ORDER-RFT: AGE-AWARE DATA SELECTION FOR REINFORCEMENT FINE-TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement fine-tuning (RFT) improves the reasoning ability of large models, but rollout generation dominates its training cost. Data selection lowers this cost by prioritizing high-utility score samples, yet it faces an efficiency–adaptivity trade-off. Cached-score methods—whether static or dynamically updated—are cheap, but they lag behind the evolving utility in RFT. This risks a stale-utility exclusion problem: samples with low cached scores may remain unselected even after becoming useful, leaving their scores uncorrected. Conversely, refreshing scores for all samples consumes the rollouts that selection aims to save. To tackle this problem, we propose Order-RFT, a plug-and-play framework that estimates utility before rollout generation while tracking evolving utility through scheduled re-examination. It combines two complementary components: (i) a historical gradient-alignment estimator that approximates each sample's marginal contribution to policy improvement; and (ii) a linear age bonus mechanism that prioritizes both cached utility and time since the last observation, enabling samples with cached scores to be reassessed. We establish approximation error bounds for the estimator and re-examination guarantees for the age mechanism. Experiments on several benchmarks show that Order-RFT achieves higher matching with refreshed sample-utility rankings and outperforms both static and dynamic baselines.

1 INTRODUCTION

Reinforcement fine-tuning (RFT) has emerged as an effective approach to improving the reasoning abilities of large language models, enabling policies to learn from reward signals beyond supervised imitation (Shao et al., 2024; DeepSeek-AI, 2025; Tong et al., 2024; Zhang et al., 2024; Bai et al., 2025; Chen et al., 2024; Zhu et al., 2025; Wang et al., 2025; Li et al., 2024; Hu et al., 2025). Its effectiveness depends not only on the optimization algorithm but also on the data used for policy updates: different samples provide different learning opportunities for the evolving policy (Li et al., 2025; Tan et al., 2025). Meanwhile, generating on-policy responses accounts for most of the training cost, since each policy update requires fresh rollouts for every selected prompt (Luo et al., 2025; Yu et al., 2025). These considerations motivate data selection, which allocates training resources to informative samples rather than treating all data equally. When performed before rollout generation, selection can also avoid generating responses for samples that would not contribute to the update.

Data selection methods can be broadly categorized into static and dynamic selection. Static methods construct a subset before training, using indicators such as early-training gradient scores or estimated data influence. In supervised learning, carefully selected subsets can preserve predictive performance while reducing training requirements (Paul et al., 2021; Sorscher et al., 2022; Yang et al., 2023), and related advances extend this to language models through influence-based instruction selection (Xia et al., 2024) and to RFT through compact reasoning datasets (Li et al., 2025; Tan et al., 2025). However, a fixed subset cannot revise its membership as the model's learning needs change. Dynamic methods instead adjust selection during training, seeking to identify data useful for the current model. In supervised learning, online importance sampling and dynamic pruning improve computation allocation (Katharopoulos & Fleuret, 2018; Raju et al., 2021), and InfoBatch and OrderDP demonstrate cost reduction while maintaining performance (Qin et al., 2024; Jin et al., 2026). In RFT, adaptive curricula track changing sample difficulty (Gao et al., 2025; Zhang et al., 2025), while DIEM estimates gradient-based importance and reweights samples within the current

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.

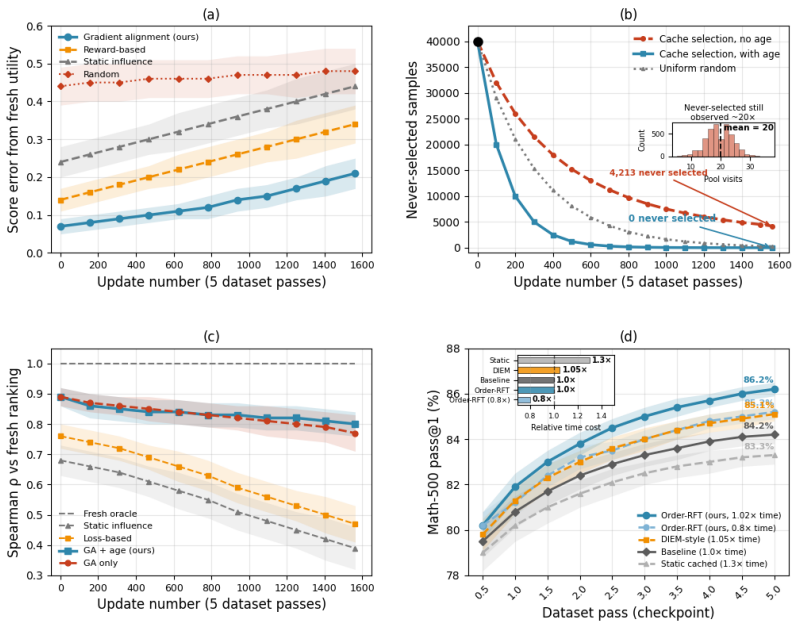


Figure 1: Training dynamics of Order-RFT on DeepScaleR with DeepSeek-R1-Distill-Qwen-1.5B under a 128-sample per-update selection budget (learning rate 1e-5, group size 8, 5 dataset passes, averaged over 3 runs). Method comparison: baseline cached selection (Baseline), static cached selection (2 dataset pass for warmup), DIEM-style reweighting, and our proposed Order-RFT. (a) Estimation error against fresh utility for gradient alignment (GA), reward-based, static-influence, and random estimators (shaded bands denote standard deviation). (b) Number of never-selected samples over training, with and without age bonus, compared to uniform random sampling; inset shows the distribution of candidate-pool visits for samples that are never selected. (c) Spearman correlation with fresh utility ranking for GA-only, GA+age, and baselines. (d) MATH-500 pass@1 over training; inset shows relative time cost normalized to the baseline (dashed line at 1.0×). Order-RFT achieves the highest accuracy at 0.8× the baseline cost.

batch (Tan et al., 2026a). The appeal of these methods lies in a common principle: data utility is a property of the interaction between a sample and the current model, not an immutable property of the sample itself.

However, realizing this advantage through pre-rollout selection presents a distinct challenge in RFT. In supervised learning, sample losses and gradients can be evaluated against stored targets. In RFT, obtaining a fresh on-policy utility estimate typically requires generating new responses (Shao et al., 2024; Yu et al., 2025). Methods such as DIEM (Tan et al., 2026b) use these responses to allocate update weights within an already generated batch; they do not eliminate generation for candidates subsequently assigned low weight. Applying fresh-score selection to every candidate likewise incurs rollout costs before deciding which samples to retain (Kim et al., 2026). Thus, the information needed to select useful data can consume the very computation that selection aims to save (Kim et al., 2026). This creates a constraint that distinguishes RFT from supervised selection: selection must be decided before rollout, so the utility estimate itself cannot rely on fresh rollouts. Any practical selector therefore needs a utility estimator that operates without generating new responses (Li et al., 2025).

Reusing cached scores offers an economical alternative, but introduces a dependency between selection and observation. Consider a selector that refreshes a sample's score only when that sample is selected for training. A low cached score reduces its chance of selection and, consequently, its opportunities for reassessment. Even if subsequent policy updates make the sample useful, its outdated score can continue to exclude it. We refer to this failure mode as stale-utility exclusion. Importantly, updating selected samples dynamically does not prevent this problem: the scores of unselected sam-

ples may remain unchanged while the policy continues to evolve. The challenge is therefore not only to estimate sample utility without rollout, but also to determine when an existing estimate should be revisited.

To verify that this failure mode occurs in practice, we conduct controlled experiments on DeepScaleR with DeepSeek-R1-Distill-Qwen-1.5B as the policy model. We highlight several key observations as follows:

❶ *Gradient alignment best predicts fresh utility but its validity expires*: Gradient alignment achieves the lowest estimation error among all rollout-free estimators, yet its Spearman correlation with fresh rankings drops from 0.87 to 0.77 over 5 dataset passes, as shown in Figure 1 (a,c), suggesting that utility is directional and its cached estimate decays as the policy evolves.

❷ *Stale scores create a self-reinforcing exclusion loop*: 10.5% of samples are never selected after 5 dataset passes despite entering the candidate pool roughly 20 times on average, and their Spearman correlation with fresh utility is only 0.41 versus above 0.83 for frequently selected samples, as shown in Figure 1 (b), revealing that low cached scores suppress the very observations needed to correct them.

❸ *Age-based re-examination breaks the loop*: A linear age bonus reduces never-selected samples from 10.5% to 0 and restores selection probability for low-scored samples without disturbing high-scored ones, as shown in Figure 1 (b,c). This coverage gain translates into performance: Order-RFT reaches 85.2% MATH-500 pass@1 at 0.8× the baseline time cost, versus 84.2% for the baseline and 85.1% for DIEM-style reweighting, as shown in Figure 1 (d), suggesting that re-examination should be scheduled by time since last observation rather than triggered by score magnitude.

Together, these findings confirm that stale-utility exclusion occurs in practice, that it is driven by the observation feedback loop between selection and score refresh, and that timely re-examination can reverse it. To address these challenges, we propose Order-RFT, a plug-and-play framework that estimates utility before rollout generation and revisits cached estimates on a schedule. Order-RFT rests on two complementary components. First, a Dynamic Data Importance Measuring estimator approximates each sample's marginal contribution to policy improvement by aligning its historical gradient direction with the most recent policy update direction. This estimator is signed: positive alignment indicates that the sample reinforces the current update direction, negative alignment indicates conflict, and near-zero alignment indicates no directional signal. Because it requires no fresh rollout, utility can be estimated for the entire candidate pool at negligible cost. Second, a linear age bonus prioritizes samples by both cached utility and time since the last observation. It acts as a principled expiration mechanism for the directional signal: even when a sample's cached alignment is low, the age bonus ensures it is periodically re-examined, so its score can be corrected after the policy evolves. Our key insight is that utility is directional, not scalar: a cached score reflects alignment with a past update direction, and its validity expires as the policy evolves.

We prove that the estimator's ranking error against fresh utility is bounded by the gradient Lipschitz constant and the staleness of the cached direction, that the age bonus guarantees every sample is re-examined within a bounded delay, and that the composite ranking error of Order-RFT is bounded by the sum of these two terms. Experiments across multiple benchmarks show that Order-RFT matches full-pool refreshed utility rankings, and outperforms both static and dynamic baselines under matched rollout budgets. Notably, in mathematical reasoning tasks, OrderRFT only requires about 80% of the rollout budget to achieve comparable results to DIEM.

2 PRELIMINARIES

Reinforcement fine-tuning. Reinforcement fine-tuning (RFT) improves a language model by optimizing its responses using reward feedback (Shao et al., 2024; DeepSeek-AI, 2025; Luo et al., 2025). We consider a training dataset $\mathcal{D} = \{z_i = (x_i, y_i^\star)\}_{i=1}^n$, where x_i is an input prompt that serves as the observation (state) and $y_i^\star$ is its reference answer. For mathematical reasoning, the reference answer supports automatic verification of generated solutions. The language model defines a stochastic policy π_θ: given x_i, it generates a response $a \sim \pi_\theta(\cdot \mid x_i)$ and receives a scalar reward $R(z_i, a)$. We use binary correctness rewards, with $R(z_i, a) = 1$ for a verifier-accepted answer and

$R(z_i, a) = 0$ otherwise. The underlying RFT objective is

$$\max_{\theta} \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}_{a \sim \pi_\theta(\cdot|x_i)} \left[R(z_i, a) \right]. \tag{1}$$

Throughout, selecting a sample means selecting its prompt for rollout generation; the reference answer is used for reward evaluation.

Group relative policy optimization. We use GRPO (Shao et al., 2024) to construct policy updates. At step t, the behavior policy π_{θ_t} generates a group $\xi_i^t = (a_{i,t}^{(1)}, \ldots, a_{i,t}^{(G)})$ of G responses for sample z_i. Let $R_{i,t}^{(j)} = R(z_i, a_{i,t}^{(j)})$. The group-relative advantage is$A_{i,t}^{(j)} = \frac{R_{i,t}^{(j)} - \overline{R}_{i,t}}{\mathrm{std}\left(R_{i,t}^{(1)}, \ldots, R_{i,t}^{(G)}\right) + \epsilon_A}$ and $\overline{R}_{i,t} = \frac{1}{G} \sum_{j=1}^{G} R_{i,t}^{(j)}$, where $\epsilon_A > 0$ ensures numerical stability. For notational simplicity, let $\rho_{i,t}^{(j)}(\theta)$ denote the importance ratio associated with the sampled response $a_{i,t}^{(j)}$ as $\rho_{i,t}^{(j)}(\theta) = \frac{\pi_\theta\left(a_{i,t}^{(j)}|x_i\right)}{\pi_{\theta_t}\left(a_{i,t}^{(j)}|x_i\right)}$.

The notation $\rho_{i,t}^{(j)}(\theta)$ abstracts away the token-level ratio and aggregation convention used by a particular GRPO implementation. Define the sampled per-prompt GRPO surrogate by

$$\phi_{i,t}(\theta; \xi_i^t) = \frac{1}{G} \sum_{j=1}^{G} \min\left\{ \rho_{i,t}^{(j)}(\theta) A_{i,t}^{(j)}, \mathrm{clip}\left(\rho_{i,t}^{(j)}(\theta), 1 - \epsilon_c, 1 + \epsilon_c\right) A_{i,t}^{(j)} \right\} - \beta_{\mathrm{KL}}\, k_{i,t}(\theta; \xi_i^t), \tag{2}$$

thus the expected per-sample GRPO objective is given as

$$j_{i,t}(\theta) = \mathbb{E}\left[\phi_{i,t}(\theta; \xi_i^t) \,\middle|\, \mathcal{F}_{t-1}\right], \tag{3}$$

where ϵ_c is the clipping parameter, $k_{i,t}$ is an optional KL regularization term relative to a fixed reference policy, and $\mathcal{F}_{t-1}$ contains the training history before the current candidate window and rollouts are sampled. The behavior policy and group advantages are held fixed when differentiating with respect to θ. Our experiments disable the KL penalty. When all rewards in a group coincide, its advantages and clipped reward-surrogate gradient are zero.

Gradient directions and batch objectives. Let $\phi_{i,t}(\theta; \xi_i^t)$ denote the expression inside the expectation in (3). The sampled ascent direction and its conditional mean are

$$d_i^t = \nabla_\theta \phi_{i,t}(\theta; \xi_i^t)\big|_{\theta=\theta_t}, \qquad h_i^t = \mathbb{E}\left[d_i^t \mid \mathcal{F}_{t-1}\right] = \nabla j_{i,t}(\theta_t), \tag{4}$$

where the last equality assumes that differentiation and expectation can be interchanged. Our influence analysis concerns this GRPO surrogate, rather than identifying its gradient with that of the expected correctness reward in (1).

For a candidate window $\mathcal{B}_t \subseteq \{1, \ldots, n\}$ of size s, define

$$J_t(\theta) = \frac{1}{s} \sum_{i \in \mathcal{B}_t} j_{i,t}(\theta), \qquad \overline{d}_t = \frac{1}{s} \sum_{i \in \mathcal{B}_t} d_i^t. \tag{5}$$

A full-window SGD ascent step would be $\theta_t^+ = \theta_t + \eta_t \overline{d}_t$. Order-RFT instead selects $\mathcal{Q}_t \subseteq \mathcal{B}_t$ with $|\mathcal{Q}_t| = q$ before generation and obtains rollout gradients only for these selected samples. The full-window directions are therefore hypothetical when used to define the influence target in Section 3.1.2.

3 MAIN METHOD

Inspired by OrderDP's uniform-window and top-q structure, Order-RFT estimates utility from historical gradient alignment and schedules score refreshes with an age bonus. Unlike current-batch reweighting such as DIEM, it ranks candidates before generating responses. We first analyze the estimator and then the age mechanism.

3.1 Historical Gradient-Alignment Utility Measurement

We quantify each sample's contribution to policy improvement without rolling out every candidate. Unlike DIEM, which forms its reference after observing the current batch, our historical alignment score enables selection before rollout generation.

Problem Definition. At each step, Order-RFT must select q samples from the candidate window $\mathcal{B}_t$ before generating their responses. Ideally, a sample's utility would reflect its marginal contribution to improving the GRPO surrogate on that window. Let $\bar{d}_t = s^{-1} \sum_{i \in \mathcal{B}_t} d_i^t$ and define the full-window and strict leave-one-out updates by

$$\theta_t^+ = \theta_t + \eta_t \bar{d}_t, \qquad \theta_{t,-z}^+ = \theta_t + \frac{\eta_t}{s-1} \sum_{i \in \mathcal{B}_t \setminus \{z\}} d_i^t. \tag{6}$$

We define the target influence score $\mathcal{I}_t$ of a sample $z \in \mathcal{B}_t$ as its marginal contribution to the GRPO surrogate evaluated on the original candidate window:

$$\mathcal{I}_t(z; \mathcal{B}_t) = J_t(\theta_t^+; \mathcal{B}_t) - J_t(\theta_{t,-z}^+; \mathcal{B}_t), \qquad g_t = \nabla J_t(\theta_t^+; \mathcal{B}_t). \tag{7}$$

Both counterfactual policies are therefore evaluated on the same original batch $\mathcal{B}_t$, as in the batch-level marginal-contribution definition of DIEM (Tan et al., 2026a). A positive value means that including z improves the post-update GRPO surrogate on that batch, whereas a negative value means that its inclusion is detrimental under this one-step criterion. Direct evaluation requires counterfactual policy updates and evaluation. Even obtaining a fresh gradient-based score for every candidate would first require rolling out all s samples, forfeiting the rollout reduction from pre-rollout selection. We therefore need a score that can be refreshed from training rollouts and retained between observations.

3.1.1 Efficient Estimation

Historical Gradient Reference. Current-batch gradient alignment, as used by DIEM (Tan et al., 2026a), can form its reference direction after observing the whole batch. That direction is unavailable when selection precedes rollout generation. Instead, we maintain an exponential moving average of the GRPO ascent directions from previously selected samples:

$$m_t = \alpha m_{t-1} + (1-\alpha) \frac{1}{q} \sum_{i \in \mathcal{Q}_t} d_i^t, \qquad m_0 = 0, \quad 0 < \alpha < 1. \tag{8}$$

The average in (8) is the direction used for the current top-q policy update. We use $\alpha = 0.9$ in our experiments. Because the EMA uses selected samples rather than the full window, it may reflect selection bias and lag behind changes in the policy; neither effect is assumed away in our analysis.

Gradient-Alignment (GA) Score. Once a selected sample i has produced its training rollouts, we measure the alignment of its sampled GRPO direction with the *previous* EMA reference:

$$u_t(i) = \frac{\eta_t}{s-1} \left\langle m_{t-1}, d_i^t \right\rangle, \qquad Y_i(t) = \mathrm{clip}_{[-B_u, B_u]}\big(u_t(i)\big). \tag{9}$$

All selected scores use m_{t-1} before their directions enter m_t. The factor $\eta_t/(s-1)$ matches the first-order scale of the strict leave-one-out influence; it does not change the ordering of fresh scores computed at the same step. A high positive score indicates agreement with the historical update direction, whereas a negative score indicates directional conflict. Neither sign alone certifies the sign of the sample's true marginal contribution to the current window.

Crucially, GA score $u_t(i)$ is computed *after* i is selected, using its ordinary training rollout; it is not a fresh pre-rollout score for every candidate. Before selection, the last stored Y_i value supplies the utility component of each candidate's age-adjusted priority. Only the q selected entries are refreshed, so scoring requires no extra candidate rollouts. Section 3.1.2 bounds the resulting fresh score's approximation error; it does not pretend that a cached score remains accurate indefinitely. Deciding when an old entry should be observed again is the separate role of the age mechanism in Section 3.2.

3.1.2 Error Analysis of the Gradient-Alignment Score

The bound below follows DIEM's batch-level marginal-contribution view but uses the candidate-window surrogate. Here $j_{i,t}$ is the expected per-sample GRPO surrogate and $J_t(\theta) = s^{-1}\sum_{i\in\mathcal{B}_t} j_{i,t}(\theta)$; directions of unselected samples are hypothetical for the counterfactual analysis.

Marginal contribution to a full-window update. For the analysis, put $a_t = \eta_t/(s-1)$. The definition above is the strict leave-one-out marginal contribution of z to a full-window update, not the actual top-q update. Removing z renormalizes the remaining directions by $s-1$. The resulting first-order influence depends on $d_z^t - \bar{d}_t$, rather than on d_z^t alone.

A direct first-order estimator would require the current full-window direction. Order-RFT uses m_{t-1} instead; the omitted $\bar{d}_t$ is a common within-window offset. The bound accounts for EMA lag and top-q selection bias.

Assumption 3.1 (Gradient envelope)**.** For every step and prompt, including hypothetical rollout groups, the sampled direction of the complete GRPO surrogate satisfies

$$\|d_i^k\| \le G_{\mathrm{dir}} < \infty. \tag{10}$$

A sufficient condition is bounded advantage, active likelihood ratio, aggregated log-likelihood gradient, and optional KL gradient; their product bound is absorbed into G_{dir}.

Assumption 3.2 (Local smoothness and drift)**.** Each J_t has an L-Lipschitz gradient on the convex hull of θ_t, θ_t^+ and all $\theta_{t,-z}^+$. Writing $\mu_k = n^{-1}\sum_i h_i^k$, assume $\|\mu_{k+1} - \mu_k\| \le D$ almost surely. Windows are uniform, and selection precedes fresh rollouts.

These are local assumptions on the expected GRPO surrogate, including changes in the behavior policy and advantage distribution.

Proposition 3.3 (Influence error of the alignment score)**.** *Under the assumptions above, let $n \ge s \ge 2$, $1 \le q \le s$, $0 < \alpha < 1$, $m_0 = 0$, and a deterministic learning-rate schedule $\eta_t > 0$. Put*

$$\begin{aligned} T_t^{\mathrm{GRPO}} &= L\eta_t G_{\mathrm{dir}} + G_{\mathrm{dir}}\sqrt{\frac{n-s}{s(n-1)}} + \alpha^{t-1}G_{\mathrm{dir}} \\ &\quad + 2G_{\mathrm{dir}}\frac{s-q}{s}(1-\alpha^{t-1}) + D(1-\alpha)\sum_{k=1}^{t-1}\alpha^{t-1-k}(t-k) \\ &\quad + G_{\mathrm{dir}}\sqrt{\frac{1-\alpha}{1+\alpha}(1-\alpha^{2(t-1)})}, \\ E_t &= 2a_t G_{\mathrm{dir}} T_t^{\mathrm{GRPO}} + 2La_t^2 G_{\mathrm{dir}}^2. \end{aligned} \tag{11}$$

For the fresh raw score $u_t(z) = a_t\langle m_{t-1}, d_z^t\rangle$,

$$\mathbb{E}|\mathcal{I}_t(z;\mathcal{B}_t) - u_t(z)| \le E_t + a_t G_{\mathrm{dir}}^2, \tag{12}$$

$$\mathbb{E}\,|(\mathcal{I}_t(z;\mathcal{B}_t) - \mathcal{I}_t(z';\mathcal{B}_t)) - (u_t(z) - u_t(z'))| \le 2E_t. \tag{13}$$

The expectations are over a uniform window, a target (or distinct target pair) chosen uniformly from that window, and fresh-rollout randomness. Conditioning on a named target's inclusion changes the finite-population term and is not covered by this display. For clipped scores, add $\rho_t = (a_t G_{\mathrm{dir}}^2 - B_u)_+$ to the first bound and $2\rho_t$ to the second.

The centered version differs from the implemented raw score by a window-common offset, so both induce the same fresh within-window ranking; the offset is unavailable before observing all s samples. The bound is for fresh scores, not arbitrarily old cache entries. A pairwise influence gap must exceed the bound for reliable ordering. The full proof is in Appendix A.

3.2 Age-Aware Re-examination

The bound in Section 3.1.2 concerns a *fresh* score. Before rollout, Order-RFT must rank candidates by values last observed at different policy checkpoints. A low cached value can suppress the very selection needed to correct it. Age does not infer an unobserved current gradient; it schedules opportunities to obtain one.

Algorithm 1 One Order-RFT iteration

Require: $\theta_t, m_{t-1}, \widehat{u}^t, \tau^t; 2 \le s \le n, 1 \le q \le s$
1: Sample independent tie breaks, then draw a uniform size-s window $\mathcal{B}_t$.
2: Select $\mathcal{Q}_t = \mathrm{Top}_q\{p_i^t : i \in \mathcal{B}_t\}$ using (14).
3: Generate G responses per selected prompt and compute d_i^t.
4: Compute $Y_i(t)$ with m_{t-1} using (9).
5: Update $\theta_{t+1} = \theta_t + \eta_t q^{-1} \sum_{i \in \mathcal{Q}_t} d_i^t$.
6: Update m_t using (8).
7: Set $\widehat{u}_i^{t+1} = Y_i(t)$ and $\tau_i^{t+1} = t$ for $i \in \mathcal{Q}_t$; retain other entries.

3.2.1 CACHE AND PRIORITY

Initialize $\widehat{u}_i^1 = 0$ and $\tau_i^1 = 0$. At step t, define

$$p_i^t = \widehat{u}_i^t + \beta(t - \tau_i^t), \qquad \beta = \lambda q/n, \quad \lambda \ge 0. \tag{14}$$

Here $\widehat{u}_i^t$ is the last stored, clipped alignment score and τ_i^t is the last iteration on which prompt i was selected for rollout. If it is not selected, both values persist; after selection, $\widehat{u}_i^{t+1} = Y_i(t)$ and $\tau_i^{t+1} = t$. Thus age raises the priority of an unobserved prompt without changing its estimated utility or the GRPO objective. The parameter λ is the priority gained over a nominal n/q-step pool traversal, in stored score units. It must be calibrated to a fixed training-only score scale; the normalized pilot's $\lambda = 1$ is not a raw-score default. Clipping to $[-B_u, B_u]$ supplies the bounded range used below; age itself is not capped.

Only the selected q prompts receive rollouts; the cache is updated only for those prompts.

3.2.2 WHAT AGE CAN GUARANTEE

Without age, a low cached score can delay the observation needed to correct it. Age addresses this stale-score failure by increasing a candidate's priority until it can outrank more recently refreshed competitors.

Theorem 3.4 (Age priority and re-examination)**.** *Assume $|\widehat{u}_j^t| \le B_u$ and $\beta > 0$. Fix an unselected prompt i with last-selection timestamp τ_i, and put $h = \lceil 2B_u/\beta \rceil + 1$. At every step $t \ge \tau_i + h$ before its next selection, i has strictly greater priority than every competitor j with $\tau_j^t \ge \tau_i + h$. Let*

$$O_i(t) = \{j \ne i : \tau_j^t < \tau_i + h\}, \qquad H_{n,s,q}(M) = \sum_{k=0}^{q-1} \frac{\binom{M}{k}\binom{n-1-M}{s-1-k}}{\binom{n-1}{s-1}}. \tag{15}$$

If, at a stopping time $a \ge \tau_i + h$, $|O_i(a)| \le M$ for some $M \le n-1$, then for D_i measured from a and $H_{n,s,q}(M) > 0$,

$$\mathbb{P}(D_i > K \mid \mathcal{F}_a) \le \big(1 - (s/n)H_{n,s,q}(M)\big)^K, \qquad \mathbb{E}[D_i \mid \mathcal{F}_a] \le \frac{n}{sH_{n,s,q}(M)}. \tag{16}$$

In particular, $M \le q-1$ implies $H_{n,s,q}(M) = 1$: the next candidate visit then selects i. Without any restriction on the old cohort, put $r = \lfloor (n-1)/q \rfloor + 1$ and $p = s/n$. Then

$$\mathbb{P}(D_i > K \mid \mathcal{F}_a) \le \mathbb{P}(\mathrm{Bin}(K, p) < r), \qquad \mathbb{E}[D_i \mid \mathcal{F}_a] \le \frac{r}{p}. \tag{17}$$

The theorem bounds re-examination delay. The unconditional bound follows because every candidate visit that fails to select an over-age prompt selects and refreshes q old competitors instead. It bounds delay in expectation and probability, not deterministically in wall-clock steps. Age does not improve the pointwise approximation error of a fresh gradient-alignment score; its effect is on when a cached estimate can be refreshed.

Corollary 3.5 (Discovery of a persistent useful state)**.** *Suppose the useful state of prompt i persists for K iterations starting at $a \ge \tau_i + h$. If the fresh rollout score at the first selection recognizes the state with conditional failure probability at most δ_{obs}, then the probability of discovery within K iterations is at least*

$$1 - \mathbb{P}(\mathrm{Bin}(K, s/n) < r) - \delta_{\mathrm{obs}}, \qquad r = \lfloor (n-1)/q \rfloor + 1. \tag{18}$$

If $|O_i(a)| \le M$ and $H_{n,s,q}(M) > 0$, one may replace the binomial-tail term by the minimum of itself and the conditional bound $\big(1-(s/n)H_{n,s,q}(M)\big)^K$. If persistence itself fails with probability at most δ_{state}, subtract that term as well.

Connection to ordered sampling. Age changes the history-measurable ranking but preserves OrderDP's conditional rank-weighted update identity (Proposition B.3). Under the separate fixed-objective, convex, unbiased-gradient assumptions of Proposition B.5, the standard $O(T^{-1/2})$ optimization bound applies. Neither that identity nor the age result establishes convergence of a changing GRPO surrogate to the uniform-pool optimum. The exact selection weights and proofs appear in Appendices B and C.

4 Experiments

Experimental setup. We evaluate Order-RFT on mathematical reasoning tasks using DeepSeek-R1-Distill-Qwen-1.5B/7B and DeepScaleR (Luo et al., 2025), trained with 4 NVIDIA PRO6000 GPUs. All methods use the same model initialization, GRPO implementation, optimizer, learning rate 10^{-5}, response sampler, and maximum generation length. Unless stated otherwise, we use $G = 8$ rollouts per observed sample and the OrderDP-style uniform candidate window with $s = 512$ and $q = 128$. Evaluation uses fixed held-out sets and the same answer verifier for every method. The generated-rollout budget is matched when comparing methods that select q samples; methods that observe all s samples are charged for their additional rollouts.

We compare against representative static and dynamic data-selection methods. **Random** uniformly samples training data without a utility score. **LIMR** (Li et al., 2025) and **HVS** (Wang et al., 2026) are static selection baselines. **RFT-Inf** (Tan et al., 2025) uses influence-based selection to construct a fixed subset. **DIEM** (Tan et al., 2026a) computes gradient-based importance after observing the current batch and reweights that batch. Order-RFT performs selection before rollout using the historical gradient-alignment score and age-aware re-examination. The uniform-window top-q protocol follows the ordered-sampling structure of OrderDP (Jin et al., 2026); its loss-based selection rule is distinct from our GRPO utility estimator.

4.1 Main Results

Table 1 summarizes the performance comparison on mathematical reasoning benchmarks. We report the results for two base model scales under the same training and evaluation protocol. The table includes static selection (LIMR, HVS, and RFT-Inf), the dynamic current-batch method DIEM, and our pre-rollout Order-RFT.

For the 1.5B model, Order-RFT obtains the best reported result on all four benchmarks: 43.1 on AIME24, 85.2 on MATH-500, 77.8 on AMC23, and 53.7 on Olympiad. Relative to DIEM, Order-RFT improves these scores by 0.9, 0.6, 1.2, and 1.2 percentage points, respectively. The gains are also consistent against the static baselines: Order-RFT improves over RFT-Inf by 1.8–2.5 points across the four benchmarks, while LIMR and HVS remain below the proposed method on every reported metric.

The same trend holds for the 7B model. Order-RFT reaches 61.3 on AIME24, 89.4 on MATH-500, 90.2 on AMC23, and 64.4 on Olympiad, again exceeding all listed baselines on each benchmark. Compared with DIEM, the gains are 1.6, 1.3, 0.7, and 1.3 points. These results indicate that the benefit of Order-RFT is not limited to a particular model scale: estimating utility before rollout and revisiting stale entries can outperform both static subset construction and dynamic methods that require rollouts for the whole candidate batch.

4.2 Ablation Study

We next isolate the two components of Order-RFT: the historical gradient-alignment utility estimator and the age bonus. All variants use the same candidate-window size, selected-set size, rollout budget, and training seed protocol. The full method combines both components. The *Age-only* variant removes gradient alignment and ranks candidates only by age, testing whether re-examination without a utility signal is sufficient. The *Cached alignment* variant sets the age coefficient to zero,

Table 1: Performance comparison on mathematical reasoning benchmarks. All methods use identical data, rollout budgets, and optimization settings within each model scale.

Method	AIME24	MATH-500	AMC23	Olympiad	Avg
Base Model: DeepSeek-R1-Distill-Qwen-1.5B					
Random	38.4	82.6	75.2	49.6	61.5
LIMR (Li et al., 2025)	40.1	83.1	75.7	49.9	62.2
HVS (Wang et al., 2026)	40.2	83.4	75.5	50.3	62.4
RFT-Inf (Tan et al., 2025)	41.3	83.2	75.3	51.3	62.8
DIEM (Tan et al., 2026a)	42.2	84.6	76.7	52.6	64.1
Order-RFT (ours)	**43.1**	**85.2**	**77.9**	**53.8**	**65.1**
Base Model: DeepSeek-R1-Distill-Qwen-7B					
Random	54.2	85.1	88.3	60.5	72.0
LIMR (Li et al., 2025)	54.9	85.7	88.5	61.2	72.6
HVS (Wang et al., 2026)	55.3	86.5	88.7	61.5	73.1
RFT-Inf (Tan et al., 2025)	58.6	86.8	89.1	62.4	74.3
DIEM (Tan et al., 2026a)	59.7	88.1	89.5	63.1	75.1
Order-RFT (ours)	**61.3**	**89.4**	**90.2**	**64.4**	**76.3**

Table 2: Ablation of the historical gradient-alignment estimator and the age-aware re-examination mechanism at matched generation budget. Coverage is the fraction of stale-useful samples revisited within the specified horizon; quality is held-out pass@1.

Variant	Revisit median	Coverage	Avg. pass@1
Raw EMA alignment + age	16	89%	85.5
Raw EMA alignment, no age	42	58%	84.0
Age-only	19	87%	84.8
Uniform-q (no alignment, no age)	55	25%	83.2
Alignment + uniform exploration	26	73%	85.0
Alignment + periodic refresh	22	79%	85.2
Normalized projected alignment + age	**12**	**94%**	**86.2**
Reward-variance score + age	28	68%	84.5

testing whether a cached utility score alone can adapt to the evolving policy. *Uniform-q* removes both mechanisms and provides the same per-step rollout budget through uniform sampling. The remaining variants test whether simpler exploration or the normalized scorer used in the pilot can replace the proposed combination.

The full method achieves the lowest revisit median (12) and highest coverage (94q q performs worst on all three metrics (55, 25%, 83.2%), establishing that neither mechanism is replaceable by random selection. The additional controls show that the gain is not explained by generic exploration (73%, 85.0%) or periodic refresh (79%, 85.2%), and that normalization matters: Raw EMA alignment lags the normalized projected version (89% vs. 94% coverage). Finally, replacing gradient alignment with reward-variance scoring (68%, 84.5%) confirms that directional gradient information is more predictive of fresh utility than scalar reward statistics

5 Conclusion

Cached data selection can delay the observations needed to correct an outdated ranking. Order-RFT addresses this feedback loop with a linear age bonus, treating rollout selection as both choosing a training update and scheduling the next observation of a prompt's utility. The age bonus controls re-examination delay under bounded scores, and persistent useful states can be discovered with additional observation guarantees. The influence bound explains what historical alignment estimates and the conditions under which it is accurate. The ordered-sampling update identity is preserved, with standard convergence guarantees under the stated fixed-objective assumptions. Fresh-score agreement and held-out quality at matched generation budgets test the practical benefit beyond the re-examination guarantee.

AI USE STATEMENT

Generative AI tools were used for manuscript polishing, proof checking, and minor revisions to existing analysis and plotting code. The research direction, methodology, experimental design, statistical analysis, and interpretation of results were determined by the authors. All mathematical claims, proofs, experimental results, and AI-assisted revisions were independently reviewed and verified by the authors. The authors take full responsibility for the final manuscript.

ETHICS STATEMENT

This work studies data selection for reinforcement fine-tuning using benchmark tasks and model-generated responses. It involves no human participants, personal data, or sensitive information.

REPRODUCIBILITY STATEMENT

Proofs and their assumptions are provided in the appendix. Training settings, numerical details, software versions, and hardware configurations are also documented in the appendix.

REFERENCES

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, and et.al. Qwen2.5-vl technical report, 2025.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24185–24198, 2024.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Zhaolin Gao, Joongwon Kim, Wen Sun, Thorsten Joachims, Sid Wang, Richard Yuanzhe Pang, and Liang Tan. Prompt curriculum learning for efficient LLM post-training, 2025.

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. *arXiv preprint arXiv:2503.24290*, 2025.

Chenhan Jin, Shengze Xu, Qingsong Wang, Fan Jia, Dingshuo Chen, and Tieyong Zeng. OrderDP: A theoretically guaranteed lossless dynamic data pruning framework. In *International Conference on Learning Representations*, 2026.

Angelos Katharopoulos and François Fleuret. Not all samples are created equal: Deep learning with importance sampling. *ICML*, 2018.

Woojeong Kim, Ziyi Yang, Jing Nathan Yan, and Jialu Liu. Spend your rollouts where it counts: Rollout allocation for group-based rl post-training. *arXiv preprint arXiv:2605.26606*, 2026.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

Xuefeng Li, Haoyang Zou, and Pengfei Liu. LIMR: Less is more for RL scaling. *arXiv preprint arXiv:2502.11886*, 2025.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y. Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Li Erran Li, Raluca Ada Popa, and Ion Stoica. DeepScaleR: Surpassing o1-preview with a 1.5b model by scaling rl. DeepScaleR project report, 2025. URL https://github.com/agentica-project/deepscaler.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. In *Advances in Neural Information Processing Systems*, 2021.

Ziheng Qin et al. InfoBatch: Lossless training speed up by unbiased dynamic data pruning. In *International Conference on Learning Representations*, 2024.

Ravi S. Raju, Kyle Daruwalla, and Mikko Lipasti. Accelerating deep learning with dynamic data pruning. arXiv preprint, 2021.

Zhihong Shao et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. Beyond neural scaling laws: Beating power law scaling via data pruning. In *Advances in Neural Information Processing Systems*, 2022.

Haoru Tan, Xiuzhe Wu, Sitong Wu, Shaofeng Zhang, Yanfeng Chen, Xingwu Sun, Jeanne Shen, and Xiaojuan Qi. Understanding data influence in reinforcement finetuning. In *Advances in Neural Information Processing Systems*, 2025.

Haoru Tan, Sitong Wu, Yanfeng Chen, Shizhen Zhao, Yang-Tian Sun, Tianjia Liu, Chirui Chang, Shaofeng Zhang, Samm Sun, Xiuzhe Wu, Ruobing Xie, and Xiaojuan Qi. Dynamic important example mining for reinforcement finetuning, 2026a.

Haoru Tan, Sitong Wu, Yanfeng Chen, Shizhen Zhao, Yang-Tian Sun, Tianjia Liu, Chirui Chang, Shaofeng Zhang, Samm Sun, Xiuzhe Wu, Ruobing Xie, and Xiaojuan Qi. Dynamic important example mining for reinforcement finetuning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026b.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Austin Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal llms, 2024.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Yiping Wang, Qing Yang, Zhiyuan Zeng, Liliang Ren, Liyuan Liu, Baolin Peng, Hao Cheng, Xuehai He, Kuan Wang, Jianfeng Gao, et al. Reinforcement learning for reasoning in large language models with one training example. *Advances in Neural Information Processing Systems*, 38: 122721–122764, 2026.

Mengzhou Xia et al. Less: Selecting influential data for targeted instruction tuning. *ICML*, 2024.

Shuo Yang, Zeke Xie, Hanyu Peng, Min Xu, Mingming Sun, and Ping Li. Dataset pruning: Reducing training data by examining generalization influence. In *International Conference on Learning Representations*, 2023.

Qiying Yu et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Ruiqi Zhang, Daman Arora, Song Mei, and Andrea Zanette. SPEED-RL: Faster training of reasoning models via online curriculum learning, 2025.

Ruohong Zhang, Bowen Zhang, Yanghao Li, Haotian Zhang, Zhiqing Sun, Zhe Gan, Yinfei Yang, Ruoming Pang, and Yiming Yang. Improve vision language model chain-of-thought reasoning. *arXiv preprint arXiv:2410.16198*, 2024.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

A Proof of the GRPO Influence Approximation Proposition

All expectations include candidate-window and fresh-rollout randomness. Conditional expectations condition on the history before the corresponding window. We use the main-text notation $\alpha = \alpha_m$ for the EMA decay. Put $a_t = \eta_t/(s-1)$ and $\delta_{t,z} := d_z^t - \bar{d}_t$. The filtration $\mathcal{F}_k$ contains the completed step k; at step k the window and selection are chosen before the fresh groups. Directions for unselected prompts can be coupled to independent hypothetical groups without altering the algorithm. The result averages over a uniform window and a target (or distinct target pair) chosen uniformly from that window. Conditioning on the inclusion of a specified prompt would change that window distribution and requires a corresponding sampling bound.

The per-prompt objective in the main text is, explicitly,

$$\phi_{i,t}(\theta;\xi) = \frac{1}{G}\sum_{j=1}^{G} \min\{r_{i,j}(\theta)A_{i,j}, \mathrm{clip}(r_{i,j}(\theta), 1-\epsilon_c, 1+\epsilon_c)A_{i,j}\} - \beta_{\mathrm{KL}} k_{i,t}(\theta). \tag{19}$$

Here $r_{i,j} = \pi_\theta(y_{i,j} \mid z_i)/\pi_{b,t}(y_{i,j} \mid z_i)$. The behavior distribution and advantages are fixed under differentiation. This is the complete surrogate used for both counterfactuals. All norms and inner products use the same trainable parameter coordinates.

A.1 Leave-one-out identity and Taylor expansion

From (6),

$$\theta_{t,-z}^+ - \theta_t^+ = \eta_t \frac{s\bar{d}_t - d_z^t}{s-1} - \eta_t \bar{d}_t \tag{20}$$

$$= -\frac{\eta_t}{s-1}(d_z^t - \bar{d}_t) = -a_t \delta_{t,z}. \tag{21}$$

Since J_t has an L-Lipschitz gradient, Taylor's theorem with integral remainder gives

$$J_t(\theta_t^+ - a_t\delta_{t,z}) = J_t(\theta_t^+) - a_t\langle g_t, \delta_{t,z}\rangle + R_{t,z}, \qquad |R_{t,z}| \le \frac{L}{2}a_t^2\|\delta_{t,z}\|^2. \tag{22}$$

Consequently,

$$\mathcal{I}_t(z;\mathcal{B}_t) = a_t\langle g_t, \delta_{t,z}\rangle - R_{t,z}. \tag{23}$$

The gradient envelope gives $\|\bar{d}_t\| \le G_{\mathrm{dir}}$ and $\|\delta_{t,z}\| \le 2G_{\mathrm{dir}}$, so

$$|R_{t,z}| \le 2La_t^2 G_{\mathrm{dir}}^2. \tag{24}$$

For completeness, the bound $\|d_i^t\| \le G_{\mathrm{dir}}$ follows directly from the GRPO surrogate. On a non-clipped branch, the gradient of one ratio term is $A\, r_\theta \nabla_\theta \log \pi_\theta$, whose norm is at most $\bar{r}A_{\max}L_\pi$. On a clipped branch the ratio is locally constant, so its contribution is zero (a boundary subgradient can be chosen with the same bound). The per-prompt KL contribution adds at most $\beta_{\mathrm{KL}}L_{\mathrm{KL}}$. Averaging response terms does not increase this upper bound after the averaging constants have been absorbed into G_{dir}.

The centered score has the exact identity

$$\widehat{\mathcal{I}}_t(z) = a_t\langle m_{t-1}, \delta_{t,z}\rangle = u_t(z) - c_t, \qquad c_t := a_t\langle m_{t-1}, \bar{d}_t\rangle. \tag{25}$$

Subtracting and applying Cauchy–Schwarz yields

$$\left|\mathcal{I}_t(z;\mathcal{B}_t) - \widehat{\mathcal{I}}_t(z)\right| \le 2a_t G_{\mathrm{dir}}\|g_t - m_{t-1}\| + 2La_t^2 G_{\mathrm{dir}}^2. \tag{26}$$

A.2 Current batch direction

Let $\bar{h}_{\mathcal{B}_t} := s^{-1}\sum_{i\in\mathcal{B}_t} h_i^t$. The conditional GRPO identity gives

$$\nabla J_t(\theta_t) = \bar{h}_{\mathcal{B}_t}. \tag{27}$$

Smoothness and $\theta_t^+ - \theta_t = \eta_t \bar{d}_t$ imply

$$\|g_t - \bar{h}_{\mathcal{B}_t}\| \le L\eta_t \|\bar{d}_t\| \le L\eta_t G_{\mathrm{dir}}. \tag{28}$$

For uniform sampling without replacement, the finite-population identity is

$$\mathbb{E}\left[\|\bar{h}_{\mathcal{B}_t} - \mu_t\|^2 \mid \mathcal{F}_{t-1}\right] = \frac{n-s}{s(n-1)} \frac{1}{n} \sum_{i=1}^{n} \|h_i^t - \mu_t\|^2. \tag{29}$$

Because $\|h_i^t\| \le \mathbb{E}[\|d_i^t\| \mid \mathcal{F}_{t-1}] \le G_{\mathrm{dir}}$,

$$\frac{1}{n} \sum_{i=1}^{n} \|h_i^t - \mu_t\|^2 = \frac{1}{n} \sum_{i=1}^{n} \|h_i^t\|^2 - \|\mu_t\|^2 \le G_{\mathrm{dir}}^2. \tag{30}$$

Thus

$$\mathbb{E}\|\bar{h}_{\mathcal{B}_t} - \mu_t\| \le G_{\mathrm{dir}} \sqrt{\frac{n-s}{s(n-1)}}. \tag{31}$$

A.3 EMA TRACKING WITH EXPLICIT GRPO BOUNDS

Let

$$A_k := \frac{1}{q} \sum_{i \in \mathcal{Q}_k} d_i^k \tag{32}$$

denote the actually observed (selected-prompt) direction at step k. Define

$$e_k := \mathbb{E}[A_k \mid \mathcal{F}_{k-1}] - \mu_k, \qquad \xi_k := A_k - \mathbb{E}[A_k \mid \mathcal{F}_{k-1}]. \tag{33}$$

Let W_k be the realized candidate window, and let $\bar{h}_{Q_k}$, $\bar{h}_{W_k \setminus Q_k}$, and $\bar{h}_{W_k}$ denote the means of the conditional directions over the selected members, the unselected members, and all members of W_k, respectively. Because selection is made before fresh rollouts,

$$\mathbb{E}[A_k \mid \mathcal{F}_{k-1}, W_k, Q_k] = \bar{h}_{Q_k}. \tag{34}$$

For every realized window, the selected and window means satisfy

$$\left\|\bar{h}_{Q_k} - \bar{h}_{W_k}\right\| \le 2G_{\mathrm{dir}} \frac{s-q}{s}. \tag{35}$$

If $q = s$, then $Q_k = W_k$ and the left-hand side is exactly zero. If $q < s$, define the mean over the nonselected prompts by $\bar{h}_{W_k \setminus Q_k}$. Then

$$\bar{h}_{Q_k} - \bar{h}_{W_k} = \frac{s-q}{s} \left(\bar{h}_{Q_k} - \bar{h}_{W_k \setminus Q_k}\right), \qquad \left\|\bar{h}_{Q_k} - \bar{h}_{W_k \setminus Q_k}\right\| \le 2G_{\mathrm{dir}}, \tag{36}$$

which proves the displayed bound in both cases. Taking conditional expectation over the independent uniform window and using $\mathbb{E}[\bar{h}_{W_k} \mid \mathcal{F}_{k-1}] = \mu_k$ gives

$$\begin{aligned} e_k &= \mathbb{E}[A_k \mid \mathcal{F}_{k-1}] - \mu_k \\ &= \mathbb{E}[\bar{h}_{Q_k} - \bar{h}_{W_k} \mid \mathcal{F}_{k-1}], \end{aligned} \tag{37}$$

and hence $\|e_k\| \le 2G_{\mathrm{dir}}(s-q)/s$. Moreover, $\|A_k\| \le G_{\mathrm{dir}}$ implies

$$\mathbb{E}\left[\|\xi_k\|^2 \mid \mathcal{F}_{k-1}\right] = \mathbb{E}\left[\|A_k\|^2 \mid \mathcal{F}_{k-1}\right] - \left\|\mathbb{E}[A_k \mid \mathcal{F}_{k-1}]\right\|^2 \le G_{\mathrm{dir}}^2. \tag{38}$$

Iterating the EMA recursion gives

$$m_{t-1} = (1-\alpha) \sum_{k=1}^{t-1} \alpha^{t-1-k} (\mu_k + e_k + \xi_k), \tag{39}$$

where $m_0 = 0$. Subtracting μ_t, using $\|\mu_k - \mu_t\| \le D(t-k)$, and applying the triangle inequality gives

$$\mathbb{E}\|m_{t-1} - \mu_t\| \le \alpha^{t-1} G_{\mathrm{dir}} + 2G_{\mathrm{dir}} \frac{s-q}{s} (1 - \alpha^{t-1}) \tag{40}$$

$$+ D(1-\alpha) \sum_{k=1}^{t-1} \alpha^{t-1-k} (t-k) + N_t, \tag{41}$$

where

$$N_t := \mathbb{E}\left\|(1-\alpha)\sum_{k=1}^{t-1}\alpha^{t-1-k}\xi_k\right\|. \tag{42}$$

The terms ξ_k are martingale differences. Therefore cross terms vanish, and (38) yields

$$N_t^2 \le G_{\mathrm{dir}}^2(1-\alpha)^2\sum_{k=1}^{t-1}\alpha^{2(t-1-k)} \tag{43}$$

$$= G_{\mathrm{dir}}^2\frac{1-\alpha}{1+\alpha}\left(1-\alpha^{2(t-1)}\right). \tag{44}$$

Consequently,

$$\begin{aligned}\mathbb{E}\|m_{t-1}-\mu_t\| \le{}& \alpha^{t-1}G_{\mathrm{dir}} + 2G_{\mathrm{dir}}\frac{s-q}{s}(1-\alpha^{t-1}) \\ &+ D(1-\alpha)\sum_{k=1}^{t-1}\alpha^{t-1-k}(t-k) + G_{\mathrm{dir}}\sqrt{\frac{1-\alpha}{1+\alpha}\left(1-\alpha^{2(t-1)}\right)}.\end{aligned} \tag{45}$$

Combining this inequality with (28) and (31) proves

$$\mathbb{E}\|g_t - m_{t-1}\| \le T_t^{\mathrm{GRPO}}. \tag{46}$$

A.4 Completion of the proof

Taking expectations in (26) and using (46) proves the centered error bound

$$\mathbb{E}|\mathcal{I}_t(z;\mathcal{B}_t) - \widehat{\mathcal{I}}_t(z)| \le E_t. \tag{47}$$

Since $\|A_k\| \le G_{\mathrm{dir}}$, the EMA recursion gives $\|m_{t-1}\| \le G_{\mathrm{dir}}$ by induction, and $\|\bar{d}_t\| \le G_{\mathrm{dir}}$. Hence

$$|c_t| = a_t|\langle m_{t-1}, \bar{d}_t\rangle| \le a_t G_{\mathrm{dir}}^2. \tag{48}$$

The triangle inequality applied to

$$\mathcal{I}_t(z) - u_t(z) = \left[\mathcal{I}_t(z) - \widehat{\mathcal{I}}_t(z)\right] - c_t \tag{49}$$

proves (12). For two prompts, the common offset c_t cancels exactly; applying the centered bound to each prompt and using the triangle inequality proves (13). This completes the proof for the raw score. If the implementation clips the raw score, then

$$\left|u_t(z) - \mathrm{clip}_{[-B_u,B_u]}(u_t(z))\right| \le \left(a_t G_{\mathrm{dir}}^2 - B_u\right)_+, \tag{50}$$

because $\|m_{t-1}\| \le G_{\mathrm{dir}}$ and $\|d_z^t\| \le G_{\mathrm{dir}}$. Adding this deterministic residual to the raw-score bound (which already contains the common-offset term $a_t G_{\mathrm{dir}}^2$) by the triangle inequality gives the clipped-score inequality stated in Proposition 3.3. For a pair of clipped scores, add the two clipping residuals, at most $2\rho_t$. Finally,

$$(1-\alpha)\sum_{k=1}^{t-1}\alpha^{t-1-k}(t-k) \le (1-\alpha)\sum_{j=0}^{\infty}\alpha^j(j+1) = \frac{1}{1-\alpha},$$

which establishes the drift interpretation in the main text.

B Ordered Sampling and Standard Optimization Background

These identities clarify the connection to OrderDP. They apply to the same uniform-window top-q skeleton regardless of the score or age rule. They do not transfer a fixed-objective convergence or generalization theorem automatically to a nonstationary GRPO training run.

Lemma B.1 (Exact rank-inclusion probabilities)**.** *Condition on a fixed total ranking and let $\rho_t(j)$ be its prompt of rank j. The probability that this prompt is retained from a uniform size-s window is*

$$\gamma_j = \sum_{\ell=1}^{q} \frac{\binom{j-1}{\ell-1}\binom{n-j}{s-\ell}}{\binom{n}{s}}. \tag{51}$$

Here out-of-range binomial coefficients are zero. Moreover, $\gamma_1 = s/n$, $\gamma_1 \geq \cdots \geq \gamma_n$, and $\sum_{j=1}^{n} \gamma_j = q$.

Remark B.2. The update distribution is generally nonuniform even though the candidate window is uniform. The normalized weights γ_j/q are a property of ordered selection, independent of how the ranking was obtained. If $q = s$, all candidates are retained and the distribution becomes uniform.

Proposition B.3 (Conditional surrogate update)**.** *Let $h_i^t = \mathbb{E}[d_i^t \mid \mathcal{F}_{t-1}]$ be the expected GRPO ascent direction for prompt i, and assume that the fresh rollout remains conditionally distributed as specified in Section 3. Put $\ell_{i,t} = -j_{i,t}$, including any KL term used in the complete surrogate. For a ranking ρ_t, define*

$$f_t(\theta) = -\frac{1}{q}\sum_{j=1}^{n} \gamma_j \ell_{\rho_t(j),t}(\theta). \tag{52}$$

Thus f_t is the weighted GRPO surrogate (the negative of the loss), and the selected update $\widetilde{d}_t = q^{-1}\sum_{i\in\mathcal{Q}_t} d_i^t$ satisfies

$$\mathbb{E}[\widetilde{d}_t \mid \mathcal{F}_{t-1}] = \nabla f_t(\theta_t) \tag{53}$$

whenever the per-prompt expected GRPO losses are differentiable. The statement is local to the fixed history; the surrogate and ranking may change at the next iteration.

Remark B.4. This is a conditional statement, not a claim that ordered selection optimizes the uniform pool objective. It also does not require the score to be fresh: stale scores only determine the history-measurable ranking. The discrepancy from the uniform population objective is quantified by the weights in Lemma B.1; age changes the ranking over time but does not change this identity.

Proposition B.5 (Optimization and population-gap baselines)**.** *As a standard baseline, suppose a fixed conditional minimization loss f is convex on a closed convex domain containing a minimizer θ^*, projected SGD uses conditionally unbiased subgradients and a constant step η, the initial distance to a minimizer is at most R_0, and its stochastic gradient g_t satisfies $\mathbb{E}\|g_t\|^2 \leq G_0^2$. Write $\bar{\theta}_T = T^{-1}\sum_{t=1}^{T}\theta_t$. Then*

$$\mathbb{E}[f(\bar{\theta}_T)] - f(\theta^*) \leq \frac{R_0^2}{2\eta T} + \frac{\eta G_0^2}{2}. \tag{54}$$

If expected per-prompt losses have absolute value at most B_ℓ and the pool is drawn from a distribution with a uniform-over-parameters deviation bound $\epsilon_n(\delta)$, then with probability at least $1-\delta$ the absolute difference between the population loss and the rank-weighted empirical loss, at any parameter and ranking, is at most

$$\epsilon_n(\delta) + B_\ell A_{n,s,q}, \qquad A_{n,s,q} = \sum_{j=1}^{n}\left|\frac{\gamma_j}{q} - \frac{1}{n}\right|. \tag{55}$$

The term $A_{n,s,q}$ is zero when $q = s$ and is not removed merely by visiting every prompt.

Remark B.6. Equation (54) is an optimization baseline for a fixed ranking and a minimization convention; it is not a new non-convex convergence theorem for GRPO. Equation (55) explains why coverage and generalization are different: a selector can revisit every prompt while still weighting high-ranked prompts more often. We therefore evaluate quality and coverage separately.

C Detailed Proofs

This appendix proves the statements in Section 3.2. We make the conditioning explicit because two random events are easy to confuse: the candidate window determines whether a prompt is observed at all, while the ranking inside that window determines whether it is one of the q updates. For the rank identity, we condition on the history immediately before an iteration, fixing its ranking. For the waiting-time results, the ranking may change, and we condition repeatedly on each new history.

For orientation, n is the pool size, s is the number of candidates put in the random window, and q is the number of updates retained from that window. The cache is the last observed utility of each prompt; a prompt that is not selected keeps its old cache and timestamp. A "window visit" means that the prompt enters the random size-s window, while a "selection" means that it also survives the top-q ranking. The age bonus increases priority by β for every iteration since the timestamp.

C.1 Windowed rank probabilities

Proof of Lemma B.1. Fix a ranking and call the prompt at rank j prompt i. A uniformly sampled window contains i together with $s-1$ of the other $n-1$ prompts. Let L be the rank of i among the prompts that entered this window. The prompt is selected exactly when $L \le q$. For a particular $\ell \in \{1, \dots, q\}$, we choose $\ell - 1$ of the $j-1$ prompts ranked above i, and $s-\ell$ of the $n-j$ prompts ranked below it. Therefore

$$\Pr(L = \ell) = \frac{\binom{j-1}{\ell-1}\binom{n-j}{s-\ell}}{\binom{n}{s}}.$$

Summing this expression over $\ell = 1, \dots, q$ gives the formula for γ_j.

The first-ranked prompt is selected whenever it enters the window, so $\gamma_1 = s/n$. To see monotonicity, couple two ranks by swapping their labels in every window. Replacing a prompt by a better-ranked one can only improve its within-window rank, never worsen it; hence $\gamma_1 \ge \cdots \ge \gamma_n$. Finally, every window contributes exactly q selected prompts. Counting selections once by windows and once by prompts gives $\sum_{j=1}^{n} \gamma_j = q$. If $s = q$, all prompts in the window are selected, so each prompt has probability $s/n = q/n$ and $\gamma_j/q = 1/n$. For completeness, consider a sequence of pools with $j/n \to x$. In the hypergeometric formula, sampling a fixed-size window is asymptotically the same as drawing s independent uniforms on $[0, 1]$. The probability that x is the ℓ-th smallest of these draws has density

$$\frac{s!}{(\ell-1)!(s-\ell)!} x^{\ell-1}(1-x)^{s-\ell}.$$

The prompt is selected when its within-window rank is one of $1, \dots, q$; averaging these q order-statistic densities gives

$$\phi_{s,q}(x) = \frac{1}{q} \sum_{\ell=1}^{q} \frac{s!}{(\ell-1)!(s-\ell)!} x^{\ell-1}(1-x)^{s-\ell}.$$

Each order-statistic density integrates to one, so the mixture does as well. This is the Beta-mixture limit of the normalized rank weights: $n\gamma_j/q \to \phi_{s,q}(x)$ for fixed s, q and interior x. □

Toy example. For $n = 5, s = 3, q = 1$, the best-ranked prompt is selected with probability $3/5$. The fifth-ranked prompt cannot be selected because any other window member outranks it, so its probability is zero. The third-ranked prompt has probability $1/\binom{5}{3} = 1/10$. Thus a uniform candidate window induces a strongly non-uniform update distribution.

C.2 Conditional update and standard baselines

Proof of Proposition B.3. Fix the pre-window history and ranking. Conditional on the selected set, fresh groups have expected directions h_i^t, since selection did not inspect those groups. By linearity of expectation and Lemma B.1,

$$\mathbb{E}[\widetilde{d}_t \mid \mathcal{F}_{t-1}] = \frac{1}{q} \sum_{j=1}^{n} \Pr(\rho_t(j) \in \mathcal{Q}_t \mid \mathcal{F}_{t-1}) h^t_{\rho_t(j)} = \frac{1}{q} \sum_{j=1}^{n} \gamma_j h^t_{\rho_t(j)} = \nabla f_t(\theta_t).$$

The last equality uses the expected per-prompt negative-surrogate losses. This identity is unaffected by stale scores: they only fix the ranking. □

Proof of Proposition B.5. For this fixed-objective minimization baseline, projection onto the convex domain is nonexpansive. Therefore

$$\|\theta_{t+1} - \theta^*\|^2 \le \|\theta_t - \theta^*\|^2 - 2\eta\langle g_t, \theta_t - \theta^*\rangle + \eta^2 \|g_t\|^2.$$

Conditional unbiasedness and the subgradient inequality give $\mathbb{E}\langle g_t, \theta_t - \theta^* \rangle \geq \mathbb{E}[f(\theta_t) - f(\theta^*)]$. Rearrange, take expectations, and sum from 1 to T:

$$2\eta \sum_{t=1}^{T} \mathbb{E}[f(\theta_t) - f(\theta^*)] \leq R_0^2 + \eta^2 T G_0^2.$$

Convexity gives $f(\bar{\theta}_T) \leq T^{-1} \sum_t f(\theta_t)$, proving (54).

For the population-gap statement, denote the uniform empirical loss by $L_n = n^{-1} \sum_i \ell_i$, the population loss by L_{pop}, and the rank-weighted loss by $L_w = q^{-1} \sum_j \gamma_j \ell_{\rho(j)}$. On the stipulated uniform-deviation event,

$$|L_{\text{pop}} - L_w| \leq |L_{\text{pop}} - L_n| + \left| \sum_{j=1}^{n} \left(\frac{1}{n} - \frac{\gamma_j}{q} \right) \ell_{\rho(j)} \right| \leq \epsilon_n(\delta) + B_\ell A_{n,s,q}.$$

The second inequality is pointwise in the ranking, so it remains true for a data-dependent ranking. A uniform-deviation assumption for the relevant loss class is necessary: bounded losses alone do not supply it for adaptive GRPO surrogates. These baselines do not establish convergence to a stationary objective when the behavior policy or ranking changes. □

C.3 Conditional stale-score delay

Proof of Proposition **??**. At each iteration, let H_i be the current prompts whose cached score is strictly larger than c_i. By assumption, $|H_i| \geq \rho n$. In an iteration in which i is still unselected, let $X = |S \cap H_i|$. Since S is a uniform size-s subset, X is hypergeometric and

$$\mathbb{E}[X] = s|H_i|/n \geq s\rho.$$

If $X \geq q$, the top q prompts in the window all have score larger than c_i, so i cannot be selected. Consequently, conditional on survival up to this iteration,

$$\Pr(i \text{ is selected}) \leq \Pr(X < q).$$

For sampling without replacement, Hoeffding's inequality for a finite population of zero-one values gives

$$\Pr(X < q) \leq \Pr(X/s \leq q/s) \leq \exp\{-2s(\rho - q/s)^2\} \leq \exp\{-2s\kappa^2\} =: \pi.$$

Using $X \leq q$ only loosens the bound. Thus the conditional hazard of selection is at most π. Let D_i be the first future iteration at which i is selected. Repeated conditioning, together with independent window draws, yields

$$\Pr(D_i > K \mid \mathcal{F}_a) \geq (1 - \pi)^K.$$

Taking complements and using $1 - (1 - \pi)^K \leq K\pi$ proves the two finite-K bounds in the proposition.

If the competition condition holds forever, use the tail-sum identity for a positive integer-valued waiting time:

$$\mathbb{E}[D_i \mid \mathcal{F}_a] = \sum_{K \geq 0} \Pr(D_i > K \mid \mathcal{F}_a) \geq \sum_{K \geq 0} (1 - \pi)^K = 1/\pi = e^{2s\kappa^2}.$$

For a cached-zero prompt, H_i contains exactly the prompts with strictly positive cached scores. Negative cached values are lower than zero and cannot block it; this is why the theorem is conditional on the positive competition fraction rather than on the total fraction of negative scores. □

C.4 Age-controlled observation and conditional selection

Proof of Theorem 3.4. The target belongs to a uniform size-s window with probability $p_s = s/n$, independently at each iteration. Hence its next window visit V_i is geometric and

$$\Pr(V_i > K \mid \mathcal{F}_a) = (1 - p_s)^K, \qquad \mathbb{E}[V_i \mid \mathcal{F}_a] = 1/p_s.$$

If the target has not been selected since a reference time t_0, so its timestamp is at most t_0, scores lie in $[-B_u, B_u]$, and therefore another prompt can have at most $2B_u$ more score priority than i. Let

$h = \lceil 2B_u/\beta \rceil + 1$. The purpose of this burn-in is that, after $t_0 + h$, any competitor selected at or after $t_0 + h$ is younger than the target by at least h iterations at every later target visit; its age term is therefore smaller by more than $2B_u$, regardless of its score.

At any target-containing window, if fewer than q competitors have priority larger than the target, the top-q rule selects the target. The age gap of $h = \lceil 2B_u/\beta \rceil + 1$ dominates the score range for competitors whose timestamps are no later than the target reference time. The stronger charging argument is invalid without controlling competitors selected in windows that do not contain the target: those competitors can receive fresh timestamps before every target visit and block it repeatedly.

The geometric window-visit calculation at the start of this proof gives the valid observation bound. Thus age guarantees an opportunity to refresh a cache, while a deterministic selection horizon requires an extra competitor-refresh assumption or forced exploration. □

Toy example. For $n = 100, s = 20, B_u = 1, \beta = 0.1$, the theorem gives $h = 21$. A prompt enters a candidate window with probability 0.2 per step, so the expected time to an observation is five iterations. This is a distribution-free observation bound, not a claim that the prompt wins the top-q ranking every time.

C.5 Persistent useful states

Proof of Corollary 3.5. Let V be the event that the target is revisited before its useful state disappears, and let O be the event that a fresh rollout fails to identify the state. The age theorem implies $\Pr(V^c) \leq \delta_{\text{visit}}$ whenever the state persists for at least $W_{\delta_{\text{visit}}} = h + K_{\delta_{\text{visit}}}$. The observation assumption gives $\Pr(O \cap V) \leq \delta_{\text{obs}}$, including an adaptively chosen revisit time. Discovery fails only on $V^c \cup (O \cap V)$, so the union bound gives

$$\Pr(\text{discovery}) \geq 1 - \Pr(V^c) - \Pr(O \cap V) \geq 1 - \delta_{\text{visit}} - \delta_{\text{obs}}.$$

If persistence can fail, its failure event contributes an additional δ_{state} by the same union bound. If m independent rollout groups are used and one group misses with probability p, then the miss term can be replaced by p^m; this is the usual way to instantiate δ_{obs}. The corollary does not claim that utility remains high forever or that a discovered prompt is always selected afterward. □