BALANCING REFERENCE GUIDANCE AND FREE GENERATION IN TRAJECTORY ROLLOUTS FOR REASONING RL

Anonymous authors
Paper under double-blind review

ABSTRACT

A verified reference solution provides a correct trajectory for training a reasoning model. Alternatively, a prefix of the reference can guide the model in generating a trajectory of its own. How much reference guidance should we provide? We study this question through prefix continuation, where the model continues from a reference prefix and keeps the resulting trajectory if it passes verification, falling back to the reference otherwise. Since both procedures produce correct trajectories, we compare their distributions with the ideal distribution, the model's own distribution conditioned on successful verification. For one continuation, we derive the KL divergence in closed form, which, up to a bounded term, decreases with the product of the probability of generating a different correct trajectory and the reference surprisal, the negative log probability of the reference suffix given the prefix. Since a longer prefix tends to raise the former but lowers the latter, continuation success alone does not determine the preferred amount of guidance. From this analysis, we learn a prefix selector shared across training questions from continuation outcomes, without estimating success probabilities or additional generation. The resulting **A**daptive **R**eference **G**uidance (**ARG**) constructs correct trajectories within a fixed generation budget, and we apply it to all-failure groups in Group Relative Policy Optimization (GRPO). Experiments on Qwen3-4B and Qwen3-8B across five mathematical reasoning benchmarks show that ARG achieves the highest aggregate pass@12 among the evaluated methods with competitive average sampled accuracy.

1 INTRODUCTION

Reinforcement learning with verifiable rewards (RLVR) (Lambert et al., 2024) trains Large Language Models (LLMs) using feedback on their own generated responses (Guo et al., 2025, Team et al., 2025, Jaech et al., 2024). On difficult questions, however, the model may rarely generate a response that reaches the correct answer, leaving little feedback to learn from. When such a question comes with a reference solution written by humans or a stronger model, a straightforward approach is to train on it directly (Yan et al., 2026, Fu et al., 2026, Ma et al., 2026). However, the reference may differ substantially from the trajectories the model would produce itself, which can make it a poor learning target (Setlur et al., 2026, Qu et al., 2026). Since the model learns more easily from familiar responses (Ren et al., 2024), another line of work uses the reference as guidance so that the model generates correct trajectories by itself. Some methods transform the reference into hints (Zhang et al., 2026c, Liao et al., 2026), which requires additional LLM generation and may introduce bias or errors. Other methods, in contrast, let the model generate a *continuation* from a *reference prefix*, which is an initial segment of the reference. This approach uses the reference without modification and controls how much guidance the model receives through how much of the reference is revealed in the prefix.

However, when generating a trajectory following a prefix, deciding how much to reveal involves a trade-off. Specifically, using a longer reference prefix, which provides more guidance, makes the continuation more likely to be correct. Yet the resulting trajectory contains more content the model would not generate itself, making learning harder. Using a shorter prefix keeps the trajectory closer to the model's own generation, but increases the chance of failure. At the two extremes,

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.

revealing the entire reference reduces to training on the reference directly, while revealing nothing reduces to standard RLVR, which provides no learning signal on questions the model cannot yet solve. Balancing the two is therefore essential. The prefix should reveal just enough for the model to succeed, and leave the rest for the model to generate on its own.

To analyze this trade-off, we first ask which trajectories are ideal to learn from. Ideally, these trajectories should be correct and generated entirely by the model. We therefore define the distribution of the model's own correct samples as the *ideal distribution*. We show that the Kullback-Leibler (KL) divergence from the distribution of generated trajectories used for training to this ideal distribution decreases as $d_t S_t$ increases, up to a small bounded term. Here d_t is the probability that the model follows the reference's prefix to generate a successful trajectory, and S_t is the negative log probability of the remaining reference given the prefix. Intuitively, d_t reflects how much the prefix helps the model succeed, while S_t indicates how unfamiliar the reference suffix is. Moreover, $d_t S_t$ captures both sides of the trade-off. A longer prefix tends to increase d_t, since more guidance makes success more likely, but it decreases S_t, since less of the reference is left.

As a result, a good prefix should maximize $d_t S_t$. However, d_t is unknown, and estimating it for every prefix of every question is impractical. We avoid this estimation by learning a *prefix selector*, a distribution over a few guidance levels shared across questions. Since the selector only needs to work well on average, the outcome of each continuation already provides the feedback it needs, without any additional generation.

These analyses lead to *Adaptive Reference Guidance* (**ARG**). Given a question and its reference, ARG samples guidance levels from the prefix selector, generates continuations from the resulting prefixes within a fixed budget, and falls back to the reference when no continuation succeeds. The same continuations also update the prefix selector, which therefore requires no additional generation. To examine how the accepted trajectories affect policy learning, we apply ARG to Group Relative Policy Optimization (GRPO) (Shao et al., 2024) for groups in which every sampled response fails, resulting in no learning signals.

Empirically, on Qwen3-4B and Qwen3-8B (Yang et al., 2025) across five mathematical reasoning benchmarks, ARG achieves the highest aggregate $\mathrm{pass@12}$ of 67.33% and 70.00%, respectively. Moreover, its average sampled accuracy remains competitive, with a small gap to the strongest baseline. Our contributions are threefold.

- We analyze reference-prefix continuation through the KL divergence to the model's own correct trajectories, revealing a trade-off between reference guidance and surprisal.
- We derive an unbiased update of the prefix selector from the same continuations, adapting the guidance level across questions and during training without additional generation.
- We develop ARG from this analysis and apply it to all-failure groups in GRPO, achieving the highest aggregate $\mathrm{pass@12}$ across five math benchmarks on two model scales.

2 PRELIMINARIES

Reasoning with verifiable rewards. Let $\pi_\theta(y \mid x)$ be a language model that generates a trajectory y for a question x drawn from the training distribution $\mathcal{D}$. The trajectory includes the reasoning, the final answer, and the termination token. A verifier assigns a binary reward $R_{\mathrm{acc}}(x, y) \in \{0, 1\}$, and we call y *correct* when $R_{\mathrm{acc}}(x, y) = 1$. Let C_x be the set of correct trajectories and $p_\theta(x) = \pi_\theta(C_x \mid x)$ be the success probability. During training, θ_{old} denotes the parameters of the current model, which generates the samples at each training step. We use natural logarithms throughout.

Reference-prefix continuation with fallback. Suppose that the question comes with a correct reference $r \in C_x$. For a prefix length $0 \le t < |r|$, we write $r = h_t \oplus a_t$, where the reference prefix h_t contains the first t tokens of r, the reference suffix a_t contains the remaining tokens, and $\oplus$ denotes token concatenation. The model samples a continuation $Z \sim \pi_\theta(\cdot \mid x, h_t)$, which forms the *candidate trajectory* $\widetilde{Y}_t = h_t \oplus Z$. At $t = 0$, the prefix is empty and the model generates under the original question alone. We keep the candidate trajectory if it is correct and fall back to the reference

otherwise, which gives the accepted trajectory

$$Y = \begin{cases} \widetilde{Y}_t, & \widetilde{Y}_t \in C_x, \\ r, & \widetilde{Y}_t \notin C_x, \end{cases} \qquad Y \sim Q_t. \tag{1}$$

Thus, Q_t assigns all probability to correct trajectories for every prefix length. In contrast, training on the reference corresponds to the distribution δ_r, which assigns all probability to r.

Ideal distribution. As discussed in Section 1, we take the model's own correct trajectories as the ideal distribution. Formally, for $p_\theta(x) > 0$, sampling from $\pi_\theta(\cdot \mid x)$ and keeping only correct trajectories gives

$$\nu_\theta(y \mid x) = \frac{\pi_\theta(y \mid x)\mathbf{1}\{y \in C_x\}}{p_\theta(x)}. \tag{2}$$

This distribution also has a direct connection to likelihood learning (Phan et al., 2023, Singh et al., 2023). Specifically, for a finite trajectory space and a differentiable policy with positive probabilities,

$$\mathbb{E}_{Y \sim \nu_\theta(\cdot|x)} \left[\nabla_\theta \log \pi_\theta(Y \mid x)\right] = \nabla_\theta \log p_\theta(x). \tag{3}$$

In other words, learning the likelihood of trajectories drawn from ν_θ increases the success probability in expectation at the current model. Finally, we measure how far a distribution Q of correct trajectories is from the ideal distribution by the Kullback-Leibler (KL) divergence

$$D_{\mathrm{KL}}(Q \| \nu_\theta) = \mathbb{E}_{Y \sim Q}\left[\log \frac{Q(Y)}{\nu_\theta(Y \mid x)}\right].$$

Appendix A.2 derives Equation (3) and explains this direction of KL.

3 Analyzing and Selecting Reference Prefixes

In this section, we analyze how the prefix length affects the accepted trajectory and turn the analysis into a practical way of choosing prefixes. Section 3.1 derives the distance from the accepted trajectory to the ideal distribution, which identifies the preferred prefix of a question. Section 3.2 explains why this prefix is difficult to find during training, and Section 3.3 addresses both challenges by learning a prefix selector that performs well on average across questions.

3.1 Distance to the ideal distribution

Fix a question x with reference r and the current model $\pi_{\theta_{\text{old}}}$, and suppress the conditioning on x and the parameters by writing π, C, $p = \pi(C)$, and ν. We assume a finite trajectory space with $0 < \pi(r) < 1$. For a prefix length t, we define the *successful replacement probability* d_t and the *reference surprisal* S_t as

$$d_t = \Pr\big(\widetilde{Y}_t \in C,\ \widetilde{Y}_t \neq r\big), \qquad S_t = -\log \pi(a_t \mid h_t). \tag{4}$$

Here d_t excludes an exact copy of the reference, since the accepted trajectory is then still r. In other words, d_t is the probability that the model replaces the reference suffix with a different correct continuation, and S_t measures how unfamiliar the reference suffix is to the model after the prefix.

Theorem 1 (Distance to the ideal distribution)**.** *Under the setup above, for every prefix length t,*

$$D_{\mathrm{KL}}(Q_t \| \nu) = \underbrace{\log \frac{p}{\pi(r)}}_{D_{\mathrm{KL}}(\delta_r \| \nu)} - d_t S_t + (1 - d_t)\log(1 - d_t). \tag{5}$$

Moreover, every $t^* \in \arg\max_t d_t S_t$ satisfies $D_{\mathrm{KL}}(Q_{t^*} \| \nu) \leq \min_t D_{\mathrm{KL}}(Q_t \| \nu) + 1/e$.

Theorem 1 expresses the distance between Q_t and the ideal distribution ν in closed form with three terms. The first term equals that of training on the reference, which does not depend on the prefix. The third term is bounded within $[-1/e, 0]$. Therefore, the prefix affects the distance mainly through the second term, $d_t S_t$. Intuitively, this product formalizes the trade-off in choosing the prefix length. A longer prefix never increases S_t, since it leaves less of the reference for the continuation to replace.

In contrast, a longer prefix can increase d_t by making the continuation easier to succeed. Consequently, both a nearly complete prefix and an empty prefix on a hard question make $d_t S_t$ close to zero and leave the distance close to $D_{\mathrm{KL}}(\delta_r \| \nu)$, indicating learning from the reference alone. Moreover, a prefix with a higher success probability of the continuation need not give a smaller distance, as illustrated in Appendix A.5. Instead, maximizing $d_t S_t$ attains the smallest distance up to $1/e$ by the second statement in the theorem, and we call h_{t^*} the *preferred prefix* of the question under the current model. Appendix A gives the proof.

3.2 Challenges in finding the preferred prefix

Although Theorem 1 identifies the preferred prefix, finding it during training is difficult. The reference surprisal S_t is available for every prefix length from a single forward pass of the model on the reference. However, the successful replacement probability d_t is unknown, which leads to the following two challenges.

Challenge 1: Variation across questions and training steps. Theorem 1 concerns a fixed question and a fixed model, and d_t depends on both. Therefore, the estimates of d_t for one question do not apply to another question. Furthermore, since d_t changes as the model is updated, the estimates obtained at one training step may become outdated at the next. As a result, the search for t^* would have to be repeated for every question and every model update.

Challenge 2: Limited sampling budget. Moreover, each such search is expensive. Identifying t^* compares $d_t S_t$ across prefix lengths and therefore requires an accurate estimate of d_t at every prefix length. Since d_t is often small on hard questions, such an estimate requires many continuations even for a single prefix length, and this cost multiplies across prefix lengths.

3.3 Learning a prefix selector across questions

Both challenges arise from seeking the preferred prefix of each question at each training step. Instead, we learn a way of choosing prefixes that performs well on average over the questions that receive guidance, which addresses both challenges.

Sharing across questions and training steps. Averaging over questions requires describing prefixes in the same way for every question, which the prefix length does not achieve for references of different lengths. Since Theorem 1 measures the replaced suffix by its surprisal rather than its length, we describe a prefix through the reference surprisal. Specifically, for a question x with reference r_x, let $S_{0,x} = -\log \pi_{\theta_{\mathrm{old}}}(r_x \mid x)$ be the surprisal of the entire reference. A *guidance level* α in a set $\mathcal{A} \subset (0, 1]$ selects a prefix length $t_x(\alpha)$ whose reference surprisal is approximately $\alpha S_{0,x}$. Thus, α is the fraction of the reference surprisal left to the continuation, where $\alpha = 1$ gives the empty prefix. As with prefix lengths, the levels cannot all be tried within the sampling budget. Therefore, $\mathcal{A}$ contains only a few levels, and Section 4.1 specifies the mapping from levels to prefix lengths.

We then learn a *prefix selector* ρ_ϕ, which is a distribution over $\mathcal{A}$ with parameters ϕ shared across questions. Let $\mathcal{D}_{\mathrm{guide}}$ denote the distribution of questions that receive guidance at the current training step, which Section 4.3 specifies for our application. For $X \sim \mathcal{D}_{\mathrm{guide}}$ and $\alpha \sim \rho_\phi$, we write $d_{X,\alpha}$, $S_{X,\alpha}$, and $Q_{X,\alpha}$ for d_t, S_t, and Q_t at the prefix length $t_X(\alpha)$, and ν_X for the ideal distribution of X. The selector maximizes

$$J(\phi) = \mathbb{E}_{X \sim \mathcal{D}_{\mathrm{guide}},\, \alpha \sim \rho_\phi} \left[\frac{d_{X,\alpha} S_{X,\alpha}}{S_{0,X}} \right], \tag{6}$$

which averages $d_t S_t$ over the questions and the sampled levels in units of the surprisal of the entire reference. This normalization keeps questions with long or unfamiliar references from dominating the average, and it does not change which prefix maximizes $d_t S_t$ within any question. Since the selector is shared across questions and updated at every training step, continuations on all questions and at all steps inform the same choice, while the selected prefix still adapts to each question through its reference surprisal. This addresses Challenge 1, and Appendix B.3 bounds the sharing cost.

Learning without estimating d_t. The average objective also avoids the accurate estimates of d_t required in Challenge 2. For a continuation from the prefix h_t, we define the *selector feedback*

$$W_t = \mathbf{1}\{\widetilde{Y}_t \in C,\ \widetilde{Y}_t \neq r\} \frac{S_t}{S_0}, \tag{7}$$

and write $W_{X,\alpha}$ for the feedback at the prefix length $t_X(\alpha)$. In other words, a successful replacement receives the fraction of the reference surprisal left to the continuation, while a failure or an exact copy of the reference receives zero. Since the indicator has expectation d_t, the feedback satisfies

$$\mathbb{E}[W_t] = \frac{d_t S_t}{S_0}. \tag{8}$$

Thus, the feedback of a single continuation is an unbiased estimate of its term in J. A single estimate is noisy, and noisy estimates cannot reliably rank the prefix lengths of one question. However, since J is an average that is linear in $d_{X,\alpha}$, the errors of single continuations cancel when pooled over many questions. Moreover, since $\mathcal{A}$ contains only a few levels and the selector keeps a positive probability on every level, every level receives feedback from many questions, which addresses Challenge 2. The following theorem shows that J still measures the distance in Theorem 1 and that the selector can be learned from this feedback.

Theorem 2 (Learning the prefix selector from feedback)**.** *Fix the current model, the references, and $\mathcal{D}_{\text{guide}}$, where every question satisfies the setup of Section 3.1. Let $\mathcal{L}(\phi)$ and $\mathcal{L}_{\text{ref}}$ be the expectations of $D_{\text{KL}}(Q_{X,\alpha}\|\nu_X)/S_{0,X}$ and $D_{\text{KL}}(\delta_{r_X}\|\nu_X)/S_{0,X}$ over $X \sim \mathcal{D}_{\text{guide}}$ and $\alpha \sim \rho_\phi$. Then*

$$J(\phi) \le \mathcal{L}_{\text{ref}} - \mathcal{L}(\phi) \le J(\phi) + \mathbb{E}_X\left[\frac{1}{eS_{0,X}}\right]. \tag{9}$$

Moreover, for a differentiable prefix selector with positive probabilities and a baseline b fixed before sampling,

$$\mathbb{E}\left[(W_{X,\alpha} - b)\nabla_\phi \log \rho_\phi(\alpha)\right] = \nabla_\phi J(\phi). \tag{10}$$

The first statement shows that, compared with training on the reference, the selector reduces the average normalized distance by at least J, where each distance is divided by the surprisal of the entire reference. The gap is at most $\mathbb{E}_X[1/(eS_{0,X})]$, which is small when no reference is highly likely under the model. Therefore, maximizing J pursues the goal of Theorem 1 across questions. The second statement shows that raising the probability of the levels whose feedback exceeds the baseline b moves the selector toward a larger J on average, where b introduces no bias and serves to reduce the variance. Since these continuations also produce the accepted trajectories, the update requires no additional generation. Appendix B proves Theorem 2 and extends it to batches, changing models, and randomly sampled levels.

4 ARG: Adaptive Reference Guidance

We now build ARG on Theorems 1 and 2. Section 4.1 describes how ARG generates accepted trajectories for a question, and Section 4.2 describes how the same continuations update the prefix selector. Neither step depends on a particular reinforcement learning algorithm. The algorithm only determines the questions that receive guidance and how the model learns from the accepted trajectories, and Section 4.3 specifies both for GRPO.

4.1 Generating accepted trajectories

Consider a question x that receives guidance, with reference r. ARG first scores r once under the current model $\pi_{\theta_{\text{old}}}$, which gives the reference surprisal S_t for every prefix length. Each guidance level $\alpha \in \mathcal{A}$ then maps to a prefix length $t_x(\alpha)$. Specifically, $\alpha = 1$ gives the empty prefix. For $\alpha < 1$, we take the latest prefix length whose reference surprisal is at least αS_0 and move it to a nearby sentence or formula boundary, so that the prefix ends at a natural break. Appendix E.2 gives the boundary rules.

The prefix selector assigns a logit ϕ_α to each level and samples from

$$\rho_\phi(\alpha) = (1 - \varepsilon_{\text{sel}})\frac{\exp(\phi_\alpha)}{\sum_{\alpha' \in \mathcal{A}} \exp(\phi_{\alpha'})} + \frac{\varepsilon_{\text{sel}}}{|\mathcal{A}|}, \tag{11}$$

where $0 < \varepsilon_{\text{sel}} < 1$ keeps every level observed as the model changes. Within a fixed budget of K continuations per question, we independently sample $\alpha_j \sim \rho_\phi$ for $j = 1, \ldots, K$. For the resulting prefix length $t_j = t_x(\alpha_j)$, we sample a continuation $Z_j \sim \pi_{\theta_{\text{old}}}(\cdot \mid x, h_{t_j})$ and verify the candidate trajectory $\widetilde{Y}_j = h_{t_j} \oplus Z_j$.

Every correct candidate trajectory different from r can serve as an accepted trajectory. If there is none, ARG falls back to the reference. Therefore, the question always receives a correct trajectory. When the number of accepted trajectories is limited, as in our application, we prefer candidate trajectories that keep less of the reference. Specifically, a correct candidate trajectory from the empty prefix keeps none of the reference. Indeed, it is generated entirely by the model and follows the ideal distribution restricted to trajectories other than r. Therefore, we accept these candidate trajectories first. Among candidate trajectories from nonempty prefixes, we prefer the shortest prefix, since the surprisal of the prefix, $-\log \pi_{\theta_{\text{old}}}(h_t \mid x) = S_0 - S_t$, never decreases with t. Finally, Theorem 1 concerns a single continuation. For accepted trajectories from several continuations, Appendix C.2 gives a sufficient condition under which the distance remains no larger than that of training on the reference.

4.2 Updating the prefix selector

Every continuation provides the selector feedback $W_j = W_{t_j}$ in Equation (7), whether or not its candidate trajectory is accepted. Since the feedback does not depend on which trajectories are accepted, the second statement of Theorem 2 holds under any acceptance rule, while the first statement, like Theorem 1, concerns a single continuation.

We collect continuations for all questions in a batch while keeping the current model and the prefix selector fixed. Let $\mathcal{U}$ index these continuations, with sampled levels α_j and selector feedback W_j. For a nonempty $\mathcal{U}$, we update the prefix selector by

$$\phi \leftarrow \phi + \eta_{\text{sel}} \sum_{j \in \mathcal{U}} (W_j - b) \nabla_\phi \log \rho_\phi(\alpha_j), \tag{12}$$

where $\eta_{\text{sel}} > 0$ is the step size and b is a running average of past feedback fixed before collection. By the second statement of Theorem 2, the update follows the gradient of J in expectation. Appendix B gives the baseline update and derives the expected update as the model and the questions change during training.

4.3 Applying ARG to GRPO

GRPO (Shao et al., 2024) samples a group of $G = 8$ responses for each question and compares their rewards within the group. When every response in a group is incorrect, the group provides no signal for improving correctness, since all correctness rewards are identical. In such an *all-failure group*, adding a correct trajectory restores the learning signal. Therefore, we apply ARG to all-failure groups, using $K = 8$ continuations. These questions define $\mathcal{D}_{\text{guide}}$.

We insert at most four accepted trajectories into each group, which keeps at least four original failures and permits a balanced group, motivated by prior work on intermediate pass rates (Bae et al., 2026, Liu et al., 2026b). Following the preference in Section 4.1, we first accept correct candidate trajectories from the empty prefix. If fewer than four are accepted, we additionally accept at most one from a nonempty prefix, which limits repeated reference content in the same group. If no candidate trajectory is accepted, we insert the reference once as fallback. Each inserted trajectory replaces an original failure at a uniformly chosen position. We then recompute the group advantages A_i from the final correctness and format rewards, as specified in Appendix E.3.

The model learns each inserted trajectory under the original question through its complete likelihood, including the prefix. This matches Theorem 1, which measures the complete accepted trajectory under the original question. Let $\mathcal{I}$ index the inserted trajectories (x_i, y_i) and let T be the total number of tokens in the loss. Their contribution to the gradient is

$$\nabla_\theta \mathcal{L}_{\text{inserted}} = -\frac{1}{T} \sum_{i \in \mathcal{I}} A_i \nabla_\theta \log \pi_\theta(y_i \mid x_i), \tag{13}$$

where the trajectories, advantages, and token count are fixed during differentiation. The remaining original responses use the clipped GRPO loss with the same denominator. Moreover, for binary rewards, Appendix D shows that inserting $Y \sim Q$ into an all-failure group gives an expected update proportional to the gradient of $\log \frac{p_\theta}{1-p_\theta} - D_{\text{KL}}(Q \| \nu_\theta)$ at the current model. The gradient of the second term vanishes when Q is the ideal distribution, in which case the update only increases the odds of success. This connects the distance in Theorem 1 to the GRPO update.

After collecting continuations, we update the prefix selector by Equation (12) and take one policy update on the final groups. Algorithm 1 summarizes the training procedure.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETTING

Data and models. We use DeepMath-103k with offline reference solutions (He et al., 2026) as our training set. We use Qwen3-4B and Qwen3-8B with thinking mode disabled (Yang et al., 2025) as our base LLMs. For all methods, we evaluate on five mathematical reasoning benchmarks, AIME 2024, AIME 2025, AIME 2026, HMMT February 2025, and HMMT November 2025 (Balunovic et al., 2026, Dekoninck et al., 2026). Appendix E gives further details.

Baselines and evaluation. We compare against the initial Qwen model, vanilla supervised fine-tuning (SFT), GRPO (Shao et al., 2024), OPSD (Zhao et al., 2026), SAGE (Liao et al., 2026), BREAD (Zhang et al., 2026d), and LUFFY (Yan et al., 2026). During evaluation, we draw 12 responses per question. $\mathrm{pass@12}$ measures whether a question has at least one correct response, while $\mathrm{avg@12}$ measures the fraction of correct responses. We use $\mathrm{pass@12}$ as the primary metric.

Model	Method	AIME24	AIME25	AIME26	HMMT_Feb	HMMT_Nov	Avg.
Qwen3-4B	Base	50.00	43.33	43.33	23.33	33.33	38.67
	SFT	63.33	63.33	66.67	36.67	36.67	53.33
	GRPO	<u>76.67</u>	<u>70.00</u>	**73.33**	**53.33**	46.67	64.00
	OPSD	56.67	36.67	40.00	40.00	40.00	42.67
	SAGE	73.33	<u>70.00</u>	<u>70.00</u>	43.33	<u>56.67</u>	62.67
	BREAD	**80.00**	**73.33**	**73.33**	<u>50.00</u>	<u>56.67</u>	<u>66.67</u>
	LUFFY	**80.00**	**73.33**	<u>70.00</u>	43.33	<u>56.67</u>	64.67
	ARG (ours)	**80.00**	**73.33**	**73.33**	43.33	**66.67**	**67.33**
Qwen3-8B	Base	46.67	43.33	53.33	16.67	46.67	41.33
	SFT	<u>80.00</u>	70.00	**73.33**	46.67	56.67	65.33
	GRPO	<u>80.00</u>	63.33	<u>70.00</u>	46.67	53.33	62.67
	OPSD	66.67	60.00	50.00	36.67	43.33	51.33
	SAGE	<u>80.00</u>	70.00	<u>70.00</u>	50.00	60.00	66.00
	BREAD	<u>80.00</u>	66.67	66.67	<u>53.33</u>	**70.00**	<u>67.33</u>
	LUFFY	76.67	**76.67**	**73.33**	46.67	<u>63.33</u>	<u>67.33</u>
	ARG (ours)	**83.33**	<u>73.33</u>	**73.33**	**56.67**	<u>63.33</u>	**70.00**

Table 1. Main experimental results in $\mathrm{pass@12}$ percentages. Avg. is the mean over the five benchmarks. Within each model, **bold** and <u>underline</u> mark the best and second-best values in each column.

5.2 RESULTS

Table 1 evaluates ARG through its application to GRPO. ARG achieves the highest aggregate $\mathrm{pass@12}$ at both model scales, reaching 67.33% on Qwen3-4B and 70.00% on Qwen3-8B. Compared with GRPO, these results improve $\mathrm{pass@12}$ by 3.33 and 7.33 percentage points, respectively. ARG also exceeds the reference-informed baselines in aggregate $\mathrm{pass@12}$, which use reference solutions as off-policy responses in LUFFY, as prefixes in BREAD, and as hints in SAGE (Yan et al., 2026, Zhang et al., 2026d, Liao et al., 2026). Thus, ARG solves the largest fraction of evaluation questions under the 12-response budget. For $\mathrm{avg@12}$, ARG improves over GRPO on both models and remains competitive with the strongest baseline, as discussed in Appendix F. Sections 5.3 and 5.4 examine the contributions of the continuations and the prefix selector.

5.3 ABLATIONS

We examine the contribution of the continuations, compare with unguided resampling, and evaluate the surprisal weighting in the selector feedback.

Reference insertion. For each all-failure group, we replace a random failure in the original group with the reference without any extra generation. This ablation corresponds to training on the reference in Theorem 1. Table 2(a) shows that reference insertion achieves 56.67% in $\mathrm{pass@12}$ and

Method	pass@12	avg@12
Reference insertion	56.67	28.44
Resampling	60.00	34.00
ARG	67.33	41.11

(a) Sources of accepted trajectories on Qwen3-4B

Method	Qwen3-4B		Qwen3-8B	
	pass@12	avg@12	pass@12	avg@12
w/o weighting	63.33	34.67	64.00	31.89
ARG	67.33	41.11	70.00	42.28

(b) Surprisal weighting on both models

Table 2. Ablation results averaged over the five benchmarks. Resampling uses eight continuations from the empty prefix without reference fallback. The weighting ablation removes S_t/S_0 from the selector feedback.

28.44% in $\mathrm{avg@12}$ on Qwen3-4B, compared with 67.33% and 41.11% for ARG. These results support using the reference to guide continuations and reserving the reference itself for fallback.

Resampling. We allocate the same eight continuations to the empty prefix for each all-failure group. Correct candidate trajectories replace original failures, and groups with no correct candidate trajectory remain unchanged. Thus, this comparison evaluates ARG against additional sampling without reference guidance or fallback. Table 2(a) shows that ARG improves $\mathrm{pass@12}$ from 60.00% to 67.33% and $\mathrm{avg@12}$ from 34.00% to 41.11% on Qwen3-4B. During their respective training runs, candidate trajectories are accepted in 66.0% of all-failure groups with ARG and only 32.5% with resampling. More details are shown in Appendix G.2.

Without surprisal weighting. We replace the selector feedback in Equation (7) with the successful replacement indicator. The guidance levels, surprisal-based prefix construction, selector update rule, and acceptance rule remain unchanged. Table 2(b) shows that removing the weighting lowers both evaluation metrics on both models, with $\mathrm{pass@12}$ decreasing by 4.00 and 6.00 percentage points on Qwen3-4B and Qwen3-8B, respectively. These results support the surprisal weighting in the selector feedback. Section 5.4 examines the selector feedback and prefix selection under ARG, while Appendix G.3 compares the prefix selection with that of the ablation.

5.4 Training dynamics

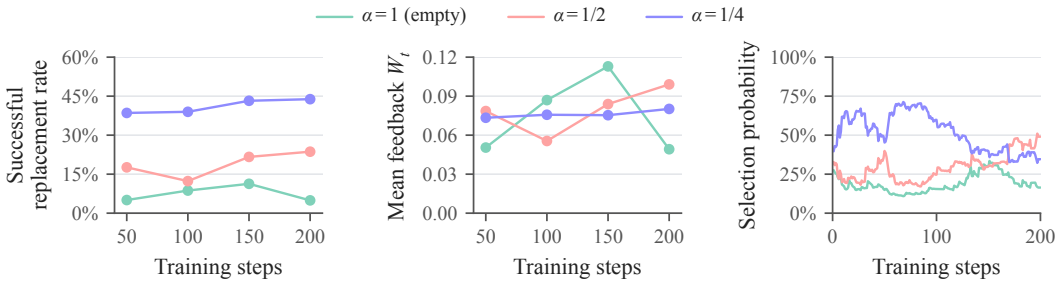
Figure 1 examines Qwen3-8B training dynamics through successful replacement rates, selector feedback, and prefix selection.

Successful replacement and selector feedback. Figure 1(a) reports the successful replacement rate for each guidance level. Each point averages all continuations at that level within a window of 50 training steps. The level $\alpha = 1/4$ has a higher successful replacement rate than $\alpha = 1/2$ in every window. However, Figure 1(b) shows that their ordering can change when the successful replacement indicator is weighted by the fraction S_t/S_0 of the reference surprisal left to the continuation. In the final window, for example, $\alpha = 1/4$ has lower mean feedback than $\alpha = 1/2$ (0.0802 versus 0.0991), although its successful replacement rate is higher (43.84% versus 23.65%). Thus, the feedback favors less reference guidance in this window, consistent with Section 3.1, where a higher success probability of the continuation need not give a smaller distance.

Prefix selection during training. The selector uses the feedback W_t from each continuation to update its selection probabilities. Figure 1(c) shows how these probabilities change during training. The selector favors $\alpha = 1/4$ through much of training and increases its probability of selecting $\alpha = 1/2$ later, assigning it the highest final probability of 49.0%. Since a larger α leaves more of the reference surprisal to the continuation, this shift supplies less reference guidance. Appendix H provides additional training-dynamics results.

6 Related Work

Learning signals beyond on-policy rollouts. LUFFY, SRFT, ReLIFT, and related methods combine reference or teacher trajectories with the model's own rollouts, through mixed-policy groups or combined supervised and reinforcement updates (Yan et al., 2026, Fu et al., 2026, Ma et al., 2026, Zhang et al., 2026b, Liu et al., 2025a, Dong et al., 2026, Lv et al., 2025). OPSD and SDPO instead use the model conditioned on solution information or feedback as a teacher for token-level supervi-



(a) Successful replacement (b) Selector feedback (c) Prefix selection

Figure 1. Training dynamics of a Qwen3-8B ARG run. Panel (a) reports successful replacement rates, panel (b) reports mean selector feedback W_t, and panel (c) shows the selection probabilities. In panels (a) and (b), a point at step s summarizes all continuations at that level across the past 50 steps, including failures and candidate trajectories not accepted for training. Panel (c) shows the state after each training step without smoothing.

sion (Zhao et al., 2026, Hübotter et al., 2026). Other methods recover learning signals from failed rollouts, either by repairing them into correct ones (Li et al., 2026b, Yang et al., 2026, Zhang et al., 2025, Ren et al., 2026) or by reshaping the rewards or advantages of groups with identical rewards (Chen et al., 2025, Nie et al., 2026, Nan et al., 2025, Le et al., 2026).

Reference-guided generation. Closer to our work, many methods use the reference only to guide the model's own generation, either by revealing a prefix of the reference for the model to continue (Liu et al., 2026a, Setlur et al., 2026, Qu et al., 2026, Zhang et al., 2026a, Su et al., 2026, Liu et al., 2025b) or by providing hints derived from the reference or its answer (Liao et al., 2026, Zhang et al., 2026c, Nath et al., 2025, Chen et al., 2026, Zhou et al., 2026a). The amount of guidance is fixed or scheduled (Xi et al., 2024, Huang et al., 2025, Li et al., 2026a), adapted to the observed success of the model (Zhang et al., 2026d, amani et al., 2026, Li et al., 2025b, Guo et al., 2026, Beliaev, 2026), or chosen by the gain of a prefix or the transfer of a hint (Zhou et al., 2026c, Xia et al., 2026). In contrast, we derive the amount of guidance from the distance between the accepted trajectory and the ideal distribution, which shows that success alone does not determine the preferred prefix, and we learn a prefix selector shared across questions from the continuations themselves.

Posterior views of learning from correct trajectories. Training on the model's own correct samples, as in STaR and ReST^{EM}, approximates learning from our ideal distribution (Zelikman et al., 2022, Singh et al., 2023), which is also the correct distribution closest to the model in KL and relates to the Bayesian view of KL-regularized RL (Dymetman, 2026, Korbak et al., 2022). Latent-variable methods approximate the posterior over rationales given the correct answer (Phan et al., 2023, Zhou et al., 2026b), and TRICE keeps a stored rationale until a new proposal is correct, similar to our fallback. Other work analyzes or modifies the function of the success probability that RL maximizes (Davis & Recht, 2025, Tajwar et al., 2026, Ramasubramanian et al., 2026), or shows that targets unlikely under the model are harder to learn from (Ren et al., 2024, Li et al., 2025a, Setlur et al., 2024).

7 Conclusion and Limitations

We study reference-prefix continuation with fallback through the distance from the accepted trajectory to the ideal distribution. Up to a small bounded term, this distance decreases with $d_t S_t$, which captures the trade-off between successful replacement and the reference surprisal. This analysis leads to ARG, which learns a prefix selector across questions from the continuations themselves. Applied to all-failure groups in GRPO, ARG achieves the highest aggregate pass@12 on Qwen3-4B and Qwen3-8B across five mathematical benchmarks, with competitive average sampled accuracy.

Limitations. ARG is only applied to GRPO, although the method does not depend on a particular reinforcement learning algorithm. Moreover, the selector chooses among a predefined set of guidance levels, and our experiments cover mathematical reasoning with one model family. Future work may extend the method to other domains with verifiable rewards.

AI USE STATEMENT

In this work, we used generative AI tools only to assist with writing. Specifically, we used them to revise parts of the text from the authors' outlines and polish the manuscript for readability. We have not used generative AI tools for any task with required disclosure, including generating data, developing the theoretical framework, formulating or proving the mathematical claims, proposing hypotheses, designing the methodology or experiments, implementing the method, translating, or interpreting the results. We have reviewed all AI-assisted work. In particular, the authors reviewed every AI-edited passage and checked that it accurately reflects the technical content, the cited work, and the experimental results. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human subjects, personal data, or newly collected data. All experiments use publicly available models, training data, and benchmarks for mathematical reasoning.

REPRODUCIBILITY STATEMENT

For the theoretical results, Section 3 states all assumptions, and Appendices A and B give complete proofs of Theorems 1 and 2. Appendices C and D prove the supporting results on multiple continuations and on learning from inserted trajectories. For the method, Section 4 describes the steps of ARG, and Appendix E gives the complete training procedure in Algorithm 1. For the experiments, Section 5.1 reports the data, models, benchmarks, and baselines. Appendix E.4 lists the optimizer and rollout settings shared by ARG and the GRPO-based baselines, the settings used for SFT and OPSD, and the question prompt. Appendix E.5 lists the reported checkpoints and the evaluation protocol. Appendices F and G report the benchmark-level results.

REFERENCES

mohammad hossein amani, Aryo Lotfi, Nicolas Baldwin, Samy Bengio, Mehrdad Farajtabar, Emmanuel Abbe, and Robert West. Rl for reasoning by adaptively revealing rationales. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 118599–118620, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/c073a9d002f3dab99a7af18642d74bdb-Paper-Conference.pdf.

Sanghwan Bae, Jiwoo Hong, Min Young Lee, Hanbyul Kim, JeongYeon Nam, and Donghyun Kwak. Online difficulty filtering for reasoning oriented reinforcement learning. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 700–719, 2026.

Mislav Balunovic, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. Matharena: Evaluating llms on uncontaminated math competitions. *Advances in Neural Information Processing Systems*, 38, 2026.

Vladislav Beliaev. Max out grpo signal: Adaptive trace prefix control for hard reasoning problems. *arXiv preprint arXiv:2607.07674*, 2026.

Justin Chen, Xiangyu Peng, Prafulla Kumar Choubey, Kung-Hsiang Huang, Jiaxin Zhang, Mohit Bansal, and Chien-Sheng Wu. Nudging the boundaries of llm reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 30315–30335, 2026.

Peter Chen, Xiaopeng Li, Ziniu Li, Xi Chen, and Tianyi Lin. Stepwise guided policy optimization: Coloring your incorrect reasoning in grpo. *arXiv preprint arXiv:2505.11595*, 2025.

Damek Davis and Benjamin Recht. What is the objective of reasoning with reinforcement learning? *arXiv preprint arXiv:2510.13651*, 2025.

Jasper Dekoninck, Nikola Jovanović, Tim Gehrunger, Kári Rögnvaldsson, Ivo Petrov, Chenhao Sun, and Martin Vechev. Beyond benchmarks: Matharena as an evaluation platform for mathematics with llms. *arXiv preprint arXiv:2605.00674*, 2026.

Yihong Dong, Xue Jiang, Yongding Tao, Huanyu Liu, Kechi Zhang, Lili Mou, Rongyu Cao, Yingwei Ma, Jue Chen, Binhua Li, et al. Rl-plus: Countering capability boundary collapse of llms in reinforcement learning with hybrid-policy optimization. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 43029–43053, 2026.

Marc Dymetman. Binary rewards and reinforcement learning: Fundamental challenges. *arXiv preprint arXiv:2605.02375*, 2026.

Yuqian Fu, Tinghong Chen, Jiajun Chai, Xihuai Wang, Songjun Tu, Guojun Yin, Wei Lin, Qichao Zhang, Yuanheng Zhu, and Dongbin Zhao. Srft: A single-stage method with supervised and reinforcement fine-tuning for reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 102430–102459, 2026.

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

Yongxin Guo, Wenbo Deng, Zhenglin Cheng, and Xiaoying Tang. G2rpo-a: Guided group relative policy optimization with adaptive guidance. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4525–4539, 2026.

Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, et al. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 138306–138322, 2026.

Zeyu Huang, Tianhao Cheng, Zihan Qiu, Zili Wang, Yinghui Xu, Edoardo M Ponti, and Ivan Titov. Blending supervised and reinforcement fine-tuning with prefix sampling. *arXiv preprint arXiv:2507.01679*, 2025.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, et al. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026.

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

Tomasz Korbak, Ethan Perez, and Christopher Buckley. Rl with kl penalties is better viewed as bayesian inference. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 1083–1091, 2022.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, et al. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

Thanh-Long V Le, Myeongho Jeon, Kim Vu, Viet Lai, and Eunho Yang. No prompt left behind: Exploiting zero-variance prompts in llm reinforcement learning via entropy-guided advantage shaping. In *International Conference on Learning Representations*, volume 2026, pp. 121956–121982, 2026.

Jiazheng Li, Hongzhou Lin, Hong Lu, Kaiyue Wen, Zaiwen Yang, Jiaxuan Gao, Yi Wu, and Jingzhao Zhang. Questa: Expanding reasoning capacity in llms via question augmentation. In *International Conference on Learning Representations*, volume 2026, pp. 144191–144215, 2026a.

Shuozhe Li, Vaishnav Tadiparthi, Kwonjoon Lee, Nakul Agarwal, Hossein Nourkhiz Mahjoub, Ehsan Moradi Pari, Lizhang Chen, Amy Zhang, and Liu Leqi. Learning robust reasoning through guided adversarial self-play. *arXiv preprint arXiv:2602.00173*, 2026b.

Yuetai Li, Xiang Yue, Zhangchen Xu, Fengqing Jiang, Luyao Niu, Bill Yuchen Lin, Bhaskar Ramasubramanian, and Radha Poovendran. Small models struggle to learn from strong reasoners. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 25366–25394, 2025a.

Ziheng Li, Zexu Sun, Jinman Zhao, Erxue Min, Yongcheng Zeng, Hui Wu, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, Xu Chen, et al. Staying in the sweet spot: Responsive reasoning evolution via capability-adaptive hint scaffolding. *arXiv preprint arXiv:2509.06923*, 2025b.

Baohao Liao, Hanze Dong, Xinxing Xu, Christof Monz, and Jiang Bian. Self-hinting language models enhance reinforcement learning. *arXiv preprint arXiv:2602.03143*, 2026.

Mingyang Liu, Gabriele Farina, and Asuman Ozdaglar. Uft: Unifying supervised and reinforcement fine-tuning. *Advances in Neural Information Processing Systems*, 38:101347–101383, 2026a.

Yihao Liu, Shuocheng Li, Lang Cao, Yuhang Xie, Mengyu Zhou, Haoyu Dong, Xiaojun Ma, Shi Han, and Dongmei Zhang. Superrl: Reinforcement learning with supervision to boost language model reasoning. *arXiv preprint arXiv:2506.01096*, 2025a.

Zehao Liu, Yuanpu Cao, Jinghui Chen, and Vasant G Honavar. Restoring the sweet spot: Pass-rate weighted self-distillation for llm reasoning. *arXiv preprint arXiv:2605.27765*, 2026b.

Ziru Liu, Cheng Gong, Xinyu Fu, Yaofang Liu, Ran Chen, Shoubo Hu, Suiyun Zhang, Rui Liu, Qingfu Zhang, and Dandan Tu. Ghpo: Adaptive guidance for stable and efficient llm reinforcement learning. *arXiv preprint arXiv:2507.10628*, 2025b.

Xingtai Lv, Yuxin Zuo, Youbang Sun, Hongyi Liu, Yuntian Wei, Zhekai Chen, Xuekai Zhu, Kaiyan Zhang, Bingning Wang, Ning Ding, et al. Towards a unified view of large language model post-training. *arXiv preprint arXiv:2509.04419*, 2025.

Lu Ma, Hao Liang, Meiyi Qiang, Lexiang Tang, Xiaochen Ma, Zhen Wong, Junbo Niu, Chengyu Shen, Runming He, Yanhao Li, et al. Learning what reinforcement learning can't: Interleaved online fine-tuning for hardest questions. In *International Conference on Learning Representations*, volume 2026, pp. 80802–80821, 2026.

Gongrui Nan, Siye Chen, Jing Huang, Mengyu Lu, Dexun Wang, Chunmei Xie, Weiqi Xiong, Xianzhou Zeng, Qixuan Zhou, Yadong Li, et al. Ngrpo: Negative-enhanced group relative policy optimization. *arXiv preprint arXiv:2509.18851*, 2025.

Vaskar Nath, Elaine Lau, Anisha Gunjal, Manasi Sharma, Nikhil Baharte, and Sean Hendryx. Adaptive guidance accelerates reinforcement learning of reasoning models. *arXiv preprint arXiv:2506.13923*, 2025.

Wenhua Nie, Jianan Wu, Junlin Liu, Ziwei Li, Zheng Lin, Zhang Zijian, Yilong Fan, Haoran Zheng, and Jyh-Shing Roger Jang. Gradient starvation in binary-reward grpo: Why group-mean centering fails and why the simplest fix works. *arXiv preprint arXiv:2605.07689*, 2026.

Du Phan, Matthew Douglas Hoffman, David Dohan, Sholto Douglas, Tuan Anh Le, Aaron Parisi, Pavel Sountsov, Charles Sutton, Sharad Vikram, and Rif A Saurous. Training chain-of-thought via latent-variable inference. *Advances in Neural Information Processing Systems*, 36:72819–72841, 2023.

Yuxiao Qu, Amrith Setlur, Virginia Smith, Ruslan Salakhutdinov, and Aviral Kumar. Pope: Learning to reason on hard problems via privileged on-policy exploration. *arXiv preprint arXiv:2601.18779*, 2026.

Shrinivas Ramasubramanian, Daman Arora, Fahim Tajwar, Guanning Zeng, Qingyang Wu, Zhongzhu Zhou, Chenfeng Xu, Haiwen Feng, Yuda Song, Aarti Singh, et al. Tail-likelihood reinforcement learning. *arXiv preprint arXiv:2609.02987*, 2026.

Xuan Ren, Biao Wu, and Lingqiao Liu. I learn better if you speak my language: Understanding the superior performance of fine-tuning large language models with llm-generated responses. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 10225–10245, 2024.

Yanwei Ren, Haotian Zhang, Likang Xiao, Xikai Zhang, Jiaxing Huang, Jiayan Qiu, Baosheng Yu, Quan Chen, and Liu Liu. Recycling failures: Salvaging exploration in rlvr via fine-grained off-policy guidance. *arXiv preprint arXiv:2602.24110*, 2026.

Amrith Setlur, Saurabh Garg, Xinyang Geng, Naman Garg, Virginia Smith, and Aviral Kumar. Rl on incorrect synthetic data scales the efficiency of llm math reasoning by eight-fold. *Advances in Neural Information Processing Systems*, 37:43000–43031, 2024.

Amrith Setlur, Zijian Wang, Andrew Cohen, Paria Rashidinejad, and Sang Michael Xie. Reuse your flops: Scaling rl on hard problems by conditioning on very off-policy prefixes. *arXiv preprint arXiv:2601.18795*, 2026.

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

Avi Singh, John D Co-Reyes, Rishabh Agarwal, Ankesh Anand, Piyush Patil, Xavier Garcia, Peter J Liu, James Harrison, Jaehoon Lee, Kelvin Xu, et al. Beyond human data: Scaling self-training for problem-solving with language models. *arXiv preprint arXiv:2312.06585*, 2023.

Mingyu Su, Jian Guan, Yuxian Gu, Minlie Huang, and Hongning Wang. Trust-region adaptive policy optimization. In *International Conference on Learning Representations*, volume 2026, pp. 51806–51828, 2026.

Fahim Tajwar, Guanning Zeng, Yueer Zhou, Yuda Song, Daman Arora, Yiding Jiang, Jeff Schneider, Ruslan Salakhutdinov, Haiwen Feng, and Andrea Zanette. Maximum likelihood reinforcement learning. *arXiv preprint arXiv:2602.02710*, 2026.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

Zhiheng Xi, Wenxiang Chen, Boyang Hong, Senjie Jin, Rui Zheng, Wei He, Yiwen Ding, Shichun Liu, Xin Guo, Junzhe Wang, et al. Training large language models for reasoning through reverse curriculum reinforcement learning. *arXiv preprint arXiv:2402.05808*, 2024.

Yu Xia, Canwen Xu, Zhewei Yao, Julian McAuley, and Yuxiong He. Learning to hint for reinforcement learning. *arXiv preprint arXiv:2604.00698*, 2026.

Jianhao Yan, Yafu Li, Zican Hu, Zhi Wang, Ganqu Cui, Xiaoye Qu, Yu Cheng, and Yue Zhang. Learning to reason under off-policy guidance. *Advances in Neural Information Processing Systems*, 38:117157–117186, 2026.

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

Matthew Yang, Hao Bai, Ian Wu, Gene Yang, Amrith Setlur, and Aviral Kumar. Int: Self-proposed interventions enable credit assignment in llm reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 85054–85091, 2026.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488, 2022.

Kaiyi Zhang, Ang Lv, Jinpeng Li, Yongbo Wang, Feng Wang, Haoyuan Hu, and Rui Yan. Stephint: Multi-level stepwise hints enhance reinforcement learning to reason. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 37846–37864, 2026a.

Wenhao Zhang, Yuexiang Xie, Yuchang Sun, Yanxi Chen, Guoyin Wang, Yaliang Li, Bolin Ding, and Jingren Zhou. On-policy rl meets off-policy experts: Harmonizing supervised fine-tuning and reinforcement learning via dynamic weighting. In *International Conference on Learning Representations*, volume 2026, pp. 120693–120726, 2026b.

Xiaoying Zhang, Yipeng Zhang, Hao Sun, Kaituo Feng, Chaochao Lu, Chao Yang, and Helen Meng. Critique-grpo: Advancing llm reasoning with natural language and numerical feedback. *arXiv preprint arXiv:2506.03106*, 2025.

Xichen Zhang, Sitong Wu, Yinghao Zhu, Haoru Tan, Shaozuo Yu, Ziyi He, and Jiaya Jia. Scaf-grpo: Scaffolded group relative policy optimization for enhancing llm reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 131946–131974, 2026c.

Xuechen Zhang, Zijian Huang, Yingcong Li, Chenshun Ni, Jiasi Chen, and Samet Oymak. Bread: Branched rollouts from expert anchors bridge sft & rl for reasoning. *Advances in Neural Information Processing Systems*, 38:96726–96752, 2026d.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

Ruiyang Zhou, Shuozhe Li, Amy Zhang, and Liu Leqi. Expo: Unlocking hard reasoning with self-explanation-guided reinforcement learning. *Advances in Neural Information Processing Systems*, 38:90989–91016, 2026a.

Xiangxin Zhou, Zichen Liu, Haonan Wang, Chao Du, Min Lin, Chongxuan Li, Liang Wang, and Tianyu Pang. Variational reasoning for language models. In *International Conference on Learning Representations*, volume 2026, pp. 154370–154407, 2026b.

Yuhang Zhou, Yixin Cao, and Guangnan Ye. From correctness to utility: Gain-based prefix evaluation for llm reasoning. *arXiv preprint arXiv:2606.07190*, 2026c.

A Proofs for the Distance to the Ideal Distribution

This appendix proves Theorem 1 and collects its consequences. Appendix A.1 fixes the setup. Appendix A.2 derives the gradient identity in Equation (3) and the distance of training on the reference. Appendix A.3 describes the accepted trajectory and the reference surprisal, and then proves Theorem 1. Appendix A.4 gives an equivalent reading of the closed form. Finally, Appendix A.5 illustrates the theorem with a finite example.

A.1 Setup

We use the setup of Section 3.1. Fix a question x, its correct reference r, and the current model, and suppress the conditioning on x. The trajectory space is finite, and π denotes the current policy $\pi_{\theta_{\text{old}}}$. Moreover, C is the set of correct trajectories, $p = \pi(C)$ is the success probability, and $\nu = \pi(\cdot \mid C)$ is the ideal distribution in Equation (2). We assume $0 < \pi(r) < 1$. For a token sequence h, we write $\pi(h)$ for the probability that a trajectory sampled from π begins with h, and $\pi(z \mid h)$ for the probability that the model continues h with z. Since the continuation follows the same policy, including its termination rule, autoregressive factorization gives $\pi(h \oplus z) = \pi(h)\pi(z \mid h)$ for every complete trajectory $h \oplus z$.

For a prefix length $0 \le t < |r|$, the reference splits as $r = h_t \oplus a_t$, and Q_t is the distribution of the accepted trajectory in Equation (1). Besides d_t and S_t in Equation (4), the proofs use the probabilities of an exact copy of the reference and of a correct candidate trajectory,

$$u_t = \pi(a_t \mid h_t), \qquad v_t = \Pr_{Z \sim \pi(\cdot \mid h_t)}\big(h_t \oplus Z \in C\big).$$

Here an exact copy means equality of the complete token sequences, and different token sequences can still follow the same mathematical strategy. By definition, $S_t = -\log u_t$, and in particular $S_0 = -\log \pi(r)$ is the reference surprisal of the empty prefix. Since autoregressive factorization gives $u_t = \pi(r)/\pi(h_t) \ge \pi(r) > 0$, we have $d_t \le 1 - u_t < 1$, and every S_t is finite.

A.2 The ideal distribution

We first justify the choice of the ideal distribution as the learning target in Section 2. Lemma A.1 shows that learning the likelihood of its trajectories increases the success probability in expectation. Lemma A.2 then gives the distance of training on the reference, which is the first term in Theorem 1.

Lemma A.1 (Gradient identity)**.** *Suppose that π_θ is differentiable in θ and assigns positive probability to every trajectory in a neighborhood of θ_{old}. Write $p_\theta = \pi_\theta(C)$. Then*

$$\mathbb{E}_{Y \sim \nu}\Big[\nabla_\theta \log \pi_\theta(Y)\big|_{\theta=\theta_{\text{old}}}\Big] = \nabla_\theta \log p_\theta\big|_{\theta=\theta_{\text{old}}}.$$

Proof. Since the trajectory space is finite, differentiating $p_\theta = \sum_{y \in C} \pi_\theta(y)$ term by term gives

$$\nabla_\theta \log p_\theta = \frac{1}{p_\theta} \sum_{y \in C} \nabla_\theta \pi_\theta(y) = \sum_{y \in C} \frac{\pi_\theta(y)}{p_\theta} \nabla_\theta \log \pi_\theta(y).$$

At $\theta = \theta_{\text{old}}$, the weight $\pi_\theta(y)/p_\theta$ of each $y \in C$ equals $\nu(y)$ by Equation (2), which proves the claim. □

Here ν is fixed at the current model, and differentiation applies only to the trajectory log likelihood.

Lemma A.2 (Distance of training on the reference)**.** *Training on the reference has distance*

$$D_{\text{KL}}(\delta_r \| \nu) = \log \frac{p}{\pi(r)} = S_0 + \log p \in [0, S_0].$$

In contrast, if ν assigns positive probability to a trajectory other than r, then $D_{\text{KL}}(\nu \| \delta_r) = \infty$.

Proof. Since δ_r assigns all probability to r and $r \in C$, Equation (2) gives $D_{\text{KL}}(\delta_r \| \nu) = -\log \nu(r) = \log(p/\pi(r))$. Substituting $S_0 = -\log \pi(r)$ gives $S_0 + \log p$. This quantity is nonnegative since $\nu(r) \le 1$, and it is at most S_0 since $p \le 1$. For the reverse direction, a trajectory $y \ne r$ with $\nu(y) > 0$ has $\delta_r(y) = 0$, which makes the term $\nu(y) \log(\nu(y)/\delta_r(y))$ infinite. □

Thus, the direction of KL in Section 2 permits a finite comparison with training on the reference.

A.3 Proof of Theorem 1

The closed form in Theorem 1 combines two ingredients. Lemma A.3 describes the distribution of the accepted trajectory, and Lemma A.4 relates the reference surprisal to the probability of the prefix.

Lemma A.3 (Distribution of the accepted trajectory)**.** *For every prefix length t, we have $d_t = v_t - u_t$ and $Q_t(r) = 1 - d_t$. Moreover, every trajectory $y \neq r$ with $Q_t(y) > 0$ is correct and has the form $y = h_t \oplus z$, and it satisfies*

$$Q_t(y) = \pi(z \mid h_t), \qquad \frac{Q_t(y)}{\nu(y)} = \frac{p}{\pi(h_t)}.$$

These trajectories have total probability d_t under Q_t.

Proof. The candidate trajectory $\widetilde{Y}_t = h_t \oplus Z$ equals r exactly when $Z = a_t$, which has probability u_t. Since $r \in C$, the event $\{\widetilde{Y}_t \in C\}$ is the disjoint union of $\{\widetilde{Y}_t = r\}$ and $\{\widetilde{Y}_t \in C,\ \widetilde{Y}_t \neq r\}$. Therefore, $v_t = u_t + d_t$. By Equation (1), the accepted trajectory equals r when the candidate trajectory is an exact copy or is incorrect. These two events are disjoint and have total probability $u_t + (1 - v_t) = 1 - d_t$.

For $y \neq r$, the accepted trajectory equals y only when $\widetilde{Y}_t = y$ and $y \in C$. Hence, $Q_t(y) > 0$ requires $y \in C$ and $y = h_t \oplus z$ for some z, in which case $Q_t(y) = \pi(z \mid h_t)$. Autoregressive factorization gives $\pi(z \mid h_t) = \pi(y)/\pi(h_t)$, where $\pi(h_t) \geq \pi(r) > 0$. Since $\nu(y) = \pi(y)/p$ for $y \in C$, the ratio $Q_t(y)/\nu(y)$ equals $p/\pi(h_t)$. Finally, these trajectories carry the remaining probability $1 - Q_t(r) = d_t$. □

Lemma A.4 (Monotonicity of the reference surprisal)**.** *For every prefix length t,*

$$S_0 - S_t = -\log \pi(h_t), \qquad 0 \leq S_t \leq S_0.$$

Moreover, S_t is nonincreasing in t.

Proof. Autoregressive factorization gives $\pi(r) = \pi(h_t)\pi(a_t \mid h_t)$. Taking negative logarithms gives $S_0 = -\log \pi(h_t) + S_t$. Since $\pi(a_t \mid h_t) \leq 1$ and $\pi(h_t) \leq 1$, both S_t and $S_0 - S_t$ are nonnegative. For $s \leq t$, every trajectory that begins with h_t also begins with h_s. Therefore, $\pi(h_t) \leq \pi(h_s)$, which gives $S_t = S_0 + \log \pi(h_t) \leq S_s$. □

With these two lemmas, the proof of Theorem 1 reduces to a direct computation.

Theorem 1 (Distance to the ideal distribution, restated)**.** *Under the setup in Appendix A.1, for every prefix length t,*

$$D_{\mathrm{KL}}(Q_t \| \nu) = \underbrace{\log \frac{p}{\pi(r)}}_{D_{\mathrm{KL}}(\delta_r \| \nu)} - d_t S_t + (1 - d_t)\log(1 - d_t).$$

Moreover, every $t^* \in \arg\max_t d_t S_t$ satisfies $D_{\mathrm{KL}}(Q_{t^*} \| \nu) \leq \min_t D_{\mathrm{KL}}(Q_t \| \nu) + 1/e$.

Proof. Write $d = d_t$. By Lemma A.3, Q_t assigns probability $1 - d$ to r, whose likelihood ratio is $Q_t(r)/\nu(r) = (1-d)p/\pi(r)$. It assigns the remaining probability d to trajectories whose likelihood ratio is $p/\pi(h_t)$. Therefore,

$$D_{\mathrm{KL}}(Q_t \| \nu) = (1 - d)\log \frac{(1-d)p}{\pi(r)} + d \log \frac{p}{\pi(h_t)}.$$

Lemma A.4 gives $\log \pi(h_t) = \log \pi(r) + S_t$, and hence $d \log(p/\pi(h_t)) = d \log(p/\pi(r)) - dS_t$. Collecting the two terms in $\log(p/\pi(r))$ proves the closed form. Its first term equals $D_{\mathrm{KL}}(\delta_r \| \nu)$ by Lemma A.2 and does not depend on t.

For the second statement, write $D_r = D_{\mathrm{KL}}(\delta_r \| \nu)$ and $c_t = (1 - d_t)\log(1 - d_t)$. Since $-a \log a \in [0, 1/e]$ for $a \in [0, 1]$, we have $c_t \in [-1/e, 0]$. For every prefix length t, the maximality of $d_{t^*} S_{t^*}$

gives

$$D_{\mathrm{KL}}(Q_{t^*}\|\nu) = D_r - d_{t^*}S_{t^*} + c_{t^*} \le D_r - d_{t^*}S_{t^*} \le D_r - d_tS_t$$
$$= D_{\mathrm{KL}}(Q_t\|\nu) - c_t \le D_{\mathrm{KL}}(Q_t\|\nu) + \frac{1}{e}.$$

Taking the minimum over the finitely many prefix lengths completes the proof. □

Since $d_tS_t \ge 0$ and $(1-d_t)\log(1-d_t) \le 0$, the closed form also gives $D_{\mathrm{KL}}(Q_t\|\nu) \le D_{\mathrm{KL}}(\delta_r\|\nu)$ for every prefix length t. Therefore, a continuation with fallback never moves the distribution of the accepted trajectory farther from the ideal distribution than training on the reference.

A.4 Consequences of Theorem 1

We now give an equivalent reading of the closed form. Corollary A.5 rewrites the distance in terms of the reference content kept in the accepted trajectory.

Corollary A.5 (Reference content kept in the accepted trajectory)**.** *Let $\bar{h}$ be the reference content kept in the accepted trajectory $Y \sim Q_t$, which is r when $Y = r$ and the prefix h_t otherwise. Then, for every prefix length t,*

$$D_{\mathrm{KL}}(Q_t\|\nu) = \log p + \underbrace{(1-d_t)S_0 + d_t(S_0 - S_t)}_{\mathbb{E}[-\log\pi(\bar{h})]} + (1-d_t)\log(1-d_t).$$

Proof. Lemma A.2 gives $\log(p/\pi(r)) = \log p + S_0$. Substituting this expression and $S_0 = (1-d_t)S_0 + d_tS_0$ into the closed form of Theorem 1 gives the display. By Lemma A.3, $\bar{h} = r$ with probability $1-d_t$ and $\bar{h} = h_t$ with probability d_t. Since $-\log\pi(r) = S_0$ and $-\log\pi(h_t) = S_0 - S_t$ by Lemma A.4, the middle two terms equal $\mathbb{E}[-\log\pi(\bar{h})]$. □

In other words, apart from $\log p$, which does not depend on the prefix, and a term in $[-1/e, 0]$, the distance to the ideal distribution is the expected surprisal of the reference content that the model has not replaced. Both extremes of the prefix length keep nearly all of the reference surprisal S_0. A prefix covering almost the entire reference keeps it through the prefix, while an empty prefix on a hard question keeps it through fallback.

A.5 A finite example

This example shows that neither the success probability of the continuation nor the successful replacement probability alone determines which prefix gives the smaller distance. Consider five complete trajectories whose displayed strings omit a shared deterministic termination token, $\mathtt{H}\oplus\mathtt{r}$, $\mathtt{H}\oplus\mathtt{s}$, $\mathtt{H}\oplus\mathtt{f}$, $\mathtt{O}\oplus\mathtt{t}$, and $\mathtt{O}\oplus\mathtt{g}$, with respective probabilities 0.08, 0.01, 0.01, 0.20, and 0.70. The correct trajectories are $\mathtt{H}\oplus\mathtt{r}$, $\mathtt{H}\oplus\mathtt{s}$, and $\mathtt{O}\oplus\mathtt{t}$, and the reference is $\mathtt{H}\oplus\mathtt{r}$. The first tokens are H and O, with probabilities 0.10 and 0.90, and the second-token probabilities are the corresponding conditional probabilities. These probabilities define a finite autoregressive distribution.

Prefix	v_t	u_t	d_t	S_t	d_tS_t	$D_{\mathrm{KL}}(Q_t\|\nu)$
Empty	0.29	0.08	0.21	2.525729	0.530403	0.571231
H	0.90	0.80	0.10	0.223144	0.022314	1.170715

Training on the reference gives $D_{\mathrm{KL}}(\delta_r\|\nu) = 1.287854$. Supplying H increases v_t while making an exact copy more likely. In this setting, both the successful replacement probability d_t and the reference surprisal S_t decrease. Therefore, the empty prefix gives the larger d_tS_t and the smaller distance despite the lower success probability of its continuation.

To isolate the role of the reference surprisal, keep the three probabilities under H unchanged and assign probabilities 0.04 and 0.86 to $\mathtt{O}\oplus\mathtt{t}$ and $\mathtt{O}\oplus\mathtt{g}$, respectively. The resulting comparison is

Prefix	v_t	u_t	d_t	S_t	d_tS_t	$D_{\mathrm{KL}}(Q_t\|\nu)$
Empty	0.13	0.08	0.05	2.525729	0.126286	0.310493
`H`	0.90	0.80	0.10	0.223144	0.022314	0.368369

Training on the reference now gives $D_{\mathrm{KL}}(\delta_r\|\nu) = 0.485508$. Here supplying `H` increases both v_t and d_t. However, the decrease in the reference surprisal outweighs the increase in d_t. Therefore, the empty prefix still gives the larger d_tS_t and the smaller distance. Thus, even the successful replacement probability alone can favor a different prefix from the one preferred by Theorem 1. Both examples concern a fixed model and illustrate why the choice of prefix should account for d_t and S_t together.

B Proofs for Learning the Prefix Selector

This appendix proves Theorem 2 and analyzes the prefix selector during training. Appendix B.1 fixes the setup and derives the properties of the selector feedback, and Appendix B.2 proves Theorem 2. Appendix B.3 examines the effects of sharing the selector across questions and of randomizing the guidance level. Finally, Appendix B.4 derives the expected update of the selector in the GRPO application as the model and the questions change, and Appendix B.5 bounds the accumulated sampling noise.

B.1 Setup and selector feedback

Fix the current model, the references, and $\mathcal{D}_{\mathrm{guide}}$ as in Section 3.3. We assume that every question satisfies the setup of Appendix A.1, which includes $0 < \pi_{\theta_{\mathrm{old}}}(r_x \mid x) < 1$ and hence $0 < S_{0,x} < \infty$. For a question x, we write ν_x for its ideal distribution. For a guidance level α, we write $Q_{x,\alpha}$, $d_{x,\alpha}$, and $S_{x,\alpha}$ for the distribution of the accepted trajectory, the successful replacement probability, and the reference surprisal at the prefix length $t_x(\alpha)$. Here $t_x(\alpha)$ is the actual prefix length, including the rounding and boundary adjustment in Section 4.1. Therefore, the nominal level α can differ from the realized ratio $S_{x,\alpha}/S_{0,x}$. We first collect the properties of the selector feedback for a single question, for which we suppress x as in Appendix A.

Lemma B.1 (Selector feedback)**.** *For every prefix length t, the selector feedback satisfies $W_t \in [0,1]$ and $\mathbb{E}[W_t] = d_tS_t/S_0$. Moreover, with $c_t = (1-d_t)\log(1-d_t)$,*

$$\frac{D_{\mathrm{KL}}(Q_t\|\nu)}{S_0} = \frac{D_{\mathrm{KL}}(\delta_r\|\nu)}{S_0} - \mathbb{E}[W_t] + \frac{c_t}{S_0}, \qquad \frac{c_t}{S_0} \in \left[-\frac{1}{eS_0}, 0\right], \tag{14}$$

and the normalized distances satisfy $0 \le D_{\mathrm{KL}}(Q_t\|\nu)/S_0 \le D_{\mathrm{KL}}(\delta_r\|\nu)/S_0 \le 1$.

Proof. The indicator in Equation (7) has expectation d_t by Equation (4). Lemma A.4 gives $0 \le S_t \le S_0$, and hence $W_t \in [0,1]$ and $\mathbb{E}[W_t] = d_tS_t/S_0$, which proves Equation (8). Dividing the closed form of Theorem 1 by S_0 gives Equation (14), and the range of c_t follows from $-a\log a \in [0, 1/e]$ for $a \in [0,1]$. Finally, since $\mathbb{E}[W_t] \ge 0$ and $c_t \le 0$, Equation (14) gives $D_{\mathrm{KL}}(Q_t\|\nu) \le D_{\mathrm{KL}}(\delta_r\|\nu)$, and Lemma A.2 gives $0 \le D_{\mathrm{KL}}(\delta_r\|\nu) \le S_0$. □

Thus, a larger expected feedback corresponds to a smaller normalized distance, up to the bounded last term. Since normalization divides d_tS_t at every prefix of the same reference by the same S_0, it preserves the ranking of d_tS_t across these prefixes. Because of the bounded last term, this ranking need not equal the ranking of the exact distances. Across questions, normalization expresses each distance relative to the reference surprisal of the empty prefix.

The ablation without surprisal weighting in Section 5.3 replaces W_t with the indicator of successful replacement, whose expectation is d_t. Since this indicator still excludes exact copies of the reference, comparing it with W_t isolates the effect of weighting a successful replacement by the reference surprisal.

For the rest of this appendix, we write

$$g_x(\alpha) = \mathbb{E}[W_{x,\alpha}] = \frac{d_{x,\alpha}S_{x,\alpha}}{S_{0,x}} \in [0,1]$$

for the expected selector feedback of question x at level α, where the equality follows from Lemma B.1.

B.2 Proof of Theorem 2

Theorem 2 measures a selector by its average normalized distance and compares it with training on the reference. Written out, these distances are

$$\mathcal{L}(\phi) = \mathbb{E}_{X\sim\mathcal{D}_{\text{guide}},\,\alpha\sim\rho_\phi}\left[\frac{D_{\mathrm{KL}}(Q_{X,\alpha}\|\nu_X)}{S_{0,X}}\right], \qquad \mathcal{L}_{\text{ref}} = \mathbb{E}_{X\sim\mathcal{D}_{\text{guide}}}\left[\frac{D_{\mathrm{KL}}(\delta_{r_X}\|\nu_X)}{S_{0,X}}\right].$$

By Lemma B.1 and $g_x(\alpha) \in [0,1]$, the quantities $\mathcal{L}(\phi)$, $\mathcal{L}_{\text{ref}}$, and $J(\phi)$ lie in $[0,1]$.

Theorem 2 (Learning the prefix selector from feedback, restated)**.** *Fix the current model, the references, and $\mathcal{D}_{\text{guide}}$, where every question satisfies the setup of Section 3.1. Let $\mathcal{L}(\phi)$ and $\mathcal{L}_{\text{ref}}$ be the expectations of $D_{\mathrm{KL}}(Q_{X,\alpha}\|\nu_X)/S_{0,X}$ and $D_{\mathrm{KL}}(\delta_{r_X}\|\nu_X)/S_{0,X}$ over $X \sim \mathcal{D}_{\text{guide}}$ and $\alpha \sim \rho_\phi$. Then*

$$J(\phi) \le \mathcal{L}_{\text{ref}} - \mathcal{L}(\phi) \le J(\phi) + \mathbb{E}_X\left[\frac{1}{eS_{0,X}}\right].$$

Moreover, for a differentiable prefix selector with positive probabilities and a baseline b fixed before sampling,

$$\mathbb{E}\left[(W_{X,\alpha} - b)\nabla_\phi \log \rho_\phi(\alpha)\right] = \nabla_\phi J(\phi).$$

Proof. For the first statement, averaging Equation (14) over $X \sim \mathcal{D}_{\text{guide}}$ and $\alpha \sim \rho_\phi$ gives

$$\mathcal{L}(\phi) = \mathcal{L}_{\text{ref}} - J(\phi) + \epsilon(\phi), \qquad -\mathbb{E}_X\frac{1}{eS_{0,X}} \le \epsilon(\phi) \le 0, \tag{15}$$

where $\epsilon(\phi) = \mathbb{E}[c_{X,\alpha}/S_{0,X}]$ and $c_{x,\alpha}$ is the last term of Theorem 1 at the prefix length $t_x(\alpha)$. Lemma B.1 bounds each $c_{x,\alpha}/S_{0,x}$, and rearranging Equation (15) gives the first statement.

For the second statement, the level is sampled independently of the question. Condition on $X = x$. Since $\alpha \sim \rho_\phi$ and $\mathbb{E}[W_{x,\alpha} \mid \alpha] = g_x(\alpha)$,

$$\mathbb{E}\left[(W_{x,\alpha} - b)\nabla_\phi \log \rho_\phi(\alpha) \mid X = x\right] = \sum_{\alpha\in\mathcal{A}}(g_x(\alpha) - b)\nabla_\phi\rho_\phi(\alpha) = \sum_{\alpha\in\mathcal{A}} g_x(\alpha)\nabla_\phi\rho_\phi(\alpha).$$

The baseline term vanishes since $\sum_\alpha \nabla_\phi\rho_\phi(\alpha) = \nabla_\phi 1 = 0$. Define $\mu_\alpha = \mathbb{E}_{X\sim\mathcal{D}_{\text{guide}}}[g_X(\alpha)] \in [0,1]$, which does not depend on ϕ. Then

$$J(\phi) = \sum_{\alpha\in\mathcal{A}} \rho_\phi(\alpha)\mu_\alpha, \qquad \nabla_\phi J(\phi) = \sum_{\alpha\in\mathcal{A}} \mu_\alpha \nabla_\phi\rho_\phi(\alpha),$$

where differentiation acts on a finite sum with fixed bounded coefficients. Averaging the conditional identity over $X \sim \mathcal{D}_{\text{guide}}$ gives the same expression, which proves the second statement. □

The first statement leads to the following guarantee for a selector that maximizes J.

Corollary B.2 (Maximizing the objective)**.** *If $J(\phi^*) \ge J(\phi)$, then $\mathcal{L}(\phi^*) \le \mathcal{L}(\phi) + \mathbb{E}_X[1/(eS_{0,X})]$.*

Proof. By Equation (15), $J(\phi^*) \ge J(\phi)$ gives

$$\mathcal{L}(\phi^*) \le \mathcal{L}_{\text{ref}} - J(\phi^*) \le \mathcal{L}_{\text{ref}} - J(\phi) \le \mathcal{L}(\phi) + \mathbb{E}_X\frac{1}{eS_{0,X}}.$$ □

Thus, a larger objective corresponds to a smaller average normalized distance, up to a bounded correction. Since the correction $\epsilon(\phi)$ can change with the selector, J and $\mathcal{L}$ can rank selectors differently within this bound. The bound is informative only when $\mathbb{E}_X[1/S_{0,X}]$ is small, and it can be large when some reference surprisals are close to zero.

In Section 4.2, the baseline is a running average of past feedback. For such a baseline, the same argument applies after conditioning on the history and the current question. Given this information, the baseline is fixed, the level still follows ρ_ϕ, and its feedback still has conditional mean $g_x(\alpha)$. Moreover, the baseline update in Appendix B.4 keeps $b \in [0,1]$, which ensures integrability.

B.3 SHARING AND RANDOMIZING GUIDANCE LEVELS

The prefix selector shares one distribution over guidance levels across questions and samples a level for each continuation. We now examine both choices. Proposition B.3 bounds the cost of sharing within the objective J. Proposition B.4 then compares the distance after randomizing the level with the average over levels in J.

For any distribution ρ over $\mathcal{A}$, write $J(\rho) = \sum_\alpha \rho(\alpha)\mu_\alpha$ with μ_α from the proof of Theorem 2, which gives $J(\rho_\phi) = J(\phi)$. The best shared distribution has value $J^*_{\text{shared}} = \max_\alpha \mu_\alpha$. In contrast, an oracle that knows g_x for every question and chooses a level for each question separately has value $J^*_{\text{individual}} = \mathbb{E}_X \max_\alpha g_X(\alpha)$.

Proposition B.3 (Cost of sharing)**.** *For every distribution ρ over $\mathcal{A}$,*

$$J^*_{\text{individual}} - J(\rho) = \big(J^*_{\text{individual}} - J^*_{\text{shared}}\big) + \big(J^*_{\text{shared}} - J(\rho)\big), \tag{16}$$

where both terms are nonnegative. Moreover, the prefix selector in Equation (11) *satisfies*

$$\sup_\phi J(\phi) = (1 - \varepsilon_{\text{sel}}) \max_\alpha \mu_\alpha + \frac{\varepsilon_{\text{sel}}}{|\mathcal{A}|} \sum_\alpha \mu_\alpha.$$

Proof. The decomposition is an identity. The first term is nonnegative since $\max_\alpha \mathbb{E}_X[g_X(\alpha)] \le \mathbb{E}_X \max_\alpha g_X(\alpha)$. The second term is nonnegative since $J(\rho)$ is a convex combination of the values μ_α. For the selector in Equation (11), write q_ϕ for its softmax component. Then $J(\phi) = (1 - \varepsilon_{\text{sel}}) \sum_\alpha q_\phi(\alpha)\mu_\alpha + (\varepsilon_{\text{sel}}/|\mathcal{A}|) \sum_\alpha \mu_\alpha$. The first sum is at most $\max_\alpha \mu_\alpha$, and it approaches this value as q_ϕ concentrates on a maximizing level. □

The first term in Equation (16) measures the cost of using the same distribution over levels for every question. It is zero if one level maximizes g_x for almost every question. The second term measures the gap between the current selector and the best shared distribution. If all μ_α are equal, every shared distribution has the same objective even when individual questions prefer different levels. Moreover, the uniform component of the selector has an explicit cost in the current objective, while it keeps every level observed as the model changes.

Through Theorem 2, the objective J corresponds to the average of the normalized distance over the selected levels. However, sampling a level and then making one continuation produces a single distribution of the accepted trajectory, which mixes the distributions for different levels. For a fixed question x and a distribution ρ over $\mathcal{A}$, this mixture is $Q_{x,\rho} = \sum_\alpha \rho(\alpha) Q_{x,\alpha}$. Let I_x be the mutual information between the sampled level and the accepted trajectory, and let $\mathcal{H}(\rho) = -\sum_\alpha \rho(\alpha) \log \rho(\alpha)$ be the entropy of ρ.

Proposition B.4 (Randomizing the guidance level)**.** *For every question x and distribution ρ over $\mathcal{A}$,*

$$D_{\text{KL}}(Q_{x,\rho} \| \nu_x) = \sum_\alpha \rho(\alpha) D_{\text{KL}}(Q_{x,\alpha} \| \nu_x) - I_x,$$

where $0 \le I_x \le \mathcal{H}(\rho)$. Consequently, with $j_x(\rho) = \sum_\alpha \rho(\alpha) g_x(\alpha)$,

$$\frac{D_{\text{KL}}(\delta_{r_x} \| \nu_x)}{S_{0,x}} - j_x(\rho) - \frac{1/e + \mathcal{H}(\rho)}{S_{0,x}} \le \frac{D_{\text{KL}}(Q_{x,\rho} \| \nu_x)}{S_{0,x}} \le \frac{D_{\text{KL}}(\delta_{r_x} \| \nu_x)}{S_{0,x}} - j_x(\rho).$$

Proof. Every $Q_{x,\alpha}$ has finite distance by Lemma B.1. Splitting the log probability ratio through $Q_{x,\rho}$ and summing over trajectories with positive probability gives

$$\begin{aligned} \sum_\alpha \rho(\alpha) D_{\text{KL}}(Q_{x,\alpha} \| \nu_x) &= \sum_{\alpha,y} \rho(\alpha) Q_{x,\alpha}(y) \left[\log \frac{Q_{x,\alpha}(y)}{Q_{x,\rho}(y)} + \log \frac{Q_{x,\rho}(y)}{\nu_x(y)}\right] \\ &= I_x + D_{\text{KL}}(Q_{x,\rho} \| \nu_x), \end{aligned}$$

since $\sum_\alpha \rho(\alpha) D_{\text{KL}}(Q_{x,\alpha} \| Q_{x,\rho})$ equals the mutual information I_x. The mutual information is nonnegative and at most the entropy of the level, which gives $0 \le I_x \le \mathcal{H}(\rho)$. For the bounds, divide the identity by $S_{0,x}$ and apply Equation (14) to each level. The upper bound uses $I_x \ge 0$ and $c_{x,\alpha} \le 0$, and the lower bound uses $I_x \le \mathcal{H}(\rho)$ and $c_{x,\alpha} \ge -1/e$. □

The mutual information is zero when all distributions with positive weight coincide. Otherwise, randomizing the level gives a smaller distance than the average over levels. This construction returns one accepted trajectory from one continuation, and Appendix C.2 analyzes acceptance after multiple continuations.

B.4 Selector updates in the GRPO application

Theorem 2 fixes the model, the references, and $\mathcal{D}_{\text{guide}}$. We now specify $\mathcal{D}_{\text{guide}}$ and the baseline in the GRPO application of Section 4.3, and then derive the expected update of the selector for a batch and across training iterations.

Questions selected for reference guidance. Fix the current model and a reference r_x for each question. Since the G responses of a group are sampled independently, the group is an all-failure group with probability $f_x = (1 - p_{\theta_{\text{old}}}(x))^G$. For $X \sim \mathcal{D}$, define $\zeta = \mathbb{E}_{\mathcal{D}}[f_X]$. When $\zeta > 0$, the questions that receive reference guidance follow

$$\mathcal{D}_{\text{guide}}(x) = \frac{\mathcal{D}(x) f_x}{\zeta}. \tag{17}$$

Given the question, the guidance levels and continuations are sampled independently of the group. Therefore, restricting guidance to questions with an all-failure group changes their relative weights while preserving $\mathbb{E}[W_{x,\alpha}] = g_x(\alpha)$ for each question and level. If $\zeta = 0$, no question receives guidance and the selector update is zero almost surely.

Baseline and update order. The selector starts with $\phi = 0$ and $b = 0$. Every term in Equation (12) uses the same selector parameters and baseline, which are fixed before the batch is collected. After a nonempty batch of continuations, the baseline is updated by

$$b \leftarrow (1 - \beta) b + \frac{\beta}{|\mathcal{U}|} \sum_{j \in \mathcal{U}} W_j, \tag{18}$$

where $0 < \beta \leq 1$. Since this update is a convex combination of values in $[0, 1]$, induction gives $b \in [0, 1]$ at every iteration. Centering the updated logits by subtracting their mean leaves the selection probabilities unchanged. An empty batch of continuations leaves ϕ and b unchanged. Table 3 specifies the numerical hyperparameters. Reference surprisals, verification outcomes, and model parameters are fixed during the selector update. This update precedes the policy update, and both use the continuations generated by the same current model. All continuations supply feedback, including K zero feedback values when no continuation gives a successful replacement and ARG falls back to the reference.

Expected batch update. Let Ψ denote the summed direction in Equation (12) before multiplication by the step size,

$$\Psi = \sum_{j \in \mathcal{U}} (W_j - b) \nabla_\phi \log \rho_\phi(\alpha_j),$$

with $\Psi = 0$ when $\mathcal{U}$ is empty.

Proposition B.5 (Expected batch update)**.** *Consider a batch of n question positions, each with marginal distribution $\mathcal{D}$. Suppose that the model, the selector, and the baseline are fixed during collection, and that every all-failure group receives K continuations whose levels and feedback follow the laws above. Then*

$$\mathbb{E}[\Psi] = nK\zeta \nabla_\phi J(\phi), \tag{19}$$

where the right-hand side is zero when $\zeta = 0$.

Proof. Write Ψ as a sum over question positions i and continuations $k = 1, \ldots, K$ of the terms $\mathbf{1}\{\text{group } i \text{ is all-failure}\}(W_{ik} - b)\nabla_\phi \log \rho_\phi(\alpha_{ik})$. Conditional on the question x at position i and on its all-failure event, each term has expectation $\sum_\alpha g_x(\alpha) \nabla_\phi \rho_\phi(\alpha)$ by the proof of Theorem 2. Therefore, position i contributes

$$\sum_x \mathcal{D}(x) f_x K \sum_\alpha g_x(\alpha) \nabla_\phi \rho_\phi(\alpha) = K\zeta \sum_x \mathcal{D}_{\text{guide}}(x) \sum_\alpha g_x(\alpha) \nabla_\phi \rho_\phi(\alpha) = K\zeta \nabla_\phi J(\phi),$$

where the first equality uses Equation (17). Summing over the n positions by linearity proves the claim. When $\zeta = 0$, no group is an all-failure group almost surely, and hence $\Psi = 0$ almost surely. □

Since the proof uses only the marginal laws, it requires no independence between question positions or between continuations. Holding J fixed, the factors n, K, and ζ only scale the expected update at a fixed step size. However, changes in the all-failure probabilities also change the question weights in J and hence the update direction.

Changing models and questions. Across training iterations, the model, the question distribution, and the selector all change. Let $(\mathcal{F}_k)_k$ be the increasing filtration of the training history, where $\mathcal{F}_k$ contains the history before iteration k, including the current model, the selector parameters ϕ_k, and the baseline. For the question positions $X_{k,1}, \ldots, X_{k,n}$ at iteration k, define their average conditional marginal distribution by

$$\mathcal{D}_k(x) = \frac{1}{n} \sum_{i=1}^{n} \Pr(X_{k,i} = x \mid \mathcal{F}_k).$$

With the current all-failure probabilities $f_{k,x}$ and expected feedback $g_{k,x}(\alpha)$, define

$$\zeta_k = \mathbb{E}_{X \sim \mathcal{D}_k}[f_{k,X}], \qquad \mathcal{D}_{\mathrm{guide},k}(x) = \frac{\mathcal{D}_k(x) f_{k,x}}{\zeta_k},$$

$$J_k(\phi) = \mathbb{E}_{X \sim \mathcal{D}_{\mathrm{guide},k}} \sum_{\alpha} \rho_\phi(\alpha) g_{k,X}(\alpha),$$

where the last two are defined when $\zeta_k > 0$. Let Ψ_k be the summed direction at iteration k.

Proposition B.6 (Expected update under changing models and questions)**.** *Suppose that, conditional on $\mathcal{F}_k$, the groups and continuations at iteration k follow the laws of Proposition B.5 under the current model, except that position i has the marginal distribution $\Pr(X_{k,i} = \cdot \mid \mathcal{F}_k)$. Then*

$$\mathbb{E}[\Psi_k \mid \mathcal{F}_k] = nK\zeta_k \nabla_\phi J_k(\phi_k), \tag{20}$$

where the right-hand side is zero when $\zeta_k = 0$.

Proof. Apply the proof of Proposition B.5 conditionally on $\mathcal{F}_k$, with the marginal distribution of position i given by $\Pr(X_{k,i} = \cdot \mid \mathcal{F}_k)$. Since the proof uses only marginal laws, the positions may have different marginal distributions. Summing their contributions replaces $n\mathcal{D}$ with $n\mathcal{D}_k$, which gives Equation (20). □

The gradient in Equation (20) holds the history-dependent model, question distribution, and prefix mapping fixed. Thus, the update follows the current objective in expectation even as training changes these quantities. The setting includes sampling with replacement from a fixed distribution, for which $\mathcal{D}_k = \mathcal{D}$, and sampling without replacement from a remaining pool of questions.

B.5 Accumulated sampling noise

Proposition B.6 describes the expected update at each iteration. We now bound how far the accumulated updates deviate from these expectations, starting with the size of each update.

Lemma B.7 (Bounded update direction)**.** *For the selector in Equation* (11) *and a baseline in* $[0, 1]$*, every term of Ψ_k has Euclidean norm at most $\sqrt{2}$. Consequently, $\|\Psi_k/(nK)\|_2 \le \sqrt{2}$.*

Proof. Let q_ϕ be the softmax component of Equation (11) and $\mathbf{e}_\alpha$ the coordinate vector for level α. Then

$$\nabla_\phi \log \rho_\phi(\alpha) = \frac{(1 - \varepsilon_{\mathrm{sel}}) q_\phi(\alpha)}{\rho_\phi(\alpha)} \big(\mathbf{e}_\alpha - q_\phi\big).$$

The prefactor is at most one since $\rho_\phi(\alpha) \ge (1 - \varepsilon_{\mathrm{sel}}) q_\phi(\alpha)$. Moreover, $\|\mathbf{e}_\alpha - q_\phi\|_2^2 = (1 - q_\phi(\alpha))^2 + \sum_{\alpha' \ne \alpha} q_\phi(\alpha')^2 \le 2(1 - q_\phi(\alpha))^2 \le 2$. Since W_j and b both lie in $[0, 1]$, we have $|W_j - b| \le 1$, which bounds each term by $\sqrt{2}$. Finally, Ψ_k has at most nK terms, including the case of no continuations. □

Define the centered errors

$$\xi_k = \frac{\Psi_k}{nK} - \zeta_k \nabla_\phi J_k(\phi_k).$$

Proposition B.8 (Accumulated sampling noise)**.** *Under the conditions of Proposition B.6 and a baseline in* $[0, 1]$*, for a deterministic number* N_{iter} *of iterations,*

$$\mathbb{E}\left\|\frac{1}{N_{\text{iter}}}\sum_{k=1}^{N_{\text{iter}}}\xi_k\right\|_2^2 \le \frac{2}{N_{\text{iter}}}. \tag{21}$$

Proof. Proposition B.6 gives $\mathbb{E}[\xi_k \mid \mathcal{F}_k] = 0$. The conditional variance identity and Lemma B.7 then give

$$\mathbb{E}\big[\|\xi_k\|_2^2 \mid \mathcal{F}_k\big] = \mathbb{E}\left[\left\|\frac{\Psi_k}{nK}\right\|_2^2\middle|\mathcal{F}_k\right] - \left\|\mathbb{E}\left[\frac{\Psi_k}{nK}\middle|\mathcal{F}_k\right]\right\|_2^2 \le 2.$$

For $i < j$, the error ξ_i is measurable with respect to $\mathcal{F}_j$. Therefore, $\mathbb{E}\langle\xi_i, \xi_j\rangle = \mathbb{E}\langle\xi_i, \mathbb{E}[\xi_j \mid \mathcal{F}_j]\rangle = 0$. Expanding the squared norm of the sum, the cross terms vanish and each of the N_{iter} diagonal terms is at most 2, which proves the bound. □

The baseline update in Equation (18) satisfies the condition on the baseline. Proposition B.8 controls the noise around the changing expected directions.

C Analysis of Multiple Continuations

Theorem 1 concerns a single continuation, whereas ARG samples K continuations for each question and accepts candidate trajectories by the rule in Section 4.1. This appendix analyzes the distribution of the accepted trajectory after multiple continuations. Appendix C.1 fixes one prefix and studies how the distance to the ideal distribution changes with the number of continuations. Appendix C.2 gives a sufficient condition under which an arbitrary acceptance rule keeps the distance no larger than that of training on the reference.

C.1 Repeated continuations from a fixed prefix

Fix a prefix length t and let H_t be the event that a trajectory begins with h_t. Since the reference is correct and has positive probability, $0 < u_t \le v_t \le 1$. For a positive integer k, draw k independent continuations from h_t. The *first-success rule* returns the first correct candidate trajectory, including an exact copy of r, and returns r if every candidate trajectory is incorrect. This rule treats an exact copy as a success. Let $Q_t^{(k)}$ be the resulting distribution of the accepted trajectory, and let $\nu_{h_t} = \pi(\cdot \mid C \cap H_t)$ be the ideal distribution restricted to trajectories that begin with h_t.

Lemma C.1 (Distribution after repeated continuations)**.** *Let* $w_k = 1 - (1 - v_t)^k$ *be the probability that at least one candidate trajectory is correct, and define* $\lambda_k = w_k/v_t$ *and* $d_t^{(k)} = d_t\lambda_k$*. Then*

$$Q_t^{(k)} = (1 - w_k)\delta_r + w_k\nu_{h_t}, \tag{22}$$

and

$$D_{\text{KL}}(Q_t^{(k)}\|\nu) = D_{\text{KL}}(\delta_r\|\nu) - d_t^{(k)}S_t + \big(1 - d_t^{(k)}\big)\log\big(1 - d_t^{(k)}\big) + d_t^{(k)}\log\lambda_k.$$

Proof. A correct trajectory $y = h_t \oplus z$ is returned at continuation j with probability $(1-v_t)^{j-1}\pi(z \mid h_t)$. Summing over $j = 1, \ldots, k$ gives $Q_t^{(k)}(y) = \lambda_k\pi(z \mid h_t)$ for every correct $y \ne r$. Since $\nu_{h_t}(y) = \pi(z \mid h_t)/v_t$ for these trajectories, this probability equals $w_k\nu_{h_t}(y)$. The same calculation gives $\lambda_k u_t$ for an exact copy, and fallback adds $(1 - v_t)^k$. Therefore, $Q_t^{(k)}(r) = (1 - w_k) + w_k u_t/v_t = 1 - d_t^{(k)}$, which proves Equation (22). For the distance, the trajectories other than r have total probability $d_t^{(k)}$ and likelihood ratio $\lambda_k p/\pi(h_t)$. Following the proof of Theorem 1 with this ratio gives the closed form. □

Compared with the closed form for one continuation, the additional term $d_t^{(k)}\log\lambda_k$ accounts for the larger probability of each trajectory other than the reference after repeated continuations. For $k = 1$, we have $\lambda_1 = 1$, and the lemma recovers Theorem 1. The mixture form in Equation (22) shows that additional continuations move probability from the reference to ν_{h_t}, which leads to the following limit.

Proposition C.2 (The limit of a fixed prefix)**.** *The distance $D_{\mathrm{KL}}(Q_t^{(k)}\|\nu)$ is nonincreasing in k and satisfies*

$$-\log \nu(H_t) \le D_{\mathrm{KL}}(Q_t^{(k)}\|\nu) \le -\log \nu(H_t) + (1-v_t)^k \log \frac{v_t}{u_t}.$$

Consequently, it converges to $-\log \nu(H_t)$ as $k \to \infty$.

Proof. For $y \in C \cap H_t$, we have $\nu(y) = \nu(H_t)\nu_{h_t}(y)$. Hence, every distribution Q supported on $C \cap H_t$ satisfies $D_{\mathrm{KL}}(Q\|\nu) = D_{\mathrm{KL}}(Q\|\nu_{h_t}) - \log \nu(H_t)$. By Lemma C.1, $Q_t^{(k)}$ is such a distribution. Therefore, it suffices to study $D_{\mathrm{KL}}(Q_t^{(k)}\|\nu_{h_t})$, and the lower bound follows from its nonnegativity. For the upper bound, convexity of KL applied to Equation (22) gives $D_{\mathrm{KL}}(Q_t^{(k)}\|\nu_{h_t}) \le (1-w_k) D_{\mathrm{KL}}(\delta_r\|\nu_{h_t})$, where $D_{\mathrm{KL}}(\delta_r\|\nu_{h_t}) = -\log \nu_{h_t}(r) = \log(v_t/u_t)$ and $1 - w_k = (1-v_t)^k$. For monotonicity, take $\ell > k$. If $w_k = 1$, then $Q_t^{(k)} = Q_t^{(\ell)} = \nu_{h_t}$. Otherwise, $a = (1-w_\ell)/(1-w_k) \in [0,1]$ satisfies $Q_t^{(\ell)} = aQ_t^{(k)} + (1-a)\nu_{h_t}$, and convexity gives

$$D_{\mathrm{KL}}(Q_t^{(\ell)}\|\nu_{h_t}) \le a D_{\mathrm{KL}}(Q_t^{(k)}\|\nu_{h_t}) \le D_{\mathrm{KL}}(Q_t^{(k)}\|\nu_{h_t}).$$

Finally, the upper bound converges to $-\log \nu(H_t)$ since $v_t \ge u_t > 0$. □

For the empty prefix, H_t contains every trajectory. Therefore, the limiting distribution is ν and the limiting distance is zero. A nonempty prefix has the same limit exactly when H_t contains all correct trajectories with positive probability. Proposition C.2 orders the numbers of continuations for each fixed prefix. However, comparing different prefixes also requires their respective u_t and v_t, which determine both the rate and the limit of the decrease.

C.2 A SUFFICIENT CONDITION FOR GENERAL ACCEPTANCE RULES

To cover the acceptance rule of ARG, which excludes exact copies, mixes several prefixes, prefers some candidate trajectories over others, and may accept several of them, we consider a general procedure. Choose $K \ge 1$ reference prefixes $h_{t_1}, \dots, h_{t_K}$ before generation. Conditional on this list, draw a continuation $Z_j \sim \pi(\cdot \mid h_{t_j})$ and form the candidate trajectory $\widetilde{Y}_j = h_{t_j} \oplus Z_j$ for each j. The continuations may depend on each other, provided that each candidate trajectory has this conditional marginal law given the entire list. A procedure returns r or one of the correct candidate trajectories. If several trajectories are accepted, we consider the distribution obtained by selecting one of them with nonnegative weights that sum to one. Let Q denote the resulting distribution of the accepted trajectory, including fallback, and use the convention $0 \log 0 = 0$.

Proposition C.3 (Multiple continuations and acceptance)**.** *Under the setup above, define $d_Q = 1 - Q(r)$ and $\Lambda = \sum_{j=1}^K \exp(-S_{t_j})$. Then*

$$D_{\mathrm{KL}}(Q\|\nu) \le D_{\mathrm{KL}}(\delta_r\|\nu) + d_Q \log \Lambda + (1-d_Q)\log(1-d_Q). \tag{23}$$

In particular, $\Lambda \le 1$ guarantees $D_{\mathrm{KL}}(Q\|\nu) \le D_{\mathrm{KL}}(\delta_r\|\nu)$. A sufficient condition is $\min_j S_{t_j} \ge \log K$.

Proof. A trajectory $y \ne r$ can be selected only if it appears among the candidate trajectories. Thus,

$$Q(y) \le \sum_{j=1}^K \Pr(\widetilde{Y}_j = y) = \sum_{j=1}^K \mathbf{1}\{y \text{ begins with } h_{t_j}\} \frac{\pi(y)}{\pi(h_{t_j})} \le \frac{\pi(y)\Lambda}{\pi(r)},$$

where the last inequality uses $\pi(r)/\pi(h_{t_j}) = \exp(-S_{t_j})$ from Lemma A.4. Every $y \ne r$ with $Q(y) > 0$ is correct and has $\pi(y) > 0$. Hence, $Q(y)/\nu(y) \le p\Lambda/\pi(r)$. Since these trajectories have total probability d_Q and the reference has probability $1 - d_Q$ with $Q(r)/\nu(r) = (1-d_Q)p/\pi(r)$,

$$D_{\mathrm{KL}}(Q\|\nu) \le \log \frac{p}{\pi(r)} + (1-d_Q)\log(1-d_Q) + d_Q \log \Lambda.$$

Lemma A.2 gives $\log(p/\pi(r)) = D_{\mathrm{KL}}(\delta_r\|\nu)$, which proves Equation (23). Since $(1-d_Q)\log(1-d_Q) \le 0$, and $d_Q \log \Lambda \le 0$ when $\Lambda \le 1$, the condition $\Lambda \le 1$ gives the comparison with training on the reference. Finally, $\min_j S_{t_j} \ge \log K$ gives $\Lambda \le K \exp(-\log K) = 1$. □

The argument permits arbitrary selection among correct candidate trajectories, including the exclusion of exact copies and the priority rule of ARG. For one prefix used K times, $\Lambda = K\exp(-S_t)$, and the bound becomes $D_{\mathrm{KL}}(\delta_r\|\nu) - d_Q(S_t - \log K) + (1-d_Q)\log(1-d_Q)$. The term $\log K$ accounts for the possible concentration from selecting among several candidate trajectories. For $K = 1$ and the acceptance rule in Equation (1), we have $d_Q = d_t$, and the bound becomes the equality in Theorem 1.

In ARG, the prefixes come from guidance levels sampled before generation. The following corollary extends the comparison to this case.

Corollary C.4 (Guidance levels sampled before generation)**.** *Under the setup of Proposition C.3, suppose that the K guidance levels are sampled before generation from any distribution over $\mathcal{A}^K$, and that $\min_{\alpha\in\mathcal{A}} S_{t_x(\alpha)} \geq \log K$. Then the distribution Q of the accepted trajectory, averaged over the sampled levels, satisfies $D_{\mathrm{KL}}(Q\|\nu) \leq D_{\mathrm{KL}}(\delta_r\|\nu)$.*

Proof. Conditional on each list of levels, every prefix satisfies $S_{t_j} \geq \log K$. Therefore, Proposition C.3 bounds the conditional distance by $D_{\mathrm{KL}}(\delta_r\|\nu)$. Since Q is the average of these conditional distributions, convexity of KL in its first argument gives the same bound for Q. □

The condition in Corollary C.4 uses the actual prefix lengths after the boundary adjustment and is available from the same reference surprisals used for prefix selection.

D Analysis of Learning from Inserted Trajectories

The preceding appendices compare distributions of accepted trajectories at a fixed model. This appendix examines how the inserted trajectories contribute to the policy update in Section 4.3. Appendix D.1 derives the gradient of inserted trajectories in Equation (13). Appendix D.2 derives the expected group update under binary rewards for one inserted trajectory, and Appendix D.3 extends it to multiple inserted trajectories. Finally, Appendix D.4 relates learning the complete trajectory and learning only the continuation to their respective success events.

D.1 The gradient of inserted trajectories

Condition on the realized batch, its masks, and the recomputed group advantages. Let $\mathcal{I}$ index the inserted trajectories, and let $T > 0$ be the total number of active response tokens in the batch loss. For token j of an inserted trajectory y_i, let $\ell_{i,j}(\theta)$ be its log probability given the original question and all preceding tokens of the trajectory. Inserted trajectories use the clipped objective of Equation (29) in Appendix E.3 with the stopped current probability in the ratio denominator,

$$\mathcal{L}_{\mathrm{inserted}}(\theta) = -\frac{1}{T}\sum_{i\in\mathcal{I}}\sum_{j=1}^{|y_i|}\min\{\widetilde{\omega}_{i,j}(\theta)A_i, \mathrm{clip}(\widetilde{\omega}_{i,j}(\theta), 0.8, 1.3)A_i\},$$

$$\widetilde{\omega}_{i,j}(\theta) = \exp\big(\ell_{i,j}(\theta) - \mathrm{stopgrad}(\ell_{i,j}(\theta))\big),$$

where $\mathrm{stopgrad}$ preserves a value but assigns zero derivative to it.

Lemma D.1 (Gradient of inserted trajectories)**.** *Suppose that the advantages A_i and the token count T are held fixed during differentiation and that every token of an inserted trajectory is active. Then the gradient of $\mathcal{L}_{\mathrm{inserted}}$ is given by Equation* (13)*.*

Proof. By construction, $\widetilde{\omega}_{i,j} = 1$ and $\nabla_\theta\widetilde{\omega}_{i,j} = \nabla_\theta\ell_{i,j}$. Since the clipping interval contains one in its interior, both branches of the clipped loss have the same derivative at this ratio. Therefore, for any fixed real advantage A_i, the loss of token j has derivative $-A_i\nabla_\theta\ell_{i,j}/T$. Summing over all positions, including any supplied prefix and the termination token, gives $-A_i\nabla_\theta\log\pi_\theta(y_i \mid x_i)/T$ for each inserted trajectory. Summing over $i \in \mathcal{I}$ proves Equation (13). □

Lemma D.1 identifies the gradient under the prescribed rule of automatic differentiation. For $A_i > 0$, minimizing $\mathcal{L}_{\mathrm{inserted}}$ gives a positively weighted supervised fine-tuning (SFT) contribution. The advantage specifies its learning weight, while candidate generation and acceptance determine the distribution of the trajectory.

D.2 THE GROUP UPDATE WITH ONE INSERTED TRAJECTORY

We now use binary correctness rewards to separate the contributions of correct and incorrect responses. This simplification ignores the format reward, which leaves the advantages unchanged when all responses in a group receive the same format reward. The calculation is at the current model and precedes division by the batch token count. Fix a question x with set of correct trajectories C, and suppress the conditioning on x unless shown explicitly. Assume a finite trajectory space, a differentiable policy with positive trajectory probabilities in a neighborhood of θ_{old}, and $0 < p = \pi_{\theta_{\text{old}}}(C) < 1$. Draw $G \geq 2$ independent responses from $\pi_{\theta_{\text{old}}}$ and condition on all of them failing verification. Then replace one uniformly chosen response by $Y \sim Q$, where Q is supported on C, and where Y and the replaced position are independent of the original failures. Both Q and the collected responses are fixed during differentiation. Set

$$p_\theta = \pi_\theta(C), \qquad \nu_\theta = \pi_\theta(\cdot \mid C), \qquad \mu_\theta = \pi_\theta(\cdot \mid C^c),$$

and write $\nu = \nu_{\theta_{\text{old}}}$ for the ideal distribution at the current model.

At the current model, the ratios of the original responses equal one and clipping is inactive. The inserted trajectory follows Lemma D.1, and all response tokens are included. We consider the group ascent direction, which is the sum of the reward-dependent gradients before division by the batch token count. With one inserted trajectory, the reward mean is $1/G$ and the sample standard deviation is $s_R = 1/\sqrt{G}$. Therefore, the inserted trajectory has advantage $(G-1)/[G(s_R+\delta)]$, and each failure has advantage $-1/[G(s_R+\delta)]$, where $\delta > 0$ is the stabilizer in Equation (28). Define the positive coefficient

$$c_G = \frac{G-1}{G(1/\sqrt{G}+\delta)}.$$

Let $F_1, \ldots, F_{G-1}$ denote the original failures that remain after replacement. The realized group direction is

$$g = c_G \left[\nabla_\theta \log \pi_\theta(Y \mid x) - \frac{1}{G-1} \sum_{j=1}^{G-1} \nabla_\theta \log \pi_\theta(F_j \mid x) \right]_{\theta=\theta_{\text{old}}}. \tag{24}$$

Proposition D.2 (Inserted trajectories and the group update)**.** *Under the setup above, let $\bar{g}_Q = \mathbb{E}[g]$ be the expectation over the original failures, the replaced position, and the inserted trajectory. Then*

$$\bar{g}_Q = c_G \, \nabla_\theta \left[\log \frac{p_\theta}{1-p_\theta} - D_{\text{KL}}(Q \| \nu_\theta) \right] \Big|_{\theta=\theta_{\text{old}}}. \tag{25}$$

In particular, for $Q = \nu$,

$$\bar{g}_\nu = \frac{c_G}{p(1-p)} \, \nabla_\theta p_\theta |_{\theta=\theta_{\text{old}}}. \tag{26}$$

Proof. Conditioning independent responses on all of them belonging to C^c gives independent draws from $\mu_{\theta_{\text{old}}}$, and uniform replacement preserves this distribution for the remaining failures. Therefore, with $F \sim \mu_{\theta_{\text{old}}}$,

$$\frac{\bar{g}_Q}{c_G} = \left[\mathbb{E}_Q \nabla_\theta \log \pi_\theta(Y) - \mathbb{E}_{\mu_{\theta_{\text{old}}}} \nabla_\theta \log \pi_\theta(F) \right] \Big|_{\theta=\theta_{\text{old}}}.$$

Applying the argument of Lemma A.1 to the event C^c gives $\mathbb{E}_{\mu_{\theta_{\text{old}}}} \nabla_\theta \log \pi_\theta(F) = \nabla_\theta \log(1-p_\theta)$ at θ_{old}. For fixed Q, expanding $\log \nu_\theta(y) = \log \pi_\theta(y) - \log p_\theta$ gives

$$\nabla_\theta D_{\text{KL}}(Q \| \nu_\theta) = -\mathbb{E}_Q \nabla_\theta \log \pi_\theta(Y) + \nabla_\theta \log p_\theta.$$

Substituting both expressions proves Equation (25). When $Q = \nu$, the gradient of the KL term vanishes at θ_{old} by Lemma A.1. Since $\nabla_\theta \log[p_\theta/(1-p_\theta)] = \nabla_\theta p_\theta / [p_\theta(1-p_\theta)]$, this gives Equation (26). □

The ratio $p_\theta/(1-p_\theta)$ is the odds of passing verification. The first term in Equation (25) favors success over failure. The second term records which correct trajectories the distribution Q favors.

For training on the reference, $D_{\mathrm{KL}}(\delta_r \| \nu_\theta) = -\log \nu_\theta(r)$, and this term favors the particular reference within the set of correct trajectories. In contrast, Equation (26) shows that inserting trajectories from the ideal distribution gives an expected group update proportional to the gradient of the success probability under the original question.

For reference fallback, $Y = r$ makes the positive term in Equation (24) a weighted SFT gradient. For an accepted candidate trajectory, the positive term learns its complete sequence, including the supplied prefix. In both cases, the original failures supply the negative terms in the same group update. This combination of learning from correct trajectories with feedback on the model's own failures connects our update to prior work on integrating supervised and reinforcement fine-tuning (Yan et al., 2026, Fu et al., 2026).

D.3 Multiple inserted trajectories

We keep the preceding setup but insert a random number N_+ of correct trajectories with $1 \le N_+ < G$, where a single reference is inserted when no candidate trajectory is accepted. The inserted trajectories may depend on each other. However, they, their number N_+, and the replaced positions, which are uniformly chosen and distinct, are independent of the original failures. Conditional on $N_+ = m$, let $\widehat{Q}$ and $\widehat{M}$ be the empirical distributions of the inserted trajectories and of the remaining failures. Substituting the advantages $(G-m)/[G(s_m+\delta)]$ and $-m/[G(s_m+\delta)]$ of the inserted trajectories and the remaining failures gives the group direction

$$g = \kappa_m \left[\mathbb{E}_{\widehat{Q}} \nabla_\theta \log \pi_\theta(Y) - \mathbb{E}_{\widehat{M}} \nabla_\theta \log \pi_\theta(F) \right]_{\theta=\theta_{\mathrm{old}}},$$

where

$$\kappa_m = \frac{m(G-m)}{G(s_m+\delta)}, \qquad s_m = \sqrt{\frac{m(G-m)}{G(G-1)}}.$$

Define the average coefficient and the advantage-weighted distribution of inserted trajectories by

$$\bar{\kappa} = \mathbb{E}[\kappa_{N_+}], \qquad Q_+(y) = \frac{\mathbb{E}[\kappa_{N_+} \widehat{Q}(y)]}{\bar{\kappa}}.$$

Since $1 \le N_+ < G$, we have $\bar{\kappa} > 0$, and Q_+ is a probability distribution on C.

Proposition D.3 (Multiple inserted trajectories)**.** *Under the setup above,*

$$\mathbb{E}[g] = \bar{\kappa}\, \nabla_\theta \left[\log \frac{p_\theta}{1-p_\theta} - D_{\mathrm{KL}}(Q_+ \| \nu_\theta) \right] \Bigg|_{\theta=\theta_{\mathrm{old}}}.$$

For one inserted trajectory, $\kappa_1 = c_G$ and $Q_+ = Q$, which recovers Proposition D.2.

Proof. Conditional on the inserted trajectories and their number, uniform replacement preserves the failure distribution $\mu_{\theta_{\mathrm{old}}}$ for the remaining failures. Therefore, $\mathbb{E}[\kappa_{N_+} \widehat{M}(y)] = \bar{\kappa} \mu_{\theta_{\mathrm{old}}}(y)$, and taking expectations in the group direction gives

$$\mathbb{E}[g] = \bar{\kappa} \left[\mathbb{E}_{Q_+} \nabla_\theta \log \pi_\theta(Y) - \mathbb{E}_{\mu_{\theta_{\mathrm{old}}}} \nabla_\theta \log \pi_\theta(F) \right]_{\theta=\theta_{\mathrm{old}}}.$$

The proof of Proposition D.2 then applies with c_G replaced by $\bar{\kappa}$ and Q replaced by Q_+. Finally, $s_1 = 1/\sqrt{G}$ gives $\kappa_1 = c_G$. □

The distribution Q_+ weights each realized set of inserted trajectories by its total positive advantage κ_{N_+}. Therefore, it accounts for both acceptance and the resulting composition of the group.

D.4 Learning the prefix and the continuation

The likelihood in Equation (13) includes every token of the complete accepted trajectory under the original question. For a candidate trajectory from a nonempty prefix, the chain rule gives

$$\log \pi_\theta(h_t \oplus z \mid x) = \log \pi_\theta(h_t \mid x) + \log \pi_\theta(z \mid x, h_t). \tag{27}$$

Hence, learning covers both producing the supplied prefix and completing the trajectory, and reference fallback trains the complete reference through Equation (13). We now connect the two terms in Equation (27) to different success events. Fix a prefix length t, let H_t be the event that a trajectory begins with h_t, and let $\nu_{h_t} = \pi_{\theta_{\text{old}}}(\cdot \mid C \cap H_t)$ as in Appendix C.1. This distribution is well defined since the correct reference has positive probability. Let $Q_{Z,t}(z) = \nu_{h_t}(h_t \oplus z)$ be the distribution of its continuations. Here $\pi_\theta(C \mid H_t)$ is the probability that a continuation from h_t gives a correct candidate trajectory, which equals v_t at θ_{old}.

Proposition D.4 (Learning the prefix and the continuation)**.** *Under the assumptions of Lemma A.1, at $\theta = \theta_{\text{old}}$,*

$$\mathbb{E}_{Y\sim\nu_{h_t}} \nabla_\theta \log \pi_\theta(Y) = \nabla_\theta \log \pi_\theta(C \cap H_t), \qquad \mathbb{E}_{Z\sim Q_{Z,t}} \nabla_\theta \log \pi_\theta(Z \mid h_t) = \nabla_\theta \log \pi_\theta(C \mid H_t).$$

Moreover, the distribution $Q_t^{(k)}$ of the accepted trajectory after k continuations in Lemma C.1 satisfies

$$\mathbb{E}_{Y\sim Q_t^{(k)}} \nabla_\theta \log \pi_\theta(Y) = (1 - w_k)\nabla_\theta \log \pi_\theta(r) + w_k \nabla_\theta \log \pi_\theta(C \cap H_t).$$

Proof. The first identity follows from Lemma A.1 with C replaced by $C \cap H_t$. For the second, Equation (27) gives $\nabla_\theta \log \pi_\theta(h_t \oplus Z) = \nabla_\theta \log \pi_\theta(h_t) + \nabla_\theta \log \pi_\theta(Z \mid h_t)$, and $\pi_\theta(C \cap H_t) = \pi_\theta(h_t)\pi_\theta(C \mid H_t)$. Subtracting $\nabla_\theta \log \pi_\theta(h_t)$ from both sides of the first identity gives the second. Finally, the mixture form in Equation (22), whose weights are fixed at θ_{old}, and the first identity give the last statement. □

Under ν_{h_t}, the expected gradient of the complete trajectory addresses the joint event of producing the prefix and passing verification. In contrast, learning only the continuation addresses verification given an externally supplied prefix. The two coincide for the empty prefix. The joint event covers the correct trajectories that begin with this prefix, while the overall success event C also includes correct trajectories that begin differently. Finally, the mixture identity in Proposition D.4 shows how fallback and successful continuations contribute to the expected gradient of the accepted trajectory before any weighting.

E Implementation and Evaluation Details

We first specify the training procedure, reference boundaries, and policy loss. We then give the training configurations, question prompt, and evaluation protocol. Together, these details connect the single-continuation analysis in Theorem 1 to the complete training procedure.

E.1 Training procedure

Training uses 11,068 screened DeepMath examples with matched offline DeepSeek-R1 reference solutions (Guo et al., 2025). We start from DeepMath-103K, keep problems with difficulty at least 5.0, and drop problems whose gold answer is not a single short expression, is a multiple-choice letter, or cannot be verified by our checker when written inside `\boxed{}`. From the remaining problems, we randomly sample 16,000 in proportion to difficulty and let the base Qwen3-4B attempt each one eight times under the training prompt and sampling settings. Problems solved in all eight attempts are removed. We also drop the few problems whose reference solution does not end in the gold answer. Table 3 gives the ARG configuration. Algorithm 1 separates collection under a fixed policy from updates of the selector and policy. All continuations, including failures and candidate trajectories not selected for the policy update, provide selector feedback.

In the algorithm, $\mathcal{U}_x$ contains K distinct batch-wide continuation indices for question x. Candidate quantities are computed for every $j \in \mathcal{U}_x$. The operation Accept returns the candidate indices selected by the rule in Section 4.3.

E.2 Prefix boundaries

The level $\alpha = 1$ always uses the empty prefix. For $\alpha < 1$, we take the latest prefix length whose reference surprisal is at least αS_0 and, if this length is positive, move it to the first valid sentence or formula boundary at or after it. If no such boundary exists, we use the latest earlier valid boundary,

Algorithm 1 One ARG training update with GRPO

Input Batch $\mathcal{B}$ with references, policy θ, selector ϕ, baseline b.

$\theta_{\text{old}} \leftarrow \theta, \mathcal{U} \leftarrow \varnothing$
$\mathbf{y}_x = (y_{x,i})_{i=1}^{G}, y_{x,i} \overset{\text{iid}}{\sim} \pi_{\theta_{\text{old}}}(\cdot \mid x)$ for $x \in \mathcal{B}$
for $x \in \mathcal{B}$ with $\sum_{i=1}^{G} R_{\text{acc}}(x, y_{x,i}) = 0$ **do**
    $S_t \leftarrow -\log \pi_{\theta_{\text{old}}}(a_t \mid x, h_t)$ for $0 \le t < |r|$, $\mathcal{J}_x \leftarrow \varnothing$
    if $0 < S_0 < \infty$ **then**
        $\alpha_j \overset{\text{iid}}{\sim} \rho_\phi, t_j \leftarrow t_x(\alpha_j)$
        $Z_j \sim \pi_{\theta_{\text{old}}}(\cdot \mid x, h_{t_j}), \widetilde{Y}_j \leftarrow h_{t_j} \oplus Z_j$
        $W_j \leftarrow \mathbf{1}\{\widetilde{Y}_j \in C_x,\ \widetilde{Y}_j \neq r\}\, S_{t_j}/S_0$
        $\mathcal{U} \leftarrow \mathcal{U} \cup \mathcal{U}_x, \mathcal{J}_x \leftarrow \text{Accept}((\widetilde{Y}_j, t_j)_{j \in \mathcal{U}_x}, r)$
    end if
    Replace rows in $\mathbf{y}_x$ by $(\widetilde{Y}_j)_{j \in \mathcal{J}_x}$, or by r if $\mathcal{J}_x = \varnothing$.
end for
Recompute final rewards R_i, then $A_i \leftarrow (R_i - \bar{R})/(s_R + \delta)$.
if $\mathcal{U} \neq \varnothing$ **then**
    $\phi \leftarrow \phi + \eta_{\text{sel}} \sum_{j \in \mathcal{U}} (W_j - b) \nabla_\phi \log \rho_\phi(\alpha_j)$
    $b \leftarrow (1 - \beta) b + \frac{\beta}{|\mathcal{U}|} \sum_{j \in \mathcal{U}} W_j$
end if
Update θ using Equation (13) and the remaining-response loss in Appendix E.3.

Setting	ARG configuration
Questions per update	32
Ordinary responses per question G	8
Continuations per all-failure group K	8
Guidance levels $\mathcal{A}$	$\{1, 1/2, 1/4\}$
Total acceptance cap in Algorithm 1	4
Maximum accepted trajectories with empty prefixes	4
Maximum accepted trajectories with nonempty prefixes	1
Selector exploration ε_{sel}	0.2
Selector step size η_{sel}	0.3
Baseline update rate β	0.1
Training budget	200 updates
Learning rate	10^{-6}
Warmup updates	10
Weight decay	0.01
Gradient norm clipping	1
Policy optimization epochs per batch	1
Clipping ratio interval	$[0.8, 1.3]$
Policy loss averaging	Batch token mean
Total reward	Accuracy plus 0.2 format
Advantage stabilizer δ	10^{-6}
Training temperature	1
Training top-p and top-k	1 and no truncation
Question length limit	3072 tokens
Trajectory length budget including the prefix	8192 tokens
Additional KL loss	None
Entropy loss coefficient	0

Table 3. ARG configuration for both model sizes.

or the empty prefix if there is none. A valid boundary falls on a token boundary, closes all mathematical delimiters, and leaves a nontrivial suffix containing the final boxed answer. The selector feedback uses the reference surprisal S_t at this adjusted prefix length, and the prefix counts toward the maximum trajectory length.

E.3 Policy loss

Final rewards R_i combine binary verifier accuracy with 0.2 times a binary format reward indicating an extracted boxed answer. Thus, a correct response receives 1.2, while an incorrect response receives 0.2 or zero according to its format. We recompute advantages within each final group as

$$A_i = \frac{R_i - \bar{R}}{s_R + \delta}, \tag{28}$$

where $\bar{R}$ and s_R are the mean and sample standard deviation of the group's final rewards, and $\delta = 10^{-6}$ ensures numerical stability. All-equal rewards give exactly zero advantages.

For an ordinary response y_i, let $y_{i,j}$ be token j and define

$$\omega_{i,j}(\theta) = \frac{\pi_\theta(y_{i,j} \mid x_i, y_{i,<j})}{\pi_{\theta_{\text{old}}}(y_{i,j} \mid x_i, y_{i,<j})}.$$

Let $\mathcal{O}$ index the original responses that remain in their groups, and let T be the total number of active response tokens in the final batch. Their loss contribution uses the asymmetric clipping interval $[0.8, 1.3]$,

$$\mathcal{L}_{\text{original}}(\theta) = -\frac{1}{T} \sum_{i \in \mathcal{O}} \sum_{j=1}^{|y_i|} \min\left\{\omega_{i,j}(\theta) A_i, \text{clip}(\omega_{i,j}(\theta), 0.8, 1.3) A_i\right\}. \tag{29}$$

Inserted trajectories use the stopped current probability in the ratio denominator, as derived in Appendix D. With unit token weights over each complete inserted trajectory, this gives Equation (13). Loss-active tokens include supplied prefixes and termination tokens when present, and both contributions share the same batch token denominator.

E.4 Training configurations and question prompt

The GRPO, SAGE, BREAD, LUFFY, and ARG runs share the optimizer and student rollout settings in Table 4. The resampling and weighting ablations use the same settings. SFT learns from the offline references. For OPSD, we follow Zhao et al. (2026) in using one student response per question, a training generation limit of 1024 tokens, and temperature 1.1. Table 4 specifies the remaining settings used in our experiments. All listed runs use 32 questions per update and one optimization epoch per collected batch. Their data and parameter-initialization seeds are zero, and the learning rate is constant after warmup.

Setting	**GRPO-based runs**	**SFT**	**OPSD**
Learning rate	10^{-6}	10^{-6}	5×10^{-7}
Warmup updates	10	10	0
Weight decay	0.01	0.01	0
Gradient norm clipping	1	1	0.1
Responses per question	8	Reference	1
Response length limit	8192	8192	1024
Generation temperature	1	N/A	1.1
Generation top-p	1	N/A	0.95
Generation top-k	Unrestricted	N/A	20

Table 4. Recorded configurations for both model sizes. GRPO-based runs here comprise GRPO, SAGE, BREAD, LUFFY, ARG, resampling, and the weighting ablation. The reference-insertion arm is specified by its replacement rule in Section 5.3.

The ordinary student question prompt and the evaluation prompt contain a single user message with the question, a blank line, and the instruction below. They use no system message or few-shot example. Auxiliary teacher or reference conditioning is specific to each training method. The

Qwen3 chat template is applied with thinking mode disabled, which appends an empty thinking block before response generation. The rendered question prompt is shown schematically below.

```
<|im_start|>user
<question>

Please reason step by step, and put your final answer within \boxed{}.
<|im_end|>
<|im_start|>assistant
<think>

</think>
```

Here `<question>` denotes the question text, and the line break before `<|im_end|>` is added for display. For ARG generation, the selected reference prefix supplies the beginning of the candidate trajectory. Evaluation supplies no reference prefix.

E.5 Checkpoint and evaluation records

Table 5 lists the reported checkpoints. GRPO, SAGE, BREAD, LUFFY, ARG, resampling, and the weighting ablations use the fixed budget of 200 updates. Reference insertion also reports its step-200 checkpoint. For SFT and OPSD, we select the reported checkpoints by comparing mean $\mathrm{pass@12}$ and $\mathrm{avg@12}$ on the five evaluation benchmarks.

Method	**4B checkpoint**	**8B checkpoint**
Base	0	0
SFT	25	25
OPSD	60	100
GRPO, SAGE, BREAD, LUFFY, ARG	200	200
Resampling	200	N/A
w/o weighting	200	200
Reference insertion	200	N/A

Table 5. Reported checkpoint steps for both model sizes. Base uses the released parameters. SFT and OPSD use benchmark-selected checkpoints, and the remaining trained entries report step 200.

All 20 reported model and method entries use the same 150 evaluation questions, comprising 30 questions from each of the five benchmarks. The benchmark files, answer extractor, and verifier implementation match across entries. Table 6 gives the common decoding settings. Each question contributes 12 scored responses.

Setting	**Evaluation value**
Temperature	1
Top-p	0.95
Top-k	20
Maximum new tokens	16,384
Maximum context length	32,768
Scored responses per question	12
Random seed	1234
Parameter precision	bfloat16
Generation engine	vLLM 0.11.0
Thinking mode	Disabled

Table 6. Evaluation configuration shared by all reported entries.

The scorer extracts the final balanced boxed answer. A response without a boxed answer is incorrect. Extracted answers are checked with Math-Verify 0.9.0 and a normalized string comparison. For each question, $\mathrm{pass@12}$ records whether at least one of its 12 responses is correct, while $\mathrm{avg@12}$ is the fraction of those responses that are correct. Both metrics are averaged over questions and reported

as percentages. Since the benchmarks have equal sizes, the mean over benchmarks also equals the mean over all 150 questions.

F Additional Main Results

Table 7 reports avg@12 for the same checkpoints and the same 12 scored responses per question as Table 1, following the scoring protocol in Appendix E.5.

Model	Method	AIME24	AIME25	AIME26	HMMT_Feb	HMMT_Nov	Avg.
Qwen3-4B	Base	22.78	22.22	13.33	11.94	9.17	15.89
	SFT	19.44	20.56	16.94	9.72	9.17	15.17
	GRPO	47.22	42.22	<u>46.67</u>	<u>26.94</u>	29.17	38.44
	OPSD	22.22	18.89	20.83	14.44	13.06	17.89
	SAGE	44.17	39.17	40.83	25.00	28.06	35.44
	BREAD	**56.39**	**47.78**	**47.78**	**29.17**	**35.56**	**43.33**
	LUFFY	50.28	43.61	41.67	25.00	<u>33.06</u>	38.72
	ARG (ours)	<u>52.50</u>	<u>46.67</u>	46.39	<u>26.94</u>	<u>33.06</u>	<u>41.11</u>
Qwen3-8B	Base	25.28	21.67	18.61	9.72	9.44	16.94
	SFT	33.06	27.78	28.06	12.22	17.50	23.72
	GRPO	48.89	35.83	36.67	22.22	30.56	34.83
	OPSD	33.89	29.44	26.67	14.17	19.44	24.72
	SAGE	54.72	40.00	<u>44.17</u>	<u>27.22</u>	35.28	40.28
	BREAD	**60.56**	<u>42.50</u>	43.06	**27.78**	**39.44**	**42.67**
	LUFFY	50.83	42.22	43.89	25.28	31.94	38.83
	ARG (ours)	<u>55.28</u>	**46.94**	**46.39**	<u>27.22</u>	<u>35.56</u>	<u>42.28</u>

Table 7. Main experimental results in avg@12 percentages. Avg. is the mean over the five benchmarks. Within each model, **bold** and <u>underline</u> mark the best and second-best values in each column.

ARG improves aggregate avg@12 over GRPO by 2.67 and 7.45 percentage points on Qwen3-4B and Qwen3-8B, respectively, and achieves the second-highest aggregate avg@12 on both models. The difference between the two metrics is consistent with where ARG intervenes. Specifically, ARG modifies the GRPO update only for all-failure groups, where all G responses to a training question are incorrect, and leaves the update for all other questions unchanged. Therefore, the additional learning signal concentrates on questions that the model currently fails to solve. When a question changes from unsolved to solved by a few of the 12 responses, pass@12 counts the entire question, whereas avg@12 counts only the fraction of correct responses. As a result, the effect of ARG appears most clearly in pass@12, which measures the fraction of questions that the model solves within the sampling budget.

G Additional Ablation Results

We provide benchmark-level results, then examine how the ablations change the accepted trajectories and the prefix selector during training. Reference insertion and resampling use Qwen3-4B, while the surprisal-weighting comparison covers both model sizes.

G.1 Evaluation results

Tables 8 and 9 report pass@12 and avg@12 separately for all five benchmarks. All entries use the checkpoint after 200 training steps and the same evaluation protocol as the main results. Each question contributes 12 scored responses, as specified in Appendix E.5.

For the ablations in Table 9, ARG has higher average sampled accuracy on every benchmark.

G.2 Accepted candidate trajectories during training

To measure how often continuations supply the trajectories used for training, we count all-failure groups that accept at least one candidate trajectory. Reference fallback is excluded from this count.

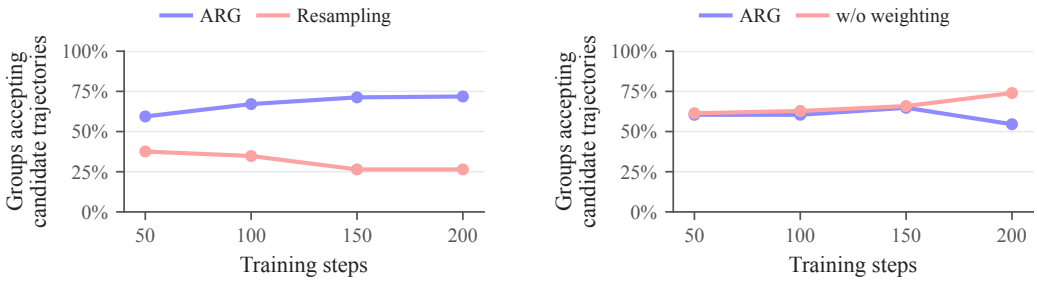
Model	Method	AIME24	AIME25	AIME26	HMMT_Feb	HMMT_Nov	Avg.
Qwen3-4B	Reference insertion	70.00	60.00	63.33	43.33	46.67	56.67
	Resampling	76.67	63.33	63.33	43.33	53.33	60.00
	w/o weighting	76.67	70.00	66.67	50.00	53.33	63.33
	ARG	80.00	73.33	73.33	43.33	66.67	67.33
Qwen3-8B	w/o weighting	83.33	70.00	66.67	50.00	50.00	64.00
	ARG	83.33	73.33	73.33	56.67	63.33	70.00

Table 8. Benchmark-level ablation results in pass@12 percentages. Avg. is the mean over benchmarks. Here w/o weighting denotes ARG without surprisal weighting.

Model	Method	AIME24	AIME25	AIME26	HMMT_Feb	HMMT_Nov	Avg.
Qwen3-4B	Reference insertion	36.67	34.17	34.17	17.78	19.44	28.44
	Resampling	44.44	35.56	40.00	21.67	28.33	34.00
	w/o weighting	43.61	40.83	39.17	21.39	28.33	34.67
	ARG	52.50	46.67	46.39	26.94	33.06	41.11
Qwen3-8B	w/o weighting	45.28	33.89	38.33	18.33	23.61	31.89
	ARG	55.28	46.94	46.39	27.22	35.56	42.28

Table 9. Benchmark-level ablation results in avg@12 percentages, computed from the same scored responses as Table 8. Avg. is the mean over benchmarks.

Figure 2 reports the proportion in consecutive windows of 50 training steps, using the total group counts within each window. Each curve describes the all-failure questions encountered in its own training run.



(a) Qwen3-4B resampling

(b) Qwen3-8B surprisal weighting

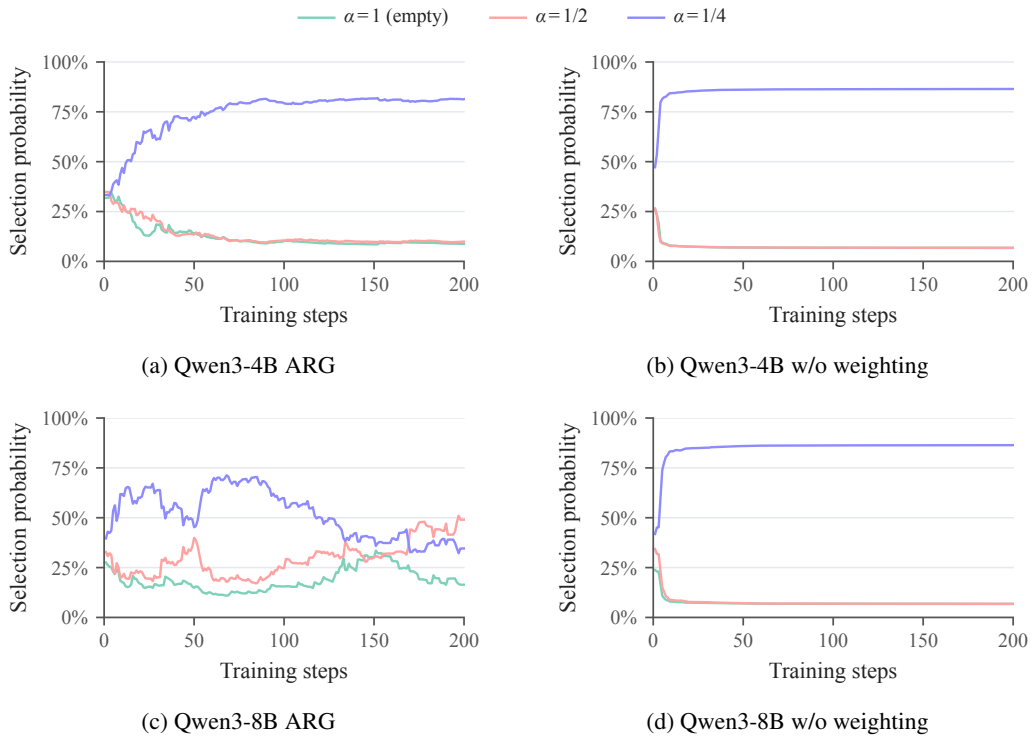
Figure 2. Proportion of all-failure groups that accept candidate trajectories. A point at step s aggregates steps $s-49$ through s. Reference fallback is excluded. The curves come from their respective training runs.

On Qwen3-4B, the proportion increases from 59.4% to 71.8% for ARG across the first and last windows, while resampling decreases from 37.6% to 26.4%. Over the full runs, ARG accepts candidate trajectories in 318 of 482 all-failure groups and uses reference fallback in the remaining 164. Resampling accepts candidate trajectories in 145 of 446 all-failure groups, leaving the remaining 301 unchanged. Resampling retains all correct candidate trajectories. Both methods use eight continuations per all-failure group, and ARG also uses reference guidance and fallback.

The weighting ablation gives a different comparison. On Qwen3-8B, it accepts candidate trajectories in 64.8% of all-failure groups, compared with 60.2% for ARG, despite its lower evaluation scores. On Qwen3-4B, these proportions are similar at 65.9% and 66.0%. Thus, the proportion of groups that accept candidate trajectories alone does not explain the evaluation results.

G.3 Surprisal weighting and prefix selection

The weighting ablation uses the successful replacement indicator as feedback, while retaining the same guidance levels, prefix construction, selector hyperparameters, and acceptance rule. Figure 3 compares the resulting selection probabilities. On Qwen3-4B, both selectors favor $\alpha = 1/4$ for most of training. On Qwen3-8B, the unweighted selector favors $\alpha = 1/4$ throughout training, whereas the probabilities under ARG change more over time. The final probabilities under ARG are 49.0% for $\alpha = 1/2$ and 34.6% for $\alpha = 1/4$, compared with 6.8% and 86.4% without weighting.



(a) Qwen3-4B ARG

(b) Qwen3-4B w/o weighting

(c) Qwen3-8B ARG

(d) Qwen3-8B w/o weighting

Figure 3. Selection probabilities of the prefix selector with and without surprisal weighting. Curves show the state after each training step without smoothing. Steps with no selector update retain the previous probabilities. The level $\alpha = 1$ uses an empty prefix, while smaller levels leave less of the reference surprisal to the continuation.

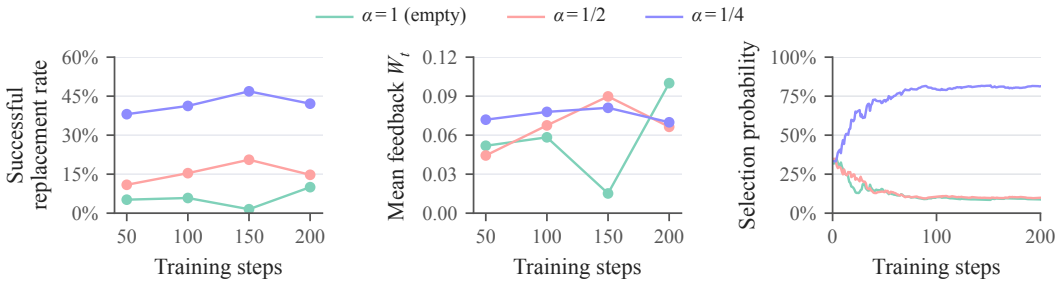
H Training Dynamics

We supplement Section 5.4 with the corresponding analysis on Qwen3-4B, the sources of accepted trajectories for both models, and general training metrics. Comparisons with the ablated methods appear in Appendix G.

H.1 Successful replacement, feedback, and prefix selection

Figure 4 reports successful replacement rates, mean selector feedback, and prefix selection on Qwen3-4B. As in the Qwen3-8B plots, each point in the first two panels uses all continuations sampled at that level within a window of 50 training steps. Successful replacement requires a correct candidate trajectory different from the reference. We compute W_t with the actual S_t/S_0 after prefix-boundary adjustment, including zero feedback from failures and exact reference copies. The third panel shows the selection probabilities after each training step without smoothing.

The level $\alpha = 1/4$ has the highest successful replacement rate in each window. Weighting reduces its advantage over $\alpha = 1/2$, and their mean-feedback ordering varies across windows. Over the



(a) Successful replacement (b) Selector feedback (c) Prefix selection

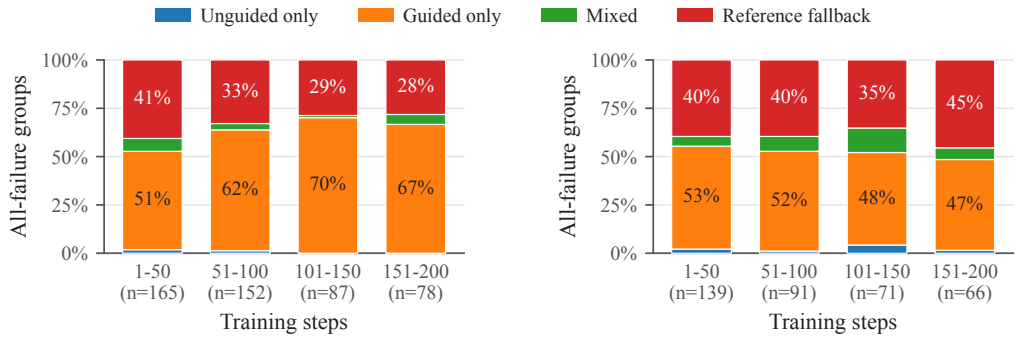
Figure 4. Training dynamics of a Qwen3-4B ARG run. In panels (a) and (b), a point at step s averages the continuations at each level from steps $s - 49$ through s. Panel (c) shows the selection probabilities after each training step without smoothing. The colors and corresponding axis ranges match Figure 1.

full run, the mean feedback is 0.0754 for $\alpha = 1/4$ and 0.0588 for $\alpha = 1/2$. These observations accompany the persistent preference for $\alpha = 1/4$ shown in Figure 4(c).

The windowed statistics summarize continuations collected as the policy, the all-failure questions, and the selection probabilities change. Since the selector updates sequentially from these continuations, its state reflects their order and the preceding updates. Steps with no selector update retain the previous probabilities. This gives a view of the feedback observed during training alongside the resulting selection history.

H.2 Sources of accepted trajectories

Figure 5 records which trajectories enter policy learning. We call a candidate trajectory guided if it comes from a nonempty prefix and unguided if it comes from the empty prefix. Each all-failure group belongs to one of four categories, according to whether it accepts only guided candidate trajectories, only unguided ones, both types, or the reference as fallback. Thus, the figure describes groups, with each group counted once even when it accepts multiple trajectories.



(a) Qwen3-4B (b) Qwen3-8B

Figure 5. Sources of accepted trajectories among all-failure groups in consecutive windows of 50 training steps. The mixed category contains groups that accept both guided and unguided candidate trajectories. Fallback supplies the complete reference when no candidate trajectory is accepted. Counts below the windows give the number of all-failure groups.

Candidate trajectories are accepted in 66.0% and 60.2% of all-failure groups for Qwen3-4B and Qwen3-8B, respectively. The remaining groups use reference fallback. Most groups that accept candidate trajectories use guided ones, while a smaller fraction accept unguided ones. Fallback remains present throughout training, and its proportion varies across windows as the all-failure questions change. These records complement the selection curves by showing the trajectories used for the policy updates.