A UNIFIED KL–χ^2 GEOMETRY OF APPROXIMATE POLICY MIRROR DESCENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Regression-based reinforcement learning for language models approximates policy mirror descent (PMD). The exact KL-regularized update requires a log-partition function that is difficult to estimate from a few rollouts. Practical methods modify this baseline, as in the two-temperature formulation of A*PO and the mean baseline of Kimi K2. We study how these choices change the policy-improvement objective. For regression relative to the rollout policy, we characterize a mixed KL–χ^2 objective with the same population optimum. The induced χ^2 strength is zero at matched baseline and update temperatures, increases continuously with baseline temperature, and reaches its maximum at the mean baseline. We bound this strength using reward variation and characterize its dependence on the two temperatures and rollout pass rate for binary rewards. The bounds identify penalty strengths that cannot be reached through baseline temperature alone. We then optimize the mixed objective directly, making the χ^2 strength an explicit parameter. Experiments with Qwen2.5 models on MATH and DAPO-Math-17k show that baseline choice and explicit penalty strength affect training stability. On MATH, positive χ^2 coefficients sustain training in settings where KL-only updates fail across all three model sizes.

1 INTRODUCTION

Reinforcement learning with verifiable rewards (RLVR) improves language models by rewarding responses that pass a correctness check (Lambert et al., 2024; DeepSeek-AI, 2025). Policy mirror descent provides a way to turn these rewards into a regularized policy update. With KL regularization, its exact solution uses an exponential tilt of the current policy and a log-partition function that normalizes the response probabilities. Regression methods fit the corresponding log-probability ratios to rewards centered by this baseline. Estimating the log-partition function from a few rollouts is difficult, especially at small update temperatures. Practical methods therefore use alternative baselines, which become part of the regression target.

A*PO separates the temperature that scales the fitted log-ratio from the temperature used to compute a soft reward baseline (Brantley et al., 2025). A larger baseline temperature smooths this baseline, and its infinite-temperature limit is the mean reward. Mean-centered regression appears in Kimi K1.5 and forms the foundation of Kimi K2's RL algorithm (Kimi Team, 2025a;b). PMD-MEAN uses this mean baseline and exhibits more stable training than the matched-temperature PMD-PART method in the experiments of Xu et al. (2026). These choices motivate studying how the baseline changes the regularization of the policy update.

Prior work connects this change to a mixed KL–χ^2 objective. Xu et al. (2026) show that the population optimum of mean-centered regression maximizes expected reward with both penalties. Sakhi et al. (2026) connect advantage centering and temperature decoupling to Lambert-W policy targets. The control of large policy ratios provided by χ^2 regularization also underlies χPO (Huang et al., 2025). We build on these results to characterize the map from baseline temperature to the induced χ^2 coefficient. For a fixed update temperature and rollout reward distribution, we ask which coefficients are attainable through baseline temperature alone, what limits this attainable range, and how it compares with choosing the coefficient directly.

We analyze regression relative to the rollout policy, with the baseline computed under that same policy. This includes the online two-temperature variant evaluated here. For each baseline temperature at

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.

or above the update temperature, we identify the unique χ^2 coefficient for which regression and the mixed objective share a population optimum. Increasing the baseline temperature increases this coefficient continuously from zero at matched temperatures to its maximum at the mean baseline. Bounds in terms of reward variation give a finite attainable range, with a universal upper bound for rewards in $[0, 1]$. Explicit regularization provides access to coefficients beyond this range. For binary verifier rewards, we derive how the induced coefficient depends on the two temperatures and the prompt's rollout pass rate. The strength can therefore vary during training even when both temperatures are fixed. At the small update temperatures used in RLVR, however, substantially different coefficients can correspond to nearly identical fully optimized policies.

We then optimize the mixed objective directly, choosing the χ^2 coefficient independently of the baseline. Experiments with Qwen2.5 models on MATH and DAPO-Math-17k compare these explicit objectives with regression under different baseline choices. Across all three MATH model sizes, the positive coefficients tested sustain training where KL-only updates fail. Mean-baseline regression also sustains training and gives the highest MATH500 accuracy among our trained 3B and 7B models. The DAPO results further show that training stability and reasoning accuracy depend on the chosen penalty strength.

2 Background and related work

We first review regression-based policy updates in RLVR, then summarize the baseline choices and implicit-regularization results that motivate our analysis.

2.1 Regression policy updates in RLVR

A prompt x is drawn from a distribution ρ, a response y is sampled from a policy $\pi(\cdot \mid x)$, and a verifier assigns reward $r(x, y) \in [0, 1]$. Our experiments use binary correctness rewards. The objective is $\mathbb{E}_{x\sim\rho, y\sim\pi(\cdot|x)} r(x, y)$, treating each complete response as an action.

Policy mirror descent. Policy mirror descent updates a policy through regularized improvement steps (Lan, 2023). In our response-level setting, its KL-regularized step at iteration t maximizes $\mathbb{E}_\pi r - \beta_u \,\mathrm{KL}(\pi\|\pi_t)$ separately for each prompt. Its exact solution is

$$\pi_{t+1}(y \mid x) = \pi_t(y \mid x) \exp\left(\frac{r(x, y) - V_{\beta_u}(x)}{\beta_u}\right), \qquad V_b(x) = b \log \mathbb{E}_{y\sim\pi_t(\cdot|x)} e^{r(x,y)/b}.$$

The update temperature $\beta_u > 0$ is the coefficient of the KL penalty. The soft value V_{β_u} normalizes the updated policy. Taking logarithms gives the regression target $\beta_u \log(\pi_{t+1}/\pi_t) = r - V_{\beta_u}$. This construction underlies the regression updates studied in A*PO and PMD-MEAN (Brantley et al., 2025; Xu et al., 2026).

Two-temperature regression. Practical regression methods need not use the same temperature for the baseline and the fitted log-ratio. For a group of n responses $y_1, \ldots, y_n \sim \pi_t(\cdot \mid x)$, the baseline estimate and group regression loss are

$$\begin{aligned} \widehat{V}_{b,n}(x) &= b \log\left(\frac{1}{n}\sum_{i=1}^{n} e^{r(x,y_i)/b}\right), \\ \widehat{L}_{b,\beta_u}(\theta) &= \frac{1}{2n}\sum_{i=1}^{n}\left[\beta_u \log\frac{\pi_\theta(y_i \mid x)}{\pi_t(y_i \mid x)} - r(x, y_i) + \widehat{V}_{b,n}(x)\right]^2. \end{aligned} \tag{1}$$

The loss is averaged over prompts. The rollout policy, baseline estimate, and responses remain fixed during fitting. We write $b = \beta_b$ for the baseline temperature and analyze $\beta_b \geq \beta_u > 0$. Appendix D.2 specifies the implementation's loss scaling and reduction.

Trust regions and clipped updates. The regression updates above are one way to control policy improvement. TRPO instead constrains the average KL change (Schulman et al., 2015), while PPO uses a clipped surrogate to reduce the incentive for large changes in sampled probability ratios (Schulman et al., 2017). GRPO combines this construction with group-relative

rewards (Shao et al., 2024). The χ^2 penalty studied here acts directly on squared deviations of complete-response ratios from one. Our experiments omit ratio clipping to isolate the effects of baseline and penalty choice.

2.2 Regression methods and baseline choices

Equation (1) encompasses several regression methods that differ primarily in how they construct the target baseline and reference policy.

Relative rewards and optimal advantages. REBEL fits differences in log-policy ratios to reward differences between two responses (Gao et al., 2024). Differencing cancels a common prompt baseline. In pointwise regression, the baseline remains part of each response's target. The original A*PO formulation uses a fixed reference policy to define its soft baseline and log-ratio target (Brantley et al., 2025). OAPL uses a lagged inference policy both to generate responses and as the log-ratio reference, periodically synchronizing it with the trained policy (Ritter et al., 2026). A*PO and OAPL use separate baseline and update temperatures in their experiments. Equation (1) uses the common rollout and reference distribution that also defines our population analysis.

PMD-PART and T^2-PART. PMD-PART sets $\beta_b = \beta_u$. For $\beta_u < \beta_b < \infty$, we call the update Two-Temperature PMD-PART (T^2-PART). Our experiments use $\beta_b = 0.5$. This is an online A*PO variant whose rollout policy is also the log-ratio reference. In original A*PO, the reference stays fixed as the rollout policy changes.

Kimi K1.5/K2 and PMD-MEAN. Kimi K1.5 derives squared log-ratio regression from KL-regularized policy mirror descent and reports that the empirical mean reward works well as an approximation to the soft baseline, motivated by the infinite-temperature limit (Kimi Team, 2025a). Kimi K2 explicitly adopts mean-centered regression as the foundation of its RL algorithm (Kimi Team, 2025b). In our notation, this core objective is PMD-MEAN, which takes $\beta_b \to \infty$ while retaining a finite update temperature β_u. Its baseline is $\widehat{V}_{\infty,n}(x) = n^{-1} \sum_i r(x, y_i)$. These methods motivate studying the full range from matched temperatures through finite temperature mismatch to the mean endpoint.

Estimating the baseline. At small b, the largest sampled rewards dominate the empirical soft value. With few responses, a group can miss rare successful responses, so this plug-in baseline can differ substantially from its population value (Xu et al., 2026). As $b \downarrow 0$, it approaches the largest reward in the group. Appendix C gives the soft-value limits and sampled-baseline identities.

RLOO uses a leave-one-out reward baseline to reduce variance while preserving unbiased policy-gradient estimates from independent on-policy samples (Kool et al., 2019; Ahmadian et al., 2024). Our PMD-MEAN runs use an inclusive group mean, while Xu et al. (2026, Equation (17)) use a leave-one-out mean in their sampled formulation. For squared regression, Appendix C.1 derives the target rescaling caused by self-inclusion and its interpretation as a change in update temperature.

2.3 Implicit regularization and scope of our analysis

Xu et al. (2026) characterize the mean-baseline population optimum through the Lambert-W function and show that it maximizes reward with mixed KL–χ^2 regularization. They give bounds on the induced coefficient, binary-reward asymptotics, and a convergence analysis accounting for estimation and fitting errors. Their mean-endpoint results provide a starting point for our analysis of finite baseline temperatures.

Sakhi et al. (2026) characterize Lambert policy targets for general advantage centering and explain how temperature decoupling changes the induced target. Their analysis also treats the finite-group effect of including a response in its own baseline. The mixed-divergence geometry and statistical benefits of χ^2 regularization appear in χPO, which studies protection against overoptimization in offline preference learning (Huang et al., 2025).

We study how baseline temperature restricts the strength of the induced χ^2 penalty. For a fixed update temperature and rollout reward distribution, Section 3 characterizes the full attainable interval, bounds

its size using reward variation, and derives its dependence on pass rate for binary rewards. These results quantify the regularization available through the soft-baseline family when the rollout policy also defines the regression reference. Our experiments compare this implicit choice with selecting the coefficient directly in an explicit objective. The population equivalence concerns fully fitted policies, while the experiments examine training with sampled responses and finite fitting.

3 THE REGULARIZATION SPECTRUM OF TWO-TEMPERATURE REGRESSION

We characterize the χ^2 regularization induced by the baseline temperature in two-temperature regression. At the population level, where expectations under the rollout policy are exact, we show that each baseline temperature corresponds to a unique mixed KL–χ^2 objective with the same population optimum. As the baseline temperature increases from the update temperature to the mean-baseline limit, the induced χ^2 coefficient varies continuously from zero to a finite maximum. We then characterize what limits this attainable range and how the coefficient depends on the rollout reward distribution. Our analysis builds on the mean-baseline mixed-divergence objective of Xu et al. (2026) and the connection between advantage centering, temperature decoupling, and Lambert targets in Sakhi et al. (2026). Proofs of all results in this section are given in Appendix A.

At the population level, we fix an RLVR prompt x, a finite response space, and a full-support rollout policy $\mu(\cdot \mid x)$. Throughout, assume the verifier reward $r(x, y) \in [0, 1]$ is nonconstant in y. Write $\mathbb{E}_\mu = \mathbb{E}_{y\sim\mu(\cdot|x)}$ and $w(y) = \pi(y \mid x)/\mu(y \mid x)$, suppressing x below. The rollout policy defines the baseline and the log-ratio reference. For update temperature $\beta = \beta_u > 0$ and baseline temperature $b = \beta_b \in [\beta, \infty]$, set

$$V_b = b \log \mathbb{E}_\mu e^{r/b}, \qquad V_\infty = m := \mathbb{E}_\mu r, \qquad L_b(\pi) = \tfrac{1}{2}\mathbb{E}_\mu[(\beta \log w - r + V_b)^2].$$

Here L_b is the population loss of the two-temperature regression family. Optimization ranges over normalized policies and uses exact population baselines. To characterize the regularization induced by L_b, define $\chi^2(\pi\|\mu) = \mathbb{E}_\mu[(w-1)^2]$ and consider the mixed objective

$$J_{\beta,\xi}(\pi) = \mathbb{E}_\pi r - \beta\,\mathrm{KL}(\pi\|\mu) - \tfrac{\xi}{2}\chi^2(\pi\|\mu).$$

Our first result shows that varying the baseline temperature traces out a unique and continuous range of mixed KL–χ^2 objectives.

Proposition 1 (Exact coefficient spectrum). *For each $b \in [\beta, \infty]$, there is a unique pair (ξ_b, w_b) with $\xi_b \geq 0$ and $w_b > 0$ satisfying*

$$\beta \log w_b(y) + \xi_b w_b(y) = r(y) - V_b, \qquad \mathbb{E}_\mu w_b = 1. \tag{2}$$

The policy $\pi_b = \mu w_b$ uniquely minimizes L_b and uniquely maximizes J_{β,ξ_b}. At fixed β, μ, r, the map $b \mapsto \xi_b$ is continuous and strictly increasing from $\xi_\beta = 0$ to $\xi_\infty > 0$. Its attainable coefficients are exactly $[0, \xi_\infty]$.

Matching temperatures gives pure KL regularization, while raising the baseline temperature increases the induced χ^2 coefficient, with the mean baseline attaining its maximum. Proposition 1 identifies the attainable interval $[0, \xi_\infty]$ but does not yet control its size. The next result shows that reward variation imposes a finite ceiling on this interval.

Proposition 2 (A finite-temperature limit on induced regularization). *Let $\sigma^2 = \mathrm{Var}_\mu(r)$ and $D_b = V_b - m$. At every positive update temperature,*

$$\xi_b \leq \min\left\{ V_\beta - V_b, \frac{\sqrt{D_b^2 + \sigma^2} - D_b}{2} \right\} \leq \frac{\sigma}{2} \leq \frac{1}{4}.$$

In fact, $\xi_b < 1/4$. The bound $1/4$ is sharp as a supremum.

Thus reward variation limits the strength supplied by temperature decoupling. Under our convention $-\xi\chi^2/2$, explicit coefficients above $1/4$ lie beyond this family for rewards in $[0, 1]$. At $b = \infty$, the bound $\xi_\infty \leq V_\beta - m$ recovers the mean-baseline bound of Xu et al. (2026, Theorem 3.1, Equation (6)). Appendix A.2 gives the exact decomposition of the baseline gap that yields this bound.

The induced strength can therefore change during training even when the temperatures are fixed. At the mean endpoint and small β, it is largest near 50% success. Xu et al. (2026, Equation (7) and Appendix B.1) establish this endpoint limit. The finite-b expansion describes the intermediate temperatures. Figure 2 illustrates the spectrum, with numerical checks in Appendix B.

3.1 Explicit Regularization for Training

The preceding results apply to general bounded rewards. Because our RLVR experiments use binary correctness rewards, we next specialize the spectrum to this setting and then use the resulting characterization to motivate the explicit objective optimized in Section 4.

Binary verifier rewards. Let $p = \Pr_{y\sim\mu(\cdot|x)}\{r(x,y)=1\} \in (0,1)$ denote the rollout pass rate. Then $V_b(p) = b\log(1-p+pe^{1/b})$ and $V_\infty(p) = p$. Equation (2) gives one ratio for successful responses and one for failures, so normalization reduces to $pw_1 + (1-p)w_0 = 1$. This reduction makes the dependence of the induced coefficient on the rollout pass rate explicit. At small update temperatures, it admits the following expansion.

Proposition 3 (Coefficient at small update temperatures)**.** *For fixed $p \in (0,1)$ and $b \in (0,\infty]$, as $\beta \downarrow 0$,*

$$\xi_b = p\big(1 - V_b(p)\big) + \beta p \log p + o(\beta).$$

The mean-baseline endpoint tends to $p(1-p)$. At every $\beta > 0$ and $b \geq \beta$, $\xi_b \leq \sqrt{p(1-p)}/2$, so the coefficient vanishes as $p \to 0$ or $p \to 1$, uniformly over admissible temperatures.

Thus even with fixed temperatures, the regularization induced by regression can change as the rollout pass rate evolves. Combined with Proposition 2, this also shows that baseline temperature restricts the range of χ^2 strengths available to the update. We therefore also consider choosing the coefficient directly.

Explicit regularization. For a batch of N responses, let $q_i = \pi_\theta(y_i \mid x_i)/\pi_t(y_i \mid x_i)$ be the sequence-level ratio and $r_i = r(x_i, y_i)$ the verifier reward. For a fixed coefficient, the baseline-free objective averaged uniformly over these responses is

$$\widehat{J}_{\beta_u,\xi}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left[q_i r_i - \beta_u(q_i \log q_i - q_i + 1) - \frac{\xi}{2}(q_i - 1)^2\right]. \tag{3}$$

When ξ is fixed independently of the sampled responses, the expectation of (3) recovers the mixed objective $J_{\beta,\xi}$ from Proposition 1, averaged over prompts. We use the convention $-\xi\chi^2/2$. By Proposition 2, the settings $\xi = 0.3$ and $\xi = 1$ lie outside the range attainable through baseline temperature alone for rewards in $[0,1]$. We additionally consider a simple adaptive rule based on the sampled pass rate,

$$\xi_t(x) = \widehat{p}_t(x) = \frac{1}{n}\sum_{j=1}^{n} r(x, y_j).$$

We compute this value once per rollout group and hold it fixed during fitting, with no gradient through the coefficient. This pass-rate heuristic takes values in $[0,1]$ and can exceed the attainable regression range. The temperature-induced coefficient is logged separately. The fixed-coefficient expectation above applies when the coefficient is specified independently of the sampled responses. In the adaptive setting, we use the same responses to set the coefficient and evaluate the objective.

3.2 From Population Solutions to Practical Training

The population analysis characterizes how baseline choice changes the induced regularization, but differences in regularization need not translate into large differences between fully optimized policies. This distinction is especially pronounced at the small update temperatures used in our experiments. At $p = 1/2$ and $\beta_u = 0.001$, the coefficient ranges from zero to approximately 0.24965 at the mean baseline. Yet every ideal policy in this grid has success probability above 0.99995. Appendix A.4 proves their common success-conditioned limit as $\beta \downarrow 0$.

The binary specialization also shows that the induced regularization weakens as the rollout policy becomes highly successful.

Remark 1 (Regularization at high pass rates). As the rollout pass rate approaches one, both baselines approach one and their gap vanishes because $p \leq V_b(p) \leq V_\beta(p) \leq 1$. Proposition 2 then gives $0 \leq \xi_b \leq V_\beta(p) - V_b(p) \leq 1 - p$. This decrease follows from the reward distribution and baseline choice. A prescribed convergence schedule can instead increase the penalty over training. Under exact

population optimization, the feasible rollout policy has zero penalty, giving $\mathbb{E}_{\pi_b} r \geq J_{\beta,\xi_b}(\pi_b) \geq J_{\beta,\xi_b}(\mu) = p$. Sampling error and incomplete fitting can change this behavior, and an explicit coefficient permits independent control of the penalty during training.

Remark 2 (Connection to practical updates). Figure 2 predicts little change in the fully optimized policy across the displayed baseline temperatures. Empirical training nevertheless depends on this choice. The A⋆PO ablation at update temperature 0.001 reports nonmonotone performance differences across baseline temperatures in its fixed-reference setting (Brantley et al., 2025, Figure 5). In our online regression comparisons, matched temperatures and $b = 0.5$ lead to failures, while the inclusive mean baseline sustains training over the reported intervals (Section 4).

Proposition 1 identifies a common population optimum for regression and its corresponding mixed objective. Their gradients during fitting need not agree. Finite sampling perturbs the baseline and restricts regression to the sampled responses, while shared neural parameters and a finite number of optimizer steps further shape the fitted policy. These are possible sources of the gap. The paired comparison in Section 4.1 examines fitting by holding the baseline rule fixed and changing the objective. The experiments do not isolate the cause of deterioration.

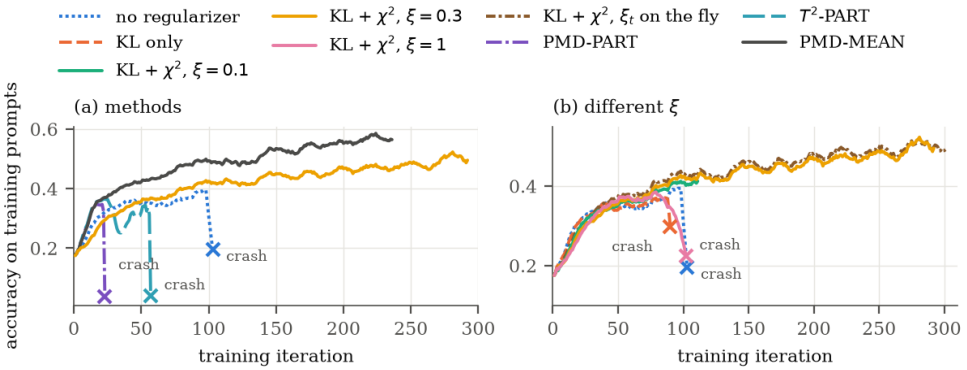

Figure 1. Qwen2.5-Math-7B training accuracy on DAPO-Math-17k. The left panel compares regression baselines and the right panel varies the explicit χ^2 coefficient. Curves use a moving average with window size 10. Crosses mark the last available points of reported failed runs. Run endpoints and compute budgets differ across methods (Appendix D.3).

4 Experiments

We study how the baseline temperature and explicit χ^2 strength affect RLVR training when multiple optimizer steps are taken on each frozen rollout batch. Our experiments ask three questions. First, how do regression baselines and explicit KL–χ^2 objectives behave across model scales and regularization strengths? Second, when the explicit surrogate uses the coefficient implied by a particular regression baseline and sampled reward group, does it reproduce the corresponding training behavior? Third, how do these stability effects change under a different dataset and rollout configuration? We evaluate the first two questions on MATH and the third on DAPO-Math-17k. Our implementation uses verl (Sheng et al., 2024) and extends the public implementation of A⋆PO (Brantley et al., 2025). Because some settings become unstable, we report failures explicitly. Tables give the failure iteration, crosses mark the final available point, and Appendix D.3 distinguishes collapse from time limits and manual stopping.

4.1 Reasoning on MATH

We use MATH as our primary setting to study both the effect of baseline and regularization choices and the correspondence between regression and its implied explicit surrogate.

Training and evaluation. Following Brantley et al. (2025), we train Qwen2.5 base models (Yang et al., 2024a) of sizes 1.5B, 3B, and 7B on the 7,500 MATH training problems (Hendrycks et al., 2021) with binary correctness rewards. Each iteration samples $n = 4$ responses for each of 256

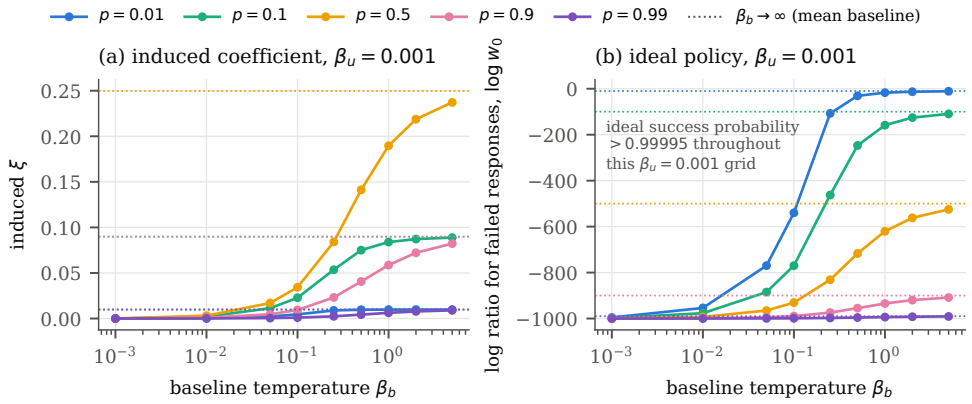

Figure 2. Binary population solutions at $\beta_u = 0.001$. (a) Induced coefficient as the baseline temperature varies. Dotted lines show the mean-baseline endpoint. (b) Log ratio of failed responses under the ideal policy. Its success probability exceeds 0.99995 throughout this grid, despite the different coefficients. Appendix B gives the grid, solver checks, and a comparison at $\beta_u = 0.5$.

prompts. After response-level balancing, the batch is divided into mini-batches of 128 responses, giving $K = 8$ optimizer steps at learning rate 10^{-6}. Each response is used once per rollout iteration, so later mini-batches are fitted after the policy has changed from the rollout policy. We use one run per configuration, with gradient-norm clipping at 1.0 and no ratio clipping or entropy bonus. Appendix D.2 details the training budgets, loss reduction, and regression loss scale.

The explicit comparisons include no regularizer ($\beta_u = \xi = 0$), KL only ($\beta_u = 10^{-3}$, $\xi = 0$), and mixed objectives with $\xi \in \{0.1, 0.3, 1\}$ at $\beta_u = 10^{-3}$. The regression comparisons use the same update temperature, with $\beta_b = \beta_u$ for PMD-PART, $\beta_b = 0.5$ for T^2-PART, and an inclusive group mean for PMD-MEAN. The online two-temperature variant uses π_t as the log-ratio reference. Appendix D.2 gives the remaining hyperparameters.

We select the checkpoint with the best greedy MATH500 validation accuracy and evaluate it on MATH500 (Lightman et al., 2023), Minerva Math (Lewkowycz et al., 2022), OlympiadBench (He et al., 2024), and AMC 2023. For AMC's 40 questions, we report mean accuracy over 32 generations. Evaluation uses the Qwen2.5-Math answer grader (Yang et al., 2024b). Published GRPO (Shao et al., 2024) and A*PO comparator results are taken from Brantley et al. (2025).

With this common setup, we first compare how the regression baseline and explicit regularization strength affect training across model scales.

How do baseline choice and χ^2 strength affect training? Both substantially affect training stability, while their effect on accuracy is less monotonic. PMD-PART and T^2-PART fail at every evaluated MATH model size, whereas PMD-MEAN remains stable and achieves the highest MATH500 accuracy among our 3B and 7B runs. A similar pattern appears for the explicit surrogate. Removing the χ^2 term, either entirely or while retaining only the KL term, leads to failure at all three model sizes, while every evaluated positive χ^2 coefficient remains stable over the reported MATH runs. Among these stable settings, however, no single coefficient is uniformly best. The adaptive coefficient gives the highest 1.5B MATH500 accuracy and is competitive at 3B, while $\xi = 0.1$ performs best among the explicit surrogates at 7B. Thus the main effect of increasing χ^2 regularization is improved stability, while the coefficient that gives the highest accuracy depends on model scale. The complete benchmark results in Appendix D show the same lack of a uniformly optimal coefficient across evaluation sets.

Does the corresponding explicit surrogate reproduce the regression behavior? We pair the Qwen2.5-1.5B regression at $\beta_u = 0.001$ and $\beta_b = 0.5$ with a centered surrogate using the corresponding coefficient computed from each empirical reward group. Both runs use the same baseline rule and eight optimizer steps per rollout.

Method	1.5B	3B	7B
base	49.0	59.8	68.8
GRPO†	58.0	66.0	73.2
A⋆PO †	57.8	66.2	76.2
PMD-PART	*crashed* (15)	*crashed* (34)	*crashed* (33)
T^2-PART	*degraded* (209)	*crashed* (172)	*crashed* (35)
PMD-MEAN	<u>58.8</u>	**67.6**	**76.6**
Explicit KL–χ^2 surrogate			
no regularizer	*crashed* (42)	*crashed* (48)	*crashed* (51)
KL only	*crashed* (50)	*crashed* (188)	*crashed* (66)
$\xi = 0.1$	54.8	64.2	<u>75.4</u>
$\xi = 0.3$	57.6	64.8	73.4
$\xi = 1$	57.2	65.4	74.2
ξ_t on the fly	**59.4**	<u>66.6</u>	73.2

Table 1. MATH500 greedy accuracy (%) after training on MATH. Parentheses give the reported failure iteration, with conventions detailed in Appendix D.3. "Degraded" identifies the accuracy deterioration followed by manual stopping. Bold and underlined values rank the methods we trained within each model size. † denotes published comparator results from Brantley et al. (2025). Full benchmark results are in Appendix D.

Writing v_i for the shared inclusive soft baseline and $A_i = r_i - v_i$, their per-response losses are

$$\ell_{R,i} = (\beta_u \log q_i - A_i)^2, \qquad \ell_{S,i} \quad = 2\beta_u \left[-q_i A_i + \beta_u (q_i \log q_i - q_i + 1) + \frac{\widehat{\xi}_i}{2} (q_i - 1)^2 \right].$$

The factor $2\beta_u$ matches their loss gradients at $q_i = 1$ under common response weights, while $\widehat{\xi}_i$ is the empirical coefficient implied by the sampled reward group. Appendix D.4 gives the complete construction and implementation details.

Despite their population correspondence, their training dynamics diverge. Regression reaches a best greedy validation accuracy of 55.6% but accuracy drops to 4.8% at iteration 130 and recovers, then falls again to 4.6% at iteration 210. The corresponding surrogate reaches a similar peak accuracy of 56.0% but remains stable through all 261 iterations, ending at 52.8% (Figure 3). Regression also shows repeated episodes of extremely small sequence ratios and response-length excursions, while the surrogate largely avoids these behaviors. Thus matching the regression-induced χ^2 coefficient does not reproduce the finite-step optimization behavior of regression, despite matching its population optimum. Appendix D.4 gives the complete diagnostics. These validation scores use the training reward function, while Table 1 uses the Qwen2.5-Math answer grader.

4.2 Reasoning on DAPO-Math-17k

We next move to DAPO-Math-17k, which changes both the dataset and rollout configuration, to test the robustness of the stability patterns observed on MATH.

Training and evaluation. We train Qwen2.5-Math-7B (Yang et al., 2024b) on DAPO-Math-17k (Yu et al., 2025) with $n = 16$ responses per prompt. We evaluate on AIME and the February HMMT benchmarks from 2024 and 2025, reporting Avg@32 and Pass@32. The A⋆PO comparison uses a fixed reference policy, while the PMD and explicit-surrogate runs use the rollout policy as reference. For the PMD and explicit-surrogate runs, checkpoints are selected by AIME 2024 validation accuracy and training has a two-day wall-clock limit. The fixed-coefficient surrogate runs use four H200 GPUs, while PMD-PART, T^2-PART, PMD-MEAN, and the adaptive-coefficient surrogate use two. Training lengths also differ because some runs were stopped early.

How do these stability effects change under a different dataset and rollout configuration? Several qualitative patterns from MATH persist on DAPO-Math-17k, but the stable range of explicit regularization becomes narrower. PMD-PART and T^2-PART again fail, while PMD-MEAN remains

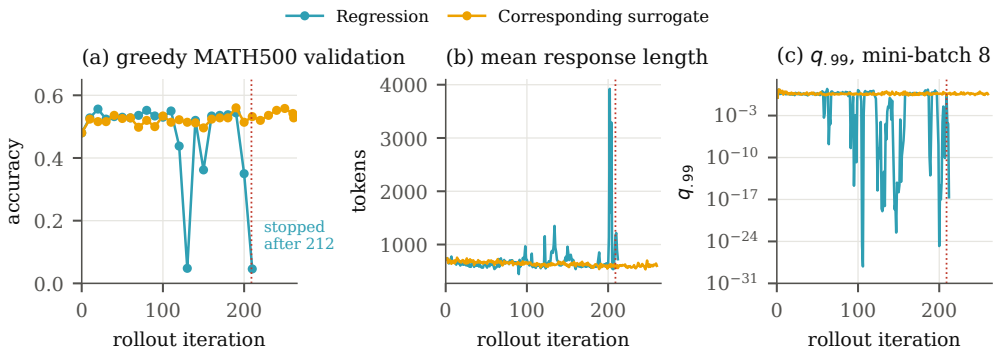

Figure 3. Paired MATH training with Qwen2.5-1.5B. (a) Greedy validation accuracy. (b) Mean response length. (c) Sequence-ratio 99th percentile on the eighth mini-batch, measured on 128 responses after seven steps and before the eighth, relative to the frozen rollout policy. Curves are unsmoothed. The dotted line marks the regression's deterioration criterion at iteration 209. Regression is stopped after 212, while the surrogate completes 261 iterations.

stable and achieves the highest average benchmark accuracy among our runs at 16.7%. Positive explicit χ^2 regularization also supports substantially longer training than PMD-PART and T^2-PART. The adaptive coefficient reaches 13.6% average accuracy and fixed $\xi = 0.3$ reaches 12.9% (Table 2). However, unlike on MATH, increasing the explicit coefficient is not uniformly stabilizing. The $\xi = 1$ run collapses on DAPO, whereas the corresponding MATH runs remain stable over the plotted intervals. Figure 1 therefore shows both a persistent benefit from moving away from weakly regularized updates and a stronger sensitivity to the amount of χ^2 regularization in this setting. Together, these results suggest that the qualitative role of baseline choice and χ^2 regularization transfers across the two training setups, while the coefficient that yields stable training is setting-dependent.

Method	AIME 24	AIME 25	HMMT 24	HMMT 25	Average
base	17.4	7.4	8.2	1.0	8.5
A*PO	32.0	**16.9**	**13.6**	1.5	16.0
PMD-PART			*crashed* (25)		
T^2-PART			*crashed* (58)		
PMD-MEAN	**35.5**	16.6	12.3	2.3	**16.7**
KL–χ^2 (ξ=0.1)	19.8	10.3	10.3	1.5	10.5
KL–χ^2 (ξ=0.3)	26.4	12.2	11.2	1.7	12.9
KL–χ^2 (ξ_t on the fly)	28.9	12.3	10.7	**2.4**	13.6

Table 2. Avg@32 accuracy (%) for Qwen2.5-Math-7B trained on DAPO-Math-17k. The average is over the four displayed benchmarks. Bold marks the best trained model in each column. Pass@32 results are in Appendix D.

5 Conclusion and Future Work

Two-temperature regression induces a continuous range of χ^2 coefficients, from zero at matched temperatures to a maximum at the mean baseline. Our analysis characterizes this range and shows how reward variation and binary pass rates limit the regularization available through the two temperatures.

At small update temperatures, distinct coefficients can produce nearly identical fully fitted policies. The RLVR experiments show that baseline choice and the objective used for fitting still affect training. Positive explicit coefficients sustain training in the MATH comparisons, and regression differs from a surrogate using its corresponding empirical coefficient in the paired run. Future work will study how policy parameterization and stochastic optimization affect the relationship between baseline choice and training stability.

AI USE STATEMENT

We used generative AI tools to assist with literature review, checking mathematical claims and proofs, and polishing the manuscript. The authors are responsible for the final claims, proofs, and experimental results.

REPRODUCIBILITY STATEMENT

Section 3 states the population assumptions, and Appendix A provides complete proofs. Section 4 describes the training and evaluation setup. Appendix D.2 lists the reported hyperparameters.

REFERENCES

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting REINFORCE-style optimization for learning from human feedback in LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12248–12267. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.662. URL https://aclanthology.org/2024.acl-long.662/.

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.

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

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. In *Advances in Neural Information Processing Systems*, volume 37, pp. 52354–52400, 2024. doi: 10.52202/079017-1660. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/5df1d92b478716877f774b82943477b3-Abstract-Conference.html.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. *arXiv preprint arXiv:2402.14008*, 2024.

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. *arXiv preprint arXiv:2103.03874*, 2021.

Audrey Huang, Wenhao Zhan, Tengyang Xie, Jason D. Lee, Wen Sun, Akshay Krishnamurthy, and Dylan J. Foster. Correcting the mythos of KL-regularization: Direct alignment without overoptimization via chi-squared preference optimization. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2407.13399.

Kimi Team. Kimi k1.5: Scaling reinforcement learning with LLMs. *arXiv preprint arXiv:2501.12599*, 2025a.

Kimi Team. Kimi K2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*, 2025b.

Wouter Kool, Herke van Hoof, and Max Welling. Buy 4 REINFORCE samples, get a baseline for free! In *Deep Reinforcement Learning Meets Structured Prediction, ICLR 2019 Workshop*, 2019. URL https://openreview.net/forum?id=r1lgTGL5DE.

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

Guanghui Lan. Policy mirror descent for reinforcement learning: Linear convergence, new sampling complexity, and generalized problem classes. *Mathematical Programming*, 198:1059–1106, 2023. doi: 10.1007/s10107-022-01816-5. URL https://doi.org/10.1007/s10107-022-01816-5.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35, 2022.

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. *arXiv preprint arXiv:2305.20050*, 2023.

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.

Otmane Sakhi, Aleksei Arzhantsev, Imad Aouali, and Flavian Vasile. Off-policy learning to reason works because it is more pessimistic than you think. *arXiv preprint arXiv:2605.28150*, 2026. URL https://arxiv.org/abs/2605.28150.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1889–1897. PMLR, 2015. URL https://proceedings.mlr.press/v37/schulman15.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.

Zhenghao Xu, Qin Lu, Changlong Yu, and Tuo Zhao. Approximation of log-partition function in policy mirror descent induces implicit regularization for LLM post-training. *arXiv preprint arXiv:2602.05933*, 2026.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024a.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024b.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

A PROOFS FOR THE TEMPERATURE-INDUCED REGULARIZATION SPECTRUM

We use the population setting and notation of Section 3, with $m = \mathbb{E}_\mu r$ and $\sigma^2 = \mathrm{Var}_\mu(r) > 0$.

A.1 EXACT COEFFICIENT SPECTRUM

Proof of Proposition 1. For finite b, define $\nu_b = \mu e^{r/b}/\mathbb{E}_\mu e^{r/b}$. Differentiation gives $dV_b/db = -\,\mathrm{KL}(\nu_b\|\mu) < 0$, and $V_b \to m$ as $b \to \infty$. Thus V_b decreases continuously from V_β to m.

For $V \in [m, V_\beta]$ and $\xi \geq 0$, let $w_{\xi,V}(y)$ solve $\beta \log w + \xi w = r(y) - V$. The left side is strictly increasing from $-\infty$ to $+\infty$, so the solution is unique and positive. Its mass $M(\xi, V) = \mathbb{E}_\mu w_{\xi,V}$ is continuous and strictly decreasing in ξ, with

$$M(0, V) = \exp\left(\frac{V_\beta - V}{\beta}\right) \geq 1, \qquad \lim_{\xi\to\infty} M(\xi, V) = 0.$$

For any $\delta > 0$, $\beta \log \delta + \xi\delta \to \infty$ as $\xi \to \infty$, so each ratio eventually lies below δ. The finite response space then gives the stated limit of M. Hence a unique $\xi(V) \geq 0$ normalizes the ratios. Taking $V = V_b$ gives (ξ_b, w_b), with $\xi_\beta = 0$ and $\xi_b > 0$ for $b > \beta$.

To prove global optimality, write $\pi_b = \mu w_b$. Expanding the regression loss around its residual $-\xi_b w_b$ gives, for any positive normalized π,

$$L_b(\pi) - L_b(\pi_b) = \frac{\beta^2}{2}\mathbb{E}_\mu\left[\left(\log\frac{\pi}{\pi_b}\right)^2\right] + \beta\xi_b\,\mathrm{KL}(\pi_b\|\pi).$$

This extends the mean-baseline identity of Xu et al. (2026, Lemma D.1, Equation (37)), with $\lambda = \beta\xi_\infty$ in their notation. Both terms are nonnegative, and the first vanishes only at π_b. Policies with a zero probability have infinite loss, so π_b uniquely minimizes L_b.

The mixed objective J_{β,ξ_b} is strictly concave. At π_b, its coordinate derivatives are

$$r(y) - \beta(\log w_b(y) + 1) - \xi_b(w_b(y) - 1) = V_b + \xi_b - \beta.$$

Their common value proves optimality on the probability simplex, and strict concavity gives uniqueness. If the same policy maximized $J_{\beta,\xi}$ at another coefficient, subtracting the stationarity equations would make $(\xi - \xi_b)w_b(y)$ constant. Nonconstant rewards imply nonconstant w_b, so $\xi = \xi_b$.

Finally, the scalar equations extend smoothly near every normalized solution because their derivative in w is $\beta/w + \xi > 0$, including at $\xi = 0$. Implicit differentiation of the mass equation gives

$$\frac{d\xi}{dV} = -\frac{\mathbb{E}_\mu[w/(\beta + \xi w)]}{\mathbb{E}_\mu[w^2/(\beta + \xi w)]} < 0.$$

The implicit-function theorem gives continuity, including the endpoints $V = m$ and $V = V_\beta$. Since V_b strictly decreases with b, ξ_b strictly increases and covers exactly $[0, \xi_\infty]$. □

A.2 ATTAINABLE COEFFICIENT BOUNDS

Proof of Proposition 2. Let $X_b = \mathbb{E}_\mu[(w_b - 1)^2]$ and $\pi_\beta = \mu e^{(r-V_\beta)/\beta}$. Multiplying Equation (2) by w_b and averaging gives

$$\beta\,\mathrm{KL}(\pi_b\|\mu) + \xi_b(1 + X_b) = \mathbb{E}_{\pi_b} r - V_b.$$

Using $\beta\,\mathrm{KL}(\pi_b\|\pi_\beta) = \beta\,\mathrm{KL}(\pi_b\|\mu) - \mathbb{E}_{\pi_b} r + V_\beta$, we obtain

$$V_\beta - V_b = \beta\,\mathrm{KL}(\pi_b\|\pi_\beta) + \xi_b(1 + X_b).$$

Nonnegativity gives $\xi_b \leq V_\beta - V_b$. The first term is the decrease in the KL-regularized reward from π_β to π_b, and $1 + X_b = \mathbb{E}_\mu w_b^2$.

For the variance bound, put $D_b = V_b - m \geq 0$ and $t = \sqrt{X_b}$. Centering the reward and applying Cauchy–Schwarz yields

$$\xi_b(1 + t^2) + \beta\,\mathrm{KL}(\pi_b\|\mu) + D_b = \mathbb{E}_\mu[(w_b - 1)(r - m)] \leq \sigma t.$$

Consequently,

$$\xi_b \le \sup_{t \ge 0} \frac{\sigma t - D_b}{1 + t^2} = \frac{\sqrt{D_b^2 + \sigma^2} - D_b}{2} \le \frac{\sigma}{2}.$$

The supremum occurs at $t = (D_b + \sqrt{D_b^2 + \sigma^2})/\sigma$. For $r \in [0, 1]$, $r^2 \le r$ gives $\sigma^2 \le m(1-m) \le 1/4$. Nonconstant rewards imply $\pi_b \ne \mu$, so the positive KL term makes $\xi_b < \sigma/2 \le 1/4$. Proposition 3 with $p = 1/2$ and $b = \infty$ shows that $1/4$ is sharp as a supremum. □

A.3 Binary coefficient at small update temperatures

Proof of Proposition 3. Fix $p \in (0, 1)$ and $b \in (0, \infty]$, and write $V = V_b(p) \in (0, 1)$. For sufficiently small β, $b \ge \beta$. The failure equation gives

$$w_0 = \exp\left(-\frac{V + \xi_b w_0}{\beta}\right) \le e^{-V/\beta}.$$

Set $\varepsilon = (1-p)w_0$, so $w_1 = (1-\varepsilon)/p$. Substitution into the success equation yields

$$\xi_b = \frac{p}{1-\varepsilon}\left[1 - V + \beta \log p - \beta \log(1-\varepsilon)\right].$$

Since $\varepsilon = O(e^{-V/\beta})$, expansion gives $\xi_b = p(1-V) + \beta p \log p + O(e^{-V/\beta})$, proving the stated $o(\beta)$ remainder. At the mean endpoint $V = p$, the limit is $p(1-p)$. Proposition 2 gives $\xi_b \le \sqrt{p(1-p)}/2$, which tends to zero at both pass-rate endpoints independently of the temperatures. □

A.4 Binary policy limit

For fixed $p \in (0, 1)$, both a fixed $b \in (0, \infty]$ and the matched-temperature path $b = \beta$ give $\pi_b \to \mu(\cdot \mid r = 1)$ as $\beta \downarrow 0$.

Proof. For fixed b, the preceding proof gives $w_0 \to 0$ and $w_1 \to 1/p$. For matched temperatures, $\xi_\beta = 0$ and the success probability is $pe^{1/\beta}/(1 - p + pe^{1/\beta}) \to 1$. Responses with the same reward have the same ratio in both cases, preserving their relative probabilities under μ. □

B Numerical verification of the spectrum

Setup. We solve Equation (2) for $p \in \{0.01, 0.1, 0.5, 0.9, 0.99\}$, $\beta_u \in \{0.001, 0.1, 0.5\}$, and $\beta_b \in \{0.001, 0.01, 0.05, 0.1, 0.25, 0.5, 1, 2, 5, \infty\}$ with $\beta_b \ge \beta_u$. This gives 110 binary cases. A bracketed scalar root search in log space solves the normalization condition. Figure 4 adds the larger-temperature comparison to the two panels in Figure 2.

Checks. Normalization and stationarity residuals are at most 1.8×10^{-15}. The success probabilities agree within 9.6×10^{-9} with independent one-dimensional optimization of the regression loss and the mixed objective. The coefficient is nonnegative, zero at matched temperatures, nondecreasing in the baseline temperature, and bounded as in Proposition 2. Separate homogeneous-reward checks give zero coefficient. At $\beta_u = 0.001$, the expansion in Proposition 3 has maximum absolute error approximately 4.4×10^{-7}.

At $\beta_u = 0.001$, the minimum ideal success probability on the grid is approximately 0.999955, attained at $p = 0.01$ with the mean baseline. Its failure probability is about 4.49×10^{-5}. At $\beta_b = 0.5$, the same pass rate gives failure probability 3.52×10^{-14}. These values quantify the near-saturation result in Figure 2.

Coefficients from sampled rewards. Replacing p with the sampled pass rate $\widehat{p}$ gives a plug-in estimate of the population coefficient. It is exact for the categorical problem in which the uniformly weighted empirical group defines sampling, the baseline, and the log-ratio reference. The neural update instead uses the rollout policy as its reference. In particular, homogeneous groups have zero empirical coefficient even when the population pass rate lies strictly between zero and one.

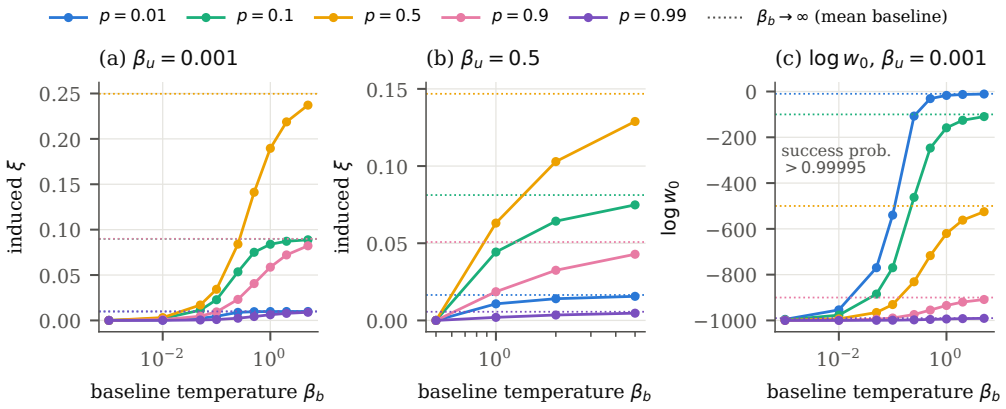

Figure 4. Population coefficient spectrum at update temperatures (a) $\beta_u = 0.001$ and (b) $\beta_u = 0.5$, and (c) the log ratio of failed responses at $\beta_u = 0.001$. Dotted lines mark the mean-baseline endpoint. Panels (a) and (c) also appear in Figure 2.

C Sampled baselines

Finite groups. For n independent responses and $M = \max_i r(x, y_i)$, factoring out $e^{M/b}$ gives $M - b\log n \le \widehat{V}_{b,n} \le M$. Thus $\widehat{V}_{b,n} \to M$ as $b \downarrow 0$. For binary rewards with pass rate p, a group has no success with probability $(1-p)^n$. At the other endpoint, expansion gives $\widehat{V}_{b,n} = n^{-1}\sum_i r(x, y_i) + O(b^{-1})$ as $b \to \infty$.

C.1 Including a response in its own mean baseline

Fix a prompt and sample $Y_1, \dots, Y_n \sim \mu$ independently, with $n \ge 2$. Let $\widehat{m} = n^{-1}\sum_j r(Y_j)$, let $\widehat{m}_{-i}$ exclude response i, and put $\alpha = (n-1)/n$. The mean estimate is unbiased, but the centered targets satisfy

$$r(Y_i) - \widehat{m} = \alpha\big(r(Y_i) - \widehat{m}_{-i}\big), \qquad \mathbb{E}[r(Y_i) - \widehat{m} \mid Y_i = y] = \alpha(r(y) - m).$$

The same shrinkage appears in Sakhi et al. (2026, Appendix C.1). If $\widehat{L}_{\text{mean}}$ uniformly averages the squared regression terms with factor $1/2$, then for each fixed candidate policy π,

$$\mathbb{E}[\widehat{L}_{\text{mean}}(\pi)] = \alpha^2 L_\infty^{\beta/\alpha}(\pi) + \frac{n-1}{2n^2}\sigma^2,$$

where L_∞^γ denotes the population mean-baseline loss at update temperature γ.

Proof. The realized-target identity follows from $n\widehat{m} = r(Y_i) + (n-1)\widehat{m}_{-i}$. Conditional on $Y_i = y$, the baseline has mean $[r(y) + (n-1)m]/n$ and variance $(n-1)\sigma^2/n^2$. Applying the mean–variance decomposition to the squared residual and averaging over $y \sim \mu$ proves the expected-loss identity. □

Thus the expected sampled loss has effective temperature $n\beta/(n-1)$, equal to $4\beta/3$ for our MATH groups and $16\beta/15$ for DAPO. Fitting to a realized group also depends on its sampled responses. For a general group baseline, the same conditional decomposition gives the expected target $r(y) - \mathbb{E}[\widehat{V}_b \mid Y_i = y]$, with an added variance term independent of the fixed candidate policy.

D Complete experimental results and training details

The tables below give the benchmark evaluations of the selected checkpoints. Published comparator results are marked with †. The bold and underlined entries rank the methods trained in this study. A dash indicates that a result is not reported.

Table 3. Full MATH benchmark results for Qwen2.5-1.5B. MATH500, Minerva Math, and OlympiadBench use greedy accuracy (%). AMC uses average accuracy over 32 generations.

Method	MATH500	Minerva	Olympiad	AMC
base	49.0	17.3	18.5	22.1
GRPO†	58.0	22.8	21.1	32.2
A⋆PO †	57.8	22.4	22.3	31.9
PMD-PART		*crashed* (15)		
T^2-PART		*degraded* (209)		
PMD-MEAN	<u>58.8</u>	<u>23.2</u>	<u>20.2</u>	28.2
Explicit KL–χ^2 surrogate				
no regularizer		*crashed* (42)		
KL only		*crashed* (50)		
$\xi = 0.1$	54.8	21.0	**20.8**	26.1
$\xi = 0.3$	57.6	17.6	19.0	**28.7**
$\xi = 1$	57.2	22.1	19.3	25.4
ξ_t on the fly	**59.4**	**25.0**	19.7	<u>28.6</u>

Table 4. Full MATH benchmark results for Qwen2.5-3B. MATH500, Minerva Math, and OlympiadBench use greedy accuracy (%). AMC uses average accuracy over 32 generations.

Method	MATH500	Minerva	Olympiad	AMC
base	59.8	26.8	25.8	33.5
GRPO†	66.0	25.0	28.9	34.6
A⋆PO †	66.2	25.7	28.0	35.5
PMD-PART		*crashed* (34)		
T^2-PART		*crashed* (172)		
PMD-MEAN	**67.6**	<u>32.4</u>	<u>28.5</u>	**41.7**
Explicit KL–χ^2 surrogate				
no regularizer		*crashed* (48)		
KL only		*crashed* (188)		
$\xi = 0.1$	64.2	25.7	**29.5**	35.0
$\xi = 0.3$	64.8	26.1	26.0	30.3
$\xi = 1$	65.4	**33.5**	27.3	35.1
ξ_t on the fly	<u>66.6</u>	26.1	27.6	<u>40.9</u>

Table 5. Full MATH benchmark results for Qwen2.5-7B. MATH500, Minerva Math, and OlympiadBench use greedy accuracy (%). AMC uses average accuracy over 32 generations.

Method	MATH500	Minerva	Olympiad	AMC
base	68.8	35.7	35.5	42.3
GRPO†	73.2	33.5	34.0	55.9
A⋆PO †	76.2	34.6	34.3	56.2
PMD-PART		*crashed* (33)		
T^2-PART		*crashed* (35)		
PMD-MEAN	**76.6**	<u>38.6</u>	<u>36.6</u>	**54.3**
Explicit KL–χ^2 surrogate				
no regularizer		*crashed* (51)		
KL only		*crashed* (66)		
$\xi = 0.1$	<u>75.4</u>	**39.0**	36.1	47.7
$\xi = 0.3$	73.4	34.6	35.9	46.0
$\xi = 1$	74.2	36.0	**37.1**	<u>49.6</u>
ξ_t on the fly	73.2	<u>38.6</u>	36.4	48.5

Table 6. Complete DAPO-Math-17k evaluation results (Avg@32, %) for Qwen2.5-Math-7B.

Method	AIME 24	AIME 25	HMMT 24	HMMT 25	Average
base	17.4	7.4	8.2	1.0	8.5
A*PO	32.0	**16.9**	**13.6**	1.5	16.0
PMD-PART			*crashed* (25)		
T^2-PART			*crashed* (58)		
PMD-MEAN	**35.5**	16.6	12.3	2.3	**16.7**
KL–χ^2 (ξ=0.1)	19.8	10.3	10.3	1.5	10.5
KL–χ^2 (ξ=0.3)	26.4	12.2	11.2	1.7	12.9
KL–χ^2 (ξ_t on the fly)	28.9	12.3	10.7	**2.4**	13.6

Table 7. Complete DAPO-Math-17k evaluation results (Pass@32, %) for Qwen2.5-Math-7B.

Method	AIME 24	AIME 25	HMMT 24	HMMT 25	Average
base	53.3	40.0	20.0	20.0	33.3
A*PO	53.3	**50.0**	23.3	**16.7**	35.8
PMD-PART			*crashed* (25)		
T^2-PART			*crashed* (58)		
PMD-MEAN	**66.7**	40.0	**30.0**	10.0	**36.7**
KL–χ^2 (ξ=0.1)	56.7	40.0	26.7	**16.7**	35.0
KL–χ^2 (ξ=0.3)	60.0	40.0	23.3	10.0	33.3
KL–χ^2 (ξ_t on the fly)	63.3	33.3	23.3	**16.7**	34.2

D.1 MATH TRAINING CURVES

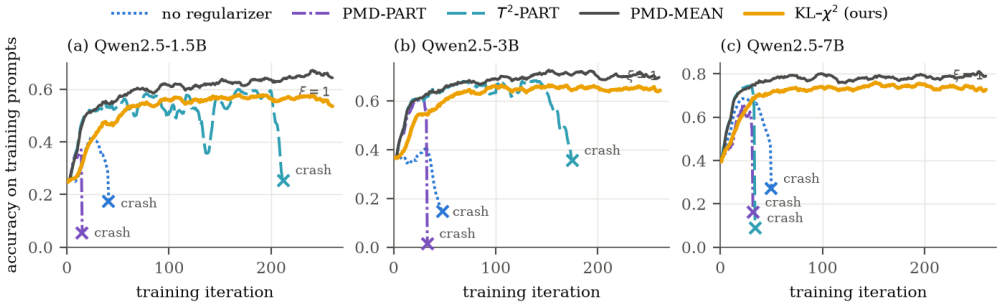

Figure 5. MATH training accuracy with Qwen2.5 (a) 1.5B, (b) 3B, and (c) 7B. Curves use a moving average with window size 10 and $K = 8$ optimizer steps per rollout batch. Crosses mark the last available point of each failed run. The explicit-surrogate curve uses $\xi = 1$ in each panel. Figure 6 shows the coefficient sweep.

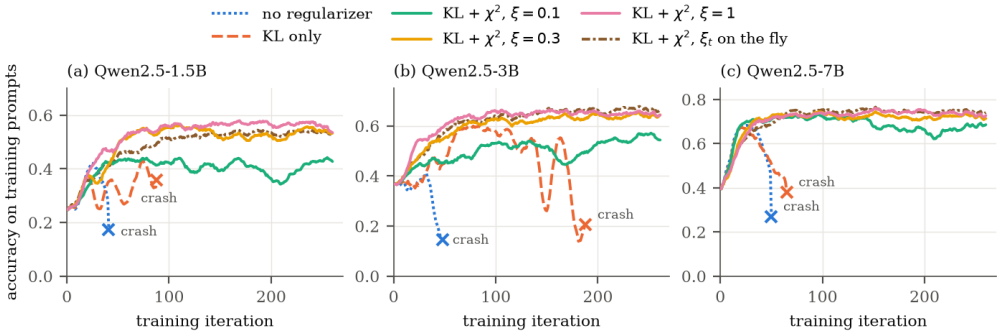

Figure 6. MATH training accuracy for the explicit-surrogate runs with fixed coefficients and the adaptive ξ_t setting. Model sizes follow Figure 5.

D.2 HYPERPARAMETERS

Method	Baseline temperature	Update temperature	Explicit ξ
PMD-PART	10^{-3}	10^{-3}	–
T^2-PART	0.5	10^{-3}	–
PMD-MEAN	∞	10^{-3}	–
Baseline-free surrogate	–	10^{-3}	$\{0, 0.1, 0.3, 1\}$ or $\widehat{p}_t$
No regularizer	–	0	0
Centered paired surrogate	0.5	10^{-3}	$\widehat{\xi}$ from each group

The following settings apply to the PMD and explicit-surrogate runs. Each configuration uses one seed, a learning rate 10^{-6}, gradient-norm clipping at 1, and the rollout policy π_t as the sequence-ratio reference. There is no ratio clipping or entropy bonus. Training generation uses temperature 1 and top-$p = 1$, while validation is greedy. Each prompt has $n = 4$ responses on MATH and $n = 16$ on DAPO. The dynamic microbatch token budget is 20,480. Each rollout batch has $256n$ responses and each mini-batch has $32n$, giving $K = 8$ optimizer steps. Response-level balancing can split a prompt group across mini-batches. One epoch of MATH is 29 iterations. Most runs were configured for 9 epochs (261 iterations). The 3B fixed-coefficient surrogate runs, including the two zero-χ^2 controls, were configured for 15 epochs (435 iterations). The MATH figures display the first 261 iterations. The selected 3B checkpoints for $\xi = 0.1$, 0.3, and 1 are at iterations 40, 110, and 70, respectively. DAPO runs have a two-day wall-clock limit and use two or four H200 GPUs, with endpoints listed below.

Reward and loss reduction. Both datasets use a boxed-answer correctness check and give zero reward to responses that reach the length limit without an end-of-sequence token. The per-response regression loss omits the factor $1/2$ in Equation (1), while the explicit surrogate uses the $\xi/2$ factor in Equation (3).

Our implementation forms token-balanced microbatches and averages each microbatch's per-sequence loss before accumulation. Let M be the local mini-batch size, m_0 the configured local microbatch size, and B_j the set of responses in microbatch j, with $m_j = |B_j|$. All sizes count response sequences. For per-sequence loss ℓ_i, the accumulated objective is

$$\frac{m_0}{M} \sum_j \frac{1}{m_j} \sum_{i \in B_j} \ell_i.$$

The PMD and explicit-surrogate runs use this reduction, so unequal microbatch sizes give unequal sequence weights during fitting. Equations (1) and (3) express the objectives under uniform response weighting.

D.3 RUN ENDPOINTS AND FAILURE LABELS

Failure labels combine automatic detection and author-classified collapse. When enabled, the automatic detector stops on NaN gradient norm, entropy loss, or mean reward. From the twentieth logged iteration onward, it also stops when the five-iteration mean training reward falls below half the running peak of the ten-iteration mean. Training logs and stopping reasons distinguish these failures from time limits and manual stopping, as listed below.

DAPO method	Last cached iteration	GPUs	Run outcome
PMD-PART	23	2	Failure reported at 25
T^2-PART	57	2	Failure reported at 58
PMD-MEAN	236	2	Two-day time limit
No regularizer	103	4	Reward collapse, then manually stopped
KL only	90	4	Reward collapse, then manually stopped
$\xi = 0.1$	111	4	Manually stopped
$\xi = 0.3$	292	4	Manually stopped
$\xi = 1$	102	4	Reward and response-length collapse, then manually stopped
$\xi_t = \widehat{p}_t$	301	2	Two-day time limit

The DAPO unregularized and KL-only histories first show zero mean reward at iterations 99 and 89. The $\xi = 1$ run retains finite gradient norms but loses reward and response length, ending at iteration 102. These differences in training length and compute budget limit direct accuracy comparisons.

For MATH 1.5B with KL only, validation accuracy deteriorates at iteration 50 while training reward remains near 0.35. The table uses 50 as the reported failure iteration. The run was manually stopped at iteration 88, where its curve ends. For the 7B regression failures, the last cached points are 32 and 34, preceding the reported stop iterations 33 and 35. Crosses in the figures mark these last available points.

For MATH 1.5B with $\beta_b = 0.5$, the training-reward criterion is met at iteration 209. The run remains numerically finite and is stopped manually after iteration 212. Tables label this outcome as degraded. The comparison below gives its validation results and diagnostics.

D.4 Details of regression and its corresponding surrogate

Configuration. The paired runs follow the MATH configuration in Appendix D.2, with Qwen2.5-1.5B, $\beta_u = 0.001$, and $\beta_b = 0.5$. The prompt and response caps are 2,048 and 8,192 tokens. AdamW uses momentum parameters $(0.9, 0.999)$, numerical constant 10^{-8}, and weight decay 0.01. Sequence log probabilities include the end-of-sequence token. Both objectives use the dynamic microbatch reduction defined above.

The surrogate uses the same inclusive soft baseline as regression and the coefficient obtained by solving the normalization equation for the uniformly weighted empirical reward group. For k correct responses among four, these coefficients are approximately

$$(\widehat{\xi}_{k=0}, \ldots, \widehat{\xi}_{k=4}) = (0,\ 0.130346,\ 0.141208,\ 0.091119,\ 0).$$

The baseline, coefficient, and rollout reference remain fixed through the eight fitting steps. The surrogate has multiplier $2\beta_u = 0.002$, with no coefficient floor. This matches the losses' derivatives with respect to $\log q_i$ at $q_i = 1$ under common response weights. Dynamic microbatching gives unequal weights during fitting, while $\widehat{\xi}$ is computed for the uniformly weighted group.

Evaluation and stopping. Both runs start from the same model parameters with a fresh optimizer and share the sampling seed and prompt order. Their sampled responses can differ after their policies diverge. The intended horizon is 261 rollout iterations. Greedy MATH500 validation runs before training, every ten iterations, and at completion. Table 8 reports validation accuracy under the strict end-of-sequence and boxed-answer reward. The benchmark tables instead use the Qwen2.5-Math answer grader. Their values come from a separate evaluation pass.

Regression validation accuracy drops to 4.8% at iteration 130 and recovers before falling again to 4.6% at iteration 210. The run meets the reward-deterioration criterion in Appendix D.3 at iteration 209 and is stopped manually after 212. Its last validation is at 210. The logs show no non-finite training statistic, out-of-memory error, or exception. The surrogate completes all 261 iterations without triggering its automatic stop.

Table 8. Greedy validation accuracy (%) for the paired MATH runs. Best checkpoints are selected by this validation metric. The last evaluations occur at different iterations because regression is stopped early. These values use the training reward function and are separate from the benchmark evaluations in Table 1.

	Regression	Centered surrogate
Initial accuracy	48.0	48.0
Best accuracy (iteration)	55.6 (20)	56.0 (190)
Last accuracy (iteration)	4.6 (210)	52.8 (261)
Last training iteration	212	261
Outcome	Manually stopped after deterioration	Completed

Regression diagnostics. The sequence-ratio 99th percentile on the eighth mini-batch repeatedly falls below 10^{-3} and stays below that value from iteration 186. These measurements follow seven optimizer steps and precede the eighth. Before the final deterioration, mean response length reaches about 3,900 tokens with 15–21% of responses at the cap.

For a failed response, the regression target for its log ratio is $-v/\beta_u$. In mixed groups with one, two, or three successes, it is approximately -477, -717, or -878, respectively. It is zero in an all-failure group.

Surrogate diagnostics. The batch-mean coefficient ranges from 0.035 to 0.069. Homogeneous groups receive zero coefficient and account for 43–70% of groups. The mean coefficient over mixed groups is approximately 0.11–0.13. Over the last thirty iterations, the eighth-mini-batch ratio 99th percentile is about 3–6, the empirical χ^2 about 0.5–2, the effective-sample-size fraction about 0.33–0.61, and the mean response length about 560–660 tokens.

At iteration 174, the eighth mini-batch has a maximum ratio of about 2,208 and empirical χ^2 of about 3.8×10^4. An individual optimizer step has a gradient norm of about 3,739 before clipping

to 1.0. The gradient panel in Figure 7 instead averages over the eight steps. Reward and response length remain stable through this event. Thus gradient clipping is an active part of this result, and completing training does not imply uniformly small ratios.

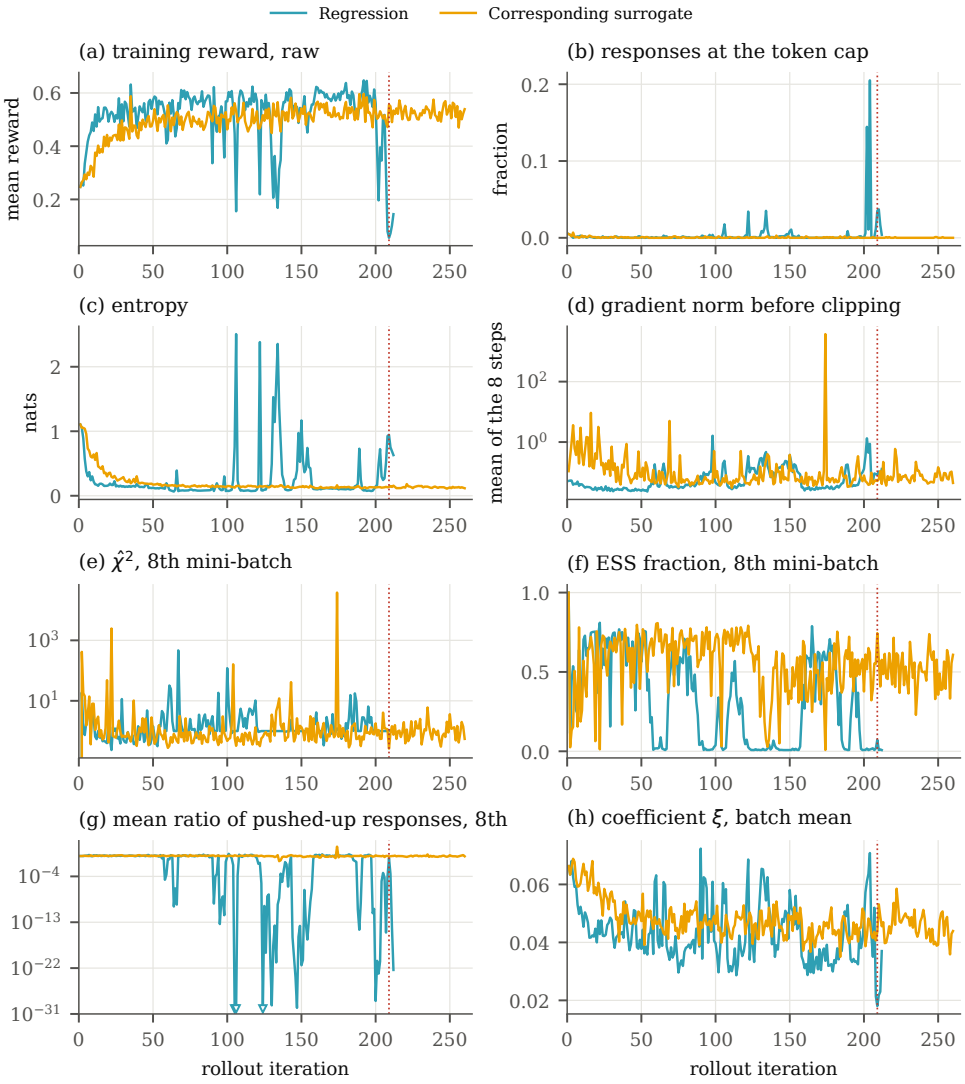

Figure 7. Further diagnostics of the paired runs, using unsmoothed per-iteration values. (a) Training reward. (b) Fraction of responses at the response cap. (c) Entropy. (d) Gradient norm before clipping, averaged over eight optimizer steps. (e) Empirical χ^2, (f) effective-sample-size fraction, and (g) mean sequence ratio among responses with positive advantage, each measured on the eighth mini-batch before its optimizer step. In (g), downward triangles mark values at or below the display floor 10^{-30}. (h) Batch-mean coefficient, used explicitly by the surrogate and computed from the regression's empirical groups for comparison. The dotted line marks iteration 209.