ISAR: CALIBRATING HISTORICAL ADVANTAGES FOR EFFICIENT TRAJECTORY REUSE

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning improves language models through repeated generation and policy optimization, but autoregressive rollouts make training expensive. Reusing historical trajectories reduces generation demand, yet aggressive reuse can impair learning when their reward-relative signals become poorly matched to the evolving policy. We introduce ISAR, which recalibrates advantages by estimating the current policy's prompt-specific expected reward from fresh and historical trajectories. It combines full-trajectory, self-normalized importance weighting, candidate exclusion, and a circular replay buffer with a token-level clipped actor objective. Our analysis characterizes historical advantage evolution under policy improvement and shows how mixed-group averaging attenuates the response to that improvement. In search-agent training with Qwen3.5-9B, ISAR obtains 46.0 average exact match versus 44.5 for GRPO, using 60.5% fewer new tokens and 50.2% less total training time at the same update horizon. It further reduces these costs by 18.2% and 12.3% relative to the most token-efficient and fastest strong baselines, respectively. Full-parameter DeepSeek-V4-Flash training on operations research tasks reduces generation by 54.9% and time by 38.7%, with similar aggregate quality. Component controls show that calibration recovers quality lost under uncalibrated replay, supporting current-reference estimation as a practical component of efficient reuse.

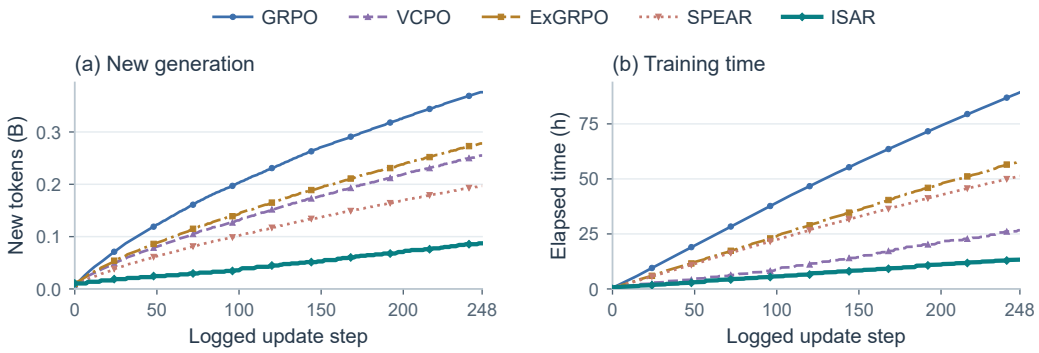


Figure 1: **Generation and elapsed training cost.** Qwen3.5-9B DAPO-Math records: (a) cumulative newly generated tokens and (b) elapsed time in hours, against the recorded update index.

1 INTRODUCTION

Reinforcement learning (RL) has become an important component of language-model post-training, improving instruction following and verifiable reasoning (Ouyang et al., 2022; Shao et al., 2024; DeepSeek-AI, 2025). These improvements require repeated cycles of generation and optimization. Standard autoregressive decoding evaluates the model sequentially for each output token (Leviathan et al., 2023); long responses incur substantial computation and memory traffic, and decoding can be limited by memory bandwidth (Pope et al., 2023; Zhong et al., 2024). In agentic tasks, rollouts

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.

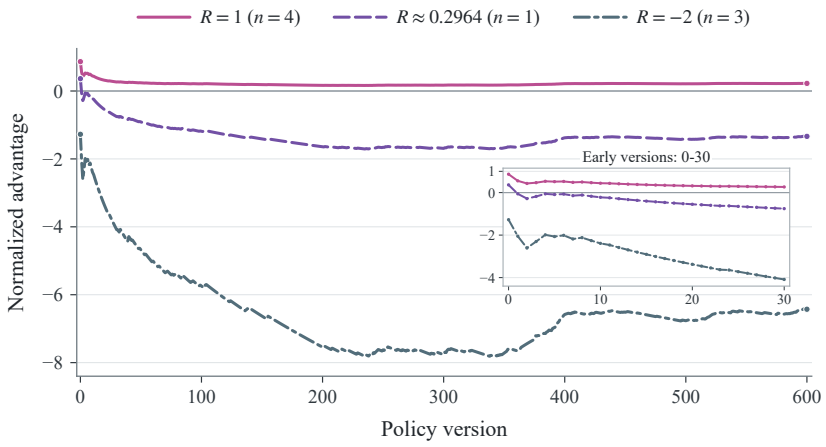


Figure 2: **Advantage evolution of fixed historical responses.** At each training checkpoint, fresh responses to the same prompts are mixed with an unchanged historical cohort at a fixed ratio. GRPO advantages are recomputed using the mean and standard deviation of each mixed group and summarized separately for successful and unsuccessful historical responses.

additionally involve tool calls and external observations. Obtaining more learning progress from each newly generated trajectory is therefore a central efficiency problem.

Advantage estimation connects these costly trajectories to policy improvement. Actor–critic methods learn value estimates (Konda & Tsitsiklis, 1999), while proximal policy optimization (PPO) commonly combines such estimates with generalized advantage estimation and a clipped surrogate (Schulman et al., 2017; 2016). For outcome-supervised language-model training, group relative policy optimization (GRPO) retains a PPO-style update but replaces the learned critic with rewards from responses to the same prompt (Shao et al., 2024). RLOO excludes each candidate from its reward baseline (Ahmadian et al., 2024). These critic-free designs tie the reward reference to the composition of sampled data; GRPO's additional standardization also changes the relative weighting of prompts (Liu et al., 2025).

Recent work reduces training cost through asynchronous execution, repeated optimization, and historical-data reuse. AReaL and LlamaRL coordinate generation and learning under policy lag (Fu et al., 2025; Wu et al., 2025). ReMix controls mixed-policy updates (Liang et al., 2025), and ExGRPO combines one historical response with $K - 1$ fresh responses to form an experiential group (Zhan et al., 2026). SPEAR recalibrates historical advantages using recent group baselines (Qin et al., 2025); VCPO combines a minimum-variance baseline with effective-sample-size-based learning-rate adaptation (Huang et al., 2026). These advances make reuse practical, but reducing fresh generation does not ensure that the reused data continue to support effective learning. Aggressive reuse can degrade quality or destabilize training. A remaining question is how to maintain a useful reward reference when historical trajectories dominate the available responses.

Our starting point is a fixed-cohort observation(Figure 2). We track the same historical answers across checkpoints, generate fresh answers to their prompts, and recompute the historical subset's standardized GRPO advantages at a fixed mixing ratio. In the observed regime, both successful and unsuccessful historical answers show declining signed advantages: positive signals weaken, while negative signals become more negative. This concerns a fixed subset, whose mean can change even though each full comparison group is centered. Appendix D.7 specifies the measurement, while Section 2 separates reward-reference evolution from group standardization.

We formalize the observation through a current-policy reward reference. For a fixed answer, $A^{\star}_{i,t} = R_i - V_t(x)$ decreases whenever the current sampling policy's expected reward $V_t(x)$ improves on that prompt. More consequentially for reuse, a mixed historical–fresh mean responds to this improvement only in proportion to its fresh fraction. Its deviation from the current reference can therefore grow as reuse increases. We characterize this bias–variance tradeoff and distinguish

it from the effects of standardization. The argument is conditional on prompt-wise improvement: practical training can regress, and signed failure advantages do not simply decay toward zero. A fixed exponential schedule cannot generally express these dynamics.

We introduce **ISAR**, which couples circular trajectory reuse with current-reference advantage recalibration (Figure 3). Full-trajectory likelihood ratios estimate a prompt-specific reward mean under a frozen current-policy snapshot through self-normalized importance sampling. Candidate exclusion prevents a response and its replayed copies from entering their own reference. The resulting advantage follows this estimated reference rather than a prescribed decay curve. The asynchronous architecture lets rollout workers supply complete trajectories while the learner revisits available partitions, retaining immutable behavior probabilities. Scheduling reduces dependence on new generation; calibration addresses the changing interpretation of reused outcomes. The actor keeps a conventional token-level clipped objective.

We evaluate this design in Search-R1-style agent training with Qwen3.5-9B and in full-parameter OR training with DeepSeek-V4-Flash. At the same update horizon, ISAR reduces total new tokens and elapsed training time by 60.5% and 50.2% relative to GRPO in the agent task, while increasing mean exact match from 44.5 to 46.0. It achieves lower costs than all compared methods, although SPEAR has the highest mean accuracy. The OR comparison supports effectiveness at a larger model scale. Four component controls show that the buffer supplies generation savings, while calibration recovers the quality lost by replay alone.

Our contributions are: (i) a conditional analysis of historical advantage evolution and mixed-reference error; (ii) a current-reference calibration method integrated with circular replay and a conventional clipped actor; and (iii) empirical quality–cost comparisons against strong baselines, with a larger-model validation and a controlled separation of replay and calibration. Extended related work appears in Appendix A.

2 Historical Advantage Evolution under Trajectory Reuse

2.1 Problem Setup and Reward References

We first consider a fixed prompt x, a complete response y, and a fixed outcome reward $R(x, y)$. At outer iteration t, $\bar{\pi}_t^s(\cdot \mid x)$ denotes a frozen snapshot of the current sampling distribution; the superscript s includes the actual decoding and stopping rules. A historical response y_i was sampled from μ_i^s and retains its recorded reward R_i. We define the current reward reference and the associated unstandardized advantage as

$$V_t(x) = \mathbb{E}_{Y \sim \bar{\pi}_t^s(\cdot|x)} R(x, Y), \qquad A_{i,t}^\star = R_i - V_t(x). \tag{1}$$

This population reference differs from both the population mean of a response's birth policy and the finite-group baseline recorded when it was generated. It provides a measurable interpretation of a historical response relative to the current policy, without requiring a learned critic.

GRPO instead constructs a finite-group reference from the responses available for a prompt (Shao et al., 2024). For a group G, its outcome-based advantage has the form

$$Z_i(G) = \frac{R_i - \widehat{b}(G)}{\widehat{\sigma}(G) + \epsilon_{\mathrm{std}}}, \qquad \widehat{b}(G) = \frac{1}{|G|} \sum_{j \in G} R_j. \tag{2}$$

Our trend measurement follows the same historical responses while changing only their fresh comparison data. The measurement uses the full-group statistic in Equation (2); the training baseline in Section 3 uses candidate exclusion. Keeping these constructions distinct is necessary for interpreting the observation.

2.2 Reference Evolution under Policy Improvement

For a fixed response and reward, the change in reference advantage is exactly

$$A_{i,t}^\star - A_{i,s}^\star = -\big(V_t(x) - V_s(x)\big). \tag{3}$$

Thus, improvement on the same prompt decreases the signed advantage of every fixed response to that prompt. With binary rewards, a successful response has advantage $1 - V_t(x)$, whereas an

unsuccessful response has advantage $-V_t(x)$. The latter becomes more negative as the reference improves; its magnitude does not decay toward zero.

Equation (3) is conditional, not a guarantee that every prompt improves during neural optimization. An explicit sufficient mechanism is the prompt-wise regularized update

$$p_u(y \mid x) \propto p(y \mid x)e^{uR(x,y)}, \qquad \frac{d}{du}\mathbb{E}_{p_u}R(x,Y) = \mathrm{Var}_{p_u}(R(x,Y)) \geq 0. \tag{4}$$

For bounded rewards and $u > 0$, p_u maximizes $\mathbb{E}_q R - u^{-1}D_{\mathrm{KL}}(q\|p)$ over the support of p; differentiating its log normalizer gives the variance identity (Appendix B.2; Geist et al., 2019). This identifies a reward-improvement path along which fixed historical advantages decrease. Shared parameters, finite data, and clipped updates do not implement it independently for every prompt. ISAR therefore estimates the reference at each stage and permits it to increase or decrease.

2.3 Mixed-Group References and Normalization

Let H be a fixed historical cohort containing n_h responses with mean reward q_H. At a checkpoint, generate $n_f \geq 1$ independent fresh responses from $\bar{\pi}_t^s$ and denote their mean by $\widehat{V}_t^f$. Write $\alpha = n_f/(n_h + n_f)$ and $\widehat{b}_t^{\mathrm{mix}} = (1-\alpha)q_H + \alpha\widehat{V}_t^f$.

Proposition 1 (Response and error of a mixed reference)**.** *Fix x, H, and the current sampling policy, and take expectation only over the fresh responses. If their reward variance is $\sigma_t^2(x) < \infty$, then*

$$\mathbb{E}[\widehat{b}_t^{\mathrm{mix}} \mid H] = (1-\alpha)q_H + \alpha V_t(x), \tag{5}$$

$$\mathbb{E}[(\widehat{b}_t^{\mathrm{mix}} - V_t(x))^2 \mid H] = (1-\alpha)^2(q_H - V_t(x))^2 + \frac{\alpha^2\sigma_t^2(x)}{n_f}. \tag{6}$$

For two checkpoints with the same cohort and mixing ratio, the expected mixed reference changes by $\alpha\big(V_t(x) - V_s(x)\big)$.

Proof sketch. The historical term is constant under the stated conditioning. The fresh mean has expectation $V_t(x)$ and variance $\sigma_t^2(x)/n_f$. Substituting these quantities into the linear mixture gives its bias and variance, whose sum yields Equation (6). Subtracting Equation (5) at two checkpoints gives the response relationship. Appendix B.3 supplies the full derivation and candidate-excluded form. □

When $q_H < V_t(x)$, the mixed mean underestimates the current reward reference in conditional expectation. Reducing the fresh fraction makes it less responsive to current improvement. However, fresh-only estimation also has a cost: its variance is $\sigma_t^2(x)/n_f$, which can be large when few new responses are available. Historical data therefore present an estimation tradeoff. The proposition motivates using their probability information; it does not establish that importance weighting always has lower error than either alternative.

Standardization introduces a separate effect. For a nondegenerate binary-reward group with success fraction m, population-form standard deviation, and $\epsilon_{\mathrm{std}} = 0$, the two possible standardized advantages are

$$Z^+(m) = \sqrt{\frac{1-m}{m}}, \qquad Z^-(m) = -\sqrt{\frac{m}{1-m}}. \tag{7}$$

Both decrease with m. A finite-group version follows by coupling fresh Bernoulli draws at different success probabilities, with the historical cohort and group size held fixed (Appendix B.4). This does not permit replacing the expectation of a normalized advantage by its value at the expected group mean. For graded rewards, changes in standard deviation can reverse the direction suggested by the mean alone. These distinctions explain why the observed trend is informative while a universal advantage-decay rule is too restrictive.

3 ISAR: Importance-Sampling Advantage Recalibration

Figure 3 shows the asynchronous realization of circular replay and advantage recalibration. A frozen reference makes the interpretation of stored rewards explicit, while versioned records preserve the probabilities needed for both reference estimation and actor correction.

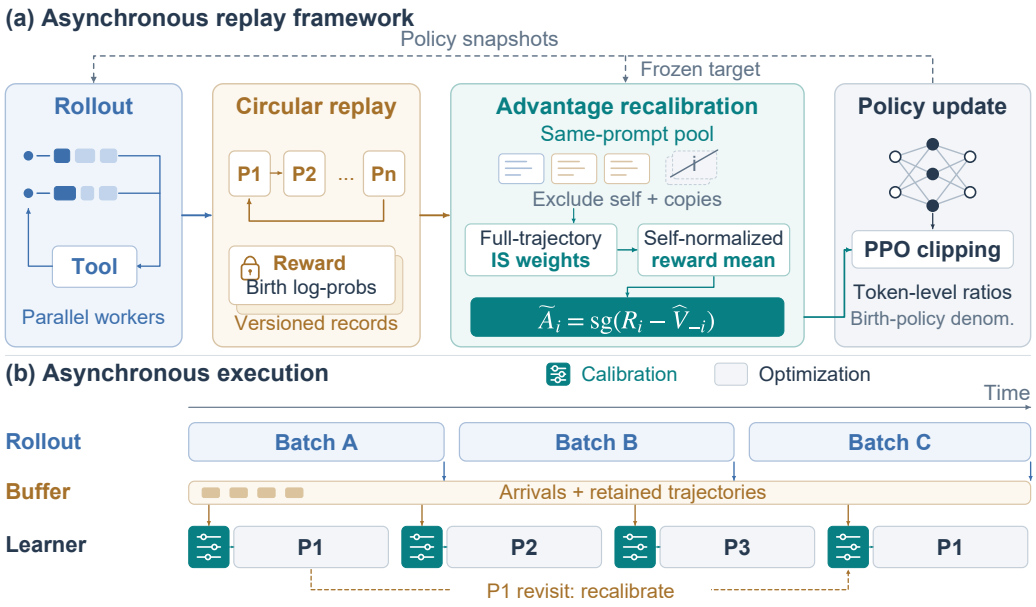


Figure 3: **Asynchronous replay with advantage recalibration.** (a) Rollout workers enqueue complete, versioned trajectories. A frozen target supplies full-trajectory weights for a candidate-excluded reward reference; the actor separately uses token-level ratios against the birth policy. (b) The circular schedule overlaps generation with calibration and optimization. Every revisit uses the current frozen reference; immutable behavior probabilities remain attached to each trajectory. Dashed arrows distribute snapshots. Worker counts, partition counts, and block lengths are schematic.

3.1 Estimating the Current Reward Reference

Let $D_{t,-i}(x)$ be the same-prompt reference pool for candidate i, excluding that candidate and all copies of its original trajectory; a fold-level implementation excludes its entire fold. For reference trajectory j, ISAR computes

$$w_{j,t} = \frac{\bar{\pi}_t^s(y_j \mid x)}{\mu_j^s(y_j \mid x)}, \qquad \log w_{j,t} = \sum_{u \in \mathcal{G}_j} \log \frac{\bar{\pi}_t^s(y_{j,u} \mid h_{j,u})}{\mu_j^s(y_{j,u} \mid h_{j,u})}, \tag{8}$$

where $\mathcal{G}_j$ indexes all policy-generated tokens, including termination. For a single response, $h_{j,u} = (x, y_{j,<u})$; for an interactive episode it also contains the recorded observations. Under a fixed environment, its transition probabilities cancel from the trajectory ratio (Appendix D.9). Tool observations are context, not generated tokens. Log ratios are summed over the trajectory, not averaged over length. The probabilities include the actual sampling transformations. A trajectory has unit weight only when its generating sampler equals the frozen target; newly enqueued asynchronous rollouts need not satisfy this equality.

The self-normalized importance-sampling (SNIS) reference and detached advantage are

$$\widehat{V}_{t,-i}(x) = \frac{\sum_{j \in D_{t,-i}(x)} w_{j,t} R_j}{\sum_{j \in D_{t,-i}(x)} w_{j,t}}, \qquad \widetilde{A}_{i,t} = \mathrm{sg}\Big(R_i - \widehat{V}_{t,-i}(x_i)\Big). \tag{9}$$

SNIS is a classical estimator (Owen, 2013); its role here is to put responses from heterogeneous generators on a common current reward reference. The estimate lies within the reference rewards' range when its denominator is positive, but self-normalization introduces finite-sample bias. Appendix B.5 gives the error identity, moment conditions, and support assumptions. In particular, birth-policy likelihoods do not automatically correct reward-dependent retention, and long trajectories can concentrate the weights.

Candidate exclusion follows the local baseline-separation principle of RLOO (Ahmadian et al., 2024). We exclude duplicate trajectory identities before computing both weighted sums and stop

gradients through all reference statistics. This prevents direct self-inclusion; it does not remove a response's earlier influence on the current network. Numerical evaluation uses log-space normalization (Appendix C.1). No learned risk model or ESS gate is part of the core estimator.

3.2 Calibration and the Actor Objective

Relative to a candidate-excluded mixed mean, the correction can be written

$$\widetilde{A}_{i,t} = R_i - \widehat{b}^{\mathrm{mix}}_{t,-i}(x_i) - \Delta_{i,t}, \quad \Delta_{i,t} = \widehat{V}_{t,-i}(x_i) - \widehat{b}^{\mathrm{mix}}_{t,-i}(x_i). \tag{10}$$

A higher current reference decreases the signed advantage; a lower reference increases it. Thus calibration can follow both progress and regression. The core advantage is unstandardized; matched component controls retain the same remaining advantage processing, separating calibration from the normalization issue identified by Liu et al. (2025).

The actor retains token-level ratios against each trajectory's generation-time model policy:

$$\rho_{i,u}(\theta) = \frac{\pi_\theta(y_{i,u} \mid h_{i,u})}{\mu_i(y_{i,u} \mid h_{i,u})}. \tag{11}$$

With fixed nonnegative reduction weights $c_{i,u}$ summing to one over included tokens, the PPO-style reward objective (Schulman et al., 2017) is

$$\mathcal{J}_{\mathrm{reward}}(\theta) = \sum_{i,u} c_{i,u} \min\Big\{ \rho_{i,u}\widetilde{A}_{i,t}, \mathrm{clip}(\rho_{i,u}, 1-\epsilon_-, 1+\epsilon_+)\widetilde{A}_{i,t} \Big\}. \tag{12}$$

The underlying recipe's masks, loss reduction, and any KL term are fixed in the component controls. Baseline weights $w_{j,t}$ estimate an outcome expectation, whereas actor ratios $\rho_{i,u}$ define a local training surrogate. Their distinct roles mean that recalibration does not make the token surrogate an exact full-trajectory policy gradient.

The unstandardized advantage error satisfies

$$\widetilde{A}_{i,t} - A^\star_{i,t} = V_t(x_i) - \widehat{V}_{t,-i}(x_i). \tag{13}$$

Let $\delta_i = \widehat{V}_{t,-i}(x_i) - V_t(x_i)$ and $s_{i,u} = \nabla_\theta \log \pi_\theta(y_{i,u} \mid h_{i,u})$. For the same fixed data, actor, and reduction weights, the reward gradients with estimated and exact references satisfy, away from clipping ties,

$$\|G(\widehat{V}) - G(V)\| \le \left(\sum_{i,u} c_{i,u}\delta_i^2 \right)^{1/2} \left(\sum_{i,u} c_{i,u}\rho_{i,u}^2 \|s_{i,u}\|^2 \right)^{1/2}. \tag{14}$$

The scalar clipped-gradient coefficient is continuous and $\rho_{i,u}$-Lipschitz in the advantage, including sign changes; weighted Cauchy–Schwarz gives the bound. Appendix B.6 provides the proof. The reference error matters through a factor that may grow for stale data, but this is not a convergence result. An independent baseline cancels in an ideal unclipped, exactly corrected expected gradient, and expected reward need not minimize gradient variance (Greensmith et al., 2004). Controlled comparisons must establish calibration's practical value.

3.3 Partitioned Circular Replay

The buffer stores complete trajectories, prompts, rewards, immutable generation-time probabilities, sampler metadata, and trajectory and policy-version IDs. Rollout workers enqueue completed records while the learner traverses available optimization partitions in cyclic order. The target snapshot is fixed within a learner round, and reference scores are refreshed for its scheduled reuses. Optimization partitions schedule computation; cross-fitting folds identify excluded data and need not coincide. Algorithm 1 in Appendix C specifies this flow and its implementation requirements.

Frozen-target scores are cached by target-policy and trajectory IDs and refreshed when the target changes. Rescoring known tokens can use batched teacher forcing, but still incurs model, memory, and communication cost. Savings arise when avoided sequential generation and environment interaction outweigh rescoring and replay overhead. We measure total elapsed training time at the same update horizon, alongside newly generated tokens; overlapping asynchronous operations are not added together as if they were serial costs.

4 EXPERIMENTS

We study whether calibrated trajectory reuse reduces the cost of reinforcement learning while preserving task quality. Our main experiment inserts ISAR into an established search-agent training loop. We then extend the evaluation to full-parameter training of DeepSeek-V4-Flash on operations research (OR) tasks. A compact component study separates the contribution of the replay buffer from that of advantage calibration.

4.1 EXPERIMENTAL SETUP

Training tasks and baselines. The main experiment uses Qwen3.5-9B in text-only thinking mode (Qwen Team, 2026). We follow Search-R1's NQ and HotpotQA training protocol and fixed local Wikipedia search environment, using a Qwen3.5-compatible veRL implementation (Jin et al., 2025b;c; veRL Contributors, 2026). We evaluate exact match (EM) on the seven-task suite and report the equally weighted task average. Comparisons include fresh-only GRPO (Shao et al., 2024), SPEAR (Qin et al., 2025), ExGRPO (Zhan et al., 2026), ReMix (Liang et al., 2025), and VCPO (Huang et al., 2026). Methods share the initialization, task interface, external evaluator, and resource allocation; their defining selection, reward-shaping, baseline, and update rules are retained. Model and task adaptations are documented in Appendix D.2.

Quality and resource measurements. We report held-out quality and total training cost at a common optimizer-update horizon B, with the same hardware allocation within each experiment. Cumulative new training tokens $N^{\mathrm{new}}(B)$ and elapsed time $T(B)$ cover the entire run. Generation includes all newly generated thinking, tool-request, and answer tokens, including buffer initialization and discarded attempts. The training-time boundary additionally covers rescoring, environment execution, replay management, optimization, and synchronization. Evaluation-only work is a separate accounting category. These measurements compare complete training runs, not first crossings of a common quality target. Reporting details appear in Appendix D.1.

4.2 EFFICIENT AGENTIC RL WITH TRAJECTORY REUSE

Integration into the training loop. The agent alternates between reasoning, issuing a search query, observing retrieved passages, and producing an answer. ISAR replaces part of the fresh rollout batch with complete historical episodes from the circular buffer, then recalibrates their advantages before the policy update. Each episode retains its original model inputs, generated tokens, observations, reward, and behavior probabilities. Importance weights cover policy-generated tokens across all turns; retrieved passages provide context and are masked out. The actor objective retains its separate token-level importance ratios.

Task quality and training efficiency. Table 1 separates final task quality from total training cost. ISAR reaches 46.0 mean EM, compared with 44.5 for GRPO and 46.6 for the highest-scoring baseline, SPEAR. At the common update horizon, ISAR uses 0.435×10^9 new tokens and 49.8 hours, versus 1.101×10^9 tokens and 100.0 hours for GRPO: 60.5% fewer new tokens, 50.2% less training time, and a $2.01\times$ total-time ratio. The results show a quality–cost tradeoff: ISAR improves on GRPO's average but remains below SPEAR's average.

For a comparator m, the reductions in new generation and elapsed time are

$$\rho_N^{(m)} = 1 - \frac{N_{\mathrm{ISAR}}^{\mathrm{new}}(B)}{N_m^{\mathrm{new}}(B)}, \qquad \rho_T^{(m)} = 1 - \frac{T_{\mathrm{ISAR}}(B)}{T_m(B)}. \tag{15}$$

Among the strong baselines, ReMix has the lowest new-token cost (0.532×10^9), whereas VCPO has the lowest elapsed time (56.8 hours). ISAR reduces these costs by a further 18.2% and 12.3%, respectively. These comparisons use different reference methods and concern total training costs. They do not establish a time-to-quality advantage. Quality and cost are reported jointly in Table 1; the DAPO-Math records in Figure 1 belong to a separate task and do not replace these measurements.

Table 1: **Task quality and training efficiency under the Search-R1 task protocol.** Qwen3.5-9B, with thinking enabled for every method. Final EM and total costs use the common update horizon B and hardware allocation. Avg. is the seven-task mean; ΔEM is its difference from GRPO, computed before rounding the mean. New tokens include thinking, tool requests, and answers. Savings are $100(1 - N_m/N_{\mathrm{GRPO}})$ and $100(1 - T_m/T_{\mathrm{GRPO}})$; speedup is the total-time ratio T_{GRPO}/T_m. Bamboogle uses 125 questions.

(a) Final task quality: exact match (%)↑

	In-domain		Cross-dataset generalization					
Method	NQ	HotpotQA	TriviaQA	PopQA	2Wiki	MuSiQue	Bamboogle	Avg.
GRPO	40.8	45.0	69.1	41.8	36.6	20.7	57.6	44.5
SPEAR	40.9	46.9	70.4	43.2	38.6	24.6	61.6	46.6
ExGRPO	41.8	46.3	70.4	43.0	38.3	23.3	59.2	46.0
ReMix	40.6	46.4	71.0	43.0	36.8	21.9	58.4	45.4
VCPO	40.7	46.0	69.9	42.8	38.5	25.7	60.0	46.2
ISAR	40.9	46.5	69.8	43.2	39.4	25.2	56.8	46.0

(b) Final quality difference and total training cost

Method	ΔEM (pp)	New tokens (10^9)↓	Token saving (%)↑	Time (h)↓	Time saving (%)↑	Speedup ↑
GRPO	0	1.101	0.0	100.0	0.0	1.00×
SPEAR	+2.1	0.893	18.9	86.0	14.0	1.16×
ExGRPO	+1.5	0.764	30.6	85.5	14.5	1.17×
ReMix	+0.9	0.532	51.7	64.8	35.2	1.54×
VCPO	+1.7	0.938	14.8	56.8	43.2	1.76×
ISAR	+1.5	0.435	**60.5**	49.8	50.2	**2.01**×

Table 2: **Full-parameter OR reinforcement learning at scale.** GRPO and ISAR continue training from the same CPT plus Clean-CoT SFT checkpoint of DeepSeek-V4-Flash. Quality is measured at the common RL horizon using zero-shot greedy Pass@1; Avg. is the equal-weight benchmark mean. Costs cover the entire RL stage under the same update budget and hardware allocation, excluding shared CPT/SFT preparation. Speedup is the total-time ratio relative to GRPO.

	Final benchmark quality (%)↑					Total RL cost		
RL method	NL4OPT	OptiBench	B4O-Feas.	B4O-ORG.	Avg.	New tokens (10^9)↓	Time (h)↓	Speedup ↑
GRPO	92.4	71.2	76.4	61.4	75.4	0.881	240.0	1.00×
ISAR	91.9	71.1	78.9	61.4	75.8	**0.397**	147.1	**1.63**×

4.3 LARGE-SCALE VALIDATION ON OR TASKS

Common initialization and terminal feedback. We initialize every DeepSeek-V4-Flash run from the same CPT plus Clean-CoT SFT checkpoint described in the SLAI technical report (AI Training Platform Team, 2026). All model parameters are updated during the subsequent OR-RL stage. Each rollout generates one complete solver program; execution then supplies a terminal reward,

$$R(x, y) = 2\,\mathbf{1}\{\text{the program solves the instance correctly}\} - 1. \tag{16}$$

A compilable but incorrect program receives -1. Correctness includes the required output contract and task-specific solution checks. This is a single-response solver-verified task, extending the multi-turn search evaluation to a different interaction structure and model scale. Training instances are separated from evaluation instances and their derived variants.

Table 3: **Roles of replay and reference calibration.** The four configurations share the remaining advantage processing and actor update. Final quality and total generation use the common training horizon. The two fresh-only configurations match GRPO; both replay configurations use the same new-token budget.

Configuration	Buffer	Calibration	Final EM (%)↑	Total new tokens (10^9)↓
Fresh-only matched control	No	No	44.5	1.101
Replay only	Yes	No	36.8	0.435
Calibration only	No	Yes	44.5	1.101
Full ISAR	Yes	Yes	**46.0**	0.435

Evaluation at scale. We follow the report's zero-shot, greedy Pass@1 evaluation on NL4OPT, OptiBench, B4O-Feasible, and B4O-ORGEval (Ramamonjison et al., 2023; Yang et al., 2025; Wang et al., 2025; AI Training Platform Team, 2026). Solver correctness and formulation-graph matching are distinct evaluation criteria. Table 2 compares fresh-only GRPO and ISAR after continued RL from the shared trained checkpoint, with the same update horizon and resource allocation. The RL cost boundary starts from that checkpoint; The shared CPT and SFT preparation is excluded from the RL costs.

ISAR obtains a reported mean score of 75.8, compared with 75.4 for GRPO. It uses 0.397×10^9 new tokens and 147.1 hours, versus 0.881×10^9 tokens and 240.0 hours: reductions of 54.9% and 38.7%, with a $1.63\times$ total-time ratio. Task-level changes are mixed: B4O-Feasible increases from 76.4 to 78.9, while NL4OPT and OptiBench decrease by 0.5 and 0.1 points; B4O-ORGEval is unchanged. These point estimates support lower total RL cost with similar reported aggregate quality. Implementation details appear in Appendix D.3.

4.4 Ablation Study

Table 3 compares four configurations on the Qwen3.5-9B search task. Replay-only uses the same buffer and candidate-excluded reference pool with unit baseline-estimation weights; calibration-only uses fresh episodes. All configurations share thinking mode, remaining advantage processing, candidate exclusion, actor objective, and optimizer settings.

The two fresh-only controls match GRPO at 44.5 mean EM and 1.101×10^9 new tokens. Replay alone reduces generation to 0.435×10^9 tokens, but its reported mean EM is approximately 20% lower than full ISAR (about 36.8 versus 46.0). Adding calibration recovers quality at the same replay budget. Calibration alone leaves quality and generation unchanged in the fresh-only setting. These controls support calibration's contribution under reuse (Appendix D.4).

5 Discussion and Conclusion

ISAR treats historical advantage calibration as estimation of a moving, prompt-specific reward reference. Under explicit improvement conditions, fixed historical outcomes receive decreasing signed reference advantages, while a mixed-group mean responds only through its fresh component. Full-trajectory importance weighting supplies an alternative reference, and candidate exclusion separates each response from its own local estimate. The experiments show substantial reductions in total generation and elapsed training time, with quality close to the strongest baselines; the replay ablation isolates calibration's practical contribution.

Limitations. The analysis does not guarantee monotone neural-policy improvement or convergence. Long trajectories, insufficient support, outcome-dependent buffer selection, and adaptive reuse can undermine reference estimation; cross-fitting does not remove a sample's past influence on the target model. Nor does reward-mean accuracy imply a variance-optimal policy gradient.

AI USE STATEMENT

Generative AI was used for language polishing. All content was reviewed and verified by the authors, who take full responsibility for the paper.

ETHICS STATEMENT

The method seeks to reduce the generation and computation required for language-model RL. Lower training cost can improve access to model development, but can also reduce the cost of adapting models for harmful uses. Search-agent correctness and solver-verified OR performance do not establish factual reliability, fairness, or safe deployment in open environments. Generated solver programs require isolated execution with appropriate resource and access limits. Dataset and model licenses, benchmark contamination, and rights to redistribute derived trajectories must be checked for any release. Reported token and elapsed-time savings characterize these training configurations; they are not direct measurements of energy use or environmental impact.

REPRODUCIBILITY STATEMENT

Section 2 specifies the conditioning used in the analysis, with complete derivations in Appendix B. Section 3 and Algorithm 1 define the reference estimator, actor objective, and replay flow; Appendix C details probability semantics, exclusions, and numerical evaluation. Appendix D describes the task interfaces, component controls, cost accounting, and source-specific reference results. Reported averages and savings are computed from the supplied table entries, including explicit Bamboogle counts.

REFERENCES

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting REINFORCE style optimization for learning from human feedback in LLMs, 2024. URL https://arxiv.org/abs/2402.14740v2.

AI Training Platform Team. SLAI T-Rex: Full-parameter post-training of the DeepSeek-V4 family on Ascend SuperPOD. Training Platform Technical Report Part I, Shenzhen Loop Area Institute, 2026. Author-supplied report; reference values from Tables 12–13.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Wei Fu, Jiaxuan Gao, Xujie Shen, Chen Zhu, Zhiyu Mei, Chuyi He, Shusheng Xu, Guo Wei, Jun Mei, Jiashu Wang, Tongkai Yang, Binhang Yuan, and Yi Wu. AReaL: A large-scale asynchronous reinforcement learning system for language reasoning, 2025. URL https://arxiv.org/abs/2505.24298.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized markov decision processes. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2160–2169. PMLR, 2019. URL https://proceedings.mlr.press/v97/geist19a.html.

Evan Greensmith, Peter L. Bartlett, and Jonathan Baxter. Variance reduction techniques for gradient estimates in reinforcement learning. *Journal of Machine Learning Research*, 5:1471–1530, 2004. URL https://jmlr.org/papers/v5/greensmith04a.html.

Luke J. Huang, Zhuoyang Zhang, Qinghao Hu, Shang Yang, and Song Han. Stable asynchrony: Variance-controlled off-policy RL for LLMs, 2026. URL https://arxiv.org/abs/2602.17616v2.

Bowen Jin. Search-R1 released NQ/HotpotQA training data and seven-task evaluation bundle. Hugging Face dataset: PeterJinGo/nq_hotpotqa_train, 2025. URL https://huggingface.co/datasets/PeterJinGo/nq_hotpotqa_train. Split sizes and source counts accessed 22 September 2026.

Bowen Jin, Jinsung Yoon, Priyanka Kargupta, Sercan O. Arik, and Jiawei Han. An empirical study on reinforcement learning for reasoning-search interleaved LLM agents, 2025a. URL https://arxiv.org/abs/2505.15117.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan O. Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-R1: Training LLMs to reason and leverage search engines with reinforcement learning, 2025b. URL https://arxiv.org/abs/2503.09516.

Bowen Jin et al. Search-R1: Native training code, versioned recipes, and experiment logs, 2025c. URL https://github.com/PeterGriffinJin/Search-R1. The v0.2 GRPO recipe and experiment-log documentation; accessed 22 September 2026.

Vijay R. Konda and John N. Tsitsiklis. Actor-critic algorithms. In *Advances in Neural Information Processing Systems*, volume 12, pp. 1008–1014, 1999. URL https://proceedings.neurips.cc/paper_files/paper/1999/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19274–19286. PMLR, 2023. URL https://proceedings.mlr.press/v202/leviathan23a.html.

Siheng Li, Zhanhui Zhou, Wai Lam, Chao Yang, and Chaochao Lu. RePO: Replay-enhanced policy optimization, 2025. URL https://arxiv.org/abs/2506.09340v1.

Jing Liang, Hongyao Tang, Yi Ma, Jinyi Liu, Yan Zheng, Shuyue Hu, Lei Bai, and Jianye Hao. Squeeze the soaked sponge: Efficient off-policy reinforcement finetuning for large language model, 2025. URL https://arxiv.org/abs/2507.06892v3.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-Zero-like training: A critical perspective, 2025. URL https://arxiv.org/abs/2503.20783v2.

Yu Luo, Shuo Han, Yihan Hu, Dong Li, and Jianye Hao. Ratio-variance regularized policy optimization for efficient LLM fine-tuning, 2026. URL https://arxiv.org/abs/2601.03320v1.

Luca Martino, Víctor Elvira, and Francisco Louzada. Effective sample size for importance sampling based on discrepancy measures. *Signal Processing*, 131:386–401, 2017. doi: 10.1016/j.sigpro.2016.08.025. URL https://arxiv.org/abs/1602.03572.

Junhyuk Oh, Yijie Guo, Satinder Singh, and Honglak Lee. Self-imitation learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3878–3887. PMLR, 2018. URL https://proceedings.mlr.press/v80/oh18b.html.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022. URL https://arxiv.org/abs/2203.02155.

Art B. Owen. *Monte Carlo Theory, Methods and Examples*. Author-published online book, 2013. URL https://artowen.su.domains/mc/. Chapter 9: Importance sampling.

Hyun Bin Park and Du-Seong Chang. Headroom-drift replay: A primitive for principled replay control in GRPO, 2026. URL https://arxiv.org/abs/2609.03941v1.

Reiner Pope, Sholto Douglas, Aakanksha Chowdhery, Jacob Devlin, James Bradbury, Anselm Levskaya, Jonathan Heek, Kefan Xiao, Shivani Agrawal, and Jeff Dean. Efficiently scaling transformer inference. In *Proceedings of Machine Learning and Systems*, volume 5, 2023. URL https://proceedings.mlsys.org/paper_files/paper/2023/hash/c4be71ab8d24cdfb45e3d06dbfca2780-Abstract-mlsys2023.html.

Yulei Qin, Xiaoyu Tan, Zhengbao He, Gang Li, Haojia Lin, Zongyi Li, Zihan Xu, Yuchen Shi, Siqi Cai, Renting Rui, Shaofei Cai, Yuzheng Cai, Xuan Zhang, Sheng Ye, Ke Li, and Xing Sun. Learn the ropes, then trust the wins: Self-imitation with progressive exploration for agentic reinforcement learning, 2025. URL https://arxiv.org/abs/2509.22601.

Qwen Team. Qwen3.5-9B model card, 2026. URL https://huggingface.co/Qwen/Qwen3.5-9B. Official post-trained checkpoint and model documentation; accessed 22 September 2026.

Rindranirina Ramamonjison, Timothy Yu, Raymond Li, Haley Li, Giuseppe Carenini, Bissan Ghaddar, Shiqi He, Mahdi Mostajabdaveh, Amin Banitalebi-Dehkordi, Zirui Zhou, and Yong Zhang. NL4Opt competition: Formulating optimization problems based on their natural language descriptions. In *Proceedings of the NeurIPS 2022 Competitions Track*, volume 220 of *Proceedings of Machine Learning Research*, pp. 189–203, 2023. URL https://proceedings.mlr.press/v220/ramamonjison23a.html.

John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *International Conference on Learning Representations*, 2016. URL https://arxiv.org/abs/1506.02438.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL https://arxiv.org/abs/1707.06347v2.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Eric Veach and Leonidas J. Guibas. Optimally combining sampling techniques for monte carlo rendering. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '95, pp. 419–428. ACM, 1995. doi: 10.1145/218380.218498. URL https://graphics.stanford.edu/papers/combine/.

veRL Contributors. Add models with the FSDP backend, 2026. URL https://github.com/verl-project/verl/blob/main/docs/advance/fsdp_extension.rst. Official training and rollout compatibility documentation; accessed 22 September 2026.

Xu Wan, Yansheng Wang, Wenqi Huang, and Mingyang Sun. Buffer matters: Unleashing the power of off-policy reinforcement learning in large language model reasoning, 2026. URL https://arxiv.org/abs/2602.20722v2.

Zhuohan Wang, Ziwei Zhu, Ziniu Li, Congliang Chen, Yizhou Han, Yufeng Lin, Zhihang Lin, Angyang Gu, Xinglin Hu, Ruoyu Sun, and Tian Ding. ORGEval: Graph-theoretic evaluation of LLMs in optimization modeling, 2025. URL https://arxiv.org/abs/2510.27610.

Bo Wu, Sid Wang, Yunhao Tang, Jia Ding, Eryk Helenowski, Liang Tan, Tengyu Xu, Tushar Gowda, Zhengxing Chen, Chen Zhu, Xiaocheng Tang, Yundi Qian, Beibei Zhu, and Rui Hou. LlamaRL: A distributed asynchronous reinforcement learning framework for efficient large-scale LLM training, 2025. URL https://arxiv.org/abs/2505.24034v2.

Fengkai Yang, Zherui Chen, Xiaohan Wang, Xiaodong Lu, Jiajun Chai, Guojun Yin, Wei Lin, Shuai Ma, Fuzhen Zhuang, Deqing Wang, Yaodong Yang, Jianxin Li, and Yikun Ban. Your group-relative advantage is biased, 2026. URL https://arxiv.org/abs/2601.08521v1.

Zhicheng Yang, Yiwei Wang, Yinya Huang, Zhijiang Guo, Wei Shi, Xiongwei Han, Liang Feng, Linqi Song, Xiaodan Liang, and Jing Tang. OptiBench meets ReSocratic: Measure and improve LLMs for optimization modeling. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2407.09887.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale, 2025. URL https://arxiv.org/abs/2503.14476v2.

Runzhe Zhan, Yafu Li, Zhi Wang, Xiaoye Qu, Dongrui Liu, Jing Shao, Derek F. Wong, and Yu Cheng. ExGRPO: Learning to reason from experience. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2510.02245.

Hongzhi Zhang, Jia Fu, Jingyuan Zhang, Kai Fu, Qi Wang, Fuzheng Zhang, and Guorui Zhou. RLEP: Reinforcement learning with experience replay for LLM reasoning, 2025. URL https://arxiv.org/abs/2507.07451v1.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization, 2025. URL https://arxiv.org/abs/2507.18071v2.

Yinmin Zhong, Shengyu Liu, Junda Chen, Jianbo Hu, Yibo Zhu, Xuanzhe Liu, Xin Jin, and Hao Zhang. DistServe: Disaggregating prefill and decoding for goodput-optimized large language model serving. In *18th USENIX Symposium on Operating Systems Design and Implementation (OSDI 24)*, pp. 193–210. USENIX Association, 2024. URL https://www.usenix.org/conference/osdi24/presentation/zhong-yinmin.

A Extended Related Work

Advantage estimation and policy optimization. Actor–critic algorithms use value estimation to support policy updates (Konda & Tsitsiklis, 1999). PPO supplies a clipped surrogate and a multiple-epoch optimization procedure, with GAE frequently used to construct advantages from a learned value function (Schulman et al., 2017; 2016). For LLM post-training, GRPO replaces a separately trained critic with a within-prompt group reference, while RLOO uses the other responses to form each candidate's baseline (Shao et al., 2024; Ahmadian et al., 2024). These designs establish the foundations of our estimator and actor objective. ISAR's question concerns how a current reward reference can be estimated when many available responses were generated by earlier policies. It does not claim novelty for subtracting a baseline, omitting a critic, or excluding a candidate from a local mean.

Grouping, historical references, and reward normalization. RePO estimates advantages separately for fresh and replay groups (Li et al., 2025); ExGRPO constructs an experiential group from one historical response and $K-1$ fresh responses (Zhan et al., 2026). SPEAR already recalibrates historical advantages using a median of recent group baselines (Qin et al., 2025). HA-DW studies group-relative estimation and uses historical difficulty information to alter advantage weights (Yang et al., 2026). These are direct precedents for treating historical advantage signals as quantities that merit reconsideration. Our target is specifically $\mathbb{E}_{\bar{\pi}_t^s} R(x, Y)$, estimated with trajectory likelihoods from a potentially heterogeneous set of generators. The relevant comparison is whether this target and estimator improve calibration and training under a matched reference-data budget. Analyses of GRPO normalization (Liu et al., 2025) further motivate separating the reward mean from the standard deviation and from length-dependent loss reduction.

Selecting and reusing trajectories. Self-imitation learning uses an agent's own successful experience as a learning signal (Oh et al., 2018). More recent LLM methods explore historical-data replay, group construction, and repeated optimization. ReMix combines data from different policy stages while controlling the resulting updates (Liang et al., 2025); RLEP explores successful-experience reuse (Zhang et al., 2025); and BAPO emphasizes buffer management and reuse of useful historical experience (Wan et al., 2026). Headroom-Drift Replay ranks stored groups by remaining learning value and checks their compatibility with the current policy (Park & Chang, 2026). Selection and recalibration are related but distinct decisions. A circular buffer is an implementation component of ISAR, while its scientific evaluation concerns the current-reference estimate and its incremental benefit under controlled reuse.

Update control and execution efficiency. R^2VPO relates policy-ratio variance to update regularization and applies the resulting objective to data reuse (Luo et al., 2026). Its regularized gradient can itself be expressed through an adjusted advantage coefficient. Thus, "adjusting advantages" does not distinguish ISAR. The distinction is the estimation of a prompt-specific current expected reward, rather than a ratio-variance penalty on the actor objective. VCPO addresses heavy-tailed importance weights using an ESS-dependent learning rate and a minimum-variance off-policy baseline (Huang et al., 2026). Its variance criterion differs from our target of the current prompt-specific expected reward; the two baselines need not coincide. GSPO changes the granularity and normalization of actor likelihood ratios (Zheng et al., 2025); its actor weighting is not interchangeable with the full-density ratio used to estimate our reward mean. DAPO shows that sampling, clipping, and loss reduction also affect large-scale RL training (Yu et al., 2025). AReaL and LlamaRL address asynchronous execution and policy lag (Fu et al., 2025; Wu et al., 2025). These works motivate reporting both algorithmic data demand and realized system costs.

Importance weighting and the meaning of a baseline. Importance sampling and self-normalized ratio estimation are established statistical tools (Owen, 2013). Multiple-proposal methods further show how samples from several known distributions can be combined (Veach & Guibas, 1995). We use these foundations rather than presenting them as new estimators. Weight-based effective sample size is a useful concentration diagnostic but is not an exact count of independent training information (Martino et al., 2017). Finally, classical variance-reduction analysis distinguishes a value reference from a variance-optimal baseline (Greensmith et al., 2004). This distinction is important here: the

scientific claim must rest on measured calibration, controlled training effects, and practical reuse efficiency, not solely on the availability of an importance-sampling identity.

B Proofs and Additional Analysis

B.1 Notation and Conditioning

The conditioning differs between the observational and estimation analyses. Proposition 1 treats the historical cohort as a realized, fixed collection and averages over newly generated responses. The importance-sampling calculations below instead average over draws from specified logging policies with a fixed target. Neither calculation by itself proves unbiasedness for an arbitrary realized buffer whose contents also influenced the target network.

Table 4: Notation for reward references and policy updates.

Symbol	Meaning
x, y_i, R_i	Prompt, fixed response or complete transcript, and fixed outcome reward.
μ_i, μ_i^s	Generation-time model policy and its actual sampling distribution.
$\bar{\pi}_t, \bar{\pi}_t^s$	Frozen current model snapshot and current target sampler.
$V_t(x)$	Current target sampler's expected outcome reward.
$b_k(x), \widehat{b}_{k,i}(x)$	A birth policy's population mean and a finite birth-group estimate, respectively.
$A_{i,t}^\star$	Unstandardized reference advantage $R_i - V_t(x)$.
H, q_H, n_h, n_f, α	Fixed historical cohort, its mean reward, historical and fresh sizes, and fresh fraction.
$D_{t,-i}(x)$	Candidate-excluded or fold-excluded reference pool for prompt x.
$w_{j,t}, \rho_{i,u}$	Full-response baseline density ratio and token-level actor ratio.
$\widehat{V}_{t,-i}, \widetilde{A}_{i,t}$	SNIS reference estimate and detached training advantage.
n	Number of optimization partitions, distinct from group size and number of cross-fitting folds.

B.2 A Sufficient Policy-Improvement Mechanism

Fix x and suppress it in the notation. Let p be a reference policy and let the reward be bounded, so the following normalizing constant and its derivatives exist. For $u \geq 0$, define

$$Z(u) = \mathbb{E}_{Y\sim p} e^{uR(Y)}, \qquad p_u(y) = \frac{p(y)e^{uR(y)}}{Z(u)}. \tag{17}$$

For $u > 0$, this is the maximizer, over distributions supported on p, of

$$\mathbb{E}_{Y\sim q} R(Y) - \frac{1}{u} D_{\mathrm{KL}}(q\|p). \tag{18}$$

Indeed, substituting $\log p_u = \log p + uR - \log Z$ rewrites Equation (18) as $u^{-1}\log Z(u) - u^{-1}D_{\mathrm{KL}}(q\|p_u)$, maximized at $q = p_u$. This is a one-prompt specialization of regularized policy improvement (Geist et al., 2019).

Differentiating the log normalizer gives $d\log Z(u)/du = \mathbb{E}_{p_u} R$. Differentiating once more yields

$$\frac{d}{du}\mathbb{E}_{p_u} R = \mathbb{E}_{p_u} R^2 - (\mathbb{E}_{p_u} R)^2 = \mathrm{Var}_{p_u}(R) \geq 0. \tag{19}$$

For a fixed historical outcome R_i, $d(R_i - \mathbb{E}_{p_u} R)/du = -\mathrm{Var}_{p_u}(R)$. Strict decrease requires nonzero reward variance along the path. This derivation specifies a sufficient mechanism; it does not identify the actual Adam or clipped-GRPO iterate with an exact mirror update. It also preserves the reference policy's support and does not establish discovery of unsupported responses.

B.3 Mixed-Reference Response and Mean-Squared Error

Write the fresh rewards as $F_1, \ldots, F_{n_f}$, conditionally independent with common mean V_t and variance σ_t^2. Then

$$\widehat{b}_t^{\mathrm{mix}} - V_t = (1-\alpha)(q_H - V_t) + \alpha(\widehat{V}_t^f - V_t). \tag{20}$$

The second term has conditional expectation zero. Squaring and taking expectation removes the cross term, and the variance of the fresh mean is σ_t^2/n_f. This proves Equations (5)–(6). Subtracting the mean expression at checkpoints s, t with the same H and α proves the claimed response coefficient. No independence between the two checkpoint measurements is needed for this difference of expectations.

The fresh-only estimate has MSE σ_t^2/n_f. When $0 \le \alpha < 1$, mixed averaging has lower conditional MSE precisely when

$$(q_H - V_t)^2 < \frac{1+\alpha}{1-\alpha}\frac{\sigma_t^2}{n_f}. \tag{21}$$

Equality gives a tie. This condition clarifies why an old cohort can help when its mean remains close to the current value and hurt when the discrepancy grows. It also makes a fresh-only comparator necessary when evaluating a more elaborate estimator.

For a historical candidate i, leave-one-out replaces H by $H \setminus \{i\}$ and α by $\alpha_{-i} = n_f/(n_h + n_f - 1)$. The same calculation applies whenever the reference set is nonempty. If only one historical response was present, excluding it leaves a fresh-only estimate. For a fresh candidate, the independent fresh reference count decreases to $n_f - 1$; the finite-sample formulas must use that count rather than the original group size. A reference containing no independent fresh responses has no fresh-noise term and cannot respond to a fixed target change through newly sampled rewards.

For a dynamic queue, let q_t and α_t denote the historical reward mean and fresh fraction at time t. The conditional expected mixed reference $m_t = (1-\alpha_t)q_t + \alpha_t V_t$ satisfies

$$\begin{aligned} m_t - m_s = {} & \alpha_t(V_t - V_s) + (1-\alpha_t)(q_t - q_s) \\ & + (\alpha_t - \alpha_s)(V_s - q_s). \end{aligned} \tag{22}$$

The terms isolate current-policy change, historical-cohort turnover, and a change in mixing ratio. This is why a fixed-cohort probe and a training-buffer trace answer different questions.

B.4 Binary Standardization and a Graded-Reward Counterexample

For binary rewards, a group with empirical success fraction $m \in (0,1)$ has population-form standard deviation $s(m) = \sqrt{m(1-m)}$. More generally, let its denominator be $d(m) = c\,s(m) + \epsilon$ with $c > 0$ fixed by the standard-deviation convention and $\epsilon \ge 0$. A fixed group size $N > 1$ gives $c = 1$ for divisor N and $c = \sqrt{N/(N-1)}$ for divisor $N-1$. The two advantages are $z^+(m) = (1-m)/d(m)$ and $z^-(m) = -m/d(m)$, with derivatives

$$\frac{dz^+}{dm} = -\frac{c(1-m)/(2s(m)) + \epsilon}{d(m)^2} < 0, \tag{23}$$

$$\frac{dz^-}{dm} = -\frac{cm/(2s(m)) + \epsilon}{d(m)^2} < 0. \tag{24}$$

Equation (7) is the case $c = 1, \epsilon = 0$.

Proposition 2 (Finite-group binary trend)**.** *Fix a historical cohort containing both reward values, a fresh count n_f, and the denominator convention above. Suppose the fresh rewards are independent Bernoulli draws with parameter p. The expected standardized advantage of every fixed successful or unsuccessful historical response is nonincreasing in p.*

Proof. Let $U_1, \ldots, U_{n_f}$ be independent uniform variables and construct $F_j(p) = \mathbf{1}\{U_j \le p\}$. For $p' \ge p$, every $F_j(p') \ge F_j(p)$, so the mixed-group success fraction is no smaller under p'. The historical cohort contains both outcomes, ensuring $m \in (0,1)$ for every fresh draw. Both advantage functions are decreasing by Equation (24), yielding a pointwise ordering under this coupling and hence the expectation ordering. □

Equivalently, if h of the n_h historical responses succeed, the exact expectation is

$$\mathbb{E}[Z_i \mid H, p] = \sum_{k=0}^{n_f} \binom{n_f}{k} p^k (1-p)^{n_f-k} z^{R_i} \left(\frac{h+k}{n_h+n_f} \right). \tag{25}$$

This finite sum is generally different from evaluating z^{R_i} at the expected success fraction. All-equal groups require the implemented zero-variance convention; with a positive denominator constant their full-group advantages are zero. Filtering such groups changes the conditioning and must be analyzed separately.

The conclusion does not extend to arbitrary graded rewards from the mean alone. Consider a fixed historical candidate with reward 1. The group $(1, 0, 0, 1)$ gives mean $1/2$, population-form standard deviation $1/2$, and candidate advantage 1. The group $(1, 1/2, 1/2, 1/2)$ gives mean $5/8$, standard deviation $\sqrt{3}/8$, and candidate advantage $\sqrt{3}$. The mean increased while this candidate's standardized advantage also increased. These are analytic counterexample values, not empirical results. A fixed prompt distribution is similarly necessary: improving the aggregate mean across prompts does not imply improvement for every prompt.

B.5 What Self-Normalized Importance Sampling Establishes

Fix a target distribution p, fixed logging distributions $q_1, \ldots, q_m$, and independent draws $Y_j \sim q_j$. Assume each q_j covers the target support and that all stated second moments are finite. Write $w_j = p(Y_j)/q_j(Y_j)$ and $V = \mathbb{E}_p R$. The ordinary estimate $\widehat{V}^{\mathrm{OIS}} = m^{-1} \sum_j w_j R_j$ obeys

$$\mathbb{E}\widehat{V}^{\mathrm{OIS}} = V, \qquad \mathrm{Var}(\widehat{V}^{\mathrm{OIS}}) = \frac{1}{m^2} \sum_{j=1}^{m} \mathrm{Var}_{q_j}(w_j R_j). \tag{26}$$

Each mean follows by integrating $q_j(p/q_j)R$; independence makes the variances add. These are the usual importance-sampling calculations, here written for heterogeneous loggers (Owen, 2013).

For the self-normalized estimate, define $\bar{w} = m^{-1} \sum_j w_j$ and $H_m = m^{-1} \sum_j w_j (R_j - V)$. Whenever $\bar{w} > 0$,

$$\widehat{V}^{\mathrm{SNIS}} - V = \frac{H_m}{\bar{w}}. \tag{27}$$

The numerator has expectation zero under these assumptions, but its random denominator is generally dependent on it. Zero mean in the numerator therefore does not prove zero mean of the ratio. On the event $\bar{w} \geq 1/2$, its absolute error is at most $2|H_m|$.

For completeness, suppose $R \in [a, b]$, let $L = b - a$, and assign any value in $[a, b]$ if all weights vanish. The normalized estimate otherwise lies in $[a, b]$. Splitting on $\bar{w} \geq 1/2$ and applying Chebyshev's inequality gives

$$\begin{aligned} \mathbb{E}(\widehat{V}^{\mathrm{SNIS}} - V)^2 &\leq \frac{4}{m^2} \sum_j \mathbb{E}_{q_j}\left[w_j^2 (R_j - V)^2 \right] + L^2 \Pr(\bar{w} < 1/2) \\ &\leq \frac{4}{m^2} \sum_j \mathbb{E}_{q_j}\left[w_j^2 (R_j - V)^2 \right] + \frac{4L^2}{m^2} \sum_j \mathrm{Var}_{q_j}(w_j). \end{aligned} \tag{28}$$

Here $\mathbb{E}\bar{w} = 1$ and independence controls both variances. The bound is deliberately conservative. Uniformly bounded second moments make it vanish as m grows; large or infinite moments need not. It offers no ordering against the fresh-only or mixed estimator without further information about the distributions.

Clipping and selection. Replacing w by $u = \min(w, c)$ changes the estimator. For one logger, let $\tau = \mathbb{E}(w - u) < 1$. Its population normalized reference $V_c = \mathbb{E}[uR]/\mathbb{E}[u]$ satisfies

$$|V_c - V| = \frac{|\mathbb{E}[(w-u)(V-R)]|}{1-\tau} \leq \frac{L\tau}{1-\tau}. \tag{29}$$

This is a possible bias effect, not the weight rule in Equation (9). Similarly, selecting only successes can eliminate information about the probability of failure; a ratio against the original generator does not automatically correct that selection. All-success reference data produce a binary SNIS value of one regardless of their positive weights. Calibration cannot create missing outcome diversity.

Multiple proposal distributions. Another classical construction samples fixed counts m_k from known policies q_k, with $m = \sum_k m_k$, and weights by p/q_{mix} where $q_{\text{mix}} = \sum_k (m_k/m) q_k$ (Veach & Guibas, 1995). Its ordinary estimate is unbiased under the fixed-proposal assumptions and mixture support because

$$\sum_k \frac{m_k}{m} \mathbb{E}_{q_k}\left[\frac{p(Y)R(Y)}{q_{\text{mix}}(Y)}\right] = \int p(y)R(y)\,dy. \tag{30}$$

This alternative may require scoring responses under several historical models. It is a statistical comparator, not a claimed component of the present implementation. Its computational cost and available logger likelihoods must be matched in any experimental comparison.

Adaptive training. The fixed-target independence assumptions do not follow merely from naming a snapshot an anchor. A response may already have influenced that snapshot through earlier gradient updates. Excluding the response from its local SNIS sum does not remove this influence. Repeated copies of a trajectory are also not independent logging draws. The fixed-target calculation is useful as a controlled diagnostic, while the actual adaptive-training estimate requires direct measurement of its behavior. A disjoint evaluation reference measures current estimation error without claiming to remove past dependence.

B.6 Ideal Baseline Cancellation and Practical Sensitivity

For an idealized whole-response policy gradient, fix θ, a behavior distribution μ covering π_θ, and a baseline $B(x)$ independent of the candidate conditional on x. Assuming differentiation under the integral is valid,

$$\mathbb{E}_{Y\sim\mu}\left[\frac{\pi_\theta(Y \mid x)}{\mu(Y \mid x)} B(x) \nabla_\theta \log \pi_\theta(Y \mid x)\right] = \mathbb{E}[B(x) \mid x] \nabla_\theta \int \pi_\theta(y \mid x) dy = 0. \tag{31}$$

This states why a stale independent baseline alone need not bias an exactly corrected, unclipped expected gradient. Value accuracy and variance optimality are separate questions (Greensmith et al., 2004).

Now hold fixed the actual candidate data, actor parameters, reduction weights, and denominator policies in Equation (12). Away from clipping ties, let $s_{i,u} = \nabla_\theta \log \pi_\theta(y_{i,u} \mid h_{i,u})$. The gradient contribution for a scalar advantage a is

$$g_{i,u}(a) = \rho_{i,u} a\, s_{i,u} \mathbf{1}\{(a \geq 0 \ \wedge \ \rho_{i,u} < 1 + \epsilon_+) \ \vee \ (a < 0 \ \wedge \ \rho_{i,u} > 1 - \epsilon_-)\}. \tag{32}$$

Clipping is sign-dependent: an upper-ratio violation suppresses a positive-advantage contribution, while a lower-ratio violation suppresses a negative one. It does not suppress every contribution outside the ratio interval.

Proposition 3 (Reference sensitivity of a fixed clipped update)**.** *Let $\delta_i = \widehat{V}_{t,-i}(x_i) - V_t(x_i)$ and let $G(\widehat{V})$ and $G(V)$ be the reward gradients using estimated and population references on the same data. For fixed nonnegative $c_{i,u}$ summing to one, and away from clipping ties,*

$$\|G(\widehat{V}) - G(V)\| \leq \left(\sum_{i,u} c_{i,u} \delta_i^2\right)^{1/2} \left(\sum_{i,u} c_{i,u} \rho_{i,u}^2 \|s_{i,u}\|^2\right)^{1/2}. \tag{33}$$

The same unchanged differentiable KL term cancels in the difference.

Proof. For a fixed ρ, the scalar coefficient multiplying s in Equation (32) is continuous at $a = 0$ and piecewise linear, with slopes either zero or ρ. It is therefore ρ-Lipschitz, including when the advantage changes sign. Equation (13) yields $\|g_{i,u}(\widetilde{A}_i) - g_{i,u}(A_i^\star)\| \leq \rho_{i,u}|\delta_i|\|s_{i,u}\|$. The triangle inequality and weighted Cauchy–Schwarz inequality give Equation (33). □

The comparator $G(V)$ is the same practical token surrogate with an exact reward reference, not necessarily the true current-policy gradient. The bound neither proves a smaller realized difference for every estimator nor converts an unweighted MSE improvement into a guaranteed training gain. Its weighting factor can be large for stale data. It also assumes that masks and sampling weights are held fixed; baseline-dependent gates or prioritization require additional terms.

B.7 Interpreting Generation Savings and Total Cost

For a simplified serial cost model at the same optimizer-update horizon, let generation comprise a fraction γ of a GRPO update's cost. Let α be the fresh fraction at an equal token workload and let h be the additional calibration overhead relative to the original update cost. Under comparable lengths and fixed per-unit costs,

$$\frac{T_{\mathrm{ISAR}}(B)}{T_{\mathrm{GRPO}}(B)} \approx (1-\gamma) + \alpha\gamma + h. \tag{34}$$

This explanatory model shows why nominal replay fraction does not translate directly into elapsed-time savings. Rescoring and optimization remain costly. Asynchronous overlap, variable response lengths, environment latency, and startup costs require direct measurement. Different update counts needed to reach a target quality would introduce another factor, which is not estimated by the current fixed-horizon comparison.

C Algorithm and Implementation Details

Algorithm 1 ISAR with asynchronous circular replay

Require: Initial policy; rollout workers; circular buffer $\mathcal{B}$ with n partitions; reference-pool, optimization, publication, and replacement settings.
Rollout workers (concurrent): Generate and score under assigned snapshots; enqueue complete records with immutable behavior probabilities, rewards, contexts, and trajectory IDs.

for each learner round t **do**
  Freeze the reference target sampler $\bar{\pi}_t^s$.
  for each available scheduled partition in cyclic order **do**
    Form prompt-indexed pools; rescore under the frozen target.
    Exclude each candidate's identity or fold; compute Equation (9).
    Update the actor with Equation (12), retaining birth-policy denominators.
  end for
  Advance the buffer and publish snapshots on the configured schedules.
  Record generation, rescoring, optimization exposure, and elapsed time.
end for

C.1 Stable Evaluation of the Reference

For each reference response, the input to the estimator is a reward and a pair of complete-response log probabilities. These probabilities must correspond to the same prompt, tokenization, emitted tokens, and stopping event. Define $\ell_{j,t} = \log \bar{\pi}_t^s(y_j \mid x) - \log \mu_j^s(y_j \mid x)$. For a valid candidate-excluded pool, let $a_{t,-i} = \max_{j \in D_{t,-i}} \ell_{j,t}$. Then Equation (9) is evaluated as

$$u_{j,t} = \exp(\ell_{j,t} - a_{t,-i}), \qquad \widehat{V}_{t,-i} = \frac{\sum_{j \in D_{t,-i}} u_{j,t} R_j}{\sum_{j \in D_{t,-i}} u_{j,t}}. \tag{35}$$

Subtracting a common maximum changes neither normalized weights nor the resulting baseline. It avoids exponentiating large positive log weights without imposing an additional statistical clipping threshold. Adding a constant to every reward shifts this reference by the same constant and leaves the unstandardized advantages unchanged. Scaling all rewards by a constant scales those advantages by that constant.

Exclusion is applied before computing the normalizer. If one trajectory dominates the weights, computing a full-pool sum and subtracting its contribution can lose precision. Direct masked reduction or a stable fold-excluded reduction preserves the intended calculation. An empty pool, invalid probability, or all-zero target support does not define a valid ratio estimate.

C.2 Probability Semantics

The superscript s is substantive. A temperature-scaled or truncated sampler has a different normalized conditional distribution from the unmodified model softmax. The target and behavior probabilities in the reward-reference estimator use the actual sampling distributions. The actor ratio follows

the training implementation's model-policy convention, as in Equation (11). If the sampler and model coincide, the distinction disappears; otherwise it is recorded rather than silently ignored.

Full-response weights multiply all generated-token conditional ratios, including the appropriate termination event. Length normalization, geometric-mean ratios, and isolated prefix ratios are not equivalent estimators of the same outcome expectation. A prefix weight can be paired with an appropriately defined conditional target, but cannot replace the full density ratio of an arbitrary terminal reward without further justification. Responses labeled fresh have unit baseline weight only when their actual generating sampler equals the frozen target; asynchronous generation may violate that equality.

If top-k or top-p truncation assigns zero behavior probability to responses having positive target probability, the support condition in Appendix B.5 fails. Numerical handling cannot restore missing support.

C.3 Cross-Fitting and Trajectory Identity

A trajectory is identified by its original sampling event, not by its latest occurrence in an optimization batch. Every replayed copy inherits that identity. The exclusion group for candidate i contains all occurrences of that identity; a fold-level implementation also excludes all identities in the same fold. Folds are constructed independently of the candidate's current reward and advantage, and the rule is fixed before the local baseline is computed.

The exclusion applies to the complete local estimation pipeline. In the core estimator this means the weighted reward sum and its normalizing sum. If a normalized variant is studied, it also applies to the scale. Any added local statistic, combination coefficient, or gate must follow the same rule if a cross-fitted interpretation is claimed. This requirement does not add a gate to the core method.

Optimization partitions and cross-fitting folds serve different purposes. A partition is the scheduled portion of buffer data processed by the learner. A fold identifies information excluded from a candidate's reference. Equating them is one possible implementation, but changes the available reference sample count and should not be assumed implicitly. Likewise, excluding a candidate from a pool that helped train the anchor does not create a statistically independent anchor.

C.4 Circular Storage and Optimization Partitions

The buffer is organized as circular storage and divided into approximately equal partitions. Rollout workers enqueue complete records while the learner traverses available partitions in cyclic order. Each stored record carries its prompt ID, response or transcript, reward, generation-time policy ID, behavior log probabilities, decoding configuration, original trajectory ID, and optimization-use count. Current-target scoring is associated with a separate target-policy ID and is refreshed when that ID changes. The learner's reference snapshot and the snapshots assigned to workers can differ; neither overwrites a trajectory's birth probabilities.

The birth log probabilities remain unchanged when a trajectory reenters optimization. The use count records optimization exposure rather than the number of times a reference calculation reads the record. These two forms of reuse have different costs. Prompt IDs index the data available for reward-reference estimation, while the configured reference-pool scope determines whether that data are local to the active partition or gathered across partitions.

The buffer capacity, partition count, fresh-to-historical ratio, number of optimization passes, and replacement cadence are different parameters. A partition count of n does not by itself imply that generation is reduced by $1 - 1/n$, nor that every response receives n independent learning opportunities.

C.5 Training Objective and Controlled Variants

For token averaging, $c_{i,u} = 1/\sum_i L_i$ on included tokens. For equal response averaging, $c_{i,u} = 1/(BL_i)$ for a batch of B included responses. Different masks or fresh/replay coefficients induce corresponding normalized weights. These examples specify how to read Equation (12). A sequence-average denominator in the loss does not change the full-response baseline weight.

Each actor denominator is the policy that generated the candidate, even after repeated uses. The frozen current sampler supplies the baseline numerator. During an outer iteration, the actor may move away from that sampler. Baseline values, target scores, behavior scores, and rewards are detached from the gradient. The same detachment, masks, reduction, and KL settings are used when only the reward reference is varied.

The replay pair in Table 3 differs only in the weights used to estimate the reward reference. Candidate exclusion, the reference pool, actor importance ratios, and buffer policy remain matched. The fresh-only pair follows the corresponding no-buffer path with all remaining settings fixed. Removing reward-standard-deviation normalization alone does not reproduce every design choice in Dr. GRPO, whose length reduction also matters (Liu et al., 2025). Named external baselines retain their published designs; the unit-weight control is labeled as a local component ablation.

C.6 Diagnostics and Resource Logging

With normalized weights $\bar{w}_j = w_j / \sum_k w_k$, a standard weight-concentration diagnostic is

$$\mathrm{ESS} = \frac{1}{\sum_j \bar{w}_j^2}. \tag{36}$$

This quantity is interpreted as a diagnostic, not an exact count of independent or useful training trajectories (Martino et al., 2017). The maximum normalized weight, log-weight range, number of distinct trajectory IDs, and reward diversity provide complementary information. Repeated copies can inflate a naive weight-based sample count without creating new observations.

D Experimental Protocols and Environments

This appendix defines the reporting and measurement protocol. The shared protocol below specifies the search-agent main experiment, the OR scale-up study, and four component controls. The fixed-cohort observation protocol supports the theory and method discussion; it does not add variants to the primary ablation.

D.1 Quality, Cost, and Run-Level Reporting

The main and OR comparisons use the same optimizer-update horizon B and hardware allocation within each experiment. Final quality and total costs correspond to complete runs at that horizon; development-selected checkpoints are a separate result. Task scores use equal weights, with means and differences computed before display rounding. The current aggregates use the supplied task-score entries. No statistical non-inferiority conclusion is drawn.

The current cost tables report entire-run costs and do not use a common development-quality threshold.

The new-token metric counts policy-generated tokens through the end of training, including warm-up, rejected or truncated rollouts, and baseline-only generation. Retrieved passages and solver output are observations, not generated policy tokens. Replayed tokens count toward optimization exposure. Model-input processing, rescoring, tool execution, buffer work, optimization, and synchronization remain within the wall-clock boundary. Evaluation-only work is a separate accounting category; blocking evaluation contributes to total job time. Device-hours are defined from the device allocation over time; overlapping profiler intervals cannot be summed as elapsed time. An early-terminated run does not represent a completed run at horizon B.

Savings and speedups are ratios of supplied aggregate cost entries, not averages of seed-level ratios. The most time-efficient and token-efficient strong baselines are identified separately. The OR extension compares with fresh-only GRPO. Bamboogle correct counts are 72, 77, 74, 73, 75, and 71 out of 125 for GRPO, SPEAR, ExGRPO, ReMix, VCPO, and ISAR, respectively. These counts determine its EM scores and the updated seven-task aggregates.

D.2 Search-Agent Protocol and Model Adaptation

The policy checkpoint is `Qwen/Qwen3.5-9B`, used with text-only inputs and thinking enabled for training and evaluation (Qwen Team, 2026). Raw generated token IDs preserve thinking, tool requests, and final answers before display parsing. Retrieved observations are masked from the policy loss and generated-token count, while their processing remains part of elapsed time.

We retain Search-R1's task protocol, retrieved-token masking, and corrected group indexing, using a veRL stack whose training and rollout engines support Qwen3.5 (Jin et al., 2025c; veRL Contributors, 2026). The legacy Search-R1 dependency pins are not assumed to support the new architecture.

The original v0.2 script provides starting references: five responses per prompt, an actor learning rate of 10^{-6}, a maximum-turn parameter of four, and top-three retrieval.

Table 5: **Search data inventory.** Release counts are from the Search-R1 dataset metadata accessed on 22 September 2026 (Jin, 2025). Development reserves are protocol assignments. The remaining training pool has at most 167,615 questions before any additional deduplication. Counts describe the released bundle, whose constituent sources may use different original split names.

Dataset / split	Questions	Role
Released NQ + HotpotQA training pool	169,615	Before development split / deduplication
NQ development reserve	1,000	Proposed holdout from training source
HotpotQA development reserve	1,000	Proposed holdout from training source
NQ evaluation	3,610	In-domain evaluation
HotpotQA evaluation	7,405	In-domain evaluation
TriviaQA evaluation	11,313	Cross-dataset evaluation
PopQA evaluation	14,267	Cross-dataset evaluation
2WikiMultiHopQA evaluation	12,576	Cross-dataset evaluation
MuSiQue evaluation	2,417	Cross-dataset evaluation
Bamboogle evaluation	125	Small complementary benchmark
Released evaluation total	51,713	Seven-task bundle

All methods use the same Wiki-18 corpus, E5 checkpoint, index, task prompt, answer extraction, and external EM evaluator. Search-R1's outcome-reward recipe and later format-reward studies are separate configurations (Jin et al., 2025a). SPEAR retains self-imitation, reward shaping, and its schedules; ExGRPO retains experience and question selection; ReMix retains its mixed-policy objective and training schedule. VCPO retains its baseline and learning-rate adaptation, with the source of policy lag and its measured distribution stated explicitly. Any component-only adaptation is named as such.

The historical fraction of the update inputs is

$$r = \frac{H}{H + F}, \tag{37}$$

where H and F are historical and fresh episodes. A nonempty candidate-excluded reference pool is required for calibration. Baselines retain valid native replay constructions. The controlled ablation preserves the same prompt schedule, queue partitions, replacement rule, and candidate exclusion within each replay pair.

Stored episodes preserve the exact model input at every turn, observations, all generated thinking and action tokens, immutable behavior probabilities, rewards, and trajectory IDs. Reference weights are accumulated in log space across policy-generated tokens; observations are masked. Candidate copies are excluded from their own reference statistics. Target rescoring uses recorded contexts and a frozen target snapshot with a declared probability convention consistent with decoding temperature and sampling support. Actor importance ratios retain their generation-time denominator. Reference calibration and actor correction are separate operations. Weight concentration, replay age, and repeated optimization-use counts are diagnostic quantities, not gates in the core estimator.

D.3 OR INITIALIZATION, REWARD, AND EVALUATION

The shared initialization is the DeepSeek-V4-Flash CPT plus Clean-CoT SFT checkpoint from AI Training Platform Team (2026). The report describes 10,000 Clean-CoT candidates and 8,881 accepted exported samples; neither count specifies the size of the subsequent OR-RL dataset. OR-RL training instances, development instances, final benchmarks, and their generated derivatives remain disjoint. The DP (data in the problem), DT (data in a table), and DPS (separate problem and data files) interfaces are inherited from the report.

Each policy call emits a complete program. The sandbox executes it with the prescribed solver and input files, then returns one terminal reward as defined in Equation (16). A positive reward requires end-to-end task correctness. Mere parsing, compilation, code execution, or finding a feasible but incorrect solution does not suffice. Contract failure, an incorrect result, or a verifier-defined unsuccessful completion receives -1. Infrastructure outages are separately identified rather than silently reclassified as policy failures. The environment provides no intermediate feedback for a model revision within the same episode. Fresh-only GRPO retains its native advantage construction and stabilizing settings.

The verifier specification fixes the accepted solver status, numerical tolerances, objective direction, handling of infeasible/unbounded instances, timeout behavior, and output contract. Evaluation follows the report's zero-shot greedy Pass@1 protocol on NL4OPT, OptiBench, and the two Bench4Opt views (Ramamonjison et al., 2023; Yang et al., 2025; Wang et al., 2025). Runtime validity, feasibility, objective agreement, and ORGEval graph matching remain separate fields. ORGEval is an operational structural check, not a general proof of mathematical equivalence. Its evaluation score is not silently added to the binary training reward.

The report's CPT/SFT infrastructure and throughput measurements are not RL measurements. The OR comparison uses fresh-only GRPO and ISAR, with full-parameter updates from the same initialization, update horizon, and hardware allocation. The accounting isolates RL-stage cost, with shared CPT/SFT preparation disclosed separately. This comparison establishes the scope for assessing large-model effectiveness relative to GRPO.

D.4 COMPONENT CONTROLS AND STABILITY

The four configurations form a buffer-by-calibration comparison. The no-calibration reference is the candidate-excluded unit-weight mean,

$$\widehat{b}^{\mathrm{unit}}_{t,-i}(x) = \frac{1}{|D_{t,-i}(x)|} \sum_{j \in D_{t,-i}(x)} R_j. \tag{38}$$

For the replay pair, the same records are supplied to this estimator or the full SNIS estimator. The no-replay pair instead contains only newly sampled records. The remaining advantage processing, token masks, actor ratios, loss reduction, optimizer, and update horizon are identical within the component study. Native external methods retain their full recipes in the main experiment.

For the fresh-only limit, the frozen reference target equals the policy that sampled all fresh records, giving unit weights before actor optimization. Calibration is not recomputed against a different moving target midway through this equivalence check. The result concerns the matched fresh-only controls, not arbitrary native GRPO implementations or asynchronous sampling. Independent training runs may still differ through stochastic execution.

The component comparison reports final task quality and total new tokens. The fresh-only control and calibration-only variant match GRPO's mean EM and total generation. Replay-only and full ISAR each use 0.435×10^9 new tokens. This is a relative decrease from full ISAR, not a 20-percentage-point decrease or a decrease relative to GRPO. These endpoint results do not determine when the deterioration emerges or establish a training-collapse event.

D.5 SUPPLEMENTARY RESULTS AND PUBLISHED REFERENCES

Table 1(a) reports the seven-task breakdown in the main experiment.

Table 6: **Historical search references on Qwen2.5-7B-Instruct.**

Published run	NQ	Hotpot	Trivia	PopQA	2Wiki	MuSiQue	Bamboogle	Avg.
Qwen2.5-7B-Instruct; source-specific settings								
Search-R1 (PPO)	39.3	37.0	61.0	39.7	41.4	14.6	36.8	38.5
Search-R1 (GRPO)	42.9	38.6	62.3	42.7	34.6	16.2	40.0	39.6
SPEAR, 300 steps	35.7	46.9	62.7	34.5	43.4	17.2	44.8	40.7
SPEAR, 550 steps	43.7	48.1	66.8	42.8	45.1	21.1	50.4	45.4

The published rows use Qwen2.5-7B-Instruct and retain their source-specific protocols. They document the task lineage and do not provide measured or estimated Qwen3.5-9B results. Search-R1's PPO and GRPO rows are distinguished using its original tables. Published training steps are not converted into token counts or elapsed time, and no historical score or nominal replay fraction supplies the missing ISAR measurements. The structured provenance record retains the source version, model, and table for each filled value.

D.6 Run Configuration Record

Table 7: **Run metadata for reproducing quality and efficiency comparisons.**

Field	Required record
Initialization	Model revision; exact checkpoint; shared pre-RL evaluation
Data and environment	Split IDs; deduplication; retriever or solver/verifier versions; interaction limits
Training	Horizon; batch and update conventions; optimizer; native method components; tuning budget
Replay	Fresh/historical counts; partitions; replacement rule; realized replay age and repeated uses
Evaluation	Final test protocol; sample counts; aggregation rule; seed IDs; checkpoint-selection rule
Resources	Hardware and allocation; training stack; token counters; time boundary; tool resources
Outcomes	Per-seed quality; total costs; completion status; checkpoint records

D.7 Fixed-Cohort Advantage Measurement

The experimental unit is a prompt together with its fixed historical response IDs. A cohort is sampled once and retained for all checkpoints. Stored rewards are evaluated under the same reward function throughout the measurement. The number of historical and fresh responses per prompt, prompt weights in aggregate statistics, and the success/failure subset definitions remain fixed across checkpoints. The cohort is not replaced by the latest contents of the training buffer.

At each checkpoint, the frozen sampling policy generates the specified fresh responses to the same prompts. The combined group supplies the full-group mean and standard deviation in Equation (2). Every historical response receives a newly computed signed advantage, and the success and failure subsets are summarized separately. The measurement records the unstandardized residual $R_i - \widehat{b}_t^{\mathrm{mix}}$, the group standard deviation, and the standardized advantage. This separates reference movement from changing scale.

A prompt-wise current reward estimate provides the conditioning variable for the theory. An aggregate improvement over prompts alone is not sufficient to label every prompt as improving. When stratifying by improvement or difficulty, the stratification is based on a disjoint measurement or a rule declared independently of the measured historical advantage. The reported cohort size and uncertainty refer to prompts or independent runs, rather than treating repeated observations of one response as independent samples.

For binary rewards, the exact finite-sample sum in Equation (25) is an analytic reference for a fixed group design. General-reward measurements instead report the mean and scale terms directly. Any

theoretical line is labeled as an analytic prediction, and empirical confidence intervals are derived only from the corresponding measured runs.

D.8 Total Training Cost Accounting

The reported comparisons use complete runs at the common optimizer-update horizon B, with the same hardware allocation within each experiment. They do not use a first-crossing target quality. The reporting boundary includes buffer initialization, rejected or truncated attempts, fresh data generated solely for baseline estimation, target rescoring, tool execution, optimization, and synchronization. Evaluation-only work is a separate accounting category. Optimizer tokens include repeated uses; new-token counts include a trajectory only when it is generated.

For a resource allocation that varies over time, device-hours are

$$H_m = \frac{1}{3600} \int_0^{T_m} n_m^{\mathrm{GPU}}(s)\, ds. \tag{39}$$

Elapsed time and device-hours differ when allocations differ. Overlapping profiler intervals cannot be summed to reconstruct elapsed time. The main comparisons use ratios of the supplied aggregate costs; averages of per-seed ratios require the underlying independent runs. Interrupted runs must not be treated as cheaper completed runs at horizon B.

The current aggregate table entries cannot establish time-to-quality crossings by interpolation or by assuming identical learning curves.

D.9 Operations Research and Interactive Tasks

The OR extension starts from the shared CPT plus Clean-CoT SFT checkpoint described in Appendix D.3. Its RL episode consists of one complete generated solver program followed by terminal correctness feedback. OR-RL identifies the separate reinforcement-learning training instances; evaluation uses NL4OPT, OptiBench, B4O-Feasible, and B4O-ORGEval. Full-parameter updates are compared from the same initialization.

The multi-turn trajectory factorization below applies to the search-agent experiment. For the single-response OR experiment, it reduces to the probability of the generated program, with solver execution providing the terminal observation and reward.

For an interactive episode, a trajectory comprises policy-generated actions and environment observations. Under fixed environment transitions P, its probability factorizes as

$$p_\pi(\tau \mid x) = p(s_0 \mid x) \prod_{v=0}^{H-1} \pi^s(a_v \mid h_v) P(s_{v+1} \mid h_v, a_v), \tag{40}$$

including the relevant termination event. The trajectory ratio between two policies cancels the unchanged environment factors, leaving a product over policy-generated actions, and over their generated tokens. Observations and tool outputs are conditioned upon rather than treated as tokens sampled by the policy. The factorization requires identical environment dynamics and reward semantics; a changed tool, verifier, or action interface breaks that identification. An episode generated by several policy versions also requires its per-action behavior probabilities rather than a single fictional birth snapshot.