GIBBS-GAP POLICY OPTIMIZATION FOR OFF-POLICY RL WITH VERIFIABLE REWARDS

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning with verifiable rewards often trains language models on responses generated by a different, possibly stale behavior policy. Correcting this mismatch with learner-to-behavior importance weights can introduce high variance for long responses. We introduce Gibbs-Gap Policy Optimization (GGPO), which minimizes a log-mean-exp contrast of KL-adjusted rewards among responses to the same prompt, without learner-to-behavior importance weights. We derive GGPO from KL-regularized policy optimization: on policy, its population loss is exactly the KL-regularized optimality gap, and under suitable coverage the learner's single-response expected reward is at least the behavior policy's best-of-K reward, minus terms for the raw finite-group loss, regularization, and finite training data. We characterize reference refresh through exact Gibbs-target composition and a reward bound for averaged window outputs. This refresh is empirically important: in an ablation, fixed-reference GGPO scores 32.96 on the five-benchmark problem-weighted average, versus 46.58 with the default refresh schedule. Experiments on mathematical reasoning further show that GGPO outperforms GRPO and AGRO under training–inference mismatch and on fixed offline data, while maintaining its performance as controlled policy staleness grows.

1 INTRODUCTION

Reinforcement learning with verifiable rewards (RLVR) improves language-model reasoning using automatic checks of task outcomes (DeepSeek-AI et al., 2025). Group-based methods generate several responses to each prompt and learn from their reward differences (Shao et al., 2024; Yu et al., 2025). These groups also define a natural performance target: sample several responses, select the highest-reward one, and train a learner whose individual responses approach that reward. The difficulty is that the policy generating the group need not be the policy being trained. The behavior policy π_{beh} can lag behind the learner π_θ during asynchronous generation or sequential updates on a rollout batch (Noukhovitch et al., 2025; Zheng et al., 2025a); offline training keeps the sampled responses fixed while the learner changes. Even synchronized policy versions can differ because training and inference use different implementations or numerical precision (MiniMax et al., 2025; Zhong et al., 2026).

Importance sampling corrects for behavior-policy data using learner-to-behavior likelihood ratios $\pi_\theta/\pi_{\mathrm{beh}}$. In long autoregressive responses, these ratios can have high variance. Practical methods clip, truncate, or mask ratio-based contributions (Schulman et al., 2017; Yu et al., 2025; Zhong et al., 2026; Ling Team et al., 2025; Zheng et al., 2025b), introducing choices of granularity and thresholds that trade variance against bias. This motivates objectives that learn directly from behavior-policy response groups while retaining a quantitative connection to the learner's reward.

We introduce Gibbs-Gap Policy Optimization (GGPO), which learns from contrasts in KL-adjusted rewards among responses to the same prompt, without learner-to-behavior importance weights. Under bounded coverage, its finite-group loss controls the gap between the learner's single-response expected reward and best-of-K selection from the behavior policy. Groups of order K suffice, and a uniform generalization bound extends the comparison to finite data.

KL-regularized duality identifies a Gibbs policy with constant adjusted reward. GGPO measures departures from this condition; on policy, its population loss equals the regularized optimality gap. Its finite-group guarantee does not require accurate estimation of this population loss.

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.

Refreshing the reference to the current learner allows successive rounds to build on earlier progress. We characterize exact target composition and bound the reward of averaged outputs under imperfect updates, coverage, and bounded adjusted-reward spread.

Off-policy consistency itself has precedents. AGRO (Tang et al., 2025) minimizes the variance of the same adjusted reward and identifies the same regularized optimum under common support. We compare their objectives and guarantees in Section 2, and evaluate GGPO against GRPO and AGRO under training–inference mismatch, controlled policy staleness, and offline training.

Our contributions are:

- **An off-policy group objective with a direct reward guarantee.** GGPO minimizes the log-mean-exp contrast of KL-adjusted rewards within behavior-policy response groups, without learner-to-behavior importance weights. On policy, its population value is exactly the KL-regularized optimality gap; off policy, coverage bounds that gap. Its raw finite-group loss bounds the gap between the learner's single-response expected reward and the behavior policy's best-of-K reward with explicit group-size dependence, and a uniform empirical bound and oracle inequality extend this to finite data.
- **A characterization of periodic reference refresh.** Exact refresh windows compose: T windows at temperature β implement one Gibbs update at temperature β/T. For imperfect windows, we bound the reward of the averaged window outputs by a fixed-reference term decaying as $1/T$ plus the window losses, under coverage and sampling conditions. An ablation shows that periodic refresh substantially improves GGPO under the default training setting.
- **Evaluation across behavior–learner mismatch regimes.** On five mathematical-reasoning benchmarks and two model sizes, GGPO outperforms GRPO and its closest objective-level comparator AGRO under training–inference mismatch. It holds its accuracy as the synchronization delay grows from $N = 8$ to $N = 128$, while GRPO and AGRO decline. At $N = 128$, token-level TIS improves GRPO, while GGPO achieves higher accuracy under the same synchronization delay. On a fixed offline rollout pool, GGPO has the highest MATH-500 accuracy at every checkpoint from 160 updates onward.

2 Related work

KL-regularized and direct policy objectives. Gibbs variational identities connect KL regularization to exponential tilting (Richemond & Maginnis, 2017; Geist et al., 2019). Direct objectives derived from related policy identities include DPO (Rafailov et al., 2023), InfoNCA/NCA (Chen et al., 2024), REBEL (Gao et al., 2024), and A^*-PO (Brantley et al., 2025). ConSPO (Zhang et al., 2026a) contrasts verified positive and negative responses using length-normalized sequence log probabilities in an InfoNCE-style objective. GGPO instead applies a log-mean-exp gap to KL-adjusted rewards, with behavior and reference policies distinct. Appendix B gives objective-level comparisons.

Group RLVR and behavior–learner mismatch. GRPO (Shao et al., 2024) uses within-group advantages in a PPO (Schulman et al., 2017) surrogate; DAPO (Yu et al., 2025), Dr.GRPO (Liu et al., 2025), and GSPO (Zheng et al., 2025b) modify sampling, normalization, or PPO-style ratio aggregation. MaxRL (Tajwar et al., 2026) derives success-normalized on-policy updates from a maximum-likelihood formulation. Its centered estimator is related to a special binary-reward limit of GGPO's update coefficients, detailed in Appendix A.6. Under behavior–learner mismatch, asynchronous RLHF (Noukhovitch et al., 2025) studies staleness and IMPALA (Espeholt et al., 2018) uses V-trace. In LLM training, Zhong et al. (2026) studies truncated old-trainer-to-rollout correction weights alongside the PPO ratio. Other methods mask ratio-based contributions (Ling Team et al., 2025; Li et al., 2026) or restructure learner-to-behavior ratios (Zhang et al., 2026b). These corrections trade importance-weight variance against clipping, truncation, or masking bias (Huang et al., 2026). GGPO does not evaluate learner-to-behavior correction ratios, while its population guarantee depends on behavior coverage.

In concurrent work, Marek & Ryabinin (2026) center token-level policy scores under the sampling distribution to remove an additional drift term induced by training–inference mismatch. This additive correction can be combined with importance weighting. GGPO derives a response-group objective

from KL-regularized duality and contrasts KL-adjusted rewards using a separately maintained reference. Its update uses likelihoods of sampled responses under the learner and reference, rather than the sampler's next-token distribution.

Objectives without learner-to-behavior importance weighting. PCL/Trust-PCL (Nachum et al., 2017; 2018) optimize squared soft consistency residuals on off-policy trajectories; Kimi k1.5 (Kimi Team et al., 2025) uses squared Gibbs-consistency regression with an iteratively updated reference; and Contrastive Policy Gradient (Flet-Berliac et al., 2024) uses a pairwise quadratic objective under fixed sampling distributions. OAPL (Ritter et al., 2026) regresses an estimated optimal advantage using the behavior policy as its KL anchor, while AsymRE (Arnal et al., 2025) scales positive and negative feedback asymmetrically. DRO (Richemond et al., 2024) applies a squared penalty to the optimality residual while jointly estimating the value offset. Consistency-based objectives without learner-to-behavior correction ratios therefore have several precedents.

Relation to AGRO. AGRO (Tang et al., 2025) is our closest objective-level comparator and an earlier example with an off-policy consistency guarantee. It minimizes the behavior variance of the same KL-adjusted reward; Theorem 2 of Tang et al. (2025) establishes that the regularized optimal policy is the unique global minimizer under common support, shared by GGPO.

GGPO's distinction is the direct alignment of its loss value with the KL-regularized objective. Its exponential form is derived from the exact duality residual: at the population level and when behavior equals learner, the loss equals the regularized optimality gap for every positive candidate policy. Off policy, the connection becomes a coverage-dependent upper bound. AGRO's variance loss, scaled by $1/\beta$, is the local quadratic approximation to this population contrastive loss when adjusted-reward spread is small relative to β (Appendix A.9). GGPO's finite-group loss additionally admits a direct best-of-K reward bound and an empirical generalization bound. The finite-group variance comparison also yields a reward bound in terms of AGRO's population variance objective (Eq. (46)). Appendices A.6 and A.9 give the gradient and finite-group weighting comparisons. The experiments compare the practical consequences under matched settings.

3 From KL regularization to a contrastive loss

We treat a complete response as one action in a contextual bandit. Fix a prompt x, a finite action space $\mathcal{A}$, reward $r(x, a)$, temperature $\beta > 0$, and a reference policy π_{ref} with positive support. Define

$$V_\beta(\pi; x) = \mathbb{E}_{a\sim\pi} r(x, a) - \beta D_{\mathrm{KL}}(\pi\|\pi_{\mathrm{ref}}). \tag{1}$$

Let $V^*_\beta(x) = \max_\pi V_\beta(\pi; x)$ and $\pi^*(\cdot \mid x) = \arg\max_\pi V_\beta(\pi; x)$ over all policies π on $\mathcal{A}$. The unique maximizer is the Gibbs policy

$$\pi^*(a \mid x) = \frac{\pi_{\mathrm{ref}}(a \mid x)e^{r(x,a)/\beta}}{\mathbb{E}_{a'\sim\pi_{\mathrm{ref}}} e^{r(x,a')/\beta}}. \tag{2}$$

The Gibbs variational identity (Richemond & Maginnis, 2017) suggests enforcing a constant adjusted reward $R_\theta(x, a) = r(x, a) - \beta \log[\pi_\theta(a \mid x)/\pi_{\mathrm{ref}}(a \mid x)]$ across responses to a prompt, a condition testable on behavior-policy samples.

To quantify this condition on behavior-policy samples, define the population contrastive loss

$$\mathcal{J}_{\mathrm{beh}}(\pi_\theta; x) = \beta \log \mathbb{E}_{a\sim\pi_{\mathrm{beh}}} e^{R_\theta(x,a)/\beta} - \mathbb{E}_{a\sim\pi_{\mathrm{beh}}} R_\theta(x, a). \tag{3}$$

The following proposition states its dual interpretation and its relation to the original KL-regularized objective.

Proposition 1 (Population duality and regularized optimality)**.** *Suppose π_{beh}, π_θ, and π_{ref} are positive on $\mathcal{A}$. Define $h(z) = e^z - z - 1$ and $q_\theta(a \mid x) \propto \pi_{\mathrm{beh}}(a \mid x)e^{R_\theta(x,a)/\beta}$. Then*

$$\mathcal{J}_{\mathrm{beh}}(\pi_\theta; x) = \min_{\nu\in\mathbb{R}} \beta\mathbb{E}_{a\sim\pi_{\mathrm{beh}}} h\left(\frac{R_\theta(x, a) - \nu}{\beta}\right) = \beta D_{\mathrm{KL}}(\pi_{\mathrm{beh}}\|q_\theta) \geq 0. \tag{4}$$

With $C_x = \max_a \pi_\theta(a \mid x)/\pi_{\mathrm{beh}}(a \mid x)$,

$$V^*_\beta(x) - V_\beta(\pi_\theta; x) = \beta D_{\mathrm{KL}}(\pi_\theta\|\pi^*) \leq C_x \mathcal{J}_{\mathrm{beh}}(\pi_\theta; x). \tag{5}$$

Over all positive policies, the minimum contrastive loss is zero and its unique minimizer is π^*. When $\pi_{\mathrm{beh}} = \pi_\theta$, $\mathcal{J}_{\mathrm{beh}}$ equals the exact KL-regularized optimality gap.

The ratio C_x enters the analysis but is not evaluated as an update weight. Appendix A.2 proves the proposition. Training uses finite groups, whose expected loss differs from $\mathcal{J}_{\mathrm{beh}}$ and requires the separate guarantee developed next.

4 GIBBS-GAP POLICY OPTIMIZATION

4.1 FROM THE POPULATION OBJECTIVE TO FINITE GROUPS

Given m responses $a_{1:m} \sim \pi_{\mathrm{beh}}(\cdot \mid x)$, let $r_i := r(x, a_i)$ and define their KL-adjusted rewards $R_i := r_i - \beta \log[\pi_\theta(a_i \mid x)/\pi_{\mathrm{ref}}(a_i \mid x)] = R_\theta(x, a_i)$. Replacing the behavior expectations in Eq. (3) with their empirical averages yields the single-group loss

$$\widehat{\mathcal{J}}_m(\pi_\theta; x, a_{1:m}) = \beta \log\left(\frac{1}{m}\sum_{i=1}^{m} e^{R_i/\beta}\right) - \frac{1}{m}\sum_{i=1}^{m} R_i \geq 0. \tag{6}$$

This log-mean-exp gap is zero if and only if all responses in the group have the same adjusted reward.

For fixed π_{beh} and π_{ref}, its expected finite-group target is

$$H_m(\pi_\theta; \pi_{\mathrm{beh}}, x) := \mathbb{E}_{a_{1:m}\sim\pi_{\mathrm{beh}}^m} \widehat{\mathcal{J}}_m(\pi_\theta; x, a_{1:m}). \tag{7}$$

Its prompt average is $L_m(\pi_\theta) := \mathbb{E}_{X\sim\rho} H_m(\pi_\theta; \pi_{\mathrm{beh}}, X)$. We suppress the fixed β and reference in the loss notation, writing $H_m(\pi_\theta; \pi_{\mathrm{beh}}, x; \pi_{\mathrm{ref}})$ when specifying the reference is necessary. The nonlinear log moment makes $H_m \neq \mathcal{J}_{\mathrm{beh}}$ in general. For i.i.d. groups, $H_m \leq \mathcal{J}_{\mathrm{beh}}$, the two share the same zero set for every $m \geq 2$, and $H_m \to \mathcal{J}_{\mathrm{beh}}$ as the group size grows (Proposition 4). The population bound therefore does not transfer by substitution; we instead compare finite-group loss directly with a reward benchmark.

4.2 BEST-OF-K REWARD AND THE FINITE-GROUP LOSS

For each prompt, best-of-K sampling draws K independent responses from a policy π and returns the one with the highest verifier reward. Its expected reward, averaged over the prompt distribution ρ, is

$$J_K(\pi) := \mathbb{E}_{X\sim\rho}\mathbb{E}_{A_{1:K}\sim\pi(\cdot|X)^K}\left[\max_{i\leq K} r(X, A_i)\right]. \tag{8}$$

A single response from a policy π has expected reward $J(\pi) = \mathbb{E}_{X\sim\rho, A\sim\pi(\cdot|X)} r(X, A)$; in particular, $J_1(\pi) = J(\pi)$. The selection budget K is distinct from the synchronization delay N (Section 4.4). For rewards in $[0, 1]$, Proposition 5 (Appendix A.3) gives the bound

$$J_K(\pi_{\mathrm{beh}}) - J(\pi_\theta) \leq \kappa_{K,m}\{\beta(\log m + \Gamma) + CL_m(\pi_\theta)\}. \tag{9}$$

Here Γ bounds the oscillation of $\log(\pi_{\mathrm{ref}}/\pi_{\mathrm{beh}})$ and vanishes when $\pi_{\mathrm{ref}} = \pi_{\mathrm{beh}}$, while C uniformly bounds $\pi_\theta/\pi_{\mathrm{beh}}$ and equals one on policy. The tail-comparison multiplier $\kappa_{K,m}$, defined in Appendix A.3, satisfies $\kappa_{K,m} \leq \max\{1, K/(m-1)\}$. In particular, $m \geq K+1$ gives $\kappa_{K,m} = 1$. Theorem 8 in Appendix A.4 converts it into an empirical bound based on raw loss evaluated on i.i.d. unfiltered groups.

4.3 GRADIENT AND NORMALIZED TRAINING LOSS

Minimizing the group loss gives a weighted contrast between responses. Define the within-group tilt

$$q_i = \frac{e^{R_i/\beta}}{\sum_j e^{R_j/\beta}}. \tag{10}$$

Let U_m denote the uniform distribution on the m sample indices. Then $\widehat{\mathcal{J}}_m/\beta = D_{\mathrm{KL}}(U_m \| q)$, so the finite-group loss penalizes deviations of the adjusted-reward tilt from uniformity. Holding rewards, actions, and reference likelihoods fixed during an actor update gives

$$\nabla_\theta \widehat{\mathcal{J}}_m = -\beta \sum_{i=1}^{m}\left(q_i - \frac{1}{m}\right)\nabla_\theta \log \pi_\theta(a_i \mid x). \tag{11}$$

Under gradient descent, responses with $q_i > 1/m$ receive positive update coefficients, while those with $q_i < 1/m$ receive negative coefficients. The behavior policy determines which responses appear in the group; the update uses no learner-to-behavior correction weight. The coefficient can also be written

$$q_i = \mathrm{softmax}_i\left(\frac{r_i}{\beta} - \log\frac{\pi_\theta(a_i \mid x)}{\pi_{\mathrm{ref}}(a_i \mid x)}\right). \tag{12}$$

The reward term in Eq. (12) separates reward classes, while the learner-to-reference ratio redistributes weight among similarly rewarded responses. With binary rewards and small β, q concentrates on correct responses: incorrect responses receive the same limiting negative coefficient, while the weights among correct responses follow a softmax of their negative learner-to-reference log ratios. GGPO therefore gives larger coefficients to correct responses whose likelihood has grown less relative to the reference, and the net coefficient summed over correct responses is largest when only one response in the group is correct. This simple mechanism emphasizes rare successes while counteracting concentration among them, consistent with the entropy and best@32 behavior in Figures 1 and 6. At the population level, the frozen-behavior gradient of $\mathcal{J}_{\mathrm{beh}}$ contrasts the behavior distribution with its induced exponential tilt. When $\pi_{\mathrm{beh}} = \pi_\theta$, $\mathcal{J}_{\mathrm{beh}}$ equals the exact regularized gap, while its frozen-behavior gradient follows the forward-KL/cross-entropy direction toward π^*. Appendix A.6 gives the population relation, limiting cases, and coefficient analysis; Appendix C.4 reports weight-concentration diagnostics.

For a minibatch of G_{mb} retained groups, let $\widehat{\mathcal{J}}_m^g := \widehat{\mathcal{J}}_m(\pi_\theta; x_g, a^g_{1:m})$. To reuse GRPO's training hyperparameters, we use

$$L_{\mathrm{GGPO}} = \left(\frac{1}{G_{\mathrm{mb}}}\sum_{g=1}^{G_{\mathrm{mb}}} \widehat{\mathcal{J}}_m^g\right)\frac{1}{\beta\bar{\ell}\sqrt{m-1}}, \tag{13}$$

where $\bar{\ell}$ is the minibatch mean response length. This positive normalization leaves the direction of each realized minibatch gradient unchanged, removes the exterior β, and converts the sequence-level group average to a token-mean coefficient scale. Appendix A.6 gives the full calculation. Training retains only mixed-reward groups. For deterministic binary rewards, filtering and positive normalization preserve the zero set on behavior support for each prompt with $0 < \Pr_{a\sim\pi_{\mathrm{beh}}(\cdot|x)}[r(x,a)=1] < 1$ (Proposition 11). The resulting training objective averages over the retained prompt distribution. The population and empirical best-of-K bounds concern the raw, unfiltered loss.

4.4 Reference refresh and the GGPO update

Periodic reference refresh is part of the GGPO training algorithm used in our main experiments. Every T_{ref} optimizer updates, we set $\pi_{\mathrm{ref}} \leftarrow \pi_\theta$ and hold it fixed between refreshes. This resets the learner-to-reference log ratio in Eq. (12), keeping the anchor local to the current training stage (Appendix A.7). AGRO shares the same reference-dependent Gibbs target, so this motivation applies to its reference updates as well.

A fixed reference defines a single KL-regularized Gibbs target in Eq. (2), independent of group size. Reference refresh changes this target. The results below characterize exact target composition and account for nonzero within-window losses.

Refresh composes temperatures. Let π_0 be the initial policy and π_t the output of window t. If each window exactly attains its Gibbs target relative to the previous output, Eq. (2) gives

$$\pi_T(a \mid x) \propto \pi_0(a \mid x)e^{Tr(x,a)/\beta}. \tag{14}$$

Thus T exact windows at temperature β implement one Gibbs update at temperature β/T; Lemma 12 gives the precise statement and its final-iterate reward bound. The following result allows imperfect windows and uses expected finite-group loss, with reference π_{t-1} in window t.

Proposition 2 (Reference refresh with imperfect windows)**.** *Let $r \in [0,1]$, $m \geq 2$, and u_K be the best-of-K selection policy from a positive initial policy π_0. For each positive window output π_t and positive behavior $\pi_{\mathrm{beh},t}$, assume uniformly over prompts and actions that $\pi_t/\pi_{\mathrm{beh},t} \leq C$, $u_K/\pi_{\mathrm{beh},t} \leq C$, and $\mathrm{osc}_a\{r/\beta - \log(\pi_t/\pi_{t-1})\} \leq D$. Write $\bar{\pi}_T = T^{-1}\sum_t \pi_t$ and $\bar{L}_T =$*

$T^{-1}\sum_t \mathbb{E}_X H_m(\pi_t; \pi_{\mathrm{beh},t}, X; \pi_{t-1})$. *Then*

$$J_K(\pi_0) - J(\bar{\pi}_T) \le \frac{\beta \log K}{T} + \sqrt{\frac{4\beta C e^D}{1 - 1/m}\,\bar{L}_T}. \tag{15}$$

This permits larger β with more sufficiently accurate windows. At zero loss on common positive support, exact composition holds and the $\beta \log K/T$ bound also holds for the final iterate. In general the bound concerns the average iterate, not necessarily the final output. The zero-loss temperature identity alone does not quantify the finite-update gains observed in Appendix D.5. Appendix A.8 proves these statements; Theorem 13 there converts window losses into an empirical bound under explicit sampling and uniform generalization conditions.

Let N denote the number of learner minibatches consumed from each frozen behavior collection before the behavior policy is synchronized. The GGPO update proceeds as follows:

1. Freeze π_{beh}, sample m responses per prompt, and retain mixed-reward groups.
2. For each learner minibatch, compute masked sequence log likelihoods under the learner and active reference, form R_i and q_i, and minimize the normalized loss in Eq. (13).
3. Whenever the global optimizer-update counter reaches a T_{ref} boundary, refresh $\pi_{\mathrm{ref}} \leftarrow \pi_\theta$ before processing the next minibatch, independently of the behavior-collection boundary.
4. After the N learner minibatches have been consumed, synchronize $\pi_{\mathrm{beh}} \leftarrow \pi_\theta$ and collect the next set of responses.

The empirical bound in Theorem 13 uses one fixed behavior policy per reference window; the default schedule uses $T_{\mathrm{ref}}/N = 160/8 = 20$ behavior collections per reference window. GGPO treats each autoregressive response as a structured action. Appendices C.1 and C.4 specify filtering, refill, refresh-boundary, masking, EOS, padding, and truncation details; the offline setting instead keeps its rollout pool fixed throughout training.

5 Experiments

We compare GRPO, AGRO, and GGPO under default training ($N = 8$), controlled policy staleness ($N = 8$ to 128), and offline training on fixed data. Training–inference mismatch is present throughout. Appendices D and C give additional results, reference-refresh ablations, and implementation details.

5.1 Experimental setup

Models, data, and evaluation. We evaluate mathematical reasoning with Qwen3-1.7B-Base and Qwen3-4B-Base (Yang et al., 2025). Training uses ALP-Math, which merges a difficulty-filtered OpenR1-derived corpus (weqweasdas, 2025) with the math split of Guru-RL (Cheng et al., 2025). Following standard RLVR protocols (DeepSeek-AI et al., 2025), rewards are binary and verified by symbolic equivalence with Math-Verify (Kydlíček, 2025). Under the inherited training configuration described below, progress is monitored on MATH-500 (Hendrycks et al., 2021; Lightman et al., 2024). Monitoring results were not used for hyperparameter or checkpoint selection. Final evaluation reports mean@32, the mean accuracy over 32 newly sampled responses per problem, on MATH-500, AIME 2024, AIME 2025, OlympiadBench (He et al., 2024), and Minerva Math (Lewkowycz et al., 2022). Mean@32 estimates single-response mean reward under the evaluation rollout policy. Training curves also report MATH-500 best@32 (pass@32 for binary rewards), the fraction solved at least once in 32 samples. The five-benchmark average weights every problem equally (counts in Appendix C.1).

Seeds and checkpoint reporting. All five-benchmark tables report final checkpoints; default and controlled staleness runs use 8,000 optimizer updates. Each checkpoint receives one mean@32 evaluation. We report means and sample standard deviations over training seeds 42, 1234, and 2027 for the Qwen3-1.7B-Base default comparison (Table 1, Figure 1), offline accuracy (Table 5, Figure 3, left), and $N = 128$ staleness, including GRPO+TIS (Table 4). Curves summarize each logged or evaluated update. Other conditions and offline diagnostics use seed 42; reused default results retain their three-seed means, as noted in table captions.

Table 1: **Comparison under the default training setting ($N = 8$).** Final-checkpoint mean@32 accuracy after 8,000 optimizer updates on five mathematical-reasoning benchmarks (%). The final column is the problem-weighted average over the five benchmarks, which gives every evaluated problem equal weight. For Qwen3-1.7B-Base, values are the mean ± sample standard deviation across training seeds 42, 1234, and 2027. For Avg., these statistics are computed across the per-seed problem-weighted averages. Qwen3-4B-Base uses seed 42. Bold marks the best value in each model-size block.

Model and method	MATH-500	AIME 2024	AIME 2025	OlympiadBench	Minerva Math	Avg.
Qwen3-1.7B-Base + GRPO	68.86 ± 0.32	9.03 ± 0.90	5.14 ± 0.63	32.13 ± 0.27	28.33 ± 0.40	42.64 ± 0.18
+ AGRO	70.68 ± 0.33	9.27 ± 1.15	6.56 ± 0.81	35.73 ± 0.30	29.30 ± 0.44	45.07 ± 0.19
+ GGPO	$\mathbf{72.43 \pm 0.29}$	$\mathbf{11.11 \pm 0.89}$	$\mathbf{8.30 \pm 0.68}$	$\mathbf{37.11 \pm 0.25}$	$\mathbf{30.64 \pm 0.36}$	$\mathbf{46.58 \pm 0.16}$
Qwen3-4B-Base + GRPO	80.71	**16.88**	11.88	46.63	40.29	55.51
+ AGRO	80.55	**16.88**	12.40	47.21	40.71	55.81
+ GGPO	**81.83**	**16.88**	**14.38**	**48.33**	**41.85**	**56.98**

Baselines. AGRO (Tang et al., 2025) is our direct objective-level comparator: it shares GGPO's KL-adjusted reward, Gibbs target, and independently controlled reference schedule. OAPL (Ritter et al., 2026) ties its KL anchor and value target to the rollout policy; matching our independent reference schedule would require modifying its target estimator, so we discuss it as related work. Our policy-gradient baseline, GRPO (Shao et al., 2024), uses token-level learner-to-behavior PPO ratios with rollout log probabilities as the denominator and no additional off-policy correction. At $N = 128$ and in offline training, GRPO+TIS (Zhong et al., 2026) adds token-level truncated importance sampling with a cap of 2 (Appendix C.2).

Hyperparameters. By default, we set $\beta = 0.01$ for all methods, following the offline experiments of AGRO (Tang et al., 2025), and draw $m = 8$ responses per prompt. Appendix D.6 also evaluates $\beta = 0.001$, used in AGRO's on-policy experiments. The methods share the same batch geometry and prompt order, and reuse the learning rate and optimizer settings of our GRPO baseline. They also use the same dynamic difficulty-filtering rule (Yu et al., 2025; Bae et al., 2026). By default, our GRPO implementation keeps its initial reference fixed, while AGRO and GGPO refresh the reference every $T_{\text{ref}} = 160$ optimizer updates, matching our checkpointing cadence (Section 4.4). Appendices D.5 and D.6 compare reference cadences and group weights. The complete reference, batch, decoding, refill, and optimizer settings are given in Appendix C.

Training budget. Method comparisons match optimizer update counts and retained responses per update. Mixed-reward filtering and refill can nevertheless require different numbers of generated responses across methods. Appendix D.8 reports this sampling demand; response counts do not equalize generated tokens or GPU-hours, which also depend on response length and execution costs.

5.2 Default training setting

Under default training ($N = 8$; Section 4.4), all three methods' token-mean K3 stays on the order of 10^{-3} throughout training (Figure 1, panel 1). Table 1 compares their final checkpoints at both model sizes.

Results. GGPO achieves the highest problem-weighted average at both model sizes (Table 1). On Qwen3-1.7B-Base, it exceeds AGRO by 1.5 points and GRPO by 3.9 points; the single-seed Qwen3-4B-Base results preserve this ordering. GGPO initially trails both baselines on MATH-500 mean@32, then overtakes them (Figure 1, panel 4). Its best@32 rises throughout training, while GRPO and AGRO peak before 1,000 updates and decline (panel 3); its entropy settles at two to three times AGRO's (panel 2). It also leads five-benchmark pass@K for $K \in \{1, 4, 8, 16, 32\}$, with larger margins at higher budgets (Appendix D.2). The difficulty breakdown (Appendix D.1) shows faster early gains for GRPO and AGRO on partly solved problems, whereas GGPO continues to gain on problems with no correct response among the base model's 32 samples and finishes with the highest fraction solved at least once. On Qwen3-1.7B-Base, GGPO reaches its higher final MATH-500 mean@32 with fewer generated training responses (Appendix D.8).

In the analytic comparison in Figure 6, GGPO's total coefficient on correct responses is largest when only one response is correct, whereas GRPO and AGRO peak at balanced groups. Together with

downweighting equally rewarded responses whose likelihood has grown most relative to the reference (Eq. (12)), this is consistent with GGPO's continued gains on initially unsolved problems, higher entropy, and sustained best@32 gains.

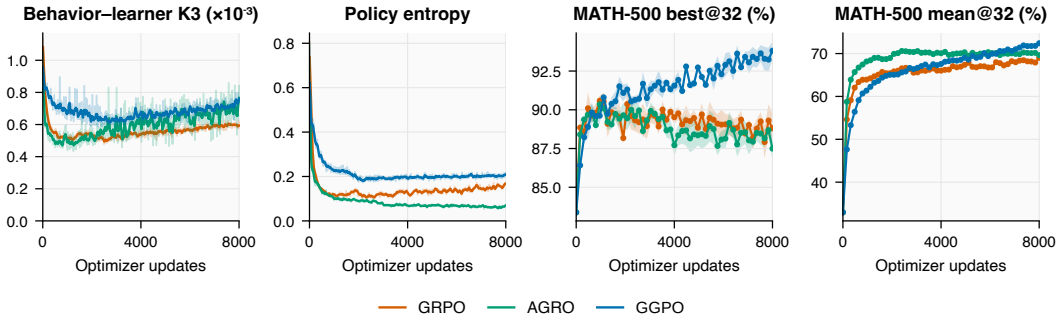

Figure 1: **Default training dynamics ($N = 8$) on Qwen3-1.7B-Base.** From left: token-mean K3 behavior–learner mismatch (Schulman, 2020) ($\times 10^{-3}$), policy entropy, MATH-500 best@32 (%), and MATH-500 mean@32 (%), each against optimizer updates (GRPO orange, AGRO green, GGPO blue). K3 is computed on mixed-reward-filtered training minibatches. In the first two panels, faint lines show logs every eight updates and solid lines nine-point moving averages. Curves and bands show means and ± 1 sample standard deviation over seeds 42, 1234, and 2027; MATH-500 is evaluated every 160 updates.

5.3 Controlled policy staleness

At delay N, each frozen behavior collection contains $32N$ prompts split into N disjoint minibatches, each used once, giving sample ages $0, \ldots, N-1$. Following controlled asynchronous RLHF evaluations (Noukhovitch et al., 2025; Huang et al., 2026), we vary $N \in \{8, 16, 32, 64, 128\}$ with other settings fixed, reusing the default run at $N = 8$. Figure 2 reports problem-weighted accuracy; Appendix D.3 gives per-benchmark results and diagnostics.

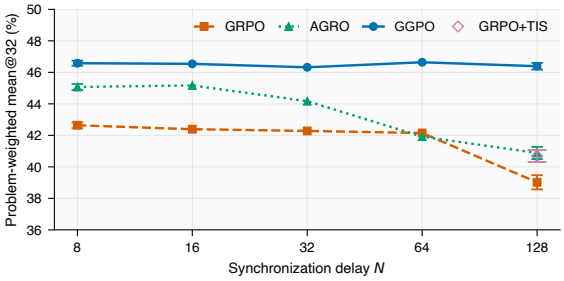

Figure 2: **Controlled policy staleness on Qwen3-1.7B-Base.** Problem-weighted final-checkpoint mean@32 accuracy over the five benchmarks after 8,000 optimizer updates, against synchronization delay N (equally spaced positions; values in Table 4). GRPO+TIS is evaluated at $N = 128$. Results at $N = 8, 128$ are three-seed means, with error bars showing ± 1 sample standard deviation; $N = 16, 32, 64$ use seed 42.

Results. GGPO maintains stable performance as policy staleness increases from $N = 8$ to $N = 128$, with problem-weighted accuracy of 46.32%–46.64%. AGRO loses 4.2 points from $N = 8$ to $N = 128$, while GRPO is essentially unchanged through $N = 64$ before declining at $N = 128$. At $N = 128$, TIS raises GRPO's accuracy from 39.02% to 40.69%, versus GGPO's 46.39% (Figure 2 and Table 4). As N increases, GGPO's K3 rises and normalized effective sample size (ESS) falls, while both remain nearly unchanged for uncorrected GRPO (Figure 7).

GRPO's nearly constant K3 is consistent with ratio clipping constraining updates. GGPO's rising K3 and stable accuracy are consistent with learning as it departs from the behavior policy. AGRO also avoids importance weights but loses accuracy at larger delays. All methods still synchronize behavior every N updates; the next experiment removes synchronization.

5.4 Offline training on fixed data

GRPO, GRPO+TIS, AGRO, and GGPO perform 400 optimizer updates on the same fixed rollout pool within each seed, without generating new responses. This tests learning as behavior–learner mismatch accumulates (protocol in Appendix C.3; all evaluated checkpoints in Table 5).

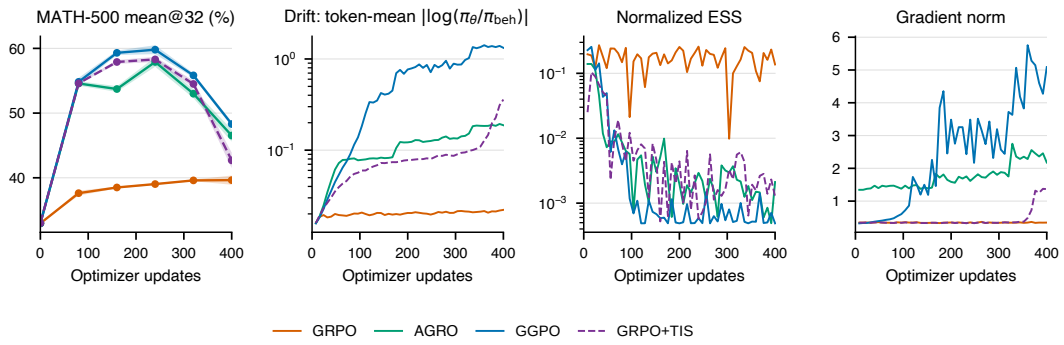

Figure 3: **Offline training on Qwen3-1.7B-Base.** Methods share the fixed rollout pool, batch geometry, prompt order, and 400-update budget within each seed (GRPO orange, GRPO+TIS purple dashed, AGRO green, GGPO blue). From left: MATH-500 mean@32 (%), mean and ± 1 sample standard deviation (Table 5); then seed-42 token-mean absolute learner-to-behavior log ratio, normalized ESS of diagnostic sequence-level importance weights per 2,048-response block (both log scale), and actor gradient norm (Appendix C.1). Mismatch diagnostics use mixed-reward-filtered training responses.

Results. GGPO improves MATH-500 accuracy from 32.96% to 59.79% within 240 optimizer updates, leading the baselines at every evaluated checkpoint from 160 updates onward (Figure 3; Table 5). GGPO, AGRO, and GRPO+TIS all peak at 240 updates and decline thereafter, alongside increasing divergence from the fixed behavior policy.

GRPO+TIS, AGRO, and GGPO show increasing drift and falling diagnostic ESS; GGPO's normalized ESS is below 10^{-3} by 160 updates. Uncorrected GRPO's drift stays near its initial training–inference mismatch.

6 Conclusion

GGPO optimizes KL-adjusted reward contrasts on behavior-policy responses without learner-to-behavior importance weights. Under suitable coverage, its finite-group loss controls the learner's single-response reward gap to behavior best-of-K, while reference refresh supports successive updates. On mathematical reasoning, it outperforms GRPO and AGRO across mismatch regimes, maintains accuracy under increasing staleness, and exceeds GRPO+TIS at $N = 128$ and on offline data from 160 updates onward.

Limitations

Our contextual-bandit analysis treats complete responses as actions and assumes bounded behavior coverage, whose factor C_x can grow exponentially with response length. Coverage and adjusted-reward spread are not estimated on trained policies; their worst-case constants may yield loose bounds. At our small β, the theory describes dependence on loss, coverage, group size, and refresh rather than predicting accuracy. It excludes token-level credit assignment, multi-turn MDPs, and contrastive temperatures separated from the KL coefficient β.

The reference-refresh bound concerns an average iterate under bounded adjusted-reward spread and additional comparator coverage; its empirical version assumes conditionally fresh prompt groups. Empirical bounds require raw loss on unfiltered groups and controlled class complexity, whereas training uses filtering and normalization. They do not establish convergence of the implemented optimizer. Very low offline diagnostic ESS indicates concentrated importance weights on filtered responses, not an estimate of global coverage.

At small β, exponential weighting approaches a hard maximum, suppressing distinctions among lower reward levels and potentially limiting fine-grained reward use. Our experiments cover only Qwen3 base models, mathematical reasoning, and binary rewards, with only one lower reward level.

AI USE STATEMENT

Generative AI tools were used to assist with literature searches, drafting and polishing the paper, code development, and auxiliary proof derivations, and to monitor experiment execution and logs. All research questions, algorithm design choices, theoretical claims, and experimental decisions were made by the authors. All theoretical statements and reported numbers were checked by the authors. The authors reviewed and edited all AI-generated text and take full responsibility for the content of this paper.

REPRODUCIBILITY STATEMENT

Training code, configurations, and data-processing scripts are available at `https://anonymous.4open.science/r/GCPO-301C/`. The pretrained models and source datasets used in our experiments are publicly available. Section 5 and Appendices C–D provide the experimental details and additional results, while Appendix A states the assumptions and gives complete proofs of all theoretical results.

REFERENCES

Charles Arnal, Gaëtan Narozniak, Vivien Cabannes, Yunhao Tang, Julia Kempe, and Rémi Munos. Asymmetric REINFORCE for off-policy reinforcement learning: Balancing positive and negative rewards. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2506.20520.

Sanghwan Bae, Jiwoo Hong, Min Young Lee, Hanbyul Kim, Jeongyeon Nam, and Donghyun Kwak. Online difficulty filtering for reasoning oriented reinforcement learning. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 700–719, 2026. doi: 10.18653/v1/2026.eacl-long.30. URL https://aclanthology.org/2026.eacl-long.30/.

Kianté Brantley, Mingyu Chen, Zhaolin Gao, Jason D. Lee, Wen Sun, Wenhao Zhan, and Xuezhou Zhang. Accelerating RL for LLM reasoning with optimal advantage regression. *arXiv preprint arXiv:2505.20686*, 2025. URL https://arxiv.org/abs/2505.20686.

Huayu Chen, Guande He, Lifan Yuan, Ganqu Cui, Hang Su, and Jun Zhu. Noise contrastive alignment of language models with explicit rewards. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2402.05369.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. URL https://arxiv.org/abs/2107.03374.

Zhoujun Cheng, Shibo Hao, Tianyang Liu, Fan Zhou, Yutao Xie, Feng Yao, Yuexin Bian, Yonghao Zhuang, Nilabjo Dey, Yuheng Zha, et al. Revisiting reinforcement learning for LLM reasoning from a cross-domain perspective. *arXiv preprint arXiv:2506.14965*, 2025. URL https://arxiv.org/abs/2506.14965.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, et al. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. URL https://arxiv.org/abs/2501.12948.

Lasse Espeholt, Hubert Soyer, Rémi Munos, Karen Simonyan, Volodymir Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, Shane Legg, and Koray Kavukcuoglu. IMPALA: Scalable distributed deep-RL with importance weighted actor-learner architectures. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1407–1416, 2018. URL https://arxiv.org/abs/1802.01561.

Yannis Flet-Berliac, Nathan Grinsztajn, Florian Strub, Bill Wu, Eugene Choi, Chris Cremer, Arash Ahmadian, Yash Chandak, Mohammad Gheshlaghi Azar, Olivier Pietquin, and Matthieu Geist. Contrastive policy gradient: Aligning LLMs on sequence-level scores in a supervised-friendly

fashion. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 21353–21370, 2024. doi: 10.18653/v1/2024.emnlp-main.1190. URL https://aclanthology.org/2024.emnlp-main.1190/.

Zhaolin Gao, Jonathan D. Chang, Wenhao Zhan, Owen Oertell, Gokul Swamy, Kianté Brantley, Thorsten Joachims, J. Andrew Bagnell, Jason D. Lee, and Wen Sun. REBEL: Reinforcement learning via regressing relative rewards. *arXiv preprint arXiv:2404.16767*, 2024. URL https://arxiv.org/abs/2404.16767.

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, 2019. URL https://proceedings.mlr.press/v97/geist19a.html.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. OlympiadBench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 2024. URL https://arxiv.org/abs/2402.14008.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://arxiv.org/abs/2103.03874.

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

Kimi Team et al. Kimi k1.5: Scaling reinforcement learning with LLMs. *arXiv preprint arXiv:2501.12599*, 2025. URL https://arxiv.org/abs/2501.12599.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, 2023. URL https://arxiv.org/abs/2309.06180.

Hynek Kydlíček. Math-Verify: Math verification library. Software repository, 2025. URL https://github.com/huggingface/Math-Verify.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://arxiv.org/abs/2206.14858.

Yingru Li, Jiacai Liu, Jiawei Xu, Yuxuan Tong, Ziniu Li, Qian Liu, and Baoxiang Wang. Trust region masking for long-horizon LLM reinforcement learning. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2512.23075.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2305.20050.

Ling Team, Anqi Shen, Baihui Li, et al. Every step evolves: Scaling reinforcement learning for trillion-scale thinking model. *arXiv preprint arXiv:2510.18855*, 2025. URL https://arxiv.org/abs/2510.18855.

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. *arXiv preprint arXiv:2503.20783*, 2025. URL https://arxiv.org/abs/2503.20783.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1711.05101.

Martin Marek and Max Ryabinin. Score centering stabilizes off-policy reinforcement learning. *arXiv preprint arXiv:2609.20807*, 2026. URL https://arxiv.org/abs/2609.20807.

MiniMax, Aili Chen, Aonian Li, Bangwei Gong, et al. MiniMax-M1: Scaling test-time compute efficiently with lightning attention. *arXiv preprint arXiv:2506.13585*, 2025. URL https://arxiv.org/abs/2506.13585.

Ofir Nachum, Mohammad Norouzi, Kelvin Xu, and Dale Schuurmans. Bridging the gap between value and policy based reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://arxiv.org/abs/1702.08892.

Ofir Nachum, Mohammad Norouzi, Kelvin Xu, and Dale Schuurmans. Trust-PCL: An off-policy trust region method for continuous control. In *International Conference on Learning Representations*, 2018. URL https://arxiv.org/abs/1707.01891.

Michael Noukhovitch, Shengyi Huang, Sophie Xhonneux, Arian Hosseini, Rishabh Agarwal, and Aaron Courville. Asynchronous RLHF: Faster and more efficient off-policy RL for language models. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2410.18252.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://arxiv.org/abs/2305.18290.

Pierre H. Richemond and Brendan Maginnis. A short variational proof of equivalence between policy gradients and soft Q learning. *arXiv preprint arXiv:1712.08650*, 2017. URL https://arxiv.org/abs/1712.08650.

Pierre Harvey Richemond, Yunhao Tang, Daniel Guo, Daniele Calandriello, Mohammad Gheshlaghi Azar, Rafael Rafailov, Bernardo Avila Pires, Eugene Tarassov, Lucas Spangher, Will Ellsworth, Aliaksei Severyn, Jonathan Mallinson, Lior Shani, Gil Shamir, Rishabh Joshi, Tianqi Liu, Remi Munos, and Bilal Piot. Offline regularised reinforcement learning for large language models alignment. *arXiv preprint arXiv:2405.19107*, 2024. URL https://arxiv.org/abs/2405.19107.

Daniel Ritter, Owen Oertell, Bradley Guo, Jonathan Chang, Kianté Brantley, and Wen Sun. LLMs can learn to reason via off-policy RL. *arXiv preprint arXiv:2602.19362*, 2026. URL https://arxiv.org/abs/2602.19362.

John Schulman. Approximating KL divergence. http://joschu.net/blog/kl-approx.html, 2020. URL http://joschu.net/blog/kl-approx.html.

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

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. *arXiv preprint arXiv:2402.03300*, 2024. URL https://arxiv.org/abs/2402.03300.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. HybridFlow: A flexible and efficient RLHF framework. *arXiv preprint arXiv:2409.19256*, 2024. URL https://arxiv.org/abs/2409.19256.

Fahim Tajwar, Guanning Zeng, Yueer Zhou, Yuda Song, Daman Arora, Yiding Jiang, Jeff Schneider, Ruslan Salakhutdinov, Haiwen Feng, and Andrea Zanette. Maximum likelihood reinforcement learning. In *Proceedings of the 43rd International Conference on Machine Learning*, Proceedings of Machine Learning Research. PMLR, 2026. URL https://arxiv.org/abs/2602.02710.

Yunhao Tang, Taco Cohen, David W. Zhang, Michal Valko, and Rémi Munos. RL-finetuning LLMs from on- and off-policy data with a single algorithm. *arXiv preprint arXiv:2503.19612*, 2025. URL https://arxiv.org/abs/2503.19612.

weqweasdas. OpenR1-derived mathematics dataset with verifier scores. Hugging Face dataset repository, 2025. URL https://huggingface.co/datasets/weqweasdas/from_default_filtered_openr1_with_scores.

An Yang et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Qiying Yu et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025. URL https://arxiv.org/abs/2503.14476.

Feng Zhang, Xinhong Ma, Ziqiang Dong, Xi Leng, Jianfei Zhao, Xin Sun, Yang Yang, and Guanjun Jiang. Revisiting reinforcement learning with verifiable rewards from a contrastive perspective. *arXiv preprint arXiv:2605.12969*, 2026a. URL https://arxiv.org/abs/2605.12969.

Yuheng Zhang, Chenlu Ye, Shuowei Jin, Changlong Yu, Wei Xiong, Saurabh Sahu, and Nan Jiang. Rethinking importance sampling in LLM policy optimization: A cumulative token perspective. *arXiv preprint arXiv:2605.07331*, 2026b. URL https://arxiv.org/abs/2605.07331.

Chujie Zheng, Kai Dang, Bowen Yu, Mingze Li, Huiqiang Jiang, Junrong Lin, Yuqiong Liu, Hao Lin, Chencan Wu, Feng Hu, An Yang, Jingren Zhou, and Junyang Lin. Stabilizing reinforcement learning with LLMs: Formulation and practices. *arXiv preprint arXiv:2512.01374*, 2025a. URL https://arxiv.org/abs/2512.01374.

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. *arXiv preprint arXiv:2507.18071*, 2025b. URL https://arxiv.org/abs/2507.18071.

Tianle Zhong, Neiwen Ling, Yifan Pi, Zijun Wei, Tianshu Yu, Geoffrey Fox, Peng Wu, and Xiao Yu. Diagnosing training inference mismatch in LLM reinforcement learning. *arXiv preprint arXiv:2605.14220*, 2026. URL https://arxiv.org/abs/2605.14220.

A Proofs and extensions

We give the omitted derivations for a finite action space $\mathcal{A}$; extensions require additional technical conditions. We suppress fixed prompts in pointwise derivations and write prompt averages explicitly.

A.1 Standing assumptions

Assume $0 < Z_\beta = \mathbb{E}_{\pi_{\mathrm{ref}}} e^{r/\beta} < \infty$, finite displayed expectations, and common positive support, as holds for finite-length autoregressive softmax policies. Coverage additionally requires $C_x = \max_{a \in \mathcal{A}} \pi_\theta(a \mid x)/\pi_{\mathrm{beh}}(a \mid x) < \infty$; C_x does not involve the reference policy.

Notation. We use the following distinctions throughout; local proof variables are defined where introduced. Loss notation suppresses a fixed temperature and reference, with $H_m(\pi_\theta; \pi_{\mathrm{beh}}, x; \pi_{\mathrm{ref}})$ showing the reference when it varies.

Symbol	Meaning
$\pi_{\mathrm{beh}};\ \pi_\theta;\ \pi_{\mathrm{ref}};\ \pi^*$	Behavior, learner, and reference policies; Gibbs policy for the active reference and temperature.
$\pi_0,\ \pi_t,\ \bar{\pi}_T;\ \pi_{\mathrm{beh},t};\ u_K$	Initial learner and reference, learner after refresh window t (reference of window $t+1$), and average iterate; behavior policy of window t; best-of-K selection policy.
$J(\pi),\ J_K(\pi)$	Single-response and best-of-K expected reward of a policy π, averaged over prompts; $J(\pi;x)$, $J_K(\pi;x)$ at prompt x.
$V_\beta(\pi;x),\ R_\theta$	KL-regularized value at prompt x; learner adjusted reward $r-\beta\log(\pi_\theta/\pi_{\mathrm{ref}})$.
$\mathcal{J}_{\mathrm{beh}},\ \widehat{\mathcal{J}}_m$	Population contrastive loss and raw loss of one sampled group.
$H_m,\ L_m,\ \widehat{L}_n$	Expected group loss at a prompt, its prompt average, and its empirical average over n groups.
$\mathcal{H}^{(t,s)}_{\mathrm{train}}$	Expected minibatch loss after mixed-group filtering and batch-dependent normalization.
$q_i,\ q_\theta$	Tilt on sampled indices and the corresponding population tilt.
$m,\ n;\ K,\ k$	Responses per group and independent fresh-prompt groups; best-of-K selection budget and correct responses in a group.
$N;\ T,\ T_{\mathrm{ref}}$	Synchronization delay: learner minibatches (one optimizer update each) per frozen behavior collection; number of reference windows and optimizer updates between reference refreshes.
$G_{\mathrm{mb}},\ P,\ G$	Groups per learner minibatch; problems and groups per problem in the allocation study.
$\bar{\ell},\ B_{\mathrm{resp}}$	Mean response length and response count in a diagnostic batch.
$C_x,\ C;\ \kappa_{K,m}$	Learner-to-behavior coverage ratio at a prompt and uniformly (in Proposition 2, C also bounds comparator-to-behavior coverage); best-of-K tail-comparison multiplier, no larger than $\max\{1, K/(m-1)\}$.
$\nu;\ D,\ \Gamma_x,\ \Gamma$	Scalar dual variable; bound on the oscillation (max minus min over actions) of the adjusted reward divided by β, and oscillation of $\log(\pi_{\mathrm{ref}}/\pi_{\mathrm{beh}})$ at a prompt and uniformly.
$\eta,\ \delta,\ \Lambda,\ \gamma_n$	Cover radius, failure probability, loss envelope, and statistical remainder.
$\Delta_i,\ c_i$	Learner-to-reference sequence log ratio and normalized GGPO coefficient of response i.

A.2 Proof of population duality and coverage

This proves Proposition 1, explaining both the loss construction in Section 3 and the coverage condition in its regularized-gap guarantee. We derive the scalar dual, identify its profiled residual, and transfer the resulting bound through coverage.

Lemma 3 (Scalar duality and primal–dual gap)**.** *Fix a prompt x, temperature $\beta>0$, reward $r(x,a)$, and positive reference policy $\pi_{\mathrm{ref}}(\cdot\mid x)$. Define $V_\beta(\pi;x)=\mathbb{E}_{a\sim\pi}r(x,a)-\beta D_{\mathrm{KL}}(\pi\|\pi_{\mathrm{ref}})$, $V^*_\beta(x)=\max_\pi V_\beta(\pi;x)$, and $Z_\beta=\mathbb{E}_{a\sim\pi_{\mathrm{ref}}}e^{r(x,a)/\beta}$. For a positive candidate π_θ, define $R_\theta(x,a)=r(x,a)-\beta\log[\pi_\theta(a\mid x)/\pi_{\mathrm{ref}}(a\mid x)]$, $h(z)=e^z-z-1$, and*

$$\begin{aligned}\Phi_x(\nu)&:=\nu+\beta\big(\mathbb{E}_{a\sim\pi_{\mathrm{ref}}}e^{(r(x,a)-\nu)/\beta}-1\big),\\ \underbrace{\max_\pi V_\beta(\pi;x)}_{\textit{primal}}&=\beta\log Z_\beta=\underbrace{\min_{\nu\in\mathbb{R}}\Phi_x(\nu)}_{\textit{scalar dual}},\\ \Psi_x(\pi_\theta,\nu)&:=\Phi_x(\nu)-V_\beta(\pi_\theta;x).\end{aligned}\tag{16}$$

Then $\Psi_x(\pi_\theta,\nu)\ge 0$, $\min_\nu\Psi_x(\pi_\theta,\nu)=V^*_\beta(x)-V_\beta(\pi_\theta;x)$, and

$$\Psi_x(\pi_\theta,\nu)=\beta\mathbb{E}_{a\sim\pi_\theta}h((R_\theta(x,a)-\nu)/\beta).\tag{17}$$

Proof. The Gibbs policy $\pi^*(a\mid x)=\pi_{\mathrm{ref}}(a\mid x)e^{r(x,a)/\beta}/Z_\beta$ satisfies

$$\beta D_{\mathrm{KL}}(\pi\|\pi^*)=\beta\log Z_\beta-V_\beta(\pi;x).$$

Hence $V_\beta^*(x) = \beta \log Z_\beta$ and π^* is its unique maximizer. Moreover, $\Phi_x'(\nu) = 1 - Z_\beta e^{-\nu/\beta}$ and $\Phi_x''(\nu) = Z_\beta e^{-\nu/\beta}/\beta > 0$, so its minimum is $\Phi_x(\beta \log Z_\beta) = V_\beta^*(x)$. Hence $\Phi_x(\nu)$ upper-bounds the primal optimum for every ν, so Ψ_x is a nonnegative primal–dual gap, and $\min_\nu \Psi_x(\pi_\theta, \nu) = V_\beta^*(x) - V_\beta(\pi_\theta; x)$. Finally, $\mathbb{E}_{\pi_\theta} R_\theta = V_\beta(\pi_\theta; x)$ and $\pi_\theta e^{(R_\theta - \nu)/\beta} = \pi_{\mathrm{ref}} e^{(r-\nu)/\beta}$. Expanding h proves Eq. (17). □

Proof of Proposition 1. Fix x and use the quantities defined in Lemma 3. Replacing the learner expectation by π_{beh} and minimizing over ν gives $\nu_\theta = \beta \log \mathbb{E}_{\pi_{\mathrm{beh}}} e^{R_\theta/\beta}$: the first derivative is $1 - \mathbb{E}_{\pi_{\mathrm{beh}}} e^{(R_\theta - \nu)/\beta}$ and the second is strictly positive. Substitution gives $\mathcal{J}_{\mathrm{beh}} = \nu_\theta - \mathbb{E}_{\pi_{\mathrm{beh}}} R_\theta$, the loss in Eq. (3). Define

$$q_\theta(a \mid x) = \frac{\pi_{\mathrm{beh}}(a \mid x) e^{R_\theta(x,a)/\beta}}{\mathbb{E}_{a' \sim \pi_{\mathrm{beh}}} e^{R_\theta(x,a')/\beta}}. \tag{18}$$

Expanding its KL divergence gives $\beta D_{\mathrm{KL}}(\pi_{\mathrm{beh}} \| q_\theta) = \mathcal{J}_{\mathrm{beh}}$, proving Eq. (4). Nonnegativity of h and $\pi_\theta / \pi_{\mathrm{beh}} \le C_x$ now imply

$$V_\beta^* - V_\beta(\pi_\theta; x) \le \Psi_x(\pi_\theta, \nu_\theta) \le C_x \beta \mathbb{E}_{\pi_{\mathrm{beh}}} h((R_\theta - \nu_\theta)/\beta) = C_x \mathcal{J}_{\mathrm{beh}}.$$

When $\pi_{\mathrm{beh}} = \pi_\theta$, profiling the same learner residual gives equality with the regularized gap. Under positive support, zero loss forces R_θ constant, so $\pi_\theta \propto \pi_{\mathrm{ref}} e^{r/\beta}$ and hence $\pi_\theta = \pi^*$. Conversely, this Gibbs policy has constant adjusted reward and zero loss. □

Coverage and response length. The coverage inequality also holds under the standing common-support and finite-expectation assumptions when $C_x = \max_{a \in \mathcal{A}} \pi_\theta(a \mid x)/\pi_{\mathrm{beh}}(a \mid x) < \infty$. Across prompts it gives $\mathbb{E}_X[V_\beta^*(X) - V_\beta(\pi_\theta; X)] \le \mathbb{E}_X[C_X \mathcal{J}_{\mathrm{beh}}(\pi_\theta; X)]$; moving coverage outside the expectation requires a uniform bound. For a response $a = (a^1, \ldots, a^\ell)$,

$$\frac{\pi_\theta(a \mid x)}{\pi_{\mathrm{beh}}(a \mid x)} = \prod_{\tau=1}^{\ell} \frac{\pi_\theta(a^\tau \mid a^{<\tau}, x)}{\pi_{\mathrm{beh}}(a^\tau \mid a^{<\tau}, x)}.$$

This explains the response-length limitation in Section 6: sequence coverage can grow exponentially even though the algorithm does not use the ratio as an update weight.

A.3 Finite-group guarantees

We establish the common zero set and population limit used by the population and reference-refresh results, then state and prove the best-of-K bound of Section 4.2.

Proposition 4 (Finite-group target)**.** *Suppose that R_θ is bounded on the support of π_{beh} and that $a_1, \ldots, a_m$ are sampled independently from π_{beh}. Then, for every integer $m \ge 1$,*

$$0 = H_1 \le H_m \le H_{m+1} \le \mathcal{J}_{\mathrm{beh}}, \qquad H_m \longrightarrow \mathcal{J}_{\mathrm{beh}}. \tag{19}$$

For every $m \ge 2$, $H_m = 0$ if and only if R_θ is constant on the support of π_{beh}, equivalently if and only if $\mathcal{J}_{\mathrm{beh}} = 0$. If π_θ, π_{ref}, and π_{beh} have common positive support, this zero condition also implies $\pi_\theta = \pi^*$.

Proof. Let $X = e^{R_\theta/\beta}$ and $\bar{X}_m = m^{-1} \sum_{i=1}^m X_i$ for i.i.d. copies X_i. Then

$$H_m = \beta(\mathbb{E} \log \bar{X}_m - \mathbb{E} \log X), \qquad \mathcal{J}_{\mathrm{beh}} = \beta(\log \mathbb{E}X - \mathbb{E} \log X). \tag{20}$$

Jensen gives $0 = H_1 \le H_m \le \mathcal{J}_{\mathrm{beh}}$. Given $m+1$ samples, the leave-one-out means $A_j = m^{-1} \sum_{i \ne j} X_i$ average to $\bar{X}_{m+1}$. Thus $\log \bar{X}_{m+1} \ge (m+1)^{-1} \sum_j \log A_j$; taking expectations gives $H_{m+1} \ge H_m$, since each A_j is distributed as the size-m sample mean $\bar{X}_m$. The bounded i.i.d. sample average $\bar{X}_m$ converges to $\mathbb{E}X$, and boundedness of R_θ bounds $\log \bar{X}_m$, so dominated convergence gives $H_m \to \mathcal{J}_{\mathrm{beh}}$.

For $m \ge 2$, equality in the group Jensen inequality requires all sampled X_i to agree. Hence $H_m = 0$ implies two independent copies always agree on the sampled support, so X and R_θ are constant there. The converse is immediate; the population Jensen equality gives the same zero set for $\mathcal{J}_{\mathrm{beh}}$. Under the stated support conditions, constancy gives $\pi_\theta \propto \pi_{\mathrm{ref}} e^{r/\beta}$, proving the final claim. □

Best-of-K reward bound. Write $J_K(\pi; x)$ and $J(\pi; x) = J_1(\pi; x)$ for the expected best-of-K and single-response rewards of a policy π conditional on prompt x, so $J_K(\pi) = \mathbb{E}_X J_K(\pi; X)$ in Eq. (8), and let $L_m(\pi_\theta) = \mathbb{E}_{X\sim\rho} H_m(\pi_\theta; \pi_{\mathrm{beh}}, X)$. For $p_{x,t} = \Pr_{A\sim\pi_{\mathrm{beh}}(\cdot|x)}[r(x, A) > t]$, define

$$\kappa_{K,m} := \max\left\{1, \sup_{x,t:\, p_{x,t}>0} \frac{1-(1-p_{x,t})^K}{1-(1-p_{x,t})^{m-1}}\right\} \le \max\left\{1, \frac{K}{m-1}\right\}. \tag{21}$$

The inequality follows because $[1-(1-p)^k]/k$ decreases in k. When $K > m-1$, the upper bound is approached as a relevant behavior-policy upper-tail probability tends to zero, but the actual multiplier can be smaller.

Proposition 5 (Best-of-K and the GGPO minimizer)**.** *Let $r \in [0, 1]$, fix positive behavior and reference policies with $\Gamma_x \le \Gamma$ at every prompt, where Γ_x is the oscillation of $\log[\pi_{\mathrm{ref}}(\cdot \mid x)/\pi_{\mathrm{beh}}(\cdot \mid x)]$, and suppose every policy π_θ in a positive hypothesis class Π satisfies $\pi_\theta/\pi_{\mathrm{beh}} \le C$ uniformly. For $m \ge 2$ and $K \ge 1$,*

$$J_K(\pi_{\mathrm{beh}}) - J(\pi_\theta) \le \kappa_{K,m}\{\beta(\log m + \Gamma) + CL_m(\pi_\theta)\}. \tag{22}$$

In particular, with $\alpha_m = \inf_{\pi_\theta\in\Pi} L_m(\pi_\theta)$, a population minimizer satisfies this bound with L_m replaced by α_m; an approximate minimizer with loss at most $\alpha_m + \zeta$ incurs an additional $C\kappa_{K,m}\zeta$.

On policy, $\pi_\theta = \pi_{\mathrm{beh}}$ and $C = 1$; with $\pi_{\mathrm{ref}} = \pi_{\mathrm{beh}}$, $\Gamma = 0$.

Remark 6 (When the minimum loss is zero). For the Gibbs policy in Eq. (2), R_θ is constant within every prompt, so every group loss is zero. Hence $\alpha_m = 0$ whenever this policy belongs to the admissible class. This need not hold in a restricted class or under a prescribed coverage cap. The zero is the loss value; the constant adjusted reward is generally nonzero, and the $\beta(\log m + \Gamma)$ term remains in the best-of-K comparison.

The following prompt-level bound implies Proposition 5.

Proposition 7 (Prompt-level best-of-K bound)**.** *Fix a prompt, rewards in $[0, 1]$, $\beta > 0$, $K \ge 1$, $m \ge 2$, and positive learner, behavior, and reference policies on a common finite action space. Suppose $\pi_\theta/\pi_{\mathrm{beh}} \le C_x$, and let $\kappa_{K,m} \ge 1$ satisfy $1-(1-p_t)^K \le \kappa_{K,m}[1-(1-p_t)^{m-1}]$ for every threshold t, where $p_t = \pi_{\mathrm{beh}}\{r > t\}$. Define $\Gamma_x = \max_a \log[\pi_{\mathrm{ref}}(a \mid x)/\pi_{\mathrm{beh}}(a \mid x)] - \min_a \log[\pi_{\mathrm{ref}}(a \mid x)/\pi_{\mathrm{beh}}(a \mid x)]$. Then*

$$J_K(\pi_{\mathrm{beh}}; x) - J(\pi_\theta; x) \le \kappa_{K,m}\{\beta(\log m + \Gamma_x) + C_x H_m(\pi_\theta; \pi_{\mathrm{beh}}, x)\}. \tag{23}$$

When $\pi_{\mathrm{ref}} = \pi_{\mathrm{beh}}$, $\Gamma_x = 0$. Averaging over prompts and using $C_x \le C$ and $\Gamma_x \le \Gamma$ proves Proposition 5, including its minimizer and approximate-minimizer statements.

Proof. Suppress the prompt. Put $w_i = \pi_\theta(a_i)/\pi_{\mathrm{beh}}(a_i)$, $S = m^{-1}\sum_i w_i$, $W = \max_i w_i$, and $P_i = w_i/(mS)$. Let $u_i = r_i + \beta\log[\pi_{\mathrm{ref}}(a_i)/\pi_{\mathrm{beh}}(a_i)]$ and $z_i = u_i/\beta - \log w_i = R_i/\beta$. With U uniform on sample indices, $c = \log \mathbb{E}_U e^z$, and $g(t) = e^t - t - 1 \ge 0$, one has $\mathbb{E}_U g(z - c) = \widehat{\mathcal{J}}_m/\beta$. Define $Q_i \propto e^{u_i/\beta}$ and $A = \mathbb{E}_P e^z$. Since $Q_i = P_i e^{z_i}/A$,

$$\mathbb{E}_P g(z - c) - D_{\mathrm{KL}}(P\|Q) = Ae^{-c} - 1 - \log(Ae^{-c}) \ge 0.$$

The bound $P_i/U_i \le W/S$ therefore gives $\beta S D_{\mathrm{KL}}(P\|Q) \le W\widehat{\mathcal{J}}_m$. The finite Gibbs identity yields $\max_i u_i - \mathbb{E}_P u \le \beta\log m + \beta D_{\mathrm{KL}}(P\|Q)$. Indeed, expanding $D_{\mathrm{KL}}(P\|Q)$, using $\sum_i P_i \log P_i \ge -\log m$, and lower-bounding the log-partition by $\max_i u_i/\beta$ gives this inequality. The oscillation of $u - r$ is at most $\beta\Gamma_x$, hence $\max_i r_i - \mathbb{E}_P r \le \max_i u_i - \mathbb{E}_P u + \beta\Gamma_x$. Multiplying by S and using $S\mathbb{E}_P r = m^{-1}\sum_i w_i r_i$ together with the preceding KL bound gives

$$S\max_i r_i - \frac{1}{m}\sum_i w_i r_i \le \beta S(\log m + \Gamma_x) + W\widehat{\mathcal{J}}_m. \tag{24}$$

For $F = \max_i r_i$, symmetry and importance weighting give $\mathbb{E}_{\pi_{\mathrm{beh}}^m}[SF] = \mathbb{E}_{\pi_\theta\otimes\pi_{\mathrm{beh}}^{m-1}} F$, while $\mathbb{E}S = 1$ and $\mathbb{E}[m^{-1}\sum_i w_i r_i] = \mathbb{E}_{\pi_\theta} r$. At a threshold $t \in [0, 1]$, write $p = \pi_{\mathrm{beh}}\{r > t\}$, $v = \pi_\theta\{r > t\}$, and $d_k = 1 - (1-p)^k$. By the defining tail comparison,

$$d_K - v \le d_K(1 - v) \le \kappa_{K,m} d_{m-1}(1 - v).$$

The layer-cake identities here are

$$J_K(\pi_{\mathrm{beh}}; x) = \int_0^1 d_K \, dt, \qquad J(\pi_\theta; x) = \int_0^1 v \, dt,$$

and, because the mixed maximum has tail $1 - (1-v)(1-p)^{m-1} = v + (1-v)d_{m-1}$,

$$\mathbb{E}[SF] - \mathbb{E}_{\pi_\theta} r = \int_0^1 (1-v) d_{m-1} \, dt.$$

Integrating the threshold inequality therefore bounds $J_K(\pi_{\mathrm{beh}}; x) - J(\pi_\theta; x)$ by $\kappa_{K,m}(\mathbb{E}[SF] - \mathbb{E}_{\pi_\theta} r)$. Take expectations in Eq. (24) and use $W \le C_x$. □

A.4 Generalization and an oracle inequality

Theorem 8 below converts expected group loss into an observed-loss bound. The oracle inequality then states what approximate empirical minimization achieves in a restricted hypothesis class. Fix a prompt distribution ρ, behavior π_{beh}, reference π_{ref}, temperature $\beta > 0$, and hypothesis class Π before collecting n i.i.d. unfiltered groups. For $f_{\pi_\theta}(Z) = \widehat{\mathcal{J}}_m(\pi_\theta; Z)$, let $L_m(\pi_\theta) = \mathbb{E} f_{\pi_\theta} = \mathbb{E}_{X \sim \rho} H_m(\pi_\theta; \pi_{\mathrm{beh}}, X)$ and $\widehat{L}_n(\pi_\theta) = n^{-1} \sum_j f_{\pi_\theta}(Z_j)$. Assume $0 \le f_{\pi_\theta} \le \Lambda$ and that the induced loss family admits a finite uniform η-cover with centers in the family: each loss is within η of a center on every possible group. Write

$$t = \log \frac{2\mathcal{N}_\infty(\eta)}{\delta}, \qquad \gamma_n = 4\Lambda t/n + 3\eta.$$

A sufficient envelope is $\Lambda = 1 + \beta B$ when $\mathrm{osc}_a \log[\pi_\theta(a \mid x)/\pi_{\mathrm{ref}}(a \mid x)] \le B$ uniformly, since adjusted rewards then have range at most Λ. These assumptions apply to all possible groups, not only observed responses. Coverage remains separate.

Theorem 8 (Empirical GGPO bound)**.** *Under Proposition 5's assumptions and the sampling and cover conditions above, with probability at least $1 - \delta$, every data-selected $\pi_{\hat{\theta}} \in \Pi$ satisfies*

$$J_K(\pi_{\mathrm{beh}}) - J(\pi_{\hat{\theta}}) \le \kappa_{K,m}\{\beta(\log m + \Gamma) + C[2\widehat{L}_n(\pi_{\hat{\theta}}) + \gamma_n]\}. \tag{25}$$

The bound applies to the output of GGPO; it requires no convexity or successful empirical minimization. The reported loss uses unfiltered groups in the original units, rather than the filtered, normalized training log.

Lemma 9 (Relative uniform generalization)**.** *With probability at least $1 - \delta$, simultaneously for every $\pi_\theta \in \Pi$,*

$$L_m(\pi_\theta) \le 2\widehat{L}_n(\pi_\theta) + \gamma_n, \qquad \widehat{L}_n(\pi_\theta) \le 2L_m(\pi_\theta) + \gamma_n. \tag{26}$$

Proof. For a fixed center $g \in [0, \Lambda]$, write $P = \mathbb{E} g$ and $\widehat{P} = n^{-1} \sum_j g(Z_j)$. Since $\mathrm{Var}(g) \le \Lambda P$, Bernstein's inequality gives, with failure probability at most $2e^{-t}$,

$$|P - \widehat{P}| \le \sqrt{2\Lambda P t/n} + 2\Lambda t/(3n) \le P/2 + 5\Lambda t/(3n).$$

Solving the first direction gives $P \le 2\widehat{P} + 10\Lambda t/(3n)$. Rounding $10/3$ up to 4 and loosening the other direction to the same constant gives $P \le 2\widehat{P} + 4\Lambda t/n$ and $\widehat{P} \le 2P + 4\Lambda t/n$. Union-bound over centers. Transferring to a loss within η adds $\eta + 2\eta = 3\eta$ to either inequality. □

Proof of Theorem 8. Average Proposition 7 over prompts, use $C_x \le C$ and $\Gamma_x \le \Gamma$, and substitute the first inequality in Eq. (26). Uniformity permits any data-selected output in the fixed class. □

Proposition 10 (Oracle inequality)**.** *Let $\alpha_m = \inf_{\pi_\theta \in \Pi} L_m(\pi_\theta)$ and suppose the returned hypothesis satisfies $\widehat{L}_n(\pi_{\hat{\theta}}) \le \inf_{\pi_\theta \in \Pi} \widehat{L}_n(\pi_\theta) + \xi$. Under the reward and support conditions of Proposition 7, suppose $C_x \le C$ for every policy in Π and $\Gamma_x \le \Gamma$ at every prompt. Then, on the same uniform event,*

$$J_K(\pi_{\mathrm{beh}}) - J(\pi_{\hat{\theta}}) \le \kappa_{K,m}\{\beta(\log m + \Gamma) + C[4\alpha_m + 2\xi + 3\gamma_n]\}. \tag{27}$$

Proof. The two inequalities in Eq. (26) give

$$L_m(\pi_{\hat{\theta}}) \le 2 \inf_{\pi_\theta} \widehat{L}_n(\pi_\theta) + 2\xi + \gamma_n \le 4 \inf_{\pi_\theta} L_m(\pi_\theta) + 2\xi + 3\gamma_n.$$

Average Proposition 7 and substitute this bound. No attained minimizer is needed. □

Prompt count and optimization setting. The oracle terms separate approximation, empirical optimization, and statistics. Small ξ alone is insufficient if the minimum loss α_m is large. For an ε statistical contribution in the observed-loss bound, choose $\eta = \varepsilon/(6C\kappa_{K,m})$ and $n \geq 8\Lambda C\kappa_{K,m} \log[2\mathcal{N}_\infty(\eta)/\delta]/\varepsilon$. The oracle remainder has the same rate up to constants. These are conditional small-loss rates; entropy is evaluated at the chosen accuracy, and coverage or boundedness may depend on response length. The analysis covers a fixed offline collection or a stage with fixed behavior and reference policies, not arbitrary behavior and reference changes over a full run. Replay does not increase n. Appendix A.5 distinguishes new problems from repeated groups on the same problems.

Group size. With $m = K \geq 2$, the multiplier is at most two; with $m = K + 1$, it is one. Thus groups of order K directly support the best-of-K target and can reduce the required number of fresh prompt groups. The loss, its complexity, and the temperature bias can also change with m, so monotone improvement is not guaranteed. Total responses are nm. Appendix D.7 studies group size empirically.

A.5 Problems, groups per problem, and computation

Theorem 8 draws a fresh prompt for each group. Instead, suppose P independent verifier-equipped problems are drawn from ρ, with G independent groups per problem and m responses per group. Total response cost is PGm. Here G counts groups per problem, while G_{mb} counts groups per learner minibatch. For a hypothesis fixed independently of these data, write $h_m(x) = \mathbb{E}[f_{\pi_\theta} \mid X = x]$ and $s_m^2(x) = \mathrm{Var}(f_{\pi_\theta} \mid X = x)$. The average loss $\widehat{L}_{P,G,m}$ is unbiased for $L_m(\pi_\theta)$, and conditional independence gives

$$\mathrm{Var}(\widehat{L}_{P,G,m}) = \underbrace{\frac{\mathrm{Var}_X\, h_m(X)}{P}}_{\text{problem-sampling error}} + \underbrace{\frac{\mathbb{E}_X s_m^2(X)}{PG}}_{\text{response-sampling error}} . \tag{28}$$

Thus additional groups reduce response noise but leave a problem-sampling floor. This fixed-hypothesis identity is not itself a uniform training bound.

For uniform control, each problem with its G groups is one independent block. Block-average losses inherit the same $[0, \Lambda]$ envelope and uniform cover, so Theorem 8 applies with $n = P$. This coarse guarantee need not improve with G; substituting $n = PG$ would incorrectly count repeated problems as fresh ones. Conditional on a fixed problem set, fresh groups can instead support concentration toward that set's average loss, which targets the empirical problem distribution rather than unseen problems.

Simple illustration. Four equally likely problem types have binary behavior success probabilities $p \in \{0.005, 0.02, 0.1, 0.5\}$. Set reference equal to behavior, $\beta = 0.2$, and model success and failure as two actions, with a constant learner-to-behavior ratio within each class. Set the learner success probability to $\tilde{p} = 5p/(1 - p + 5p)$; no optimizer is run. Under this two-action model, with k successes, the exact GGPO loss is

$$f(k; m) = \beta \log[1 + (k/m)(e^{d/\beta} - 1)] - (k/m)d, \qquad d = 1 - \beta \log 5.$$

Enumerating $k \sim \mathrm{Binomial}(m, p)$ gives the moments in Eq. (28). A check with 4,000 simulated datasets per setting matches exact RMSE within 1% for all four combinations of $P \in \{16, 64\}$ and $G \in \{1, 16\}$, with $m = 16$. Figure 4 plots the exact calculations.

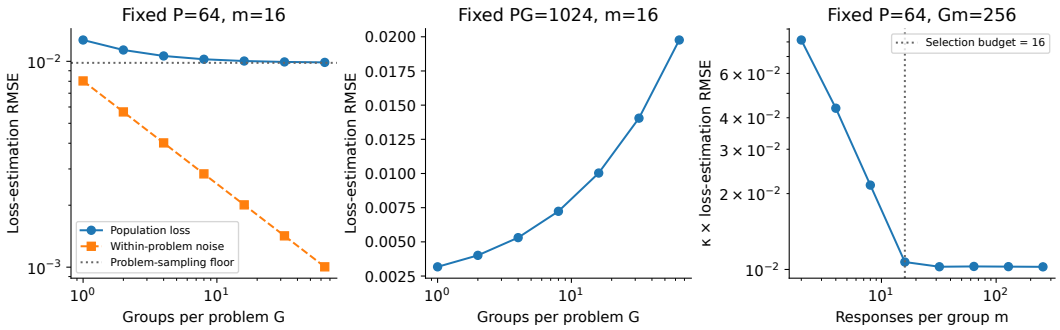

Figure 4: **Loss-estimation error under different allocations of problems and groups.** Exact RMSE of $\widehat{L}_{P,G,m}$ for the four-type illustration of Appendix A.5. Left: at fixed $P = 64$ and $m = 16$, additional groups per problem reduce the within-problem noise toward the problem-sampling floor. Middle: at fixed $PG = 1024$ and $m = 16$, more groups per problem leave fewer problems and raise the error. Right: at fixed $P = 64$ and $Gm = 256$, RMSE multiplied by $\kappa_{K,m}$ with $K = 16$; each m estimates its own L_m.

A.6 GGPO GRADIENT, NORMALIZATION, AND FILTERING

Off-policy tilt and on-policy limit. By Eq. (2), $e^{R_i/\beta}$ is proportional to $\pi^*(a_i \mid x)/\pi_\theta(a_i \mid x)$. Because responses are sampled from π_{beh}, their normalized weighted empirical distribution converges to

$$q_\theta(a \mid x) = \frac{\pi_{\text{beh}}(a \mid x)\pi^*(a \mid x)/\pi_\theta(a \mid x)}{\mathbb{E}_{a' \sim \pi_{\text{beh}}}[\pi^*(a' \mid x)/\pi_\theta(a' \mid x)]},$$

the induced tilt in Eq. (18). Holding π_{beh} fixed gives

$$\nabla_\theta \mathcal{J}_{\text{beh}} = -\beta \left(\mathbb{E}_{q_\theta} \nabla_\theta \log \pi_\theta - \mathbb{E}_{\pi_{\text{beh}}} \nabla_\theta \log \pi_\theta\right).$$

When $\pi_{\text{beh}} = \pi_\theta$, $q_\theta = \pi^*$ and the second expectation vanishes. The objective value then equals the reverse-KL regularized gap, while the frozen-π_{beh} derivative is the forward-KL/cross-entropy gradient toward π^*. Weight concentration is summarized by $\text{ESS}_q = 1/\sum_i q_i^2 \in [1, m]$.

Let $\widehat{\mathcal{J}}_m^g := \widehat{\mathcal{J}}_m(\pi_\theta; x_g, A_g)$, with $A_g = a_{1:m}^g$, be the loss of group g, and let $\bar{\ell}$ be the mean response length in the learner minibatch. The implemented loss is given in Eq. (13). If s_i is the sequence log likelihood of response i, then $\partial \widehat{\mathcal{J}}_m^g / \partial s_i = -\beta(q_i - 1/m)$. After applying the normalization above, its derivative is $-(q_i - 1/m)/(G_{\text{mb}}\bar{\ell}\sqrt{m-1})$. Writing the usual token-mean policy-gradient form as $-c_i/(G_{\text{mb}} m \bar{\ell})$ therefore gives the group coefficient

$$c_i = \frac{m(q_i - 1/m)}{\sqrt{m-1}}. \tag{29}$$

The factors $1/\beta$ and $1/\bar{\ell}$ remove the raw β prefactor and convert the group average to a token mean; the factor $1/\sqrt{m-1}$ keeps the coefficient scale similar to GRPO's standardized advantages. Thus β affects the implemented update through q, not through a remaining global multiplier.

Binary-reward limit and weight concentration. Let $\mathcal{S}$ be the k responses of a mixed-reward group with verifier reward one, and write $\Delta_i = \log[\pi_\theta(a_i \mid x)/\pi_{\text{ref}}(a_i \mid x)]$. For fixed finite Δ_i, as $\beta \to 0$ the inter-class logit gap in Eq. (12) dominates, so q concentrates on $\mathcal{S}$ with $q_i \to \text{softmax}_{i \in \mathcal{S}}(-\Delta_i)$ and $q_i \to 0$ otherwise. By Eq. (29), every response outside $\mathcal{S}$ then has $c_i \to -1/\sqrt{m-1}$, and the signed sum over correct responses is

$$\sum_{i \in \mathcal{S}} c_i \longrightarrow \frac{m-k}{\sqrt{m-1}}.$$

For GRPO's sample-standardized reward advantages, omitting the 10^{-6} stabilizer, the corresponding advantage sum is $\sqrt{k(m-k)(m-1)/m}$. For AGRO, equal learner-to-reference log ratios within the group cancel in leave-one-out centering, giving an advantage sum of $k(m-k)/(m-1)$. Normalizing each curve to its value at $k = 7$ gives Figure 6.

Because allocation inside $\mathcal{S}$ follows q rather than being uniform, a correct response can receive $c_i < 0$. In this limit that happens exactly when $(m-1)e^{-\Delta_i} < \sum_{j\in\mathcal{S}\setminus\{i\}} e^{-\Delta_j}$; equivalently, writing $\bar{\Delta}_{-i} = -\log[(k-1)^{-1}\sum_{j\in\mathcal{S}\setminus\{i\}} e^{-\Delta_j}]$, when $\Delta_i - \bar{\Delta}_{-i} > \log[(m-1)/(k-1)]$. A lone correct response ($k=1$) can never be penalized this way.

If the learner-to-reference log ratios Δ_i are equal across the successful responses $\mathcal{S}$, the binary-reward limit above yields

$$c_i \longrightarrow \frac{1}{\sqrt{m-1}}\left(\frac{mr_i}{k} - 1\right).$$

The expression in parentheses is the centered success-normalized advantage of MaxRL (Tajwar et al., 2026). MaxRL derives this update from an on-policy maximum-likelihood formulation, whereas GGPO derives its log-mean-exp loss from KL-regularized duality and permits behavior–learner mismatch without learner-to-behavior importance weights. For equally rewarded responses,

$$\frac{q_i}{q_j} = \exp[-(\Delta_i - \Delta_j)]$$

at every $\beta > 0$. Thus, reference-dependent weighting persists even at small temperatures when these log ratios differ.

The reward-only ablation sets $q_i = \mathrm{softmax}_i(r_i/\beta)$ in Eq. (29), retaining the score-gradient update and normalization. For mixed binary-reward groups, its coefficients approach the MaxRL form above as $\beta \to 0$, without requiring equal Δ_i. This removes the reference-dependent weighting that persists in full GGPO at small temperatures.

Expected implemented target. The normalization in Eq. (13) is batch dependent, so the implemented target is not simply a constant multiple of H_m. Let t index a fixed-reference window, let s index a frozen-behavior collection in that window, let $B = (A_1, \ldots, A_{G_{\mathrm{mb}}})$ be a retained learner minibatch, and let $\mathcal{P}_{t,s}$ denote its conditional distribution after mixed-group filtering and prompt refill. The expected target for that collection is

$$\mathcal{H}^{(t,s)}_{\mathrm{train}}(\pi_\theta) := \mathbb{E}_{B\sim\mathcal{P}_{t,s}}\left[\frac{G_{\mathrm{mb}}^{-1}\sum_{g=1}^{G_{\mathrm{mb}}} \widehat{\mathcal{J}}^{\,g}_{m,t}(\pi_\theta)}{\beta\bar{\ell}(B)\sqrt{m-1}}\right], \tag{30}$$

where $\widehat{\mathcal{J}}^{\,g}_{m,t}(\pi_\theta) := \widehat{\mathcal{J}}_m(\pi_\theta; x_g, A_g)$ uses the reference active in window t. Stochastic updates estimate this target conditional on the active reference and behavior collection.

For a fixed prompt x with $0 < \Pr_{a\sim\pi_{\mathrm{beh}}(\cdot|x)}[r(x,a)=1] < 1$, define $E_{\mathrm{mix}} = \{\min_i r(x,a_i) = 0,\ \max_i r(x,a_i) = 1\}$ and

$$H^{\mathrm{mix}}_m(\pi_\theta; \pi_{\mathrm{beh}}, x) := \mathbb{E}_{a_{1:m}\sim\pi^m_{\mathrm{beh}}}[\widehat{\mathcal{J}}_m(\pi_\theta; x, a_{1:m}) \mid E_{\mathrm{mix}}]. \tag{31}$$

Proposition 11 (Zero set under mixed-group filtering)**.** *Suppose $m \geq 2$, $\mathcal{A}$ is finite, the verifier reward is deterministic and binary, and $0 < \Pr_{a\sim\pi_{\mathrm{beh}}}[r(x,a)=1] < 1$. Then $H^{\mathrm{mix}}_m(\pi_\theta; \pi_{\mathrm{beh}}, x) = 0$ if and only if $R_\theta(x,a)$ is constant on the support of π_{beh}. Hence, for each such prompt, the filtered loss has the same zero set on behavior support as the unfiltered target H_m and the population objective $\mathcal{J}_{\mathrm{beh}}$. Positive response lengths and the positive normalization in Eq.* (30) *preserve this zero set over the retained prompt distribution.*

Proof. The empirical loss is nonnegative and is zero exactly when all adjusted rewards in the group agree. Any behavior-supported reward-zero action and reward-one action can co-occur in a mixed group with positive probability, so a zero conditional expectation forces their adjusted rewards to agree. Two supported actions with the same reward can each be connected through any supported action of the opposite reward. Hence all supported actions have the same adjusted reward. The converse is immediate. Finally, a positive batch-dependent multiplier cannot create or remove zeros of a nonnegative minibatch loss. □

A.7 Effective anchoring strength and the role of refresh

This calculation explains the within-reward-class weighting in Section 4.3 and why the reference reset in Section 4.4 changes those weights.

For verifier rewards, define the sequence adjusted reward $R_i = r_i - \beta\Delta_i$, where

$$\Delta_i = \sum_\tau \log \frac{\pi_\theta(a_i^\tau \mid a_i^{<\tau})}{\pi_{\mathrm{ref}}(a_i^\tau \mid a_i^{<\tau})}. \tag{32}$$

Then Eq. (12) reads

$$q_i = \mathrm{softmax}_i(r_i/\beta - \Delta_i). \tag{33}$$

In the binary-reward limit of Appendix A.6, the weights concentrate on the highest-reward subset $\mathcal{S}$, within which $q_i \propto e^{-\Delta_i}$. Thus the learner-to-reference log ratio controls how weight is distributed among equally rewarded responses and is not attenuated by β. These weights favor successful responses with smaller learner-to-reference log ratios. At an exact policy-copy boundary, $\pi_{\mathrm{ref}} = \pi_\theta$, so these log ratios are zero for every response; subsequent learner updates change them and determine the within-class weights.

Comparison with AGRO at finite loss. For a fixed sampled group, write $P = I - \mathbf{1}\mathbf{1}^\top/m$. Under the common token-mean prefactor $1/(G_{\mathrm{mb}}m\bar{\ell})$, Eqs. (29) and (48) give

$$c^{\mathrm{J}} = \frac{m}{\sqrt{m-1}}(q - \mathbf{1}/m), \qquad c^{\mathrm{A}} = \frac{m}{m-1}P(r - \beta\Delta). \tag{34}$$

Their sensitivity to learner-to-reference log ratios is

$$D_{\mathrm{J}} := -\frac{\partial c^{\mathrm{J}}}{\partial \Delta} = \frac{m}{\sqrt{m-1}}\big(\mathrm{Diag}(q) - qq^\top\big), \qquad D_{\mathrm{A}} := -\frac{\partial c^{\mathrm{A}}}{\partial \Delta} = \frac{\beta m}{m-1}P. \tag{35}$$

These derivatives describe recomputed coefficients; AGRO's advantages remain detached in its optimized surrogate. No zero-loss assumption is needed. For a mixed binary-reward group, suppose the $k \geq 2$ successful responses have equal weights, with total mass $Q = \sum_{i\in\mathcal{S}} q_i$. For any nonzero vector v supported on $\mathcal{S}$ with $\mathbf{1}^\top v = 0$,

$$\frac{\|D_{\mathrm{J}}v\|_2}{\|D_{\mathrm{A}}v\|_2} = \frac{Q\sqrt{m-1}}{k\beta}. \tag{36}$$

Thus, when reward separation gives $Q \approx 1$, GGPO can respond much more strongly to relative log-ratio changes among successful responses: for example, the ratio is approximately 13.2 at $m = 8$, $k = 2$, and $\beta = 0.1$. An exact reference copy removes accumulated relative log-ratio differences and restores equal weights within each reward class. This local comparison depends on the implemented normalizations and the current weights; D_{J} tends to zero as q concentrates on a single response. It does not imply a stronger global KL constraint or order the resulting optimizer steps. AGRO and GGPO share the same Gibbs target at a given β and reference. Appendix D.5 compares refresh cadences and examines this reweighting in the available training histories.

A.8 Temperature composition and consecutive empirical GGPO

This subsection supports Section 4.4. Exact composition characterizes the sequence of Gibbs targets attained by zero-loss windows. The empirical theorem bounds the effect of imperfect windows. The proof also establishes the simpler population-loss statement in the main text.

Fixed-reference bias and exact composition. For binary reward with reference success probability $s \in (0,1)$, the zero-loss Gibbs policy succeeds with probability $se^{1/\beta}/(1 - s + se^{1/\beta}) < 1$. Best-of-$K$ from the same reference succeeds with probability $1 - (1-s)^K$, which can be strictly larger. This gap remains at zero loss for every group size, including the infinite-group limit. Reference refresh changes the target itself.

Lemma 12 (Exact reference composition)**.** *Suppose the policies have common positive support and, in every window t, π_t has zero population finite-group loss for a group size $m_t \geq 2$ relative to reference π_{t-1}. Then temperatures $\beta_t > 0$ compose as $\pi_T(a \mid x) \propto \pi_0(a \mid x)\exp\{r(x,a)\sum_{t=1}^T \beta_t^{-1}\}$. In particular, for constant β and the best-of-K selection policy u_K from π_0,*

$$J_K(\pi_0) - J(\pi_T) \leq \frac{\beta \log K}{T}.$$

Proof. Under common positive support, Proposition 4 identifies zero population finite-group loss with the Gibbs policy. With reference π_{t-1} in window t, Eq. (2) therefore gives $\pi_t(a \mid x) \propto \pi_{t-1}(a \mid x)e^{r(x,a)/\beta}$. Iterating yields Eq. (14). More generally, temperatures $\beta_t > 0$ give effective temperature $\beta_{\text{eff}} = (\sum_{t=1}^{T} \beta_t^{-1})^{-1}$. Write $\mathcal{K}(u, v) = \mathbb{E}_X D_{\text{KL}}(u(\cdot \mid X) \| v(\cdot \mid X))$. For constant β, the final policy maximizes $J(\pi) - (\beta/T)\mathcal{K}(\pi, \pi_0)$, so for every comparator u,

$$J(u) - J(\pi_T) \le (\beta/T)\mathcal{K}(u, \pi_0).$$

Let u_K be the distribution of the selected response among K independent draws from π_0, using a fixed tie-breaking rule. Selection of action a requires at least one draw of a, hence $u_K(a \mid x) \le K\pi_0(a \mid x)$. Consequently $J(u_K) = J_K(\pi_0)$ and $\mathcal{K}(u_K, \pi_0) \le \log K$, proving $J_K(\pi_0) - J(\pi_T) \le \beta \log K/T$. □

Empirical windows. The reference assignment is the same as in the algorithm; the next result allows imperfect empirical optimization. Fix $\beta > 0$, $r \in [0, 1]$, the prompt distribution ρ, a full-support initial policy π_0, and a finite action space. Before window t, condition on the entire preceding history. Set the reference to π_{t-1} and fix a positive behavior $\pi_{\text{beh},t}$ and hypothesis class Π_t. Collect n_t conditionally independent groups, each with a fresh prompt from ρ and $m_t \ge 2$ independent responses from $\pi_{\text{beh},t}$. For any $\pi_\theta \in \Pi_t$, define

$$R_{t,\theta}(x, a) = r(x, a) - \beta \log \frac{\pi_\theta(a \mid x)}{\pi_{t-1}(a \mid x)}, \qquad L_t(\pi_\theta) = \mathbb{E}_X H_{m_t}(\pi_\theta; \pi_{\text{beh},t}, X; \pi_{t-1}),$$

where the final argument records the active reference. Here t indexes windows: L_t and $\widehat{L}_t$ abbreviate their population and empirical losses at group size m_t and sample count n_t, respectively. Assume uniformly over $x, a, \pi_\theta \in \Pi_t$ that

$$\text{osc}_a(R_{t,\theta}/\beta) \le D_t, \qquad \pi_\theta/\pi_{\text{beh},t} \le C_t, \qquad u_K/\pi_{\text{beh},t} \le \tilde{C}_t. \tag{37}$$

These bounds concern the full support, not just observed responses. The comparator coverage $\tilde{C}_t$ is additional to learner coverage C_t. The group loss has envelope $\Lambda_t = \beta D_t$. Suppose its family has a uniform η_t-cover with centers in the family of size $\mathcal{N}_{\infty,t}(\eta_t)$. Classes, covers, and bounds are fixed conditionally before collecting the window's data. Put

$$\gamma_t = 3\eta_t + \frac{4\Lambda_t}{n_t} \log \frac{2T\mathcal{N}_{\infty,t}(\eta_t)}{\delta}. \tag{38}$$

Theorem 13 (Consecutive empirical GGPO with reference refresh)**.** *Under these conditions, with probability at least $1 - \delta$, the data-selected window outputs $\pi_t \in \Pi_t$ and their uniform mixture $\bar{\pi}_T = T^{-1} \sum_{t=1}^{T} \pi_t$ satisfy*

$$J_K(\pi_0) - J(\bar{\pi}_T) \le \frac{\beta \log K}{T} + \frac{1}{T} \sum_{t=1}^{T} \sqrt{\frac{2\beta e^{D_t}(\tilde{C}_t + C_t)}{1 - 1/m_t} [2\widehat{L}_t(\pi_t) + \gamma_t]}. \tag{39}$$

If additionally $\widehat{L}_t(\pi_t) \le \inf_{\pi_\theta \in \Pi_t} \widehat{L}_t(\pi_\theta) + \xi_t$, the bracket may be replaced by $4\alpha_t + 2\xi_t + 3\gamma_t$, where $\alpha_t = \inf_{\pi_\theta \in \Pi_t} L_t(\pi_\theta)$.

Proof. First, an adjusted reward R with $\text{osc}(R/\beta) \le D$ obeys the finite-group variance bound

$$H_m \ge \frac{(1 - 1/m)e^{-D}}{2\beta} \text{Var}_{\pi_{\text{beh}}}(R). \tag{40}$$

For a realized group, set $z_i = R_i/\beta$ and $g(s) = \log(m^{-1} \sum_i e^{sz_i})$. For $s \in [0, 1]$, the normalized weight $e^{sz_i}/\sum_j e^{sz_j}$ of every group member is at least e^{-D}/m. The minimum-mean-square characterization of variance therefore gives $g''(s) = \text{Var}_s(z) \ge e^{-D} \text{Var}_{\text{group}}(z)$. Integrating with weight $1 - s$ bounds the group loss below by $\beta e^{-D} \text{Var}_{\text{group}}(z)/2$. Its expectation is Eq. (40), since empirical variance with denominator m has expectation $(1 - 1/m) \text{Var}_{\pi_{\text{beh}}}(R/\beta)$.

For the returned policy π_t, abbreviate R_t for $R_{t,\theta}$ with $\pi_\theta = \pi_t$. Let $\bar{R}_t = \mathbb{E}_{\pi_{\text{beh},t}} R_t$. Since both density ratios have $\pi_{\text{beh},t}$-mean one,

$$\mathbb{E}_{u_K} R_t - \mathbb{E}_{\pi_t} R_t = \mathbb{E}_{\pi_{\text{beh},t}} \left[\left(\frac{u_K}{\pi_{\text{beh},t}} - \frac{\pi_t}{\pi_{\text{beh},t}} \right) (R_t - \bar{R}_t) \right].$$

Cauchy–Schwarz and Eq. (37) then give pointwise

$$\mathbb{E}_{u_K} R_t - \mathbb{E}_{\pi_t} R_t \le \sqrt{(\tilde{C}_t + C_t)\,\mathrm{Var}_{\pi_{\mathrm{beh},t}}(R_t)}.$$

Indeed, $\mathbb{E}_{\pi_{\mathrm{beh},t}}[(u_K/\pi_{\mathrm{beh},t} - \pi_t/\pi_{\mathrm{beh},t})^2] \le \tilde{C}_t + C_t$, because both densities integrate to one and their cross product is nonnegative. Using Eq. (40) and Jensen over prompts bounds this residual difference by $\{2\beta e^{D_t}(\tilde{C}_t + C_t)L_t(\pi_t)/(1 - 1/m_t)\}^{1/2}$. The exact identity

$$\begin{aligned} J(u_K) - J(\pi_t) = \beta[\mathcal{K}(u_K, \pi_{t-1}) - \mathcal{K}(u_K, \pi_t) - \mathcal{K}(\pi_t, \pi_{t-1})] \\ + \mathbb{E}_X[\mathbb{E}_{u_K} R_t - \mathbb{E}_{\pi_t} R_t] \end{aligned}$$

then telescopes over windows. Dropping nonnegative terminal and step KL terms and using $\mathcal{K}(u_K, \pi_0) \le \log K$ yields the population version of Eq. (39) with $L_t(\pi_t)$ in the bracket.

Conditionally apply Lemma 9 with failure probability δ/T in each window. A union bound gives simultaneously $L_t(\pi_t) \le 2\widehat{L}_t(\pi_t) + \gamma_t$. The same two-sided generalization argument used for Proposition 10 gives $L_t(\pi_t) \le 4\alpha_t + 2\xi_t + 3\gamma_t$ under the stated empirical optimization condition. Substitution completes the proof. □

Proof of Proposition 2. The deterministic, population-loss part of the preceding proof requires no sampling or hypothesis-class assumptions. Set $m_t = m$, $D_t = D$, and $\tilde{C}_t = C_t = C$ in that argument. Jensen's inequality gives $T^{-1}\sum_t \sqrt{L_t(\pi_t)} \le \sqrt{\bar{L}_T}$, yielding Eq. (15). If $\bar{L}_T = 0$, every window has zero population loss for all prompts that contribute to its expectation, since these losses are nonnegative. The common zero set and the exact-composition argument above then also give the final-iterate reward bound. □

Temperature and group-size implications. The bound permits any $\beta > 0$: increasing T reduces the initial-reference term $\beta \log K/T$. Larger temperatures require correspondingly accurate windows because the residual term also depends on β. A sufficient range bound is $D_t \le 1/\beta + B_t$ if $\mathrm{osc}\log(\pi_\theta/\pi_{t-1}) \le B_t$ throughout the class. Thus larger β helps the reward part of the range, while reference refresh resets this log ratio at the start of a window; it does not itself enforce a range bound throughout training. The result holds for every $m_t \ge 2$. The factor $(1 - 1/m_t)^{-1}$ improves with group size, but the loss, cover size, and total sampling cost may also change. With the other constants controlled, the empirical reward remainder scales as $\sqrt{\beta\gamma_t}$ at small empirical loss, giving a conditional $1/\varepsilon^2$ prompt requirement up to entropy and other constants.

The reward of $\bar{\pi}_T$ equals that obtained by choosing a window output uniformly at random. The theorem does not establish the same bound for the final iterate or convergence of the implemented optimizer. The bound uses raw loss on unfiltered groups, even if the output is trained using filtering and normalization. Fixed offline data can support the deterministic population-loss inequality under coverage, but applying its empirical version across data-dependent reference updates requires a joint uniform class covering the references and learners, or independent held-out data. Repeated groups on the same problems do not count as fresh prompts.

A.9 Relation to AGRO: approximation and weighting

This calculation supports the comparison in Section 2: AGRO is the local quadratic approximation to the population contrastive loss when adjusted rewards have small spread. The finite-group comparison below also explains the role of exponential weighting and the slack in a uniform variance upper bound. Neither comparison globally orders the unscaled losses.

For the local relation to AGRO (Tang et al., 2025), let $\bar{R}_\theta = \mathbb{E}_{\pi_{\mathrm{beh}}} R_\theta$ and $\tilde{R} = R_\theta - \bar{R}_\theta$. The population contrastive loss can be written as the centered log moment

$$\mathcal{J}_{\mathrm{beh}} = \beta \log \mathbb{E}_{\pi_{\mathrm{beh}}} e^{\tilde{R}/\beta}. \tag{41}$$

Because $\mathbb{E}_{\pi_{\mathrm{beh}}} \tilde{R} = 0$, Taylor expansion as $\|\tilde{R}\|_\infty/\beta \to 0$ gives

$$\mathcal{J}_{\mathrm{beh}} = \frac{1}{2\beta}\mathbb{E}_{\pi_{\mathrm{beh}}} \tilde{R}^2 + O\left(\frac{\|\tilde{R}\|_\infty^3}{\beta^2}\right) = \frac{1}{2\beta}\,\mathrm{Var}_{\pi_{\mathrm{beh}}}[R_\theta] + O\left(\frac{\|\tilde{R}\|_\infty^3}{\beta^2}\right). \tag{42}$$

AGRO's off-policy population loss is $\mathcal{L}_{\mathrm{AGRO}} = \frac{1}{2}\operatorname{Var}_{\pi_{\mathrm{beh}}}[R_\theta]$, so locally $\mathcal{J}_{\mathrm{beh}} = \mathcal{L}_{\mathrm{AGRO}}/\beta$ up to higher-order terms. Equation (4) of Tang et al. (2025) gives the leave-one-out finite-group stochastic gradient of this variance objective. Our implementation divides out its exterior β factor and averages over response tokens (Eq. (48)). The corresponding coefficient relation follows by defining the exponentially tilted distribution

$$\frac{dq_\theta}{d\pi_{\mathrm{beh}}} = \frac{e^{R_\theta/\beta}}{\mathbb{E}_{\pi_{\mathrm{beh}}} e^{R_\theta/\beta}} = 1 + \frac{\tilde{R}}{\beta} + O\left(\frac{\|\tilde{R}\|_\infty^2}{\beta^2}\right). \tag{43}$$

Thus the raw GGPO coefficient multiplying the policy score is

$$-\beta\left(\frac{dq_\theta}{d\pi_{\mathrm{beh}}} - 1\right) = -\tilde{R} + O\left(\frac{\|\tilde{R}\|_\infty^2}{\beta}\right). \tag{44}$$

The leading term $-\tilde{R}$ is exactly the score-function coefficient obtained by differentiating the normalized AGRO loss $\mathcal{L}_{\mathrm{AGRO}}/\beta$, holding the behavior distribution fixed. Hence the scaled AGRO loss is the local quadratic approximation to the raw population GGPO loss, and its centered linear coefficient is the first-order approximation to the raw GGPO coefficient when $\|\tilde{R}\|_\infty/\beta \to 0$. Our default training setting uses a smaller temperature relative to verifier-reward differences, so we treat AGRO as a distinct objective rather than an approximation along the observed training trajectory.

Remark 14 (Finite-group weighting and a variance upper bound). For a fixed group of size $m \geq 2$, write $\bar{R} = m^{-1}\sum_i R_i$ and $A_m = (2m)^{-1}\sum_i (R_i - \bar{R})^2$, the half-variance with denominator m. Define $q_{s,i} = e^{sR_i/\beta}/\sum_j e^{sR_j/\beta}$ for $s \in [0,1]$. The raw GGPO loss has the exact representation and conservative bound

$$\widehat{\mathcal{J}}_m = \frac{1}{\beta}\int_0^1 (1-s)\operatorname{Var}_{q_s}(R)\, ds \leq \frac{m}{\beta} A_m. \tag{45}$$

Indeed, for $g(s) = \log(m^{-1}\sum_i e^{sR_i/\beta})$, $g''(s) = \operatorname{Var}_{q_s}(R)/\beta^2$; Taylor's integral formula gives the equality. Also $\operatorname{Var}_{q_s}(R) \leq \sum_i q_{s,i}(R_i - \bar{R})^2 \leq \sum_i (R_i - \bar{R})^2 = 2mA_m$, which gives the inequality. Locally, as adjusted-reward spread divided by β vanishes, $\widehat{\mathcal{J}}_m \sim A_m/\beta$ for nonconstant groups. Thus discarding the adaptive weights yields an upper bound whose ratio to the local loss approaches m. This is slack in this particular bound, not an unavoidable factor for every variance-based analysis or a global ordering of $\widehat{\mathcal{J}}_m$ and A_m/β. For the unbiased half-variance $A_m^{\mathrm{unb}} = mA_m/(m-1)$, the same bound reads $\widehat{\mathcal{J}}_m \leq (m-1)A_m^{\mathrm{unb}}/\beta$. These are raw objective comparisons, separate from the normalization of the implemented updates.

Let $\mathcal{V}_{\mathrm{AGRO}}(\pi_\theta) := \mathbb{E}_X[\frac{1}{2}\operatorname{Var}_{\pi_{\mathrm{beh}}}(R_\theta)]$ be the prompt-averaged population half-variance. Taking expectations gives $\mathbb{E}A_m^{\mathrm{unb}} = \mathcal{V}_{\mathrm{AGRO}}(\pi_\theta)$. Proposition 5 therefore yields, under its conditions,

$$\begin{aligned} L_m(\pi_\theta) &\leq \frac{m-1}{\beta}\mathcal{V}_{\mathrm{AGRO}}(\pi_\theta), \\ J_K(\pi_{\mathrm{beh}}) - J(\pi_\theta) &\leq \kappa_{K,m}\left\{\beta(\log m + \Gamma) + \frac{C(m-1)}{\beta}\mathcal{V}_{\mathrm{AGRO}}(\pi_\theta)\right\}. \end{aligned} \tag{46}$$

B Conceptual comparison of group objectives

Table 2 distinguishes the reference policy, PPO denominator, and importance correction: a method may evaluate a learner-to-reference log ratio without using a learner-to-behavior ratio as an importance weight.

Table 2: **Objective-level comparison.** "Learner-to-behavior likelihood term" records any explicit dependence on both learner and behavior likelihoods.

Method	Group training signal	Reference policy	PPO denominator	Learner-to-behavior likelihood term
GGPO	Exponential duality gap Adjusted-reward	Separate reference	None	None
AGRO (Tang et al., 2025)	variance Optimal-advantage	Separate reference	None	None
OAPL (Ritter et al., 2026)	regression	Behavior policy	None	Log-ratio regression
GRPO+TIS (Zhong et al., 2026)	PPO group advantage	GRPO reference	Old trainer	Trainer/rollout correction
InfoNCA (Chen et al., 2024)	Reward-softmax classification	Base/reference policy	None	None

GGPO and AGRO use the same adjusted reward and share its constancy condition; their finite-group geometries differ. A learner-to-behavior term need not be an importance weight: OAPL explicitly regresses a learner-to-behavior log ratio, whereas GRPO+TIS in Zhong et al. (2026) multiplies the PPO surrogate by a truncated old-trainer-to-rollout correction weight. Its PPO ratio compares the current learner to the old trainer policy. InfoNCA uses learner-to-reference logits and reward-softmax labels, whereas GGPO minimizes $D_{\mathrm{KL}}(U_m \| q)$ with an induced adjusted-reward tilt.

C Experimental details

C.1 Data and evaluation

Training data. ALP-Math combines the mathematics split of Guru-RL (Cheng et al., 2025) with an OpenR1-derived dataset (weqweasdas, 2025).[1] We use the 54,404 problems in `train/math_combined_54.4k.parquet` from `LLM360/guru-RL-92k`. From `weqweasdas/from_default_filtered_openr1_with_scores`, we retain the 23,036 examples whose upstream mean verifier score lies strictly between 0.2 and 0.8. We merge the two sources and deduplicate by exact user-content string after applying the common prompt format, within and across sources. We keep the first occurrence, processing Guru-RL before OpenR1. This reduces 77,440 records to 73,293 unique prompts. Length filtering retains 73,290 prompts: each input has at most 2,048 tokens after applying the chat template, including system and user instructions and the generation prompt. We independently checked the training data against all five evaluation benchmarks and found no overlapping problems. The Qwen3-1.7B-Base default-setting dataloaders shuffle with seeds 42, 1234, and 2027. The $N = 128$ condition also uses three training seeds; other settings with periodic behavior synchronization use seed 42.

Training-time evaluation uses MATH-500 from PRM800K. Final evaluation additionally uses AIME 2024, AIME 2025, OlympiadBench and Minerva Math (Lewkowycz et al., 2022). MATH-500 is the PRM800K subset of MATH (Hendrycks et al., 2021; Lightman et al., 2024), and OlympiadBench is the text-only mathematics split (He et al., 2024). Ground truths are their boxed answers. The evaluation sets contain 500 MATH-500 problems, 30 AIME 2024 problems, 30 AIME 2025 problems, 674 OlympiadBench problems, and 272 Minerva Math problems, for 1,506 problems in total. Every problem-weighted average we report uses these dataset sizes as weights. The accompanying preprocessing code pins these evaluation datasets to fixed revisions.[2]

Models and prompts. We use Qwen3-1.7B-Base and Qwen3-4B-Base with their corresponding tokenizers. The exact system prompt is "Please reason step by step, and put your final answer within `\boxed{}`." Each user problem is followed by "Let's think step by step and output the final answer within `\boxed{}`."

Reward and decoding. The binary reward uses Math-Verify (Kydlíček, 2025) to verify answers against the ground truth. We use no separate format bonus or length penalty.

[1] Dataset revisions: Guru-RL, `2e2790a962a3c099bfb5ea61389cbf98a5ea439b`; OpenR1, `da327b36b4dc077240aa8efd7f25c4017cd74ef4`.

[2] MATH-500: `6e4ed1a`; AIME 2024: `83a7f38`; AIME 2025: `563bb84`; OlympiadBench: `4faaf1e`; Minerva Math: `ee46ddc`.

Training rollouts use vLLM (Kwon et al., 2023) at sampling temperature 1.0, top-$p = 1.0$, top-k disabled, 2,048-token prompt and response maxima, and $m = 8$ responses per prompt. Training-time and final evaluation use the same sampling temperature, top-p, top-k, and length settings. Each checkpoint is evaluated once, drawing 32 responses per problem. The end-of-sequence (EOS) token is included in the response probability, and the response mask includes generated tokens through the first EOS token while excluding subsequent padding. If generation reaches the 2,048-token response maximum without EOS, all generated tokens define the maximum-length-truncated prefix; no unobserved EOS factor is included.

Trainer and evaluation policies. The bounds use π_θ, the policy defined by the trainer likelihoods. Evaluation samples from $\pi_\theta^{\mathrm{eval}}$, the rollout implementation of the same checkpoint; mean@32 estimates $J(\pi_\theta^{\mathrm{eval}})$. For a common prompt distribution ρ and reward function $r \in [0, 1]$,

$$\left|J(\pi_\theta^{\mathrm{eval}}) - J(\pi_\theta)\right| \le \mathbb{E}_{X\sim\rho}\,\mathrm{TV}\big(\pi_\theta^{\mathrm{eval}}(\cdot \mid X), \pi_\theta(\cdot \mid X)\big)\,,$$

where $\mathrm{TV}(p, q) = \frac{1}{2}\sum_a |p(a) - q(a)|$ compares distributions over complete responses. Subtracting this term transfers a learner reward lower bound to the evaluation policy.

Best-of-K evaluation. For binary rewards, best@K and pass@K both estimate $J_K(\pi_\theta^{\mathrm{eval}})$ in Eq. (8). Given c correct responses out of 32, the unbiased estimator is $1-\binom{32-c}{K}/\binom{32}{K}$ for $1 \le K \le 32$ (Chen et al., 2021), with $\binom{32-c}{K} = 0$ when $K > 32 - c$. It averages the maximum reward over all size-K subsets: at $K = 1$, it equals $c/32$ (mean@32); at $K = 32$, it equals $\mathbf{1}\{c > 0\}$. Reported scores average these values over problems.

Mixed-reward filtering and refill. In the default setting ($N = 8$), each behavior collection draws the next 256 prompts from the shuffled dataloader and generates m responses per prompt under one frozen pre-update policy. We retain groups containing at least one correct and one incorrect response and continue sampling until the collection is full. Each collection allows at most 32 generation batches, including the initial batch; training stops with an error if the target batch cannot be filled. For a prompt with success probability p, the probability that a sampled group is retained is $1 - p^m - (1 - p)^m$. The procedure therefore changes the training distribution by favoring prompts that produce mixed-reward groups. Groups with equal rewards carry no verifier signal for GRPO, AGRO, or GGPO, although reference regularization can still contribute an update. Within each matched comparison, methods use the same filtering rule, refill budget, batch geometry, and prompt order.

Mismatch diagnostics. K3 and ESS are computed on learner minibatches retained after mixed-reward filtering. The behavior–learner mismatch curve reports token-mean K3 (Schulman, 2020), $\exp(\lambda_\tau) - \lambda_\tau - 1$, for $\lambda_\tau = \log \pi_\theta(a^\tau \mid a^{<\tau}) - \log \pi_{\mathrm{beh}}(a^\tau \mid a^{<\tau})$, averaged over the logged response tokens. For the B_{resp} responses in a logged batch, the learner-to-behavior sequence importance weight is $w_i = \exp(\sum_\tau M_{i\tau}\lambda_{i\tau})$, and the reported normalized effective sample size is $\mathrm{ESS}_{\mathrm{IS}} = (\sum_i w_i)^2/(B_{\mathrm{resp}}\sum_i w_i^2) \in [1/B_{\mathrm{resp}}, 1]$. These sequence-level weights provide a mismatch diagnostic for all methods. For Figure 3, drift is the token mean of $|\lambda_\tau|$ over response tokens selected by the mask M. Its ESS uses the $B_{\mathrm{resp}} = 2{,}048$ responses of each logged block of eight learner minibatches, so the minimum normalized ESS is $1/2{,}048$. Gradient norms use each method's native actor-loss normalization and are therefore interpreted only within a method over training.

C.2 Training setup

Training uses AdamW (Loshchilov & Hutter, 2019): learning rate 10^{-6}, betas $(0.9, 0.999)$, weight decay 0.01, constant schedule, no warmup, and gradient clip 1. We use one actor epoch per collection and no entropy bonus. In the default setting ($N = 8$), each 256-prompt collection contains $m = 8$ responses per prompt and is divided into 32-prompt minibatches, giving $N = 8$ optimizer updates. Unless stated otherwise, every method uses $\beta = 0.01$, and neither AGRO nor GGPO applies a token-level current-to-pre-update ratio, clipping, learner-to-behavior importance weights, or an additive KL term.

Objectives and reference policies. GGPO optimizes the loss in Eq. (13). In every main comparison, AGRO and GGPO refresh $\pi_{\text{ref}} \leftarrow \pi_\theta$ every $T_{\text{ref}} = 160$ optimizer updates, because their regularized targets depend on the active reference, whereas our GRPO implementation retains its initial reference. Appendix D.5 compares a fixed reference, $T_{\text{ref}} = 160$, and $T_{\text{ref}} = 320$ for all three methods.

Default GRPO. GRPO standardizes rewards within each group of $m = 8$ responses by subtracting the group mean and dividing by the sample standard deviation plus 10^{-6}; the resulting advantages are held fixed during differentiation. The token-level PPO ratio (Schulman et al., 2017) divides the current learner probability by the behavior probability recorded during rollout. The surrogate clips this ratio to $[0.8, 1.2]$, applies dual clipping with $c = 10$ for negative advantages, and averages over response tokens selected by the mask. It uses the recorded rollout denominator directly, with no old-trainer recomputation or additional off-policy correction. A K3 KL penalty (`low_var_kl`) with coefficient 0.01 regularizes the learner toward the fixed initial reference; the entropy coefficient is zero.

GRPO with token-level truncated importance sampling. GRPO+TIS uses a token-level importance-weight cap of 2 in the $N = 128$ staleness comparison and offline training. Let μ be the behavior policy that generated the data and π_{old} the trainer policy at the start of an update block. At $N = 128$, each update block consists of 8 optimizer updates. Offline training uses blocks of 8 optimizer updates, refreshing π_{old} at each block boundary while keeping μ and its stored token log probabilities fixed throughout training. We compute each block's old-trainer token log probabilities once under its π_{old}. For token $a_{i\tau}$ with prompt-and-response-prefix context $h_{i\tau}$, the TIS weight and PPO ratio are

$$\begin{aligned} w_{i\tau} &= \min\left\{ \frac{\pi_{\text{old}}(a_{i\tau} \mid h_{i\tau})}{\mu(a_{i\tau} \mid h_{i\tau})},\, 2 \right\}, \\ r_{i\tau}(\theta) &= \frac{\pi_\theta(a_{i\tau} \mid h_{i\tau})}{\pi_{\text{old}}(a_{i\tau} \mid h_{i\tau})}. \end{aligned} \tag{47}$$

The TIS weight multiplies the token-level PPO clipped surrogate. The old-trainer log probabilities, TIS weights, and PPO denominator remain fixed within each block; the current-policy numerator changes with each learner update. Weights above 2 are capped at 2, without batch normalization or rejection sampling. All other settings match the GRPO baseline in the corresponding setting.

Default AGRO. AGRO (Tang et al., 2025) uses leave-one-out centered adjusted rewards. For response i in group g, let

$$s_{g,i}(\theta) = \sum_{\tau \in \text{response}} \log \pi_\theta(a^\tau_{g,i} \mid x_g, a^{<\tau}_{g,i}), \qquad z_{g,i} = r_{g,i} - \beta\big(s_{g,i}(\theta) - s^{\text{ref}}_{g,i}\big).$$

With $\bar{z}_g = m^{-1} \sum_i z_{g,i}$, the leave-one-out advantage is

$$a^{\text{LOO}}_{g,i} = z_{g,i} - \frac{1}{m-1} \sum_{j \neq i} z_{g,j} = \frac{m}{m-1}(z_{g,i} - \bar{z}_g).$$

We remove the exterior β factor from AGRO's gradient and use token-mean aggregation to reuse GRPO's training hyperparameters. For a global learner minibatch containing G groups with mean response length $\bar{\ell}$, the optimized surrogate is

$$L_{\text{AGRO}} = -\frac{1}{Gm\bar{\ell}} \sum_{g=1}^{G} \sum_{i=1}^{m} \text{sg}\big(a^{\text{LOO}}_{g,i}\big)\, s_{g,i}(\theta), \tag{48}$$

where $\beta = 0.01$ remains in the adjusted reward and $Gm\bar{\ell}$ is the total number of valid response tokens selected by the mask. The operator sg stops gradients through the entire advantage, including its current-policy log ratio; only the final $s_{g,i}(\theta)$ factor receives gradients. Rewards, reference log probabilities, and response masks are held fixed. The surrogate uses no PPO ratio, clipping, additional importance correction, additive KL loss, or entropy bonus. The reference is refreshed every 160 optimizer updates.

Numerical precision. The default GRPO, AGRO, and GGPO runs, as well as the $N = 128$ GRPO+TIS runs, store actor parameters in FP32, execute the forward pass with BF16 mixed precision, and reduce gradients in FP32. Rollout uses BF16; quantization and quantization-aware training are disabled.

Implementation and seeds. All three objectives are implemented in one fork of verl (Sheng et al., 2024) (version 0.9.0.dev) with FSDP actor sharding and vLLM (Kwon et al., 2023) rollout, so they share the data pipeline, verifier, refill logic, and evaluation code; their objectives and reference-policy handling follow the complete configurations above. Seeds and checkpoint reporting follow Section 5.1.

C.3 Experimental protocols

Default training and policy staleness. The default and controlled policy staleness runs use 8,000 optimizer updates. Each behavior collection supplies N distinct learner minibatches, with one update per minibatch and behavior synchronization after the collection; Section 5.3 states the minibatch partition and sample ages. For $N = 128$, the final collection is generated in full, and 64 of its minibatches are used to complete the 8,000-update budget.

Evaluation and checkpointing occur every 160 optimizer updates in every run, independently of T_{ref}.

Offline setting. The fixed rollout pool contains only mixed-reward groups generated once by Qwen3-1.7B-Base, which is also the initial reference policy. At generation time, we store each response's behavior token log probabilities and reuse them throughout training. Current learner token log probabilities are evaluated on the stored responses for the K3, drift, and ESS measurements; GRPO's learner-to-behavior ratios are formed against the stored behavior values. Methods share the same groups and order without new generation. For each seed, we prepare 400 fixed learner minibatches and perform one optimizer update on each, with evaluation every 80 updates and no response revisited. GRPO+TIS refreshes its old-trainer policy every 8 updates as specified in Appendix C.2. AGRO and GGPO use $T_{\mathrm{ref}} = 160$, while GRPO and GRPO+TIS keep the initial reference fixed.

C.4 GGPO implementation

Differentiation semantics. For response i, the learner log likelihood is $s_i = \sum_\tau M_{i\tau} \log \pi_\theta(a_i^\tau \mid a_i^{<\tau})$, with the response mask M selecting the valid generated tokens; the reference log likelihood is formed over the same masked sequence. Sampled tokens, verifier rewards, reference log likelihoods, and masks are held fixed during the actor update. We form the adjusted rewards and directly differentiate the log-mean-exp loss in Eq. (13) with respect to the learner log probabilities. A detached copy of q is used only to report the resulting coefficients and diagnostics, not as a separately optimized surrogate. Autograd therefore produces the gradient in Eq. (11) exactly, followed by the external normalization in Eq. (13).

End-to-end GGPO update. The update follows the steps in Section 4.4. Refill continues until there are enough mixed-reward groups for N distinct 32-prompt learner minibatches; the loss is evaluated with a stable log-sum-exp, and each minibatch receives one AdamW update.

When N does not divide T_{ref}, as at $N = 64$ and $N = 128$, a refresh boundary can fall between two minibatches from the same behavior collection. Step 3 of Section 4.4 refreshes the reference at that exact boundary; the already sampled behavior data remain unchanged, and step 2 evaluates the remaining minibatches under the new reference.

Weight-concentration diagnostics. With $\mathcal{S}$ and k as in Appendix A.6, over the $N \in \{8, 16, 32, 64\}$ diagnostic subset of the staleness ladder, the logged within-class normalized effective sample size—the ESS of q restricted to $\mathcal{S}$, divided by k, over groups with $k \geq 2$—has medians of 0.77–0.80, while the whole-group $\mathrm{ESS}_q = 1/\sum_i q_i^2$ has medians of 1.96–2.28 out of $m = 8$. The whole-group concentration reflects separation between reward classes.

D ADDITIONAL EXPERIMENTS

D.1 LEARNING BY PROBLEM DIFFICULTY

Figure 5 splits MATH-500 by the base model's empirical success rate $\hat{p}_0$ over 32 responses per problem, into the 410 problems with $0 < \hat{p}_0 \leq 0.75$ and the 63 with no correct response ($\hat{p}_0 = 0$), and tracks the three default-setting runs on each subset; the remaining 27 problems reach above 95% mean@32 for every method within 640 optimizer updates. Section 5.2 discusses the result together with the per-group gradient weight in Figure 6.

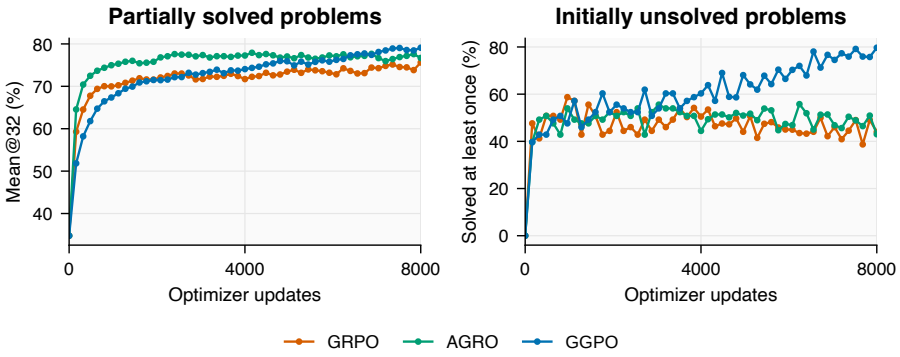

Figure 5: **MATH-500 learning curves by base-model difficulty on Qwen3-1.7B-Base.** Left: mean@32 on the 410 problems the base model solves part of the time ($0 < \hat{p}_0 \leq 0.75$). Right: fraction of the 63 initially unsolved problems (no correct response among the base model's 32 samples, $\hat{p}_0 = 0$) with at least one correct response among 32 samples. Both panels plot GRPO, AGRO, and GGPO in the default setting with seed 42 against optimizer updates.

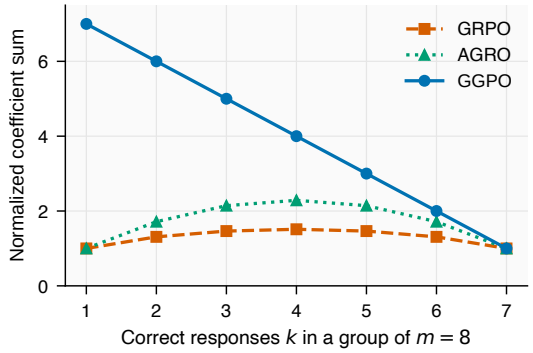

Figure 6: **Analytic coefficient sum for correct responses in a mixed-reward group.** For $m = 8$, curves show the signed sum $W_{\text{correct}} = \sum_{i:r_i=1} c_i$, normalized to its value at $k = 7$ for each method. GRPO uses standardized reward advantages, $W_{\text{correct}} \propto \sqrt{k(m-k)}$; AGRO uses leave-one-out advantages with equal learner-to-reference log ratios within the group, $W_{\text{correct}} \propto k(m-k)/(m-1)$; GGPO uses Eq. (29) in the binary-reward limit $\beta \to 0$ at fixed finite log ratios, $W_{\text{correct}} = (m-k)/\sqrt{m-1}$.

D.2 PASS@K EVALUATION

For each problem, pass@K is the probability that at least one of K independently sampled responses is correct, estimated per problem from the 32 evaluation samples using the unbiased estimator in Appendix C.1. For binary rewards, pass@K and best@K coincide; the estimated pass@1 equals mean@32. The training curves report MATH-500 best@32, while Table 3 reports problem-weighted pass@K over all five benchmarks at each method's final 8,000-update checkpoint.

Results. All three methods raise pass@K above the base model at every K, and GGPO has the highest value at every K (Table 3). Its margin over the strongest baseline grows from 1.5 points at

Table 3: **Pass@K across five benchmarks on Qwen3-1.7B-Base.** Problem-weighted pass@K (%) at the final 8,000-update checkpoint for training seed 42. The base model is evaluated under the same protocol. Bold marks the best method at each K.

Method	$K = 1$	$K = 4$	$K = 8$	$K = 16$	$K = 32$
Base model	19.24	38.57	47.43	55.43	62.68
GRPO	42.55	56.10	61.38	65.74	69.39
AGRO	45.06	57.41	61.96	65.85	69.19
GGPO	**46.60**	**60.91**	**66.22**	**70.46**	**73.90**

$K = 1$ to about 4.5 points at $K \geq 16$, while AGRO's advantage over GRPO disappears at larger sampling budgets. GGPO's advantage also appears in the MATH-500 best@32 curves and the fraction of initially unsolved problems it solves within 32 samples (Figures 1 and 5). For a rarely solved problem with success probability p and $Kp \ll 1$, the contribution to pass@K is approximately Kp, compared with p to pass@1. Improvements on such problems can therefore produce larger gains at higher sampling budgets, consistent with the difficulty analysis.

D.3 Full results under controlled policy staleness

Table 4 gives the per-benchmark final-checkpoint results across synchronization delays.

Table 4: **Full evaluation across controlled policy staleness on Qwen3-1.7B-Base.** Final-checkpoint mean@32 accuracy after 8,000 optimizer updates on five benchmarks (%), by synchronization delay N. Results at $N = 8, 128$ are three-seed means; $N = 8$ reuses Table 1, and $N = 16, 32, 64$ use seed 42. The final column gives the problem-weighted average. Sample standard deviations across seeds are reported for all columns at $N = 8, 128$.

Method (N)	MATH-500	AIME 2024	AIME 2025	OlympiadBench	Minerva Math	Avg.
GRPO ($N = 8$)	68.86 ± 0.32	9.03 ± 0.90	5.14 ± 0.63	32.13 ± 0.27	28.33 ± 0.40	42.64 ± 0.18
GRPO ($N = 16$)	68.56	8.96	5.00	31.87	28.15	42.39
GRPO ($N = 32$)	68.43	8.96	5.00	31.76	28.07	42.28
GRPO ($N = 64$)	68.28	8.85	5.00	31.62	27.98	42.15
GRPO ($N = 128$)	64.53 ± 0.85	7.71 ± 2.35	3.96 ± 1.51	28.30 ± 0.69	26.02 ± 1.01	39.02 ± 0.45
GRPO+TIS ($N = 128$)	66.46 ± 0.62	8.30 ± 1.20	4.51 ± 1.05	30.12 ± 0.47	27.09 ± 0.64	40.69 ± 0.38
AGRO ($N = 8$)	70.68 ± 0.33	9.27 ± 1.15	6.56 ± 0.81	35.73 ± 0.30	29.30 ± 0.44	45.07 ± 0.19
AGRO ($N = 16$)	70.88	9.06	6.25	35.91	29.11	45.17
AGRO ($N = 32$)	69.98	8.85	6.04	34.48	28.83	44.17
AGRO ($N = 64$)	68.29	7.29	6.25	31.07	28.10	41.92
AGRO ($N = 128$)	66.94 ± 0.67	6.35 ± 2.26	5.10 ± 1.60	30.12 ± 0.60	27.38 ± 0.92	40.88 ± 0.39
GGPO ($N = 8$)	72.43 ± 0.29	11.11 ± 0.89	8.30 ± 0.68	37.11 ± 0.25	30.64 ± 0.36	46.58 ± 0.16
GGPO ($N = 16$)	72.17	9.58	8.33	37.37	30.46	46.54
GGPO ($N = 32$)	71.70	9.27	8.96	37.51	29.70	46.32
GGPO ($N = 64$)	72.72	10.21	8.44	37.34	29.97	46.64
GGPO ($N = 128$)	71.94 ± 0.40	10.10 ± 1.10	8.33 ± 0.90	37.32 ± 0.33	30.09 ± 0.48	46.39 ± 0.21

Results. At $N = 128$, GRPO+TIS improves on GRPO across all five benchmarks, while GGPO achieves higher accuracy than GRPO+TIS on each benchmark (Table 4). GGPO also retains higher entropy than GRPO at every delay, as in the default setting (Figure 7).

D.4 Offline training details

Appendix C.3 gives the generation, logging, and refresh protocol. Table 5 reports the mean and sample standard deviation of MATH-500 accuracy across the three seeds; Figure 3 provides the corresponding curves and seed-42 diagnostics.

D.5 Reference-refresh cadence and regularization

All three methods regularize toward a reference policy; the main comparisons use the settings of Section 5.1. We compare each method under a fixed reference, $T_{\text{ref}} = 160$, and $T_{\text{ref}} = 320$, and

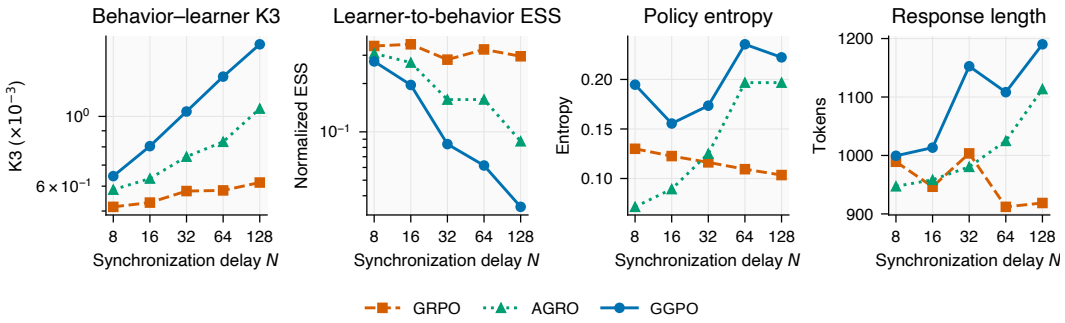

Figure 7: **Training-time diagnostics across the staleness ladder on Qwen3-1.7B-Base.** Each panel plots one metric against synchronization delay N (equally spaced positions), averaged over the last 800 optimizer updates of each run: the token-mean K3 diagnostic of behavior–learner mismatch ($\times 10^{-3}$), the normalized ESS of the diagnostic sequence-level learner-to-behavior importance weights (Appendix C.1), policy entropy, and mean response length in tokens (GRPO orange squares, AGRO green triangles, GGPO blue circles). All diagnostics use seed 42; K3 and ESS are computed on mixed-reward-filtered training responses. K3 and ESS use logarithmic vertical axes. Corresponding accuracies are in Table 4.

Table 5: **Offline MATH-500 accuracy across three seeds on Qwen3-1.7B-Base.** Mean@32 accuracy (%) at each evaluated optimizer update, reported as the mean ± sample standard deviation across training seeds 42, 1234, and 2027. Update 0 is the base model.

Optimizer updates	GRPO	GRPO+TIS	AGRO	GGPO
0	32.96	32.96	32.96	32.96
80	37.63 ± 0.38	54.58 ± 0.11	54.58 ± 0.35	54.83 ± 0.62
160	38.36 ± 0.27	57.93 ± 0.33	53.72 ± 0.44	59.31 ± 0.35
240	38.93 ± 0.31	58.28 ± 0.42	57.91 ± 0.73	59.79 ± 0.69
320	39.79 ± 0.32	54.51 ± 0.49	53.03 ± 0.68	55.83 ± 0.34
400	39.64 ± 0.72	42.71 ± 1.23	46.43 ± 0.71	48.31 ± 0.36

evaluate GGPO at $\beta = 0.1$ with a fixed reference and $T_{\text{ref}} = 160$. Table 6 reports the eleven-condition grid across all five benchmarks at the final checkpoint.

Table 6: **Reference-refresh cadence and regularization on Qwen3-1.7B-Base.** Final-checkpoint mean@32 accuracy after 8,000 optimizer updates on five benchmarks (%). "Fixed" keeps the initial reference throughout training; T_{ref} is the refresh cadence in optimizer updates. The $\beta = 0.01$ rows at each method's default cadence reuse the three-seed means from Table 1; the remaining conditions use seed 42. The final column uses the problem-weighted average of Table 1.

Method, β, reference	MATH-500	AIME 2024	AIME 2025	OlympiadBench	Minerva Math	Avg.
GRPO, $\beta = 0.01$, fixed	68.86	9.03	5.14	32.13	28.33	42.64
GRPO, $\beta = 0.01$, $T_{\text{ref}} = 160$	68.45	9.27	4.90	31.79	28.57	42.40
GRPO, $\beta = 0.01$, $T_{\text{ref}} = 320$	68.36	9.06	4.69	32.29	27.99	42.48
AGRO, $\beta = 0.01$, fixed	66.61	8.13	4.06	31.64	27.48	41.48
AGRO, $\beta = 0.01$, $T_{\text{ref}} = 160$	70.68	9.27	6.56	35.73	29.30	45.07
AGRO, $\beta = 0.01$, $T_{\text{ref}} = 320$	69.94	8.13	4.58	34.07	28.47	43.86
GGPO, $\beta = 0.01$, fixed	56.52	4.79	2.92	23.26	20.11	32.96
GGPO, $\beta = 0.01$, $T_{\text{ref}} = 160$	72.43	11.11	8.30	37.11	30.64	46.58
GGPO, $\beta = 0.01$, $T_{\text{ref}} = 320$	70.13	9.69	6.98	35.73	29.62	44.96
GGPO, $\beta = 0.1$, fixed	54.09	3.44	2.08	20.98	19.49	30.98
GGPO, $\beta = 0.1$, $T_{\text{ref}} = 160$	72.04	9.69	6.88	35.48	29.73	45.50

Results. The ablation shows that periodic reference refresh is an empirically important part of GGPO, rather than only a theoretical extension. At $\beta = 0.01$, the problem-weighted average rises from 32.96 with a fixed reference to 46.58 with the default $T_{\text{ref}} = 160$ schedule; $T_{\text{ref}} = 320$ reaches 44.96 (Table 6). Thus a fixed reference costs GGPO 13.6 points and leaves it below fixed-reference GRPO. AGRO also performs best with $T_{\text{ref}} = 160$, but its fixed-reference penalty is smaller at 3.6 points, while GRPO is essentially unchanged across the three conditions.

Table 7: **Learner-to-reference log-ratio and entropy diagnostics on Qwen3-1.7B-Base.** Mean sequence log ratio $\log[\pi_\theta/\pi_{\text{ref}}]$ over the responses in the training batch, and policy entropy, each averaged over the last 800 optimizer updates of the seed-42 run.

Method	Metric	Fixed	$T_{\text{ref}} = 160$	$T_{\text{ref}} = 320$
AGRO	Log ratio (nats)	19.2	3.8	6.6
	Entropy	0.19	0.07	0.10
GGPO	Log ratio (nats)	6.0	0.6	0.7
	Entropy	0.35	0.19	0.18

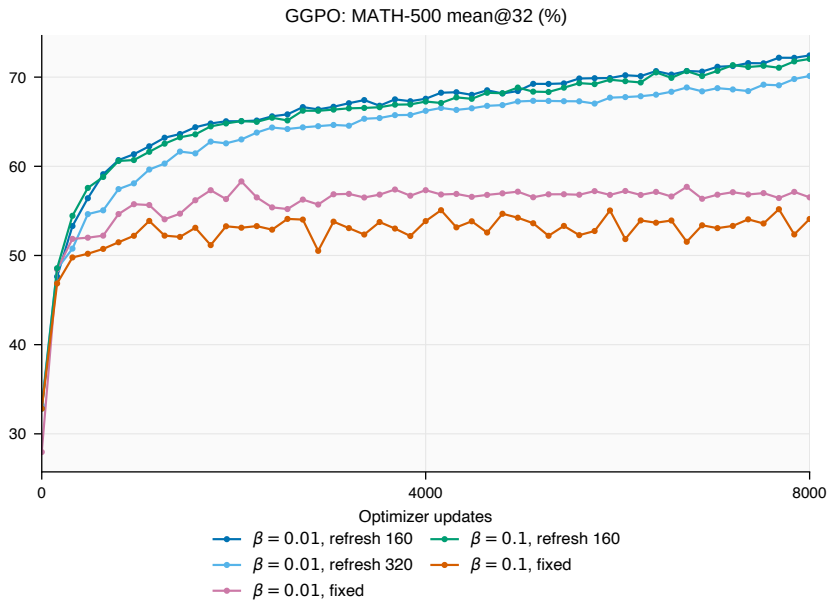

Figure 8: **GGPO training under different reference-refresh cadences and regularization strengths on Qwen3-1.7B-Base.** MATH-500 mean@32 accuracy against optimizer updates for $\beta \in \{0.01, 0.1\}$ with a fixed reference or refresh every 160 updates, and for $\beta = 0.01$ with refresh every 320 updates. All curves use seed 42. Final results across all five benchmarks are reported in Table 6.

Figure 8 shows that fixed-reference GGPO levels off after its early gains, with a lower plateau at $\beta = 0.1$. With reference refresh, accuracy continues to improve through 8,000 optimizer updates at both temperatures.

The reference shapes AGRO's and GGPO's training targets, whereas GRPO uses it only in an additive KL penalty (Section 4.4). This distinction is consistent with GRPO's small accuracy changes across refresh cadences. Appendix A.7 compares the implemented update coefficients without assuming zero training loss. AGRO's response to log-ratio changes is scaled by β, while GGPO's depends on the softmax covariance. Around equal weights on multiple successful responses, GGPO can therefore exert substantially stronger feedback against relative likelihood growth at the same β (Eq. (36)). Refreshing resets this feedback's reference point. This gives a local mechanism consistent with GGPO's larger refresh benefit.

Reference-drift diagnostics. Table 7 provides complementary empirical evidence. With the common fixed reference, GGPO's mean sequence log ratio is 6.0 nats, compared with AGRO's 19.2. The same ordering holds with $T_{\text{ref}} = 160$ (0.6 versus 3.8) and $T_{\text{ref}} = 320$ (0.7 versus 6.6). GGPO's consistently lower log ratios on the retained training responses are consistent with stronger effective reference anchoring under the implemented updates. The fixed-reference comparison is especially informative because both methods use the same initial anchor. These means are computed on each method's behavior-policy samples after mixed-reward filtering.

Weight concentration among successful responses. Figure 9 compares the distribution of GGPO's weights among successful responses with a fixed reference and with $T_{\text{ref}} = 160$ at $\beta = 0.1$. For groups with $k \geq 2$ successful responses, let $\widetilde{q}_i = q_i / \sum_{j \in \mathcal{S}} q_j$ for $i \in \mathcal{S}$, where $\mathcal{S}$ is the successful subset. The within-success normalized ESS is $[k \sum_{i \in \mathcal{S}} \widetilde{q}_i^2]^{-1} \in [1/k, 1]$; one denotes uniform weights, and smaller values indicate greater concentration. Because these responses have equal rewards, Eq. (12) gives $\widetilde{q}_i \propto e^{-\Delta_i}$, where $\Delta_i = \log[\pi_\theta(a_i \mid x)/\pi_{\text{ref}}(a_i \mid x)]$. Thus this diagnostic isolates how learner-to-reference log-ratio differences redistribute weight within the successful subset.

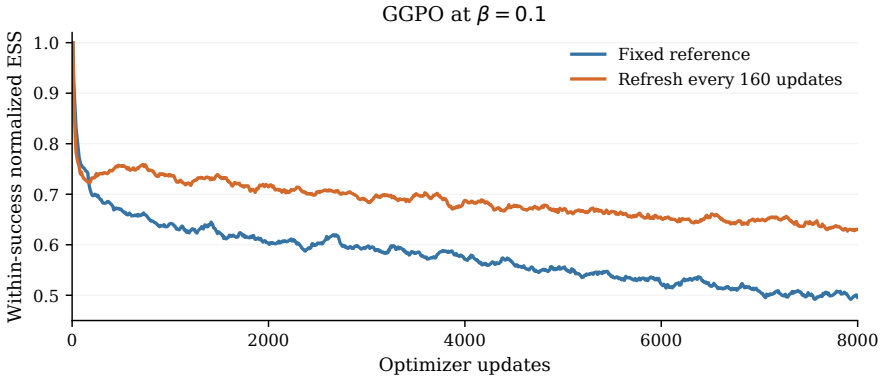

Figure 9: **Within-success weight concentration for GGPO on Qwen3-1.7B-Base.** Normalized ESS of the weights restricted to successful responses, for groups with at least two successes, at $\beta = 0.1$ with seed 42. Blue uses a fixed reference; orange refreshes every 160 optimizer updates. Higher values indicate more uniform weights among successful responses.

After the initial decline, the refreshed-reference curve remains above the fixed-reference curve, although both decrease over training. The higher ESS with refresh is consistent with limiting accumulated log-ratio differences among successful responses, as described in Appendix A.7. This provides a weight-level view of the reference dependence accompanying the accuracy trends in Figure 8.

Averaging over the final five recorded refresh windows and the matching fixed-reference interval gives normalized ESS values of 0.582 with a fixed reference and 0.690 with refresh; the corresponding fractions of groups with $k \geq 2$ are 0.758 and 0.839. The raw group loss remains nonzero, averaging 0.619 and 0.531, respectively. These diagnostics establish the presence of reference-dependent reweighting during finite-loss training but do not isolate its contribution to the accuracy gain.

D.6 Ablations on GGPO's group weights

GGPO's weights are $q_i = \text{softmax}_i(r_i/\beta - \Delta_i)$, where $\Delta_i = \log[\pi_\theta(a_i \mid x)/\pi_{\text{ref}}(a_i \mid x)]$. We vary the reward temperature β and remove the reference-dependent term $-\Delta_i$ under the default Qwen3-1.7B-Base configuration with $N = 8$.

Temperature choice. Reducing β from 0.01 to 0.001 changes GGPO's problem-weighted average from 46.58 to 46.78, while AGRO decreases from 45.07 to 44.37. We use $\beta = 0.01$ in the main comparisons.

Removing the reference-dependent term. Under the default $N = 8$ setting, we isolate the contribution of reference-dependent weighting by replacing $q_i = \text{softmax}_i(r_i/\beta - \Delta_i)$ with $q_i = \text{softmax}_i(r_i/\beta)$ at $\beta = 0.01$, while retaining the remaining GGPO update. The reward-only variant scores 45.12 with seed 42, compared with the full method's three-seed mean of 46.58. For mixed binary-reward groups, the resulting update coefficients approach the success-normalized form of MaxRL (Tajwar et al., 2026) at small β, up to GGPO's normalization (Appendix A.6). This component ablation supports reference-dependent weighting in the default $N = 8$ setting.

D.7 FINITE GROUP SIZE

We study the effect of group size at a fixed optimizer-update budget by varying the number of responses per prompt over $m \in \{4, 8, 16\}$ for Qwen3-1.7B-Base under the default GGPO configuration. Every run uses 32 groups per learner minibatch and 8,000 optimizer updates; the $m = 8$ condition reuses the default-setting run. Each update uses $32m$ retained responses, and raw generation cost also depends on mixed-group acceptance and refill. Appendix A.5 separately illustrates problem count, groups per problem, and response cost.

Table 8: **Evaluation across GGPO group sizes on Qwen3-1.7B-Base.** Problem-weighted best@K (%) over the five benchmarks at the final 8,000-update checkpoint for training seed 42. For binary rewards, best@K equals pass@K. The $m = 8$ row reuses the GGPO results in Table 3. All runs use 32 groups per learner minibatch.

Group size	$K = 1$	$K = 4$	$K = 8$	$K = 16$	$K = 32$
$m = 4$	46.51	60.73	65.97	70.12	73.17
$m = 8$	46.60	60.91	66.22	70.46	73.90
$m = 16$	46.88	61.43	66.80	71.25	75.10

Results. Increasing m from 4 to 16 improves best@K at every reported K, with gains growing from 0.37 points at $K = 1$ to 1.93 points at $K = 32$ (Table 8). Larger groups are more likely to contain a correct response on difficult prompts, providing a mixed-reward training signal that smaller groups may miss. The larger gains at higher sampling budgets are consistent with improvements on problems that the learner solves infrequently.

D.8 SAMPLING BUDGET

Figure 10 complements the update-based comparison in Section 5.2 with cumulative generated training responses for Qwen3-1.7B-Base under the default setting ($N = 8$, $m = 8$). Counts include all generated candidate responses, including those discarded by mixed-reward filtering and those generated during refill (Appendix C.1).

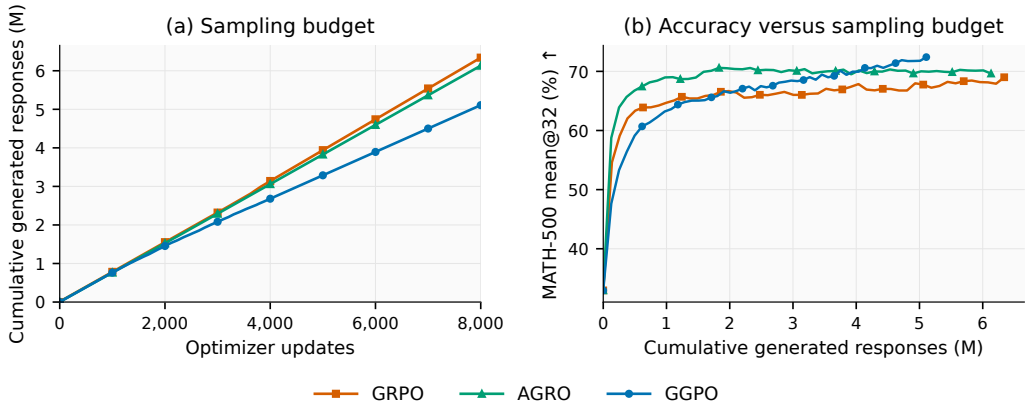

Figure 10: **Sampling budget under the default training setting on Qwen3-1.7B-Base.** Left: cumulative generated training responses, in millions, against optimizer updates. Right: MATH-500 mean@32 accuracy against the same cumulative response count. Each curve ends at 8,000 optimizer updates (GRPO orange squares, AGRO green triangles, GGPO blue circles). GGPO finishes with the highest accuracy and the lowest response count.