrect-conditioned candidate. Substituting the Gaussian densities into the objective of Equation 2 yields a quadratic log-density ratio. Prior studies suggest that response-length variability can contract during RL optimization and become small as policies produce more consistent responses (Li et al., 2026b;d), while empirical findings (Su et al., 2025; Yu et al., 2026) and Figure 1 show distinct length patterns between correct and incorrect trajectories. These observations motivate our intuition that, as correct lengths contract, the relatively wider dispersion of incorrect trajectories can provide complementary information. We therefore consider the incorrect-conditioned candidate only when $\sigma_I^2 > \sigma_C^2 + \epsilon$; otherwise, we retain the correct-conditioned reference. When this condition is satisfied,

the objective is strictly concave and admits the unique finite maximizer:

$$\ell_{\text{inc}} = \mu_C + \underbrace{\frac{\sigma_C^2}{\sigma_I^2 - \sigma_C^2}(\mu_C - \mu_I)}_{\text{discriminative correction}}. \tag{5}$$

This decomposition makes the discriminative correction explicit: μ_C provides the success reference, while incorrect rollouts provide a directional correction—longer failures move the threshold downward, whereas shorter failures move the threshold upward, with relative dispersion controlling the correction magnitude. When $\sigma_I^2 \le \sigma_C^2$, this reference does not define a valid finite candidate under the unconstrained Gaussian objective; the full closed-form derivation and boundary cases are given in Appendix A.1.

Soft directional partitioning for generating directional candidates. While incorrect-conditioned evidence should be kept separate from the marginal reference, representing all incorrect rollouts by a single Gaussian can still obscure heterogeneous failure patterns, such as co-occurring short-side and long-side failures. To capture these finer failure modes without the artifacts of a hard split, we construct soft long- and short-side references.

Let $\{l_j\}_{j=1}^{n_I}$ denote the incorrect rollout lengths. We define the soft assignment weight (or long-side responsibility) of rollout j as:

$$u_j = \text{sigmoid}\left(\frac{l_j - \mu_C}{\tau}\right), \qquad \tau = \sqrt{\sigma_C^2 + \sigma_I^2}. \tag{6}$$

The combined class spread τ provides a scale-adaptive normalization: $u_j = 1/2$ at μ_C, increasing for longer and decreasing for shorter failures. For $k \in \{\text{long}, \text{short}\}$, where long emphasizes failures above μ_C and short emphasizes failures below μ_C, let $r_{j,\text{long}} = u_j$ and $r_{j,\text{short}} = 1 - u_j$, and define:

$$w_k = \frac{1}{n_I}\sum_j r_{j,k}, \qquad m_k = \frac{\sum_j r_{j,k} l_j}{\sum_j r_{j,k}}, \qquad v_k = \frac{\sum_j r_{j,k}(l_j - m_k)^2}{\sum_j r_{j,k}}. \tag{7}$$

Unlike a hard partition at μ_C, which can create artificial mean separation and variance reduction even without directional structure, soft assignment lets observations near μ_C contribute to both sides, reducing separation when directional evidence is weak (Appendix A.5). This decomposition preserves the first two moments of the whole-incorrect distribution (Equation 8), adding directional resolution while retaining the pooled incorrect evidence.

$$w_{\text{long}} + w_{\text{short}} = 1, \qquad \sum_k w_k m_k = \mu_I, \qquad \sum_k w_k(v_k + m_k^2) = \mu_I^2 + \sigma_I^2, \tag{8}$$

Candidate generation and unified selection. Equation 5 already provides the whole-incorrect proposal ℓ_{inc}. Applying the same Gaussian construction to the directional moments $(m_{\text{long}}, v_{\text{long}})$ and $(m_{\text{short}}, v_{\text{short}})$ yields the long- and short-side proposals ℓ_{incL} and ℓ_{incS} when $v_k > \sigma_C^2 + \epsilon$. We estimate the incorrect-length distribution by the soft directional mixture:

$$\widetilde{p}_I(L \mid x) = \sum_{k \in \{\text{long},\text{short}\}} w_k \mathcal{N}(L; m_k, v_k). \tag{9}$$

Substituting $\widetilde{p}_I(L \mid x)$ and the Gaussian correct reference into the criterion in Equation 2 gives:

$$J(L) = \log \mathcal{N}(L; \mu_C, \sigma_C^2) - \log \widetilde{p}_I(L \mid x), \tag{10}$$

which measures relative support for length L between correct and incorrect distributions. We use $J(L)$ as a common criterion to compare threshold proposals, not as a training reward.

The final operational threshold $\ell^\star$ is selected as follows. When incorrect evidence is insufficient, we set $\ell^\star = \ell_{\text{mar}}$; otherwise, we form the candidate set:

$$\mathcal{C} = \{\ell_{\text{mar}}, \ell_{\text{inc}}, \ell_{\text{incL}}, \ell_{\text{incS}}\}, \tag{11}$$

restricted to available proposals. We retain locally concave candidates under the shared mixture objective, projecting each onto the empirical rollout-length support:

$$\mathcal{A} = \left\{ \text{clamp}_{[L_{\min}, L_{\max}]}(\ell) \;\middle|\; \ell \in \mathcal{C},\ J''(\ell) < 0 \right\}, \qquad \ell^\star = \begin{cases} \arg\max_{\ell \in \mathcal{A}} J(\ell), & \mathcal{A} \neq \varnothing, \\ \ell_{\text{mar}}, & \mathcal{A} = \varnothing. \end{cases} \tag{12}$$

For $n_C = 1$, zero variance incurs no penalty. The curvature gate rejects candidates outside locally concave regions of J; for a single Gaussian reference, it reduces to the concavity condition denoted in Appendix A.3. We therefore evaluate structured closed-form proposals under the shared J rather than numerically optimizing the mixture objective; full derivations are given in Appendix A.4.

Figure 1 illustrates how evaluating the available candidates under $J(\ell)$ selects, among the candidate set, the discriminative penalty boundary with the strongest relative support for correctness. ℓ_{inc} under dispersed overlap in (a), an earlier ℓ_{incL} to curb overthinking failures in (b), and a higher ℓ_{incS} to protect reasoning against premature termination in (c). The selected $\ell^\star$ serves as the penalty threshold, leaving correct rollouts with $L \le \ell^\star$ unpenalized to avoid discouraging reasoning that may already be sufficient.

4.2 Batch-Level Iso-Pressure Restoration and Final Objective

Adjusting the decision threshold can reduce compression pressure when rollout evidence supports longer reasoning. To restore this lost penalty mass without altering the chosen thresholds, we introduce batch-level *iso-pressure restoration*, derived from the Greek *isos* (equal) to preserve equivalent total penalty pressure across the batch. For each prompt x_q in training batch $\mathcal{B}$, let $\mathcal{G}_q = \{y_{q,i}\}_{i=1}^{G}$ denote its rollout group with group-specific threshold $\ell_q^\star \equiv \ell^\star(\mathcal{G}_q)$ selected in Section 4.1.

Following the dynamic length-reward scaling of SmartThinker, let $c_i \in \{0, 1\}$ denote correctness with $n_{C,q} = \sum_{i=1}^{G} c_i$. For any threshold b, mixed groups ($0 < n_{C,q} < G$) use

$$\alpha_q(b) = \frac{1 - n_{C,q}/G}{\max\limits_{i:c_i=1}[L_i - b]_+ - \dfrac{1}{G}\displaystyle\sum_{i=1}^{G} c_i[L_i - b]_+ + \epsilon_\alpha}, \quad P_q(b) = \alpha_q(b) \sum_{i=1}^{G} c_i[L_i - b]_+, \tag{13}$$

with $\alpha_q(b) = 0$ for all-correct or all-incorrect groups. Penalizing only correct rollouts prevents shorter failures from becoming relatively preferable, while $\alpha_q(b)$ normalizes token scale so the length penalty does not overwhelm GRPO's group-relative correctness signal. Since b governs both excess lengths and $\alpha_q(b)$, $P_q(b)$ measures correct-only compression pressure before gain scaling.

Taking the marginal threshold $\ell_{\text{mar},q}$ as the pre-correction reference, we quantify the batch-level lost pressure and rescale the base gain g:

$$P_{\text{lost}}^{\mathcal{B}} = \sum_{q \in \mathcal{B}} \left[P_q(\ell_{\text{mar},q}) - P_q(\ell_q^\star) \right]_+, \quad \rho_{\mathcal{B}} = 1 + \frac{P_{\text{lost}}^{\mathcal{B}}}{\displaystyle\sum_{q \in \mathcal{B}} P_q(\ell_q^\star) + \epsilon_p}, \tag{14}$$

$$g_{\text{iso}} = \min(g\rho_{\mathcal{B}}, g_{\max}), \tag{15}$$

where $\epsilon_p > 0$ is a numerical stabilizer and $g_{\max} \ge g$ caps amplification. Because $P_q(b)$ is non-increasing in b (Appendix B.2), the positive part isolates pressure lost to upward threshold corrections. This restoration is intentionally one-sided: lost pressure is restored, whereas additional pressure from downward corrections is retained. Crucially, g_{iso} scales only $[L_i - \ell_q^\star]_+$, leaving the penalty-free region defined by $\ell_q^\star$ intact, yielding the final reward:

$$r_i^{\text{final}} = \begin{cases} \max\left(1 - g_{\text{iso}}\, \alpha_q(\ell_q^\star)[L_i - \ell_q^\star]_+,\ \ \epsilon_r\right), & c_i = 1, \\ 0, & c_i = 0, \end{cases} \tag{16}$$

where $\epsilon_r > 0$ is a small reward floor that preserves a positive reward gap between correct and incorrect rollouts even under strong length penalties.

5 Experiments

5.1 Experimental Setup

We train DeepSeek-R1-Distill-Qwen-{1.5B, 7B} (DeepSeek-AI, 2025) and Qwen3-{1.7B, 4B-Thinking-2507} (Yang et al., 2025) with GRPO on DeepScaleR-Preview-Dataset (Luo et al., 2025),

Table 1: **Main comparison of accuracy and reasoning efficiency across base models.** We report accuracy (Acc.), average reasoning length (Len.), and accuracy-efficiency score (AE) on MATH500, AIME 24–26, and AMC23. Within each model block, the best and second-best results in each column are shown in **bold** and underlined, respectively. † denotes an officially released checkpoint.

Method	MATH500			AIME 24–26			AMC23			Average		
	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑
DeepSeek-R1-Distill-Qwen-1.5B												
Base	76.2	9,708	-	28.9	36,102	-	80.0	20,944	-	61.7	22,251	-
GRPO	84.8	5,393	0.827	26.7	28,810	-0.299	82.5	10,685	0.560	64.7	14,963	0.363
LASER†	76.0	4,224	0.499	31.1	12,150	0.872	75.0	7,333	0.282	60.7	7,902	0.551
ThinkPrune†	72.8	3,060	0.423	24.4	9,288	-0.051	67.5	4,672	-0.021	54.9	5,673	0.117
SmartThinker†	74.4	4,020	0.423	28.9	14,449	0.566	80.0	7,226	0.659	61.1	8,565	0.549
Ours ($\ell^\star$)	**86.0**	2,383	**1.135**	32.2	8,707	1.095	**90.0**	4,147	**1.172**	**69.4**	5,079	**1.134**
Ours ($\ell^\star$+iso)	85.4	**2,293**	1.096	**33.3**	**8,610**	**1.194**	80.0	**3,663**	0.797	66.2	**4,855**	1.029
DeepSeek-R1-Distill-Qwen-7B												
Base	94.6	3,564	-	**61.1**	13,103	-	**95.0**	5,342	-	**83.6**	7,336	-
GRPO	95.2	3,079	0.248	56.7	13,624	-0.375	**95.0**	6,686	-0.228	82.3	7,796	-0.118
LASER†	85.0	2,760	-0.292	51.1	**9,770**	-0.560	**95.0**	4,294	0.178	77.0	5,608	-0.225
Ours ($\ell^\star$)	94.4	**2,475**	0.302	54.4	10,811	−0.401	**95.0**	4,266	**0.210**	81.3	5,851	0.037
Ours ($\ell^\star$+iso)	**95.6**	2,515	**0.342**	**61.1**	9,889	**0.231**	92.5	**4,076**	0.084	83.1	**5,493**	**0.219**
Qwen3-1.7B												
Base	92.6	5,388	-	44.4	16,831	-	**92.5**	9,284	-	**76.5**	10,501	-
GRPO	92.2	5,045	0.057	44.4	15,834	0.056	**90.0**	9,019	-0.072	**75.6**	9,966	0.014
Ours ($\ell^\star$)	**93.0**	4,509	**0.188**	43.3	14,784	0.000	**90.0**	**8,207**	**-0.003**	75.4	9,167	0.062
Ours ($\ell^\star$+iso)	92.0	**4,395**	0.163	**45.6**	**14,639**	**0.218**	87.5	8,461	-0.151	75.0	**9,165**	**0.077**
Qwen3-4B-Thinking-2507												
Base	96.6	4,892	-	70.0	15,234	-	**100.0**	8,042	-	88.9	9,389	-
GRPO	98.4	4,147	0.237	78.9	16,226	0.345	**100.0**	6,720	0.163	92.4	9,031	0.249
SmartThinker†	**98.6**	**3,464**	**0.355**	76.7	**13,504**	0.394	**100.0**	5,982	0.254	91.8	**7,650**	0.334
Ours ($\ell^\star$)	97.6	3,650	0.293	77.8	14,076	0.425	**100.0**	**5,705**	**0.282**	91.8	7,810	0.333
Ours ($\ell^\star$+iso)	97.8	3,879	0.276	**81.1**	14,129	**0.544**	**100.0**	5,953	0.248	**93.0**	7,987	**0.356**

Note. Relative to Base, Ours ($\ell^\star$+iso) reduces average reasoning length by 78.2%, 25.1%, 12.7%, and 14.9%, respectively, starting with DeepSeek-R1-Distill-Qwen-1.5B and proceeding down the model blocks in table order. The corresponding changes in average accuracy are +4.5, -0.5, -1.5, and +4.1 percentage points, yielding a mean gain of +1.7 percentage points across models.

a curated dataset of 40K competition-level mathematical reasoning problems. For brevity, we refer to these models as DeepSeek-1.5B, DeepSeek-7B, Qwen3-1.7B, and Qwen3-4B, respectively. DeepSeek-1.5B is trained for 150 steps, while the remaining models are trained for 50 steps.

We evaluate on AIME 2024–2026, AMC23, and MATH500 (Lightman et al., 2024) using five samples per problem. We report accuracy (majority@5), mean response length, and the accuracy–efficiency (AE) score (Luo et al., 2025); AE uses majority-conditioned lengths, whereas mean response length is averaged over all five samples. We aggregate AIME results across its three editions; per-edition results are reported in Table 7. Training configurations, evaluation, grading, and metric computation are detailed in Appendix B.1.

We compare against the **Base Model** and **standard GRPO** to isolate the effect of RL training, and against **LASER** (Liu et al., 2026b), **ThinkPrune** (Hou et al., 2025), and **SmartThinker** (Hu et al., 2026) as representative methods for reasoning-length control, which employ adaptive length-based reward shaping, progressive length constraints, and dynamic target-length estimation, respectively. We report two ablations of our own method: Ours ($\ell^\star$) applies the selected threshold at the fixed base gain, and Ours ($\ell^\star$+iso) adds iso-pressure restoration on top of the same threshold rule.

5.2 Results

Table 1 compares our approach with existing reasoning-efficiency methods across four base models. Ours ($\ell^\star$+iso) attains the highest average AE among the methods on three of the four base models. On DeepSeek-1.5B, Ours ($\ell^\star$) achieves the highest average AE, scoring 1.134 compared with 1.029 for Ours ($\ell^\star$+iso). In contrast, GRPO improves accuracy while reducing reasoning length in some settings, with no consistent gains across models. Compared with prior length-control methods, our approach reduces reasoning length while preserving task performance. The improvement is particularly pronounced on DeepSeek-1.5B, where Ours ($\ell^\star$) achieves an average accuracy of 69.4 with only 5,079 tokens, showing the largest accuracy gain with a substantial reduction in reasoning

Table 2: **Controlled comparison with SmartThinker under matched training settings.** We retrain SmartThinker using the same hyperparameters, rollout configuration, and optimization-step budget as our method to control for difference in post-training setup; full details are provided in Appendix B.1. ‡ denotes the retrained SmartThinker baseline under this matched setting.

Model	MATH500			AIME 24–26			AMC23			Average		
	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑
DeepSeek-R1-Distill-Qwen-1.5B												
SmartThinker‡	**86.4**	**2,210**	**1.152**	28.9	**7,811**	0.762	75.0	4,072	0.481	63.4	**4,698**	0.798
Ours ($\ell^\star$+iso)	85.4	2,293	1.096	**33.3**	8,610	**1.194**	**80.0**	**3,663**	**0.797**	**66.2**	4,855	**1.029**
DeepSeek-R1-Distill-Qwen-7B												
SmartThinker‡	95.4	**2,466**	0.335	51.1	10,098	-0.638	**92.5**	5,059	-0.083	79.7	5,874	-0.129
Ours ($\ell^\star$+iso)	**95.6**	2,515	**0.342**	**61.1**	**9,889**	**0.231**	**92.5**	**4,076**	**0.084**	**83.1**	**5,493**	**0.219**
Qwen3-1.7B												
SmartThinker‡	91.8	**4,393**	0.145	37.8	15,190	-0.666	**87.5**	**8,018**	**-0.102**	72.4	9,200	-0.208
Ours ($\ell^\star$+iso)	**92.0**	4,395	**0.163**	**45.6**	**14,639**	**0.218**	**87.5**	8,461	-0.151	**75.0**	**9,165**	**0.077**
Qwen3-4B-Thinking-2507												
SmartThinker‡	**98.2**	**3,700**	**0.306**	80.0	**13,563**	0.528	97.5	**5,776**	0.143	91.9	**7,680**	0.326
Ours ($\ell^\star$+iso)	97.8	3,879	0.276	**81.1**	14,129	**0.544**	**100.0**	5,953	**0.248**	**93.0**	7,987	**0.356**

Table 3: **Ablation of incorrect-conditioned threshold candidates.** Each row progressively adds the whole-incorrect and directional candidates to the marginal threshold. Results are averaged over five benchmarks per base model—reporting the AIME editions combined as in Table 1—and **Average** reports the mean across the four models.

Candidates	DeepSeek-1.5B			DeepSeek-7B			Qwen3-1.7B			Qwen3-4B			Average	
	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑	Acc.↑	AE↑
ℓ_{mar}	63.4	**4698**	0.798	79.7	5874	-0.129	72.4	9200	-0.208	91.9	7680	0.326	76.9	0.197
$+\ell_{\mathrm{inc}}$	65.4	5327	0.927	80.0	5701	-0.065	74.9	**9093**	-0.004	92.6	**7480**	**0.382**	78.2	0.310
$+\ell_{\mathrm{incL}}$, ℓ_{incS} (**Ours ($\ell^\star$)**)	**69.4**	5079	**1.134**	81.3	5851	0.037	**75.4**	9167	0.062	91.8	7810	0.333	**79.5**	0.392
+iso (**Ours ($\ell^\star$+iso)**)	66.2	4855	1.029	**83.1**	**5493**	**0.219**	75.0	9165	**0.077**	**93.0**	7987	0.356	79.3	**0.420**

length relative to the Base. On AIME, Ours ($\ell^\star$+iso) achieves the best or tied-best accuracy across all four base models while using fewer tokens than the corresponding base models. For instance, on the 7B model, it preserves an accuracy of 61.1 while reducing reasoning length from 13,103 to 9,889 tokens, and on Qwen3-4B it improves accuracy from 70.0 to 81.1 while shortening reasoning.

To control differences in post-training configuration, we additionally retrain SmartThinker under the same experimental settings as our method[1]. As shown in Table 2, Ours ($\ell^\star$+iso) achieves higher average accuracy and AE than SmartThinker‡ across all four base models. The advantage is particularly consistent on AIME, where Ours ($\ell^\star$+iso) improves both accuracy and AE for every model. These results show that the gains persist under matched training settings and cannot be explained solely by differences in rollout budget or training configuration.

5.3 Analysis

Incorrect-conditioned evidence improves threshold selection Table 3 shows that augmenting the marginal candidate with the pooled incorrect-conditioned candidate ℓ_{inc} improves average accuracy from 76.9 to 78.2 and AE from 0.197 to 0.310, without a uniform increase in generation length. Expanding the candidate set with the directional candidates ℓ_{incL} and ℓ_{incS} further raises mean accuracy to 79.5 and AE to 0.392.

Table 4 shows that, among mixed groups observed for Ours ($\ell^\star$+iso), the combined selection rate of ℓ_{inc}, ℓ_{incL}, and ℓ_{incS} is higher in models where $\sigma_I > \sigma_C$ is more prevalent. DeepSeek-1.5B has the highest condition and combined selection rates (66.4% and 36.2%), whereas Qwen3-4B has the lowest (14.8% and 3.5%). Nevertheless, even for Qwen3-4B, the candidate ablation shows higher average AE when all three incorrect-conditioned candidates are included than when using ℓ_{mar} alone (0.333 vs. 0.326; Table 3). Within this candidate set, the directional pair is selected less frequently than ℓ_{inc}. Its combined selection rates, measured over all rollout groups, are 6.4% and

[1]SmartThinker was originally trained with a different training configuration, including fewer rollouts per prompt (8 vs. 16); we match our training setup for a controlled comparison.

Table 4: **Incorrect-Conditioned Candidate Selection and Induced Pressure Relocation.** Statistics for Ours ($\ell^\star$+iso) within a ± 15-step window around the evaluated checkpoint. *All groups* uses all rollout groups as the denominator, whereas *Mixed groups* includes only groups containing both correct and incorrect rollouts. *Total* reports the frequency of any incorrect-conditioned selection case under the corresponding denominator. $\sigma_I > \sigma_C$ is the fraction of mixed groups in which incorrect-rollout dispersion exceeds correct-rollout dispersion. $\mathcal{G}_{P\downarrow}$ and $\mathcal{G}_{\ell\downarrow}$ denote the fractions of mixed groups with decreased total penalty and with $\ell^\star < \ell_{\mathrm{mar}}$, respectively; $\overline{\Delta P}_{P\downarrow}$ and $\overline{\Delta P}_{\ell\downarrow}$ report the mean penalty changes within the corresponding subsets. $P_{\mathrm{lost}}/P_{\mathrm{base}}$ measures the fraction of reference pressure removed, and $\rho_{\mathcal{B}}$ its batch-level restoration. See Appendix B.2 for details.

Model	Incorrect-conditioned selection						Pressure relocation					
	All groups				*Mixed groups*		*Mixed groups*					
	ℓ_{inc}	ℓ_{incL}	ℓ_{incS}	Total	$\sigma_I > \sigma_C$	Total	$\mathcal{G}_{P\downarrow}$	$\overline{\Delta P}_{P\downarrow}$	$\mathcal{G}_{\ell\downarrow}$	$\overline{\Delta P}_{\ell\downarrow}$	$P_{\mathrm{lost}}/P_{\mathrm{base}}$	$\rho_{\mathcal{B}}$
DeepSeek-1.5B	16.2%	3.0%	3.4%	22.6%	66.4%	36.2%	8.3%	-30.3%	27.1%	+24.4%	3.96%	1.039
DeepSeek-7B	7.4%	1.1%	1.4%	9.8%	44.3%	21.2%	3.9%	-21.1%	16.6%	+27.1%	1.05%	1.011
Qwen3-1.7B	3.5%	0.1%	0.2%	3.8%	30.3%	9.3%	2.6%	-21.6%	6.3%	+17.6%	0.55%	1.005
Qwen3-4B	0.9%	0.0%	0.1%	0.9%	14.8%	3.5%	0.6%	-66.7%	2.8%	+9.5%	0.46%	1.005

Note: Each model has 1,984 rollout groups. n_{mix}=1239, 920, 821, 508; $n_{P\downarrow}$=103, 36, 21, 3; and $n_{\ell\downarrow}$=336, 153, 52, 14, respectively. Selected groups with $\ell^\star = \ell_{\mathrm{mar}}$ belong to neither subset.

2.5% for DeepSeek-1.5B and DeepSeek-7B, respectively, and below 1% for both Qwen3 models. The ablation further supports the contribution of these directional candidates, showing higher mean accuracy and AE across the four models when they are added to $\ell_{\mathrm{mar}}, \ell_{\mathrm{inc}}$.

Iso-pressure restoration controls pressure changes induced by threshold relocation. Threshold relocation introduced by incorrect-conditioned threshold selection can unintentionally reduce penalty pressure for some rollout groups. As shown in Table 4, while downward corrections ($\mathcal{G}_{\ell\downarrow}$, $\overline{\Delta P}_{\ell\downarrow} > 0$) align with strengthening compression, upward shifts relax penalty pressure ($\mathcal{G}_{P\downarrow}$, $\overline{\Delta P}_{P\downarrow} < 0$). Such decreases are infrequent but can exceed 20% conditionally. Restoring this deficit locally per prompt would overly concentrate pressure changes on small rollout pools, risking local gradient distortion. We instead mitigate the lost pressure across the batch: while this introduces mild cross-prompt coupling, the batch-level pressure deficit remains below 4% of the reference pressure ($P_{\mathrm{lost}}/P_{\mathrm{base}} \leq 3.96\%$) for every model, and iso-pressure restores it with only modest rescaling ($\rho_{\mathcal{B}} \leq 1.039$). Thus, the per-rollout disruption remains minimal: adding iso-pressure leaves average accuracy nearly unchanged (79.5→79.3) while improving average AE from 0.392 to 0.420 (Appendix B.2 for design considerations regarding alternative pressure adjustment schemes).

6 Conclusion

We introduced OASIS, a reasoning-efficient RL method that leverages both correct and incorrect rollout lengths for better threshold selection with batch-level pressure preservation. Across four reasoning models and multiple mathematical reasoning benchmarks, OASIS reduces generation length by 32.7% while improving accuracy by 1.7% points on average, yielding a favorable accuracy–efficiency trade-off over prior length-control methods. Moreover, OASIS preserves the discriminative length evidence in incorrect rollouts that correct-only or pooled references can obscure, using directional references to capture heterogeneous failure patterns and a common criterion to select among threshold candidates, while iso-pressure restoration recovers compression pressure lost under upward threshold corrections. Future work can extend this principle beyond length to richer trajectory signals and other verifiable multi-step tasks with distinctive failure length profiles, such as agentic tool use and long-horizon code generation.

7 AI Use Statement

We used AI-based tools as limited supplementary assistance in preparing this paper. Specifically, AI tools were used to assist with: (i) formalizing mathematical claims and checking proof-related details for theories originally developed by the authors; (ii) translating text written by the authors, including their interpretations of experimental results; (iii) producing data visualizations for figures

using experimental data produced by the authors; (iv) organizing and drafting an initial paper outline based on the authors' research ideas and theoretical development; and (v) providing limited coding assistance during implementation.

The research questions, theoretical development, methodology, experimental design, implementation and algorithmic decisions, data analysis, interpretation of experimental results, and conclusions were developed and carried out by the authors. All content produced or revised with AI assistance, including selected mathematical content, code, text, and figures, was reviewed and verified by the authors. The authors take full responsibility for the final content of this work.

8 Reproducibility Statement

We provide a detailed description of the proposed algorithm, including its complete procedure, in Appendix C.3. Section 5.1 describes the experimental setup, while Appendix B.1 provides further implementation details required for reproduction, including hyperparameters, model configurations, and metric computation procedures. Appendix B.1 also specifies the publicly available model checkpoints and datasets used in our experiments, together with links to their corresponding Hugging Face repositories. These details are provided to support independent reimplementation and evaluation of our method.

References

Pranjal Aggarwal and Sean Welleck. L1: Controlling how long a reasoning model thinks with reinforcement learning. *arXiv preprint arXiv:2503.04697*, 2025.

Siwei Chen, Siqi Chen, Xupeng Miao, and Bin Cui. QLPO: Quadrant-weighted sampling for length-aware policy optimization. In *Conference on Language Modeling*, 2026. URL https://arxiv.org/abs/2607.21793.

Zhipeng Chen, Yingqian Min, Beichen Zhang, Jie Chen, Jinhao Jiang, Daixuan Cheng, Wayne Xin Zhao, Zheng Liu, Xu Miao, Yang Lu, Lei Fang, Zhongyuan Wang, and Ji-Rong Wen. An empirical study on eliciting and improving r1-like reasoning models. *arXiv preprint arXiv:2503.04548*, 2025.

Yingqian Cui, Pengfei He, Jingying Zeng, Hui Liu, Xianfeng Tang, Zhenwei Dai, Yan Han, Chen Luo, Jing Huang, Zhen Li, Suhang Wang, Yue Xing, Jiliang Tang, and Qi He. Stepwise perplexity-guided refinement for efficient chain-of-thought reasoning in large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 18581–18597, Vienna, Austria, July 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-acl.956. URL https://aclanthology.org/2025.findings-acl.956/.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Soumya Suvra Ghosal, Souradip Chakraborty, Avinash Reddy, Yifu Lu, Mengdi Wang, Dinesh Manocha, Furong Huang, Mohammad Ghavamzadeh, and Amrit Singh Bedi. Does thinking more always help? mirage of test-time scaling in reasoning models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5747.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. Token-budget-aware LLM reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 24842–24855, Vienna, Austria, July 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-acl.1274. URL https://aclanthology.org/2025.findings-acl.1274/.

Andreas Hochlehnert, Hardik Bhatnagar, Vishaal Udandarao, Samuel Albanie, Ameya Prabhu, and Matthias Bethge. A sober look at progress in language model reasoning: Pitfalls and paths to reproducibility. In *Second Conference on Language Modeling (COLM)*, 2025. URL https://openreview.net/forum?id=90UrTTxp50.

Bairu Hou, Yang Zhang, Jiabao Ji, Yujian Liu, Kaizhi Qian, Jacob Andreas, and Shiyu Chang. Thinkprune: Pruning long chain-of-thought of llms via reinforcement learning. *arXiv preprint arXiv:2504.01296*, 2025.

Chenzhi Hu, Qinzhe Hu, Yuhang Xu, Junyi Chen, Ruijie Wang, Shengzhong Liu, Jianxin Li, Fan Wu, and Guihai Chen. Smartthinker: Progressive chain-of-thought length calibration for efficient large language model reasoning. *arXiv preprint arXiv:2603.08000*, 2026.

Gang Li, Yulei Qin, Xiaoyu Tan, Dingkang Yang, Yuchen Shi, Zihan Xu, Xiang Li, Xing Sun, and Ke Li. RoRecomp: Enhancing reasoning efficiency via rollout response recomposition in reinforcement learning. *arXiv preprint arXiv:2509.25958*, 2025. URL https://arxiv.org/abs/2509.25958.

Gang Li, Yan Chen, Ming Lin, and Tianbao Yang. Drpo: Efficient reasoning via decoupled reward policy optimization. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026a.

Jiahui Li, Jianfeng Shan, Wenpei Chen, Shunyu Wu, Jian Lou, Wenjie Feng, Dan Li, and See-Kiong Ng. Exploiting verification-generation gap: Test-time reinforcement learning with confidence-conditioned verification, 2026b. URL https://arxiv.org/abs/2606.03608.

Zheng Li, Qingxiu Dong, Jingyuan Ma, Di Zhang, Kai Jia, and Zhifang Sui. Selfbudgeter: Adaptive token allocation for efficient llm reasoning. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 21135–21156, 2026c.

Zichao Li, Jie Lou, Fangchen Dong, Zhiyuan Fan, Mengjie Ren, Hongyu Lin, Xianpei Han, Debing Zhang, Le Sun, Yaojie Lu, and Xing Yu. Tackling length inflation without trade-offs: Group relative reward rescaling for reinforcement learning, 2026d. URL https://arxiv.org/abs/2603.10535.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, 2024.

Chang Liu, Yiran Zhao, Lawrence Liu, Yaoqi Ye, Csaba Szepesvári, and Lin F Yang. Laconic: Length-aware constrained reinforcement learning for llm. *arXiv preprint arXiv:2602.14468*, 2026a.

Wei Liu, Ruochen Zhou, Yiyun Deng, Yuzhen Huang, Junteng Liu, Yuntian Deng, Yizhe Zhang, and Junxian He. Learn to reason efficiently with adaptive length-based reward shaping. In *International Conference on Learning Representations*, volume 2026, pp. 43279–43308, 2026b.

Xin Liu and Lu Wang. Answer convergence as a signal for early stopping in reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 17896–17907, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.904. URL https://aclanthology.org/2025.emnlp-main.904/.

Haotian Luo, Haiying He, Yibo Wang, Shiwei Liu, Wei Li, Xiaochun Cao, Dacheng Tao, Naiqiang Tan, and Li Shen. O1-pruner: Length-harmonizing fine-tuning for o1-like reasoning pruning. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 14242–14257, San Diego, California, United States, July 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-acl.697. URL https://aclanthology.org/2026.findings-acl.697/.

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

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time

scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 20275–20321, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.1025. URL https://aclanthology.org/2025.emnlp-main.1025/.

Keqin Peng, Yuanxin Ouyang, Xuebo Liu, Zhiliang Tian, Ruijian Han, Yancheng Yuan, and Liang Ding. Think dense, not long: Dynamic decoupled conditional advantage for efficient reasoning. *arXiv preprint arXiv:2602.02099*, 2026.

Jinyan Su, Jennifer Healey, Preslav Nakov, and Claire Cardie. Between underthinking and overthinking: An empirical study of reasoning length and correctness in llms, 2025. URL https://arxiv.org/abs/2505.00127.

Taiqiang Wu, Zenan Xu, Bo Zhou, and Ngai Wong. The art of efficient reasoning: Data, reward, and optimization. *arXiv preprint arXiv:2602.20945*, 2026a. URL https://arxiv.org/abs/2602.20945.

Yuyang Wu, Yifei Wang, Ziyu Ye, Tianqi Du, Stefanie Jegelka, and Yisen Wang. When more is less: Understanding chain-of-thought length in LLMs. In *International Conference on Learning Representations*, 2026b.

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.

Jingyang Yi, Jiazheng Wang, and Sida Li. Shorterbetter: Guiding reasoning models to find optimal inference length for efficient reasoning. In *Advances in Neural Information Processing Systems*, 2025. doi: 10.52202/085713-1302.

Zewei Yu, Lirong Gao, Yuke Zhu, Bo Zheng, Junbo Zhao, Sheng Guo, and Haobo Wang. Stop unnecessary reflection: Training lrms for efficient reasoning with adaptive reflection and length coordinated penalty, 2026. URL https://arxiv.org/abs/2602.12113.

APPENDIX

A THEORETICAL DERIVATIONS & ALGORITHMIC FORMULATION

A.1 THRESHOLD ESTIMATION DETAILS

Throughout, (μ_C, σ_C^2) and (μ_I, σ_I^2) denote the empirical mean and variance of correct and incorrect reasoning lengths within a group, n_C and n_I denote the corresponding numbers of rollouts, and $\pi_C = n_C/(n_C + n_I)$. The derivations below assume $n_C, n_I > 0$; all-correct and all-incorrect groups are handled separately as described in Appendix A.6. We suppress the conditioning on x for readability.

Closed-Form Candidate Threshold and Boundary Cases For a correct-length density $p(L \mid C) = \mathcal{N}(\mu_C, \sigma_C^2)$ and a Gaussian reference density

$$p_{\text{ref}}(L) = \mathcal{N}(\mu_{\text{ref}}, \sigma_{\text{ref}}^2), \tag{17}$$

consider the contrast

$$h(L) = \log p(L \mid C) - \log p_{\text{ref}}(L). \tag{18}$$

This form provides a unified representation of the candidate reference corrections in the main text, which differ only in their reference moments. In particular, when $p_{\text{ref}}(L) = p(L \mid I)$, Equation 18 has the same maximizer as the posterior-odds objective in Equation 2, because

$$\log \frac{P(C \mid L, x)}{P(I \mid L, x)} = \log p(L \mid C, x) - \log p(L \mid I, x) + \log \frac{P(C \mid x)}{P(I \mid x)}, \tag{19}$$

where the final term is constant with respect to L.

Substituting the Gaussian log densities into Equation 18 gives

$$\begin{aligned} h(L) &= \left[-\frac{(L-\mu_C)^2}{2\sigma_C^2} - \frac{1}{2}\log(2\pi\sigma_C^2)\right] - \left[-\frac{(L-\mu_{\text{ref}})^2}{2\sigma_{\text{ref}}^2} - \frac{1}{2}\log(2\pi\sigma_{\text{ref}}^2)\right] \\ &= \frac{(L-\mu_{\text{ref}})^2}{2\sigma_{\text{ref}}^2} - \frac{(L-\mu_C)^2}{2\sigma_C^2} + \frac{1}{2}\log\frac{\sigma_{\text{ref}}^2}{\sigma_C^2}. \end{aligned} \tag{20}$$

Thus, $h(L)$ favors lengths that are more characteristic of the correct distribution than of the reference distribution. Expanding the squared terms yields

$$\begin{aligned} h(L) &= \left(\frac{1}{2\sigma_{\text{ref}}^2} - \frac{1}{2\sigma_C^2}\right) L^2 + \left(\frac{\mu_C}{\sigma_C^2} - \frac{\mu_{\text{ref}}}{\sigma_{\text{ref}}^2}\right) L + \text{const} \\ &= AL^2 + BL + \text{const}, \end{aligned} \tag{21}$$

where

$$A = \frac{1}{2\sigma_{\text{ref}}^2} - \frac{1}{2\sigma_C^2}, \qquad B = \frac{\mu_C}{\sigma_C^2} - \frac{\mu_{\text{ref}}}{\sigma_{\text{ref}}^2}. \tag{22}$$

Hence, the curvature is determined by the relative dispersion of the correct and reference distributions, whereas the linear term reflects their relative locations after variance normalization.

When $A < 0$, $h(L)$ is strictly concave, and its stationary point is the unique finite maximizer. Differentiating Equation 21 and solving $h'(L) = 2AL + B = 0$ yields

$$\text{vertex}(\mu_C, \sigma_C^2, \mu_{\text{ref}}, \sigma_{\text{ref}}^2) = \frac{\sigma_{\text{ref}}^2\mu_C - \sigma_C^2\mu_{\text{ref}}}{\sigma_{\text{ref}}^2 - \sigma_C^2}. \tag{23}$$

Equivalently, its offset from the correct mean is

$$\text{vertex} - \mu_C = \sigma_C^2 \frac{\mu_C - \mu_{\text{ref}}}{\sigma_{\text{ref}}^2 - \sigma_C^2}. \tag{24}$$

This expression makes the threshold adjustment explicit: μ_C provides the success reference, while the reference distribution induces a signed correction. When $\mu_{\text{ref}} > \mu_C$, the correction is negative; when $\mu_{\text{ref}} < \mu_C$, it is positive. Its magnitude is modulated by the relative dispersion of the two distributions. All four candidates in Equation 11 are instances of Equation 23 with different reference moments. For a fixed reference distribution with $\sigma_{\text{ref}}^2 > 0$, the offset in Equation 24 vanishes as $\sigma_C^2 \to 0$: a group whose correct lengths are tightly concentrated receives a threshold close to the correct mean.

Boundary cases. Three regimes exhaust all possibilities. If $\sigma_C^2 < \sigma_{\text{ref}}^2$, then $A < 0$, h is strictly concave, and Equation 23 is its unique finite maximizer. If $\sigma_C^2 = \sigma_{\text{ref}}^2$, then $A = 0$, so h is affine with slope B and has no interior maximizer unless $B = 0$, in which case it is constant. If $\sigma_C^2 > \sigma_{\text{ref}}^2$, then $A > 0$, h is strictly convex, the vertex is its minimum, and $\sup_L h(L)$ is attained only as $L \to \pm\infty$. Thus, in the latter two regimes, the reference does not define a finite threshold through this unconstrained Gaussian objective. In these regimes we make a conservative boundary choice on the observed rollout-length support $[L_{\min}, L_{\max}]$, the smallest and largest rollout lengths in the group: $\ell_{\text{disc}} = L_{\max}$ when the curvature vanishes with $\mu_C < \mu_{\text{ref}}$, protecting already efficient, shorter reasoning trajectories from excessive penalties, and $\ell_{\text{disc}} = L_{\min}$ otherwise (Algorithm 1).

Fallback is subsumed by candidate selection. These degenerate regimes require no separate handling in the method. For a single-Gaussian reference, the curvature gate of Appendix A.3 reduces exactly to $\sigma_C^2 < \sigma_{\text{ref}}^2$, so a reference in a degenerate regime is not selected when another valid candidate passes the gate. If no candidate passes the curvature gate, ℓ_{mar} remains the fallback threshold. Thus, failure of the curvature gate does not by itself remove the length signal: mixed groups can still receive length shaping through $\alpha_q > 0$ when ℓ_{mar} is used as the fallback. Only all-correct and all-incorrect groups have zero length-shaping signal, as specified in Appendix A.6.

A.2 Marginal Reference: Validity and Correction Attenuation

When incorrect rollouts are insufficient for conditional estimation. If the group contains too few incorrect responses to estimate a separate incorrect-class distribution, the only distribution defined for the group is the marginal $p(L)$. Then

$$\log p(L \mid C) - \log p(L) = \log P(C \mid L) - \log P(C), \tag{25}$$

so

$$\ell_{\text{mar}} = \arg\max_L P(C \mid L), \tag{26}$$

and the marginal criterion is the valid solution.

When an incorrect class is available. Otherwise the marginal is the mixture

$$p_{\text{mar}}(L) = \pi_C\, p(L \mid C) + (1 - \pi_C)\, p(L \mid I). \tag{27}$$

Write

$$D = \mu_I - \mu_C, \qquad R = \sigma_I^2 - \sigma_C^2. \tag{28}$$

The mixture moments are

$$\mu_{\text{mar}} - \mu_C = (1 - \pi_C)D, \tag{29}$$

$$\sigma_{\text{mar}}^2 - \sigma_C^2 = (1 - \pi_C)\left[R + \pi_C D^2\right], \tag{30}$$

the second by the law of total variance. When $R > 0$, applying Equation 23 to both references,

$$\ell_{\text{inc}} - \mu_C = -\frac{\sigma_C^2 D}{R}, \tag{31}$$

$$\ell_{\text{mar}} - \mu_C = -\frac{\sigma_C^2 D}{R + \pi_C D^2}, \tag{32}$$

so the two corrections are proportional:

$$\ell_{\text{mar}} - \mu_C = \kappa\,(\ell_{\text{inc}} - \mu_C), \qquad \kappa = \frac{R}{R + \pi_C D^2} \in (0, 1], \tag{33}$$

equivalently

$$\ell_{\text{mar}} - \ell_{\text{inc}} = \frac{\sigma_C^2\, \pi_C D^3}{R(R + \pi_C D^2)}. \tag{34}$$

Equation 33 characterizes the attenuation exactly: substituting the marginal for the incorrect-conditioned reference never changes the sign of the correction and always shrinks its magnitude, by a factor κ that decreases monotonically in both the correct fraction π_C and the class separation $|D|$. Thus, at fixed conditional distributions, larger π_C suppresses the conditional correction more strongly, as does greater one-sided class separation. The displacement in Equation 34 carries the sign of D: for $D > 0$ the marginal weakens the downward correction, whereas for $D < 0$ it weakens the upward correction.

When the conditional reference fails the concavity condition. When $R \leq 0$, the incorrect-conditioned Gaussian contrast is affine ($R = 0$) or strictly convex ($R < 0$), and therefore has no finite interior maximizer. The marginal reference can nevertheless remain valid, as shown by Equation 32. In particular, if $R + \pi_C D^2 > 0$, then $\sigma^2_{\text{mar}} > \sigma^2_C$, so the marginal contrast is strictly concave and admits a finite interior maximizer. Thus, the between-class separation term $\pi_C D^2$ can restore concavity even when the incorrect-conditioned reference fails the concavity gate, providing a theoretical rationale for using the marginal reference as a fallback.

A.3 Curvature Gate

To evaluate candidate thresholds against the incorrect-length distribution, we model incorrect rollouts using a Gaussian mixture density $\tilde{p}_I(L \mid x) = \sum_k w_k \mathcal{N}(L; m_k, v_k)$ and consider the mixture contrast objective (Equation 10). For any candidate length L, let $r_k(L) = w_k \mathcal{N}(L; m_k, v_k)/\tilde{p}_I(L \mid x)$ denote the local posterior responsibility of component k, and write $z_k(L) = (L - m_k)/v_k$. Evaluating the curvature of $J(L)$ requires the second derivative of the mixture log-density:

$$\frac{d}{dL} \ln \tilde{p}_I(L \mid x) = \sum_k r_k(L) \frac{d}{dL} \ln \mathcal{N}(L; m_k, v_k) = -\mathbb{E}_r[z_k], \tag{35}$$

where $\mathbb{E}_r[f(k)] = \sum_k r_k(L) f(k)$. Because r_k is a normalized weight, its derivative is

$$\frac{dr_k}{dL} = r_k(L)\big(\mathbb{E}_r[z_k] - z_k(L)\big), \tag{36}$$

reflecting that responsibility shifts toward the component whose local log-density grows faster than the mixture average. By the product rule and $dz_k/dL = 1/v_k$,

$$\frac{d^2}{dL^2} \ln \tilde{p}_I(L \mid x) = -\sum_k \frac{dr_k}{dL} z_k - \sum_k \frac{r_k}{v_k} = \text{Var}_r[z_k] - \mathbb{E}_r\left[\tfrac{1}{v_k}\right]. \tag{37}$$

Since $\frac{d^2}{dL^2} \ln \mathcal{N}(L; \mu_C, \sigma^2_C) = -1/\sigma^2_C$,

$$J''(L) = -\frac{1}{\sigma^2_C} + \mathbb{E}_r\left[\frac{1}{v_k}\right] - \text{Var}_r[z_k(L)]. \tag{38}$$

The three terms correspond to the concavity of the correct-length Gaussian, the responsibility-weighted precision of the incorrect components at L, and the components' disagreement about L's standardized position. The curvature gate retains only candidates satisfying $J''(\ell) < 0$, our local concavity criterion for the mixture objective. For a single Gaussian reference, $\text{Var}_r[z_k] = 0$ and Equation 38 reduces exactly to $\sigma^2_C < \sigma^2_{\text{ref}}$, the concavity condition of Appendix A.1. Since the candidate thresholds are not necessarily stationary points of the joint Gaussian mixture objective, the curvature gate serves only as a local concavity filter to reject obviously unsuitable candidates rather than as evidence that the proposal is a local maximum of $J(L)$.

A.4 Why Four Candidates, and Why Not Optimize J Directly

Completeness under the moment specification. Given the observed joint signal (L, C), we restrict each reference distribution to its first two group moments; the corresponding Gaussian is the maximum-entropy distribution consistent with this information. Under this specification, we consider four distinct reference choices: the marginal candidate ℓ_{mar} applies when no incorrect class is available, the pooled candidate ℓ_{inc} captures the whole-incorrect correction, while ℓ_{incS} and ℓ_{incL} capture short- and long-failure corrections when directional resolution is useful. Further splitting either side of μ_C can change only the correction magnitude, not its sign, which is determined by $\text{sign}(\mu_C - \mu_{\text{ref}})$ in Equation 24. We do not split correct responses because a correct trace establishes that its realized length was sufficient, but provides no counterfactual evidence that the same length was necessary.

Why not maximize J directly. Although J is the common selection objective, directly maximizing it is neither available in closed form nor reliably stable at the group level: its Gaussian–mixture

form can contain small-sample-induced local maxima, and numerical optimization would be required for every rollout group at every training step. Instead, the four candidates provide closed-form proposals from moment-consistent Gaussian references. We evaluate these proposals under the full objective J and retain only proposals lying in locally concave regions through the $J'' < 0$ gate (after range clamping). This preserves the shared-mixture evidence used for selection while avoiding per-group numerical search and its sensitivity to spurious maxima.

A.5 Motivation for Soft Responsibilities over Hard Partitioning

One might consider partitioning incorrect rollouts simply by thresholding at μ_C (i.e., short vs. long). However, hard truncation introduces artificial separation even under the null. To see this, suppose the underlying rollout length is unimodal and centered at the correct mean with no directional skew: $L \sim \mathcal{N}(\mu_C, \sigma^2)$. Partitioning strictly at μ_C yields upper and lower halves. Let

$$Z = \frac{L - \mu_C}{\sigma} \sim \mathcal{N}(0, 1). \tag{39}$$

For the upper half,

$$\mathbb{E}[Z \mid Z > 0] = \sqrt{\frac{2}{\pi}}, \qquad \mathrm{Var}[Z \mid Z > 0] = 1 - \frac{2}{\pi}. \tag{40}$$

It follows that

$$\mathbb{E}[L \mid L > \mu_C] = \mu_C + \sigma\sqrt{\frac{2}{\pi}}, \qquad \mathrm{Var}[L \mid L > \mu_C] = \sigma^2\left(1 - \frac{2}{\pi}\right). \tag{41}$$

By symmetry,

$$\mathbb{E}[L \mid L < \mu_C] = \mu_C - \sigma\sqrt{\frac{2}{\pi}}, \tag{42}$$

with the same conditional variance. Hence, the conditional means are separated by $2\sigma\sqrt{\frac{2}{\pi}}$, while each conditional variance is smaller than that of the original distribution. Soft responsibilities attenuate, but do not eliminate, this effect. Under the null, monotone logistic weighting still induces separated responsibility-weighted means and reduced component variances. Compared with hard truncation, however, soft logistic weighting yields weaker mean displacement and variance contraction. These residual directional proposals are further filtered by the concavity gate: when the null variance satisfies $\sigma^2 \le \sigma_C^2$, the contracted component variances cannot satisfy the eligibility condition $\sigma_C^2 < v_k$. Thus, soft weighting reduces spurious separation, while the concavity filter rejects the resulting null-induced directional candidates in this regime.

A.6 Reward Details

For a selected threshold $\ell^\star$, define the correct-only excess

$$\mathrm{over}_i = \begin{cases} [L_i - \ell^\star]_+, & c_i = 1, \\ 0, & c_i = 0. \end{cases} \tag{43}$$

For groups containing both correct and incorrect responses, the group-level scale at threshold b is

$$\alpha_q(b) = \frac{1 - w}{\max\limits_{i:c_i=1} \mathrm{over}_i - \overline{\mathrm{over}} + \epsilon_\alpha}, \tag{44}$$

where $1 - w = n_I/n$ is the empirical error rate, $\overline{\mathrm{over}}$ is the group mean excess (with incorrect responses contributing zero), all excess statistics are computed at threshold b, and $\epsilon_\alpha > 0$ is a numerical stabilizer. The maximum over correct responses dominates a mean that includes zeros, so the denominator is nonnegative. The numerator scales the pressure with the group's error rate; the denominator normalizes raw token-scale variation, making the gain g comparable across groups and prompts.

Groups without a correct–incorrect contrast receive no length signal. For all-correct and all-incorrect groups we set the length-shaping term to zero (equivalently $\alpha_q(b) = 0$), which also avoids undefined correct-only statistics in all-incorrect groups.

Table 5: **Experimental configuration.** Settings are shared across models and methods unless otherwise specified.

Setting	Value
Training	
Prompt batch size	64
Rollouts per prompt (G)	16
PPO mini-batch size	16
Learning rate	1×10^{-6}
Max. prompt / response length	2,048 / 8,192
Sampling temperature	0.6
Entropy coefficient	0
KL coefficient	0
Optimizer	AdamW
Length reward	
Minimum incorrect samples (`min_inc`)	2
Gain (g)	1.0
Restoration ceiling ($g_{\max}$)	2.0 (Ours ($\ell^\star$+iso) only)
Numerical stabilizer (ϵ)	10^{-6}

B Experimental Setup & Implementation Details

B.1 Implementation and metric details

Models and training. We train DeepSeek-R1-Distill-Qwen-1.5B[2], DeepSeek-R1-Distill-Qwen-7B[3], Qwen3-1.7B[4], and Qwen3-4B-Thinking-2507[5] with GRPO on DeepScaleR (Luo et al., 2025)[6], a dataset of 40K competition mathematics problems. Within each model, all methods are evaluated at the same pre-specified checkpoint, avoiding method-specific checkpoint selection. DeepSeek-1.5B is evaluated later because its base AIME responses are substantially longer (36.1K vs. 13.1K–16.8K tokens for the other models). Length-reward hyperparameters are shared across models rather than tuned individually, as the group-level normalization α removes the token scale from the penalty. Table 5 summarizes the experimental configuration.

Evaluation and grading. We evaluate AIME 2024, AIME 2025, AIME 2026, AMC23, and MATH500 using five samples per problem at temperature 0.6 and top-p 1.0. Accuracy is `majority@5`; length is the mean length of the five samples, and both quantities are then averaged over problems. The evaluation budget is approximately 200K tokens, preventing harness truncation. A single rollout that collapses into repetition and saturates the generation cap can distort AE, so we clip 100K tokens when calculating AE, which occasionally occurred only in incorrect cases. We use `majority@5`, rather than `pass@`k, because it discounts one-off lucky samples and measures solution stability when methods differ in response length.

A single symbolic-equivalence grader is used for all systems, including released checkpoints. It extracts LaTeX and plain mathematical expressions from the full response, treating boxed answers, closing-sentence answers, and `\dfrac` and `\frac` variants equivalently. This avoids penalizing compressed responses for formatting changes.

AE computation. We use the accuracy–efficiency (AE) score of DeepScaleR (Luo et al., 2025). With ΔL denoting relative mean-length reduction and ΔA relative accuracy change, both relative

[2]https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-1.5B
[3]https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-7B
[4]https://huggingface.co/Qwen/Qwen3-1.7B
[5]https://huggingface.co/Qwen/Qwen3-4B-Thinking-2507
[6]https://huggingface.co/datasets/agentica-org/DeepScaleR-Preview-Dataset

to the model's base,

$$\mathrm{AE} \;=\; \begin{cases} \Delta L + 3\,\Delta A, & \Delta A \geq 0, \\ \Delta L - 5\,|\Delta A|, & \Delta A < 0. \end{cases} \tag{45}$$

Thus, AE penalizes accuracy loss more heavily than an equal-sized compression gain.

For AE, the length of each problem is averaged only over rollouts agreeing with its majority verdict: correct rollouts when the problem is correct and incorrect rollouts otherwise. The base model is processed identically. This quantity differs from the length columns of Table 1, which are plain averages over all five samples.

For the combined AIME result, accuracy and length are first averaged across AIME 2024–2026, and AE is then computed once from these combined values. Because AE is piecewise linear in relative changes, averaging edition-level AE scores would not equal the defined combined value. Per-edition results are reported in Table 7.

B.2 Iso-Pressure Details

Measurement conventions. The diagnostics of Tables 4 and 6 are recomputed from the stored training rollouts with the trainer's own rule. [7]

Pressure Geometry For group q, threshold b, and gain g, define the pressure vector over correct responses as $\boldsymbol{\pi}(b, g) = g\,\alpha_q(b)\,\mathbf{v}(b)$ with $v_i(b) = [L_i - b]_+$. At fixed b, $\boldsymbol{\pi}(b, cg) = c\,\boldsymbol{\pi}(b, g)$: gain rescales pressure. The zero-pressure set $\{i : c_i = 1,\ L_i \leq b\}$, by contrast, depends only on b; if $b_1 < b_2$, a correct response with $L_i \in (b_1, b_2]$ is penalized under b_1 and exempt under b_2 for every positive gain. Gain scaling therefore cannot reproduce threshold relocation, which is why threshold placement and pressure magnitude are controlled separately in the main text.

Pressure Conservation Group pressure is non-increasing in the threshold. The accounting below partitions the batch by the direction in which the threshold moved, so it relies on the group pressure $P_q(b)$ of Equation 13 being monotone in b. This is not immediate: raising b shrinks the excess lengths in the numerator of P_q, but it also shrinks the denominator of $\alpha_q(b)$ in Equation 44, so that $\alpha_q(b)$ itself *increases*. The two effects do not cancel, and the numerator always dominates.

Fix a group q with at least one correct and one incorrect response, write n for its size, and let

$$S(b) = \sum_{i:c_i=1} [L_i - b]_+, \qquad L_{\max} = \max_{i:c_i=1} L_i, \qquad k(b) = \#\{i : c_i = 1,\ L_i > b\}. \tag{46}$$

Then $\max_{i:c_i=1} \mathrm{over}_i = [L_{\max} - b]_+$ and $\overline{\mathrm{over}} = S(b)/n$, so the denominator of Equation 44 is $M(b) + \epsilon_\alpha$ with $M(b) = [L_{\max} - b]_+ - S(b)/n$, and

$$P_q(b) = \alpha_q(b)\,S(b) = (1 - w)\,\frac{S(b)}{M(b) + \epsilon_\alpha}. \tag{47}$$

For $b < L_{\max}$, each of the $k(b)$ positive terms of $S(b)$ is at most $L_{\max} - b$, so

$$S(b) \;\leq\; k(b)\,(L_{\max} - b). \tag{48}$$

Since $k(b) \leq n$, Equation 48 gives $S(b)/n \leq L_{\max} - b$ and hence $M(b) \geq 0$, so the denominator of Equation 47 is bounded below by $\epsilon_\alpha > 0$ and P_q is well defined. Because $1 - w$ does not depend on b, it suffices to sign the derivative of $S/(M + \epsilon_\alpha)$.

S is continuous and piecewise linear in b; on any interval containing no correct length, $S'(b) = -k(b)$ and, for $b < L_{\max}$, $M'(b) = -1 + k(b)/n$. Writing $N(b)$ for the numerator of that derivative,

$$\begin{aligned} N(b) &= S'(b)\big(M(b) + \epsilon_\alpha\big) - S(b)\,M'(b) \\ &= -k(b)\big(M(b) + \epsilon_\alpha\big) + S(b)\left(1 - \frac{k(b)}{n}\right) \\ &= S(b) - k(b)\left(M(b) + \frac{S(b)}{n}\right) - k(b)\,\epsilon_\alpha \\ &= S(b) - k(b)\big(L_{\max} - b\big) - k(b)\,\epsilon_\alpha \;\leq\; -k(b)\,\epsilon_\alpha \;\leq\; 0, \end{aligned} \tag{49}$$

[7] Entries are rounded independently from the group counts they are formed from, so a total need not follow from the rounded columns above it.

where the third equality uses $M(b) + S(b)/n = L_{\max} - b$ and the inequality is Equation 48. Thus P_q is non-increasing on every smooth piece and, being continuous, non-increasing in b throughout; it is strictly decreasing wherever $k(b) \geq 1$ and identically zero for $b \geq L_{\max}$. Consequently $b_1 < b_2$ implies $P_q(b_1) \geq P_q(b_2)$: an upward threshold correction can only remove pressure and a downward one can only add it.

Let $P_q^{\text{base}} = P_q(\ell_{\text{mar},q})$ and $P_q^{\text{new}} = P_q(\ell_q^\star)$, and partition the batch into $\mathcal{G}_\uparrow = \{q : \ell_q^\star > \ell_{\text{mar},q}\}$, $\mathcal{G}_=$, and $\mathcal{G}_\downarrow$ analogously. With $P_q^{\text{lost}} = \max(0, P_q^{\text{base}} - P_q^{\text{new}})$ we have $P_q^{\text{new}} + P_q^{\text{lost}} = P_q^{\text{base}}$ for $q \in \mathcal{G}_\uparrow \cup \mathcal{G}_=$ and $P_q^{\text{lost}} = 0$ for $q \in \mathcal{G}_\downarrow$, both by the monotonicity just established. When neither the gain cap $g_{\max}$ nor the reward floor ϵ_r is active, the total applied penalty under $g_{\text{iso}} = g\rho_{\mathcal{B}}$ is

$$\sum_i \text{penalty}_i^{\text{iso}} = g\rho_{\mathcal{B}}\, P_{\text{new}}^{\mathcal{B}} = g\left(P_{\text{new}}^{\mathcal{B}} + P_{\text{lost}}^{\mathcal{B}}\right) \tag{50}$$

$$= g\sum_q P_q^{\text{base}} + g\sum_{q\in\mathcal{G}_\downarrow}\left(P_q^{\text{new}} - P_q^{\text{base}}\right). \tag{51}$$

Pressure removed by upward corrections is exactly restored at the batch level, while additional pressure created by downward corrections is retained; if $\mathcal{G}_\downarrow$ is empty, the total matches the pre-correction budget exactly. When the floor ϵ_r truncates individual penalties the identity becomes an upper bound on applied pressure; the cap $g_{\max}$ likewise makes restoration partial by design.

Gain cap and the range of $\rho_{\mathcal{B}}$ Taken over every training step rather than the evaluation window of Table 6, the restoration factor $\rho_{\mathcal{B}}$ has a mean of 1.012–1.026 and a maximum of 1.909, still below $g_{\max} = 2.0$, so clipping never binds and does not produce the reported behavior. The evaluation windows need not fall inside the range of those means, and none of them does: DeepSeek-1.5B's (1.039) lies above it, and the other three below (1.011 on DeepSeek-7B, 1.005 on both Qwen3 models). The maximum also shows that the mean understates the intervention, since on individual steps the restored gain approaches twice the base gain.

Design Considerations for Alternative Pressure Adjustment Schemes Our batch-level restoration ($\rho_{\mathcal{B}}$) favors instantaneous spatial amortization over temporal or state-dependent coupling to prioritize training stability. While dynamically pacing length pressure across training steps is an intuitive alternative, our design explorations revealed structural drawbacks under such dependencies. First, a dual-window Exponential Moving Average (EMA) controller—comparing short-term (5-batch) and long-term (20-batch) rolling accuracy—suffered from temporal hysteresis. Because prompt difficulty naturally fluctuates across mini-batches, historical momentum lagged behind the instantaneous training states: lingering high pressure over-penalized complex reasoning on harder batches (degrading accuracy), while sluggish ramp-up delayed compression on simpler ones. Second, scaling penalty pressure in direct proportion to prompt accuracy (inverting the difficulty modulation of $\alpha_q(b)$ in Equation 13) created a destabilizing positive feedback loop under GRPO: as accuracy improved, escalating length penalties prematurely truncated essential verification steps, stalling convergence. These observations indicate that dynamic or history-dependent heuristics exacerbate mini-batch non-stationarity, leading to optimization instability. In contrast, instantaneous batch-level amortization provides a principled and robust design: it resolves upward-correction deficits strictly within the current step without temporal lag, runaway feedback loops, or localized gradient shocks.

C Extended Analysis and Results

C.1 Truncation and Selected candidate Details

Truncation details. A rollout truncated at the 8,192-token training limit is incorrect by construction and is no shorter than any correct rollout in its group. It is therefore the one case in which the observed rollout provides direct directional evidence that extending the rollout did not produce a correct outcome within the training budget. Because truncated rollouts are incorrect, an approach that uses only correct rollouts cannot exploit this information for length adjustment. Our rule instead allows these failed rollouts to contribute to the downward length correction.

However, the observed length of a truncated rollout is right-censored: its uncensored length is only known to be at least 8,192. Treating such rollouts as observations exactly at the context limit can

Table 6: **Training dynamics of threshold selection and pressure restoration.** Training-log statistics within a ± 15-step window around the evaluated checkpoint (1,984 rollout groups per run). Trunc. denotes the fraction of rollouts reaching the 8,192-token training limit, and e the group error rate. *All groups* uses all rollout groups as the denominator, whereas *Mixed groups* includes only groups containing both correct and incorrect rollouts. The incorrect-conditioned columns report selection frequencies over all groups, while *Total* reports the frequency of any such selection among mixed groups. $P_{\text{lost}}/P_{\text{base}}$ measures the fraction of reference length pressure removed by threshold relocation, and $\rho_{\mathcal{B}}$ its batch-level iso-pressure restoration factor. Ours ($\ell^\star$) uses $\rho_{\mathcal{B}} = 1$, whereas Ours ($\ell^\star$+iso) applies restoration. SmartThinker‡ always uses the marginal threshold, so the corresponding selection and restoration statistics are undefined. ‡ detnotes prior method retrained under our experimental setting. The lowest truncation rate for each model is shown in **bold**, with the runner-up underlined. See Appendix B.2 for details.

Method	Training dynamics			Incorrect-conditioned selection				Pressure relocation	
				All groups			*Mixed groups*	*Mixed groups*	
	Trunc.↓	Mix.	e	ℓ_{inc}	ℓ_{incL}	ℓ_{incS}	Total	$P_{\text{lost}}/P_{\text{base}}$	$\rho_{\mathcal{B}}$
DeepSeek-R1-Distill-Qwen-1.5B									
SmartThinker‡	12.1%	62.2%	29.9%	-	-	-	-	-	-
Ours ($\ell^\star$)	11.4%	61.5%	29.6%	15.8%	3.0%	3.6%	36.4%	3.16%	1.000
Ours ($\ell^\star$+iso)	**10.1%**	62.4%	29.4%	16.2%	3.0%	3.4%	36.2%	3.96%	1.039
DeepSeek-R1-Distill-Qwen-7B									
SmartThinker‡	17.1%	46.3%	18.5%	-	-	-	-	-	-
Ours ($\ell^\star$)	**16.7%**	47.3%	18.1%	7.1%	0.9%	1.8%	20.7%	1.75%	1.000
Ours ($\ell^\star$+iso)	17.0%	46.4%	18.0%	7.4%	1.1%	1.4%	21.2%	1.05%	1.011
Qwen3-1.7B									
SmartThinker‡	35.0%	41.4%	18.7%	-	-	-	-	-	-
Ours ($\ell^\star$)	36.0%	42.4%	18.4%	2.4%	0.3%	0.6%	7.5%	1.04%	1.000
Ours ($\ell^\star$+iso)	**34.4%**	41.4%	18.6%	3.5%	0.1%	0.2%	9.3%	0.55%	1.005
Qwen3-4B-Thinking-2507									
SmartThinker‡	23.7%	25.4%	11.8%	-	-	-	-	-	-
Ours ($\ell^\star$)	**23.6%**	25.2%	11.3%	0.8%	0.0%	0.1%	3.4%	1.39%	1.000
Ours ($\ell^\star$+iso)	25.5%	25.6%	11.6%	0.9%	0.0%	0.1%	3.5%	0.46%	1.005

compress the upper tail and underestimate the uncensored rollout-length variance. Since candidate eligibility depends on the relative variances in OASIS, this bias may reduce the measured eligibility rate and affect variance-dependent threshold corrections. Thus, while truncated failures provide valid directional evidence for length adjustment, their empirical variance should be interpreted cautiously, particularly for models with high truncation rates.

C.2 Per-benchmark Accuracy–Efficiency Comparisons

Table 1 reports the complete per-model, per-benchmark results. Here we report the comparisons that are less visible in the aggregate summary.

On MATH500, Ours ($\ell^\star$+iso) exceeds the retrained baseline on three of the four models: 0.342 versus 0.335 on DeepSeek-7B, 0.276 versus 0.255 on Qwen3-4B, and 0.163 versus 0.145 on Qwen3-1.7B; Ours ($\ell^\star$) attains 0.188 on the latter. The remaining model is DeepSeek-1.5B, where the three compared systems are within five correct problems out of 500 (86.0%, 85.4% and 86.4%) and the retrained baseline achieves the highest AE. On Qwen3-4B the released SmartThinker checkpoint reaches a higher AE (0.355) than either of our settings, so our margin there is over the retrained baseline only.

On AMC23, Ours ($\ell^\star$) has the highest AE on all four models (1.172, 0.210, 0.282, and -0.003). On DeepSeek-1.5B, Ours ($\ell^\star$+iso) retains the base accuracy of 80.0% while attaining the shortest length among all systems (3,663 tokens). AMC23 contains 40 problems, so per-benchmark orderings are sensitive to individual items.

Table 7: **Accuracy and reasoning efficiency on AIME benchmarks across base models.** We report accuracy (Acc.), average reasoning length (Len.), and accuracy-efficiency score (AE) on AIME 24–26. Within each model block, the best and second-best results in each column are shown in **bold** and underlined, respectively. † denotes an officially released checkpoint and ‡ prior method retrained under our experimental setting.

Method	AIME 2024			AIME 2025			AIME 2026		
	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑
DeepSeek-R1-Distill-Qwen-1.5B									
Base	40.0	33,860	-	23.3	38,243	-	23.3	36,204	-
GRPO (no len. reward)	33.3	32,045	-0.998	23.3	26,558	0.219	23.3	27,825	0.166
LASER†	**43.3**	12,700	0.845	26.7	11,709	1.089	23.3	12,039	0.661
ThinkPrune†	30.0	10,337	-0.588	23.3	7,417	0.787	20.0	10,110	-0.018
SmartThinker†	36.7	15,170	0.081	26.7	15,284	1.002	23.3	12,893	0.616
SmartThinker‡	40.0	**7,718**	0.738	23.3	6,830	0.800	23.3	8,886	0.745
Ours ($\ell^\star$)	**43.3**	10,629	**0.924**	**30.0**	**6,331**	**1.677**	23.3	9,161	0.741
Ours ($\ell^\star$+iso)	40.0	9,918	0.650	**30.0**	8,009	1.627	**30.0**	**7,903**	**1.624**
DeepSeek-R1-Distill-Qwen-7B									
Base	70.0	12,147	-	**50.0**	13,201	-	**63.3**	13,960	-
GRPO (no len. reward)	63.3	15,114	-0.663	**50.0**	14,334	**-0.047**	56.7	11,424	-0.360
LASER†	63.3	**9,827**	-0.280	40.0	**9,069**	-0.684	50.0	10,413	-0.801
SmartThinker‡	60.0	10,627	-0.659	36.7	10,413	-1.161	56.7	9,254	-0.236
Ours ($\ell^\star$)	63.3	10,106	-0.408	40.0	11,544	-0.923	60.0	10,783	0.011
Ours ($\ell^\star$+iso)	**73.3**	9,996	**0.301**	46.7	10,651	-0.166	**63.3**	**9,018**	**0.354**
Qwen3-1.7B									
Base	**53.3**	16,619	-	**43.3**	16,919	-	36.7	16,956	-
GRPO (no len. reward)	50.0	15,683	-0.251	36.7	16,415	-0.758	**46.7**	15,405	0.913
SmartThinker‡	46.7	15,477	-0.584	30.0	14,924	-1.435	36.7	15,169	0.106
Ours ($\ell^\star$)	**53.3**	14,376	**0.139**	40.0	14,934	-0.249	36.7	**15,042**	0.101
Ours ($\ell^\star$+iso)	50.0	**14,352**	-0.181	40.0	**14,414**	**-0.225**	**46.7**	15,153	**0.956**
Qwen3-4B-Thinking-2507									
Base	73.3	14,316	-	66.7	16,056	-	70.0	15,329	-
GRPO (no len. reward)	80.0	13,711	0.287	80.0	16,305	0.597	76.7	18,662	0.168
SmartThinker†	80.0	12,267	0.404	76.7	**14,270**	0.571	73.3	13,976	0.216
SmartThinker‡	**83.3**	12,577	**0.508**	80.0	14,345	0.699	76.7	**13,766**	0.387
Ours ($\ell^\star$)	80.0	**11,850**	0.439	73.3	14,868	0.357	**80.0**	15,509	**0.487**
Ours ($\ell^\star$+iso)	80.0	12,953	0.357	**83.3**	15,048	**0.819**	**80.0**	14,387	0.479

C.3 Algorithmic Specification

Algorithm 1 selects the final operational threshold for one rollout group, and Algorithm 2 assembles the final rewards for one training batch. When fewer than two incorrect rollouts are available, only the marginal threshold is used. With sufficient incorrect evidence, pooled and soft directional candidates are constructed whenever their corresponding closed-form thresholds are finite and locally concave. Only candidates that actually exist are inserted into $\mathcal{C}$ and evaluated under the common incorrect mixture. If no candidate passes the curvature gate, the initial marginal threshold ℓ_{mar} is retained. Note, ℓ_k denotes ℓ_{incS} when $k = \mathrm{short}$ and ℓ_{incL} when $k = \mathrm{long}$, where ℓ_{incS} and ℓ_{incL} are defined in Equation 11 in Algorithm 1. Ours ($\ell^\star$+iso) uses Algorithm 2; Ours ($\ell^\star$) follows the same procedure with $\rho_{\mathcal{B}} = 1$.

Table 8: **Per-benchmark decomposition of the candidate ablation.** The aggregate columns of Table 3 resolved into the five individual benchmarks. Each row progressively augments the marginal threshold with the whole-incorrect candidate and the directional pair.

Candidates	AIME 2024			AIME 2025			AIME 2026			AMC23			MATH500		
	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑	Acc.↑	Len.↓	AE↑
DeepSeek-R1-Distill-Qwen-1.5B															
ℓ_{mar}	40.0	**7,718**	0.738	23.3	6,830	0.800	23.3	8,886	0.745	75.0	4,072	0.481	**86.4**	**2,210**	**1.152**
$+\ell_{\mathrm{inc}}$	36.7	9,233	0.265	26.7	9,170	1.165	**30.0**	9,984	1.567	80.0	5,560	0.755	85.0	2,276	1.083
$+\ell_{\mathrm{incL}}, \ell_{\mathrm{incS}}$ (Ours ($\ell^\star$))	**43.3**	10,629	**0.924**	**30.0**	**6,331**	**1.677**	23.3	9,161	0.741	**90.0**	4,147	**1.172**	86.0	2,383	1.135
+iso (Ours ($\ell^\star$+iso))	40.0	9,918	0.650	**30.0**	8,009	1.627	**30.0**	**7,903**	**1.624**	80.0	**3,663**	0.797	85.4	2,293	1.096
DeepSeek-R1-Distill-Qwen-7B															
ℓ_{mar}	60.0	10,627	-0.659	36.7	**10,413**	-1.161	56.7	9,254	-0.236	92.5	5,059	-0.083	95.4	**2,466**	0.335
$+\ell_{\mathrm{inc}}$	63.3	**9,944**	-0.397	40.0	12,442	-0.995	56.7	9,403	-0.246	92.5	4,375	0.051	94.2	2,559	0.266
$+\ell_{\mathrm{incL}}, \ell_{\mathrm{incS}}$ (Ours ($\ell^\star$))	63.3	10,106	-0.408	40.0	11,544	-0.923	60.0	10,783	0.011	**95.0**	4,266	**0.210**	94.4	2,475	0.302
+iso (Ours ($\ell^\star$+iso))	**73.3**	9,996	**0.301**	**46.7**	10,651	**-0.166**	**63.3**	**9,018**	**0.354**	92.5	**4,076**	0.084	**95.6**	2,515	**0.342**
Qwen3-1.7B															
ℓ_{mar}	46.7	15,477	−0.584	30.0	14,924	-1.435	36.7	15,169	0.106	87.5	**8,018**	-0.102	91.8	**4,393**	0.145
$+\ell_{\mathrm{inc}}$	46.7	14,412	-0.493	**40.0**	15,416	-0.299	40.0	**14,809**	0.420	**90.0**	8,436	-0.012	92.4	4,697	0.127
$+\ell_{\mathrm{incL}}, \ell_{\mathrm{incS}}$ (Ours ($\ell^\star$))	**53.3**	14,376	**0.139**	**40.0**	14,934	-0.249	36.7	15,042	0.101	**90.0**	8,207	**-0.003**	**93.0**	4,509	**0.188**
+iso (Ours ($\ell^\star$+iso))	50.0	**14,352**	-0.181	**40.0**	**14,414**	**-0.225**	**46.7**	15,153	**0.956**	87.5	8,461	-0.151	92.0	4,395	0.163
Qwen3-4B-Thinking-2507															
ℓ_{mar}	**83.3**	12,577	**0.508**	80.0	**14,345**	0.699	76.7	13,766	0.387	97.5	5,776	0.143	**98.2**	3,700	0.306
$+\ell_{\mathrm{inc}}$	80.0	12,747	0.363	80.0	15,184	0.671	80.0	**13,377**	**0.525**	**100.0**	5,588	**0.298**	97.8	**3,425**	**0.333**
$+\ell_{\mathrm{incL}}, \ell_{\mathrm{incS}}$ (Ours ($\ell^\star$))	80.0	**11,850**	0.439	73.3	14,868	0.357	**80.0**	15,509	0.487	**100.0**	**5,705**	0.282	97.6	3,650	0.293
+iso (Ours ($\ell^\star$+iso))	80.0	12,953	0.357	**83.3**	15,048	**0.819**	**80.0**	14,387	0.479	**100.0**	5,953	0.248	97.8	3,879	0.276

Algorithm 1 Discriminative threshold selection for one rollout group.

Input: correct lengths $\mathcal{L}_C$, incorrect lengths $\mathcal{L}_I = \{l_j\}_{j=1}^{n_I}$
Output: $\ell^\star$ **Constants:** $min_inc = 2$, $\epsilon = 10^{-6}$
if $\mathcal{L}_C = \emptyset$ **then**
  return $NULL$
end if
$L_{\min}, L_{\max} \leftarrow \min(\mathcal{L}_C \cup \mathcal{L}_I),\ \max(\mathcal{L}_C \cup \mathcal{L}_I)$
$(\mu_C, \sigma_C^2) \leftarrow$ mean and biased variance of $\mathcal{L}_C$
$(\mu_M, \sigma_M^2) \leftarrow$ mean and biased variance of $\mathcal{L}_C \cup \mathcal{L}_I$
if $|\sigma_M^2 - \sigma_C^2| < \epsilon$ **then**
  $\ell_{\text{mar}} \leftarrow L_{\max}$ **if** $\mu_C < \mu_M$ **else** $L_{\min}$
else if $\sigma_C^2 > \sigma_M^2$ **then**
  $\ell_{\text{mar}} \leftarrow L_{\min}$ {convex: no interior maximum}
else
  $\ell_{\text{mar}} \leftarrow \text{clamp}_{[L_{\min}, L_{\max}]}\left(\dfrac{\sigma_M^2 \mu_C - \sigma_C^2 \mu_M}{\sigma_M^2 - \sigma_C^2}\right)$
end if
if $n_I < min_inc$ **then**
  return ℓ_{mar}
end if
$(\mu_I, \sigma_I^2) \leftarrow$ mean and biased variance of $\mathcal{L}_I$
$\mathcal{C} \leftarrow \{\ell_{\text{mar}}\}$
if $\sigma_I^2 > \sigma_C^2 + \epsilon$ **then**
  $\ell_{\text{inc}} \leftarrow \text{clamp}_{[L_{\min}, L_{\max}]}\left(\dfrac{\sigma_I^2 \mu_C - \sigma_C^2 \mu_I}{\sigma_I^2 - \sigma_C^2}\right)$
  $\mathcal{C} \leftarrow \mathcal{C} \cup \{\ell_{\text{inc}}\}$
end if
$\tau \leftarrow \sqrt{\sigma_C^2 + \sigma_I^2}$
if $\tau > \epsilon$ **then**
  $x_j \leftarrow (l_j - \mu_C)/\tau, \quad u_j \leftarrow \begin{cases} 1/(1+e^{-x_j}), & x_j \geq 0 \\ e^{x_j}/(1+e^{x_j}), & x_j < 0 \end{cases} \quad \forall j \in \{1, \ldots, n_I\}$
  $w_{\text{long}} \leftarrow n_I^{-1} \sum_j u_j, \quad w_{\text{short}} \leftarrow 1 - w_{\text{long}}$
  for $k \in \{\text{long}, \text{short}\}$ **do**
    $r_{j,k} \leftarrow u_j$ **if** $k = \text{long}$ **else** $1 - u_j$
    $S_k \leftarrow \sum_j r_{j,k}, \quad m_k \leftarrow S_k^{-1} \sum_j r_{j,k} l_j, \quad v_k \leftarrow S_k^{-1} \sum_j r_{j,k} (l_j - m_k)^2$
    if $w_k > \epsilon$ and $v_k > \sigma_C^2 + \epsilon$ **then**
      $\ell_k \leftarrow \text{clamp}_{[L_{\min}, L_{\max}]}\left(\dfrac{v_k \mu_C - \sigma_C^2 m_k}{v_k - \sigma_C^2}\right)$
      $\mathcal{C} \leftarrow \mathcal{C} \cup \{\ell_k\}$
    end if
  end for
  $\mathcal{M}_I^{\text{dir}} \leftarrow \{(w_k, m_k, \max(v_k, \epsilon)) \mid k \in \{\text{long}, \text{short}\}\}$
else
  $\mathcal{M}_I^{\text{base}} \leftarrow \{(1.0, \mu_I, \max(\sigma_I^2, \epsilon))\}$
end if
$\bar{\sigma}_C^2 \leftarrow \max(\sigma_C^2, \epsilon), \quad \ell^\star \leftarrow \ell_{\text{mar}}, \quad J^{\text{best}} \leftarrow -\infty$
for $\ell \in \mathcal{C}$ **do**
  if $\ell \in \{\ell_{\text{incL}}, \ell_{\text{incS}}\}$ **then**
    $a_h \leftarrow \log w_h + \log \mathcal{N}(\ell; m_h, \bar{v}_h) \quad \forall (w_h, m_h, \bar{v}_h) \in \mathcal{M}_I^{\text{dir}}$
  else
    $a_h \leftarrow \log w_h + \log \mathcal{N}(\ell; m_h, \bar{v}_h) \quad \forall (w_h, m_h, \bar{v}_h) \in \mathcal{M}_I^{\text{base}}$
  end if
  $a_{\max} \leftarrow \max_h a_h, \quad a_{\text{sum}} \leftarrow a_{\max} + \log \sum_h \exp(a_h - a_{\max})$
  $r_h \leftarrow \exp(a_h - a_{\text{sum}}), \quad z_h \leftarrow (\ell - m_h)/\bar{v}_h, \quad \bar{z} \leftarrow \sum_h r_h z_h$
  $J \leftarrow \log \mathcal{N}(\ell; \mu_C, \bar{\sigma}_C^2) - a_{\text{sum}}$
  $J'' \leftarrow -1/\bar{\sigma}_C^2 + \sum_h r_h/\bar{v}_h - \sum_h r_h (z_h - \bar{z})^2$
  if $J'' < 0$ and $J > J^{\text{best}}$ **then**
    $(\ell^\star, J^{\text{best}}) \leftarrow (\ell, J)$
  end if
end for
return $\ell^\star$

Algorithm 2 Reward assembly with iso-pressure restoration for a training batch.

Input: batch $\mathcal{B}$ of rollout groups with $(\mathrm{acc}_i, c_i, L_i)$
Output: final rewards r_i^{final}
Constants: $g = 1$, $g_{\max} = 2$, $\epsilon_\alpha = \epsilon_r = 10^{-6}$, $\epsilon_p = 10^{-12}$
for each group $q \in \mathcal{B}$ **do**
  $G \leftarrow |q|, \quad n_C \leftarrow \sum_{i \in q} c_i$
  $(\ell_q^\star, \ell_{\mathrm{mar},q}) \leftarrow$ run Algorithm 1 on $\{L_i : c_i = 1\}, \{L_i : c_i = 0\}$
  if $n_C \in \{0, G\}$ **then**
    $\alpha_q(b) \leftarrow 0, \quad P_q(b) \leftarrow 0$
  else
    $o_i(b) \leftarrow c_i[L_i - b]_+$
    $\alpha_q(b) \leftarrow \dfrac{1 - n_C/G}{\max_{i:c_i=1} o_i(b) - G^{-1} \sum_{i \in q} o_i(b) + \epsilon_\alpha}$
    $P_q(b) \leftarrow \alpha_q(b) \sum_{i \in q} o_i(b)$
  end if
  $P_q^{\mathrm{new}} \leftarrow P_q(\ell_q^\star), \quad P_q^{\mathrm{base}} \leftarrow P_q(\ell_{\mathrm{mar},q})$
end for
$\rho_{\mathcal{B}} \leftarrow 1 + \dfrac{\sum_{q \in \mathcal{B}} [P_q^{\mathrm{base}} - P_q^{\mathrm{new}}]_+}{\sum_{q \in \mathcal{B}} P_q^{\mathrm{new}} + \epsilon_p}$
$g_{\mathrm{iso}} \leftarrow \min(g \rho_{\mathcal{B}}, g_{\max})$
for each group $q \in \mathcal{B}$ and rollout $i \in q$ **do**
  $d_i \leftarrow g_{\mathrm{iso}} \alpha_q(\ell_q^\star)[L_i - \ell_q^\star]_+$ if $c_i = 1$, else 0
  $r_i^{\mathrm{final}} \leftarrow \max(\mathrm{acc}_i - d_i, \epsilon_r)$ if $c_i = 1$, else acc_i
end for
return $\{r_i^{\mathrm{final}}\}$