EVPO: Explained Variance Policy Optimization for Adaptive Critic Utilization in LLM Post-Training

Anonymous authors
Paper under double-blind review

Abstract

Reinforcement learning (RL) for LLM post-training faces a fundamental design choice: whether to use a learned critic as a baseline for policy optimization. Classical theory favors critic-based methods such as PPO for variance reduction, yet critic-free alternatives like GRPO have gained widespread adoption due to their simplicity and competitive performance. We show that in sparse-reward settings, a learned critic can inject estimation noise that exceeds the state signal it captures, increasing rather than reducing advantage variance. By casting baseline selection as a Kalman filtering problem, we unify PPO and GRPO as two extremes of the Kalman gain and prove that *explained variance* (EV), computable from a single training batch, identifies the exact boundary: positive EV indicates the critic reduces variance, while zero or negative EV signals that it inflates variance. Building on this insight, we propose **E**xplained **V**ariance **P**olicy **O**ptimization (**EVPO**), which monitors batch-level EV at each training step and adaptively switches between critic-based and batch-mean advantage estimation, provably achieving no greater variance than the better of the two at every step. Across four tasks spanning classical control, agentic interaction, and mathematical reasoning, EVPO consistently outperforms both PPO and GRPO regardless of which fixed baseline is stronger on a given task. Further analysis shows that adaptive gating tracks critic maturation and reveals an approximately concave threshold-sensitivity profile. Code for all experiments is available at `https://anonymous.4open.science/r/EVPO-9785`.

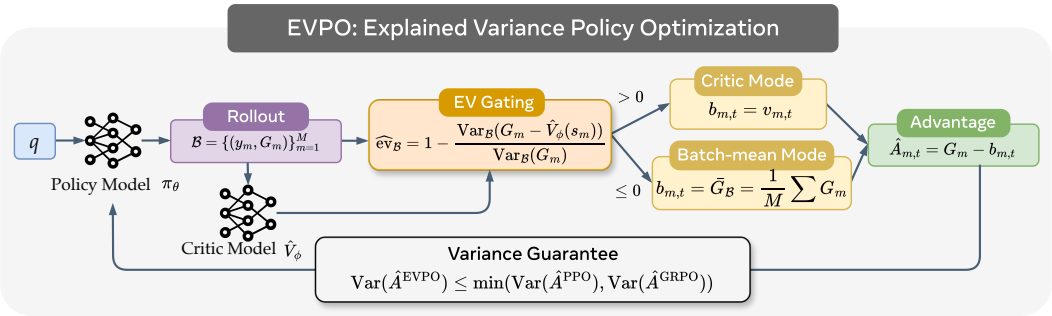


Figure 1: Overview of EVPO. The EV gating module selects between the critic baseline (EV > 0) and the batch-mean baseline (EV ≤ 0), guaranteeing advantage variance no greater than the minimum of PPO and GRPO.

1 Introduction

Reinforcement learning (RL) has become a central paradigm for LLM post-training, driving advances in mathematical reasoning (Guo et al., 2025; Singh et al., 2025; Qwen Team, 2026; Yang et al., 2025), instruction following (Zhou et al., 2023; Ding et al., 2026; Dou et al., 2026), and interactive decision-making (Shridhar et al., 2021; Wang et al., 2025; Chen et al., 2025). A prominent instantiation is

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.

reinforcement learning with verifiable rewards (RLVR) (Guo et al., 2025), where a sparse reward is assigned only at the terminal step via an automatic verifier. A fundamental algorithmic choice in this setting is whether to employ a *critic* (a learned value function) as a baseline for policy optimization.

Classical RL theory holds that a critic reduces policy-gradient variance by providing a state-dependent baseline (Sutton et al., 1999; Schulman et al., 2018), forming the foundation of actor-critic methods including PPO (Schulman et al., 2017). In practice, however, critic-free methods such as GRPO (Shao et al., 2024; Guo et al., 2025) have gained widespread adoption, as they are cheaper and can match or even surpass PPO (Hu et al., 2025). Rather than treating the critic as a binary design choice, we ask: *In sparse-reward LLM post-training, when does a learned critic help and when does it hurt?*

To answer this, we identify three progressive sub-questions. $\mathcal{Q}_1$: *Is the boundary observable?* Critic quality fluctuates during training, and we need to measure its impact on policy optimization in a controlled setting. $\mathcal{Q}_2$: *Where exactly is the boundary?* In sparse-reward LLM generation, the critic may inject more estimation noise than the state signal it captures. We need a precise characterization of when this happens. $\mathcal{Q}_3$: *Can this boundary be exploited online?* If the helpful-vs-harmful regime can be detected in real time, we can build an algorithm that adapts rather than committing to a fixed choice.

We adopt *explained variance* (EV), the fraction of return variance accounted for by the critic, as a real-time proxy for critic usefulness. Empirically ($\mathcal{Q}_1$), EV remains negative during the early stage of training, and controlled interventions (suppressing the critic or injecting noise) confirm that performance degrades once EV falls to zero or below, suggesting that its sign reliably indicates whether the critic helps or hurts (Section 2.2). To pinpoint the boundary ($\mathcal{Q}_2$), we cast baseline selection as a Kalman filtering problem (Kalman, 1960) in which the critic and the batch-mean return are two competing estimators of the true state value, with PPO and GRPO corresponding to the two extremes of the Kalman gain (Section 2.3). We prove that the sign of EV, computable from a single batch, is sufficient to determine which extreme is preferable (Theorem 1), placing the boundary at $\mathrm{EV} = 0$.

Exploiting this boundary online ($\mathcal{Q}_3$), we propose **Explained Variance Policy Optimization (EVPO)**, which implements this adaptive hard gating at each training step. EVPO maintains a standard PPO-style critic alongside the policy, and at each iteration computes batch-level EV as a scalar indicator. When EV is positive, the critic captures meaningful state-dependent signal, and EVPO uses its predictions as the baseline (the PPO mode); when EV falls to zero or below, EVPO discards the critic and falls back to the batch-mean return (the GRPO mode). We prove that this switching rule yields advantage-estimator variance no greater than the better of PPO and GRPO at every step, so EVPO never underperforms either fixed choice.

We evaluate EVPO on four tasks spanning classical control, interactive decision-making, and mathematical reasoning. Across all tasks, EVPO achieves the best final performance and maintains its advantage throughout training, regardless of whether PPO or GRPO is the stronger fixed baseline on a given task. Further analysis confirms that the adaptive gating tracks critic maturation over training and reveals an approximately concave threshold-sensitivity profile over the tested range.

Our main contributions are:

1. **Indicator and theory.** We unify PPO and GRPO as two extremes of a Kalman baseline-selection spectrum and prove that $\mathrm{ev} = 0$ is the exact boundary separating the variance-reducing from the variance-inflating critic regime (Theorem 1), where critic estimation noise equals the state-value signal it captures.
2. **Algorithm.** We propose EVPO, a lightweight method that switches between critic-based and critic-free advantage estimation via batch-level EV, with a population-level guarantee of no greater advantage variance than the better of PPO and GRPO at every step. It reuses existing critic predictions, requiring no additional learned parameters or forward passes beyond PPO.
3. **Empirical validation.** EVPO consistently outperforms fixed PPO and GRPO across four tasks spanning classical control, agentic interaction, and mathematical reasoning. It achieves the highest peak validation success rate among all compared methods across all tasks, regardless of which fixed baseline is stronger on each task. Further analyses link the gating dynamics to critic maturation and characterize sensitivity to the EV switching threshold.

2 PRELIMINARIES

This section establishes that the critic is not uniformly beneficial. We review PPO and GRPO as two baseline-selection extremes (Section 2.1), then show empirically that the sign of explained variance reliably predicts whether the critic helps or hurts (Section 2.2).

2.1 RL FOR LLM POST-TRAINING

We model autoregressive text generation as a token-level Markov Decision Process (OpenAI et al., 2024; Trung et al., 2024; Zheng et al., 2023). Given a prompt x, the policy π_θ generates a response $y = (y_1, \ldots, y_T)$ token by token, where the state at step t is $s_t = (x, y_{<t})$ and the action is $a_t = y_t \sim \pi_\theta(\cdot \mid s_t)$.

In the reinforcement learning with verifiable rewards (RLVR) setting (Guo et al., 2025; Wang et al., 2025), a scalar reward is assigned only at the terminal step: $r_T = \mathcal{R}(x, y)$, $r_t = 0$ for $t < T$. With discount factor $\gamma = 1$, the return from any state equals the terminal reward: $G_t = r_T \triangleq G$. The state-value function is $V^\pi(s) = \mathbb{E}_\pi[G \mid s_t = s]$, and the observed return G is a stochastic realization whose variance arises solely from the randomness of future actions under π.

PPO and GRPO. Both PPO (Schulman et al., 2017) and GRPO (Guo et al., 2025; Shao et al., 2024) optimize a clipped surrogate objective but differ in how they estimate advantages. Under RLVR conditions (γ=λ=1, terminal-only reward), the PPO advantage reduces to (see Appendix A.2 for the full GAE derivation):

$$\hat{A}_t^{\text{PPO}} = G - \hat{V}_\phi(s_t), \tag{1}$$

where $\hat{V}_\phi$ is a learned critic providing a *state-dependent* baseline. GRPO eliminates the critic and instead computes a *response-level* advantage from batch statistics. Concretely, for each prompt x, a batch of K responses $\{y^{(i)}\}_{i=1}^K$ is sampled from the current policy, yielding returns $\{G_i\}_{i=1}^K$ with batch mean $\bar{G}_\mathcal{B} = \frac{1}{K}\sum_{i=1}^K G_i$ and standard deviation $\sigma(G_\mathcal{B})$. The GRPO advantage is then:

$$\hat{A}_i^{\text{GRPO}} = \frac{G_i - \bar{G}_\mathcal{B}}{\sigma(G_\mathcal{B})}, \tag{2}$$

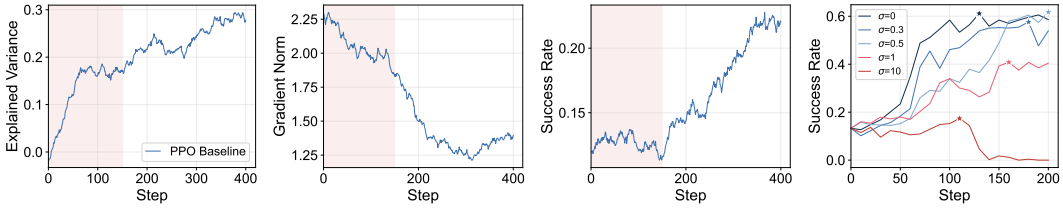
where $\bar{G}_\mathcal{B}$ and $\sigma(G_\mathcal{B})$ are the batch mean and standard deviation of returns, serving as a *constant* baseline shared across all tokens.

2.2 THE IMPACT OF CRITIC QUALITY ON POLICY OPTIMIZATION

Returning to $\mathcal{Q}_1$ (is the boundary observable?), we quantify critic quality via the *explained variance* (EV) metric, a standard indicator from regression analysis (Schulman, 2016):

$$\text{ev} = 1 - \frac{\text{Var}(G - \hat{V}_\phi(s))}{\text{Var}(G)}, \tag{3}$$

which quantifies the return variance explained by the critic: $\text{ev} > 0$ favors the critic, whereas $\text{ev} \leq 0$ favors a constant baseline because estimation noise outweighs state-dependent signal.

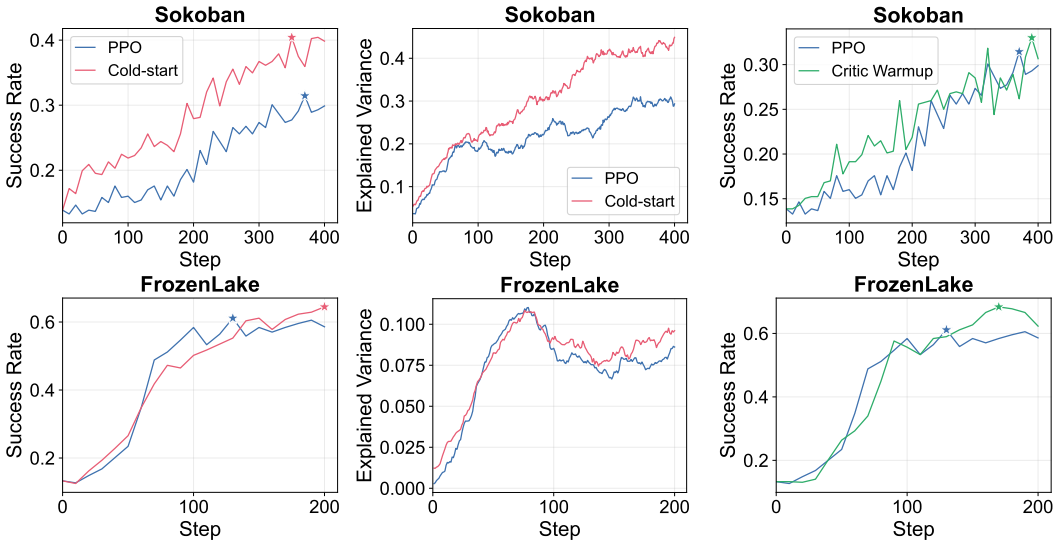


(a) Explained Variance (b) Gradient Norm (c) Train Success (d) Noise Injection

Figure 2: Training dynamics of PPO on Sokoban. **(a)** Batch-level EV over training steps. **(b)** Policy gradient norms. **(c)** Training success rate. The shaded region in (a–c) marks the early phase where EV is low. **(d)** Effect of injecting Gaussian noise into a converged critic on FrozenLake.

Observation 1: A noisy critic hurts the actor. We train a PPO policy on Sokoban and track the batch-level EV throughout training (see Appendix D.1 for details). As shown in Figure 2a, EV remains consistently low during the first $\sim$150 training steps, indicating that the critic's predictions are less accurate than a simple constant baseline during this phase. This poor critic quality has two visible consequences: gradient norms are notably elevated (Figure 2b), reflecting high-variance policy updates driven by noisy advantage estimates, and the training success rate remains low and unstable (Figure 2c), showing that the actor struggles to learn under unreliable critic guidance.

To further confirm the causal link between critic quality and actor performance, we conduct a controlled experiment: injecting Gaussian noise ($\sigma_\delta = 0.3$; the mean absolute state value is ≈ 0.5) into a *converged* critic's outputs. As shown in Figure 2d, performance degrades monotonically as the noise level increases, with complete collapse at σ=10 where EV is driven far below zero. This confirms that **negative EV is a reliable indicator of harmful critic influence**, and that the relationship between critic quality and actor performance is not merely correlational but causal.



(a) Cold-start (b) Critic Warmup

Figure 3: Reducing critic variance helps the actor. **(a)** Cold-start on Sokoban (top row) and FrozenLake (bottom row): replacing the critic output with $\bar{G}_\mathcal{B}$ during early training improves both the success rate (left column) and the explained variance of the critic (right column), confirming that the intervention raises critic quality precisely in the regime where it was initially harmful. **(b)** Critic Warmup: pre-training the critic before actor training begins, on Sokoban (top) and FrozenLake (bottom).

Observation 2: Reducing critic variance helps the actor. We conduct two complementary interventions on the same Sokoban and FrozenLake setups (see Appendix D.2 for details): (i) *Cold-start*: replacing the critic's output with the batch mean $\bar{G}_\mathcal{B}$ during the first 50/25 steps, and (ii) *Warm-up*: pre-training the critic alone for 200 steps before the actor begins updating. Both interventions yield faster convergence and more stable training curves (Figure 3), confirming that **suppressing critic variance during low-quality phases directly benefits the policy**.

These observations suggest that the choice between a learned critic and a batch-mean baseline should not be fixed a priori, but should adapt to the critic's current quality. In the following subsection, we formalize this intuition through a Kalman filtering lens.

2.3 A Kalman Filtering Perspective on Baseline Selection

To precisely locate the boundary ($\mathcal{Q}_2$), we observe that choosing a baseline is fundamentally a *state-estimation* problem: both the critic $\hat{V}_\phi(s)$ and the batch mean $\bar{G}_\mathcal{B}$ are noisy estimates of the true value $V^\pi(s)$, and the best baseline is the one with lower estimation error. This is exactly the setting

addressed by the Kalman filter (Kalman, 1960; Pei et al., 2019), which optimally fuses two noisy estimates by weighting each inversely to its error variance.

Setup. The observed return decomposes as $G = V^\pi(s) + \epsilon$ with sampling noise $R \triangleq \mathbb{E}[\mathrm{Var}(G \mid s)]$. The two competing baselines have distinct error structures: (i) the *critic* $\hat{V}_\phi(s) = V^\pi(s) + \delta(s)$ with estimation error variance $P_A \triangleq \mathrm{Var}(\delta)$, and (ii) the *batch mean* $\bar{G}_\mathcal{B}$ whose deviation from any specific state value has variance $P_B \triangleq \mathrm{Var}_s(V^\pi(s))$. The minimum-variance linear combination of these two estimators assigns the critic a weight of $1 - K^*$ with the optimal gain $K^* = P_A/(P_A + P_B)$ (see Appendix A.1 for the full derivation).

Connection to explained variance. The EV metric (Eq. (3)) directly encodes the relative reliability of the two estimators. Substituting the variance decomposition into the definition of EV yields:

$$\mathrm{ev} = \frac{P_B - P_A}{P_B + R}. \tag{4}$$

Since the denominator $P_B + R$ is always positive, the *sign* of EV is determined solely by $P_B - P_A$. This means $\mathrm{ev} = 0$ **marks the exact boundary where** $P_A = P_B$, i.e., where the critic's estimation noise equals the state signal it captures. Above this boundary ($\mathrm{ev} > 0$), the critic outperforms the batch mean; below it ($\mathrm{ev} \leq 0$), critic noise dominates. The sampling noise R compresses the magnitude of EV but preserves its sign, keeping this boundary robust under high stochasticity.

3 Methodology

Building on the Kalman filtering framework of Section 2.3, we now formalize when a critic helps or hurts (Section 3.1), unify PPO and GRPO as two extremes of the Kalman gain (Section 3.2), and derive the EVPO algorithm (Section 3.3).

3.1 The Performance Collapse Boundary

From Eq. (4), the sign of EV is governed by $P_B - P_A$. We formalize this as:

Theorem 1 (Performance Collapse Boundary). *Let $\hat{V}_\phi(s) = V^\pi(s) + \delta(s)$ with $\delta \perp \epsilon$, and let $P_A = \mathrm{Var}(\delta)$, $P_B = \mathrm{Var}_s(V^\pi(s))$, $K = P_A/(P_A + P_B)$. Then the following are equivalent:*

$$\mathrm{ev} \leq 0 \iff P_A \geq P_B \iff K \geq \tfrac{1}{2}. \tag{5}$$

In words, the critic inflates advantage variance whenever its estimation noise (P_A) exceeds the state signal it captures (P_B), and $\mathrm{ev} = 0$ is the exact tipping point. Since the sign of EV is computable from a single batch, this boundary is directly observable during training.

Computing EV from a training batch. Replacing the population variances in Eq. (3) with sample variances over a batch of M rollouts $\mathcal{B} = \{(y_m, G_m)\}_{m=1}^M$ from the same prompt gives the batch-level EV:

$$\widehat{\mathrm{ev}}_\mathcal{B} = 1 - \frac{\mathrm{Var}_\mathcal{B}(G_m - \hat{V}_\phi(s_m))}{\mathrm{Var}_\mathcal{B}(G_m)}, \tag{6}$$

computed from the critic values already obtained for advantage estimation, with no additional forward passes.

3.2 PPO and GRPO as Kalman Extremes

Under RLVR conditions (γ=λ=1, terminal-only reward), PPO and GRPO can be unified into the form $\hat{A}_t = G - b(s_t)$, differing only in the choice of baseline $b(s_t)$ (see Appendix A.2 for the detailed derivation). This difference maps directly onto the Kalman gain K:

- **PPO** (critic mode, K=0, full critic trust): $b(s_t) = \hat{V}_\phi(s_t)$. The baseline is entirely model-driven; the Kalman filter places zero weight on the evidence-driven estimator.

- **GRPO** (batch-mean mode, $K{=}1$, no critic trust): $b(s_t) = \bar{G}_{\mathcal{B}}$. (Since $1/\sigma(G_{\mathcal{B}})$ does not affect the gradient direction.) The baseline is entirely evidence-driven; the critic is discarded and the batch mean serves as a constant baseline for all tokens.

In other words, PPO and GRPO are not fundamentally different algorithms but rather two extreme operating points of the same Kalman baseline-selection spectrum. PPO implicitly assumes $P_A \ll P_B$ (the critic is always reliable), while GRPO implicitly assumes $P_A \gg P_B$ (the critic is never worth using). Neither assumption holds uniformly across training, motivating an adaptive approach.

Algorithm 1 Explained Variance Policy Optimization (EVPO)

Require: Policy π_θ, critic $\hat{V}_\phi$, task set $\mathcal{D}$, group size M, number of tasks N
for each training iteration **do**
  Sample N tasks $\{x_j\}_{j=1}^{N} \sim \mathcal{D}$
  for $j = 1, \ldots, N$ **do**
    Rollout a batch with M trajectories $\mathcal{B} = \{y_m\}_{m=1}^{M} \sim \pi_\theta(\cdot \mid x_j)$
    Obtain rewards $G_m = \mathcal{R}(x_j, y_m)$; compute critic values $v_{m,t} \leftarrow \hat{V}_\phi(s_{m,t})$
    $\widehat{\mathrm{ev}}_{\mathcal{B}} \leftarrow 1 - \mathrm{Var}_{\mathcal{B}}(G_m - v_m) \,/\, \mathrm{Var}_{\mathcal{B}}(G_m)$
    if $\widehat{\mathrm{ev}}_{\mathcal{B}} > 0$ **then**
      $b_{m,t} \leftarrow v_{m,t}$ ▷ *Critic mode*
    else
      $b_{m,t} \leftarrow \bar{G}_{\mathcal{B}} = \frac{1}{M}\sum_{m=1}^{M} G_m$ ▷ *Batch-mean mode*
    end if
    $\hat{A}_{m,t} \leftarrow G_m - b_{m,t}$
  end for
  Aggregate gradients over N tasks; update π_θ and $\hat{V}_\phi$
end for

3.3 EVPO: Adaptive Baseline Selection via EV Gating

Having established *where* the collapse boundary lies ($\mathcal{Q}_2$), we now turn to $\mathcal{Q}_3$: *can this boundary be exploited online?* The Kalman framework prescribes the optimal baseline as a continuous interpolation of the two estimators:

$$b(s) = (1 - K)\,\hat{V}_\phi(s) + K\,\bar{G}_{\mathcal{B}}, \quad K = \frac{P_A}{P_A + P_B} \in [0, 1]. \tag{7}$$

However, computing K directly is infeasible: both $P_A = \mathrm{Var}(\hat{V}_\phi(s) - V^\pi(s))$ and $P_B = \mathrm{Var}_s(V^\pi(s))$ depend on the unknown true value $V^\pi(s)$, making the continuous K non-identifiable in practice. Two remedies are available. The first is a pragmatic fallback: replace the unsamplable $V^\pi(s)$ with the samplable return G, which yields an estimate systematically biased toward $\frac{1}{2}$. We call this approach *Continual* K (Cont. K; see Appendix A.1 for the derivation and Appendix E.3 for the empirical comparison); in our experiments we find that such an overly moderate estimate fails to exploit the strengths of either PPO or GRPO.

The second remedy—adopted by EVPO—is to further reduce the domain of K from the continuous $[0, 1]$ to the binary set $\{0, 1\}$, corresponding to the two extremes: $K{=}0$ (PPO, full critic trust) and $K{=}1$ (GRPO, no critic). The decision boundary $\frac{P_A}{P_A+P_B} = \frac{1}{2}$ (i.e., $P_A = P_B$) corresponds exactly to $\mathrm{ev} = 0$ by Theorem 1. While the *value* of K is non-identifiable, its position relative to $\frac{1}{2}$ can be determined by the sign of $\widehat{\mathrm{ev}}_{\mathcal{B}}$ (Eq. (6)). This yields the **EVPO switching rule**:

$$b(s_t) = \begin{cases} \hat{V}_\phi(s_t), & \text{if } \widehat{\mathrm{ev}}_{\mathcal{B}} > 0 \quad (K < \frac{1}{2}:\ \text{trust critic}), \\ \bar{G}_{\mathcal{B}}, & \text{if } \widehat{\mathrm{ev}}_{\mathcal{B}} \le 0 \quad (K \ge \frac{1}{2}:\ \text{retreat to batch mean}), \end{cases} \tag{8}$$

where $\widehat{\mathrm{ev}}_{\mathcal{B}}$ is the batch-level EV defined in Eq. (6).

Variance guarantee. By construction, EVPO selects the lower-variance baseline at each step:

$$\mathrm{Var}(\hat{A}^{\mathrm{EVPO}}) = R + \min(P_A,\ P_B), \tag{9}$$

which is no greater than $\mathrm{Var}(\hat{A}^{\mathrm{PPO}}) = R + P_A$ or $\mathrm{Var}(\hat{A}^{\mathrm{GRPO}}) = R + P_B$ at every training step. Over the course of training, this per-step guarantee accumulates: EVPO avoids the critic's noisy cold-start phase ($P_A \gg P_B$) while fully exploiting a mature critic ($P_A \ll P_B$).

4 EXPERIMENTS

Our experiments aim to validate that the per-step variance reduction of EVPO (Eq. 9) translates into consistent performance gains over fixed baselines (Section 4.2). We further analyze whether EV serves as a reliable indicator of the P_A-vs-P_B balance throughout training (Section 4.3) and examine EV threshold sensitivity (Appendix E.5).

4.1 EXPERIMENTAL SETUP

Evaluation. We evaluate on four tasks that span diverse agent interaction paradigms. **Sokoban** (Junghanns & Schaeffer, 2001) and **FrozenLake** (Dell'Aversana, 2021) are multi-turn grid-world games requiring sequential planning under sparse rewards, with FrozenLake additionally introducing environmental stochasticity. **WebShop** (Yao et al., 2022) is an interactive web-based agentic task involving multi-step instruction following and decision making. **MATH** targets competition-level mathematical reasoning trained and evaluated on problems in DAPO-Math-17k (Yu et al., 2025).

Baseline. We compare EVPO against the **Base LLM** without RL post-training, two critic-based methods (**PPO** (Schulman et al., 2017) and **StarPO-S** (Wang et al., 2025)), and two critic-free methods (**GRPO** (Shao et al., 2024) and **DAPO** (Yu et al., 2025)). For the Math task we use Qwen2.5-7B-Instruct (Qwen et al., 2025) as the base model; the other three tasks use Qwen2.5-3B-Instruct (Qwen et al., 2025). To ensure fair comparison, the hyperparameter recipe is tuned independently per task but kept consistent across all methods on the same task. Full configurations are provided in Appendix B.

4.2 MAIN RESULTS

Table 1: Best validation success rate across four tasks. For each RL method, we report the peak of the mean validation curve over five runs, followed in parentheses by the across-run standard deviation at the same training step. **Bold** indicates the highest mean per task. Each per-run success rate is computed from 512 evaluation results, obtained by sampling 16 times for each of the 32 queries.

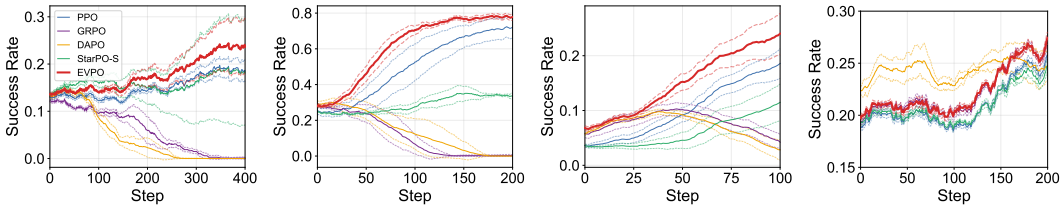
Method	Sokoban	FrozenLake	WebShop	MATH
Base LLM	0.139	0.133	0.025	0.332
Critic-based methods				
PPO	0.270 ($\pm$0.026)	0.582 ($\pm$0.060)	0.187 ($\pm$0.028)	0.361 ($\pm$0.023)
StarPO-S	0.264 ($\pm$0.151)	0.249 ($\pm$0.062)	0.151 ($\pm$0.023)	0.355 ($\pm$0.017)
Critic-free methods				
GRPO	0.164 ($\pm$0.028)	0.164 ($\pm$0.029)	0.113 ($\pm$0.029)	0.373 ($\pm$0.006)
DAPO	0.173 ($\pm$0.020)	0.224 ($\pm$0.021)	0.114 ($\pm$0.010)	0.370 ($\pm$0.014)
EVPO(*ours*)†	**0.362** ($\pm$0.088)	**0.651** ($\pm$0.034)	**0.224** ($\pm$0.043)	**0.386** ($\pm$0.005)

† Throughout the paper, EVPO refers to the five-run aggregate reported here, including when reused as the reference in subsequent analyses.

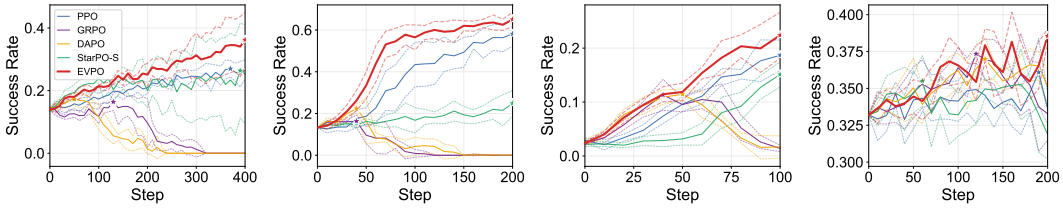
Finding 1: EVPO improves over both PPO and GRPO regardless of which fixed baseline is stronger on a given task.

The stronger fixed baseline varies across tasks: critic-based methods (PPO, StarPO-S) clearly outperform critic-free methods (GRPO, DAPO) on Sokoban, FrozenLake, and WebShop, while critic-free methods are more competitive on MATH. EVPO surpasses both families in every case

(Table 1), confirming that it is not tied to a single favorable regime but adapts to the task-specific P_A-vs-P_B balance.



(a) Sokoban (Train) (b) FrozenLake (Train) (c) WebShop (Train) (d) MATH (Train)



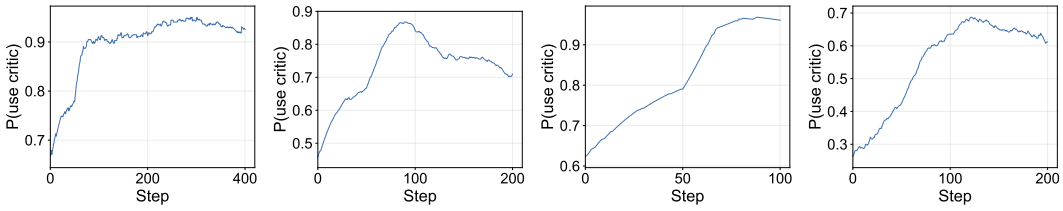
(e) Sokoban (Val) (f) FrozenLake (Val) (g) WebShop (Val) (h) MATH (Val)

Figure 4: Training and validation success rates across four tasks, aggregated over five runs. Solid curves show the across-run means, and dashed curves show mean $\pm$ one standard deviation. **Top row:** training success rate (smoothed). **Bottom row:** validation success rate (unsmoothed; stars mark each method's peak). EVPO tracks or exceeds the best competing method throughout most of training.

Finding 2: EVPO's advantage persists throughout training, not just at convergence.

As shown in Figure 4, EVPO leads or matches the strongest baseline at most training stages across all four tasks. This indicates that adaptive switching takes effect continuously—not by recovering from a poor start or catching up late, but by maintaining lower-variance advantage estimates throughout the optimization trajectory. The per-step variance guarantee (Eq. 9) predicts exactly this behavior: since EVPO selects the lower-variance baseline at every step, its cumulative benefit should manifest as a sustained advantage rather than a late-stage correction.

4.3 Gating Behavior Analysis



(a) Sokoban (b) FrozenLake (c) WebShop (d) MATH

Figure 5: Gating trigger probability ($\widehat{\mathrm{ev}}_{\mathcal{B}} > 0$, i.e., selection of critic over batch mean) across training.

Finding 3: In most tasks, gating frequency increases monotonically over training, reflecting critic maturation from $P_A > P_B$ to $P_A < P_B$.

Figure 5 tracks the critic-selection probability over training. Across Sokoban, WebShop, and MATH, the batch-mean mode is triggered frequently in early training and progressively less as training continues, consistent with the expectation that P_A decreases as the critic improves. A notable exception is FrozenLake, where gating frequency reverses around step 80, coinciding with training stagnation (Figure 4(b)). This suggests that when the policy enters a difficult region, critic quality can temporarily regress, and the adaptive gating responds by retreating to the batch-mean baseline.

4.4 EV Threshold Sensitivity

Finding 4: Threshold sensitivity exhibits an approximately concave profile over the tested range.

We sweep the EV switching threshold over $\tau \in \{-0.2, -0.1, 0, 0.1, 0.2\}$ on Sokoban and FrozenLake while keeping other configurations fixed. Training and validation performance show broadly single-peaked profiles, with quadratic fits suggesting approximate concavity. Among the tested thresholds, $\tau = -0.1$ yields the highest observed performance, while $\tau = 0$ remains competitive. Higher thresholds favor the batch-mean baseline and can discard useful critic information, whereas lower thresholds retain the critic more often and may admit unreliable value estimates. Full curves and analysis are provided in Appendix E.5.

5 Related Work

Reinforcement Learning with Verifiable Rewards for LLMs. RL-based LLM post-training uses either learned critics, as in PPO (Schulman et al., 2017) with GAE (Schulman et al., 2018), or critic-free return-based estimators. GRPO (Shao et al., 2024; Guo et al., 2025; Wang et al., 2026) replaces the learned baseline with intra-group reward normalization; REINFORCE++ (Hu et al., 2025) uses global batch normalization, while RLOO (Ahmadian et al., 2024) and DAPO (Yu et al., 2025) further demonstrate the effectiveness of critic-free estimation for single-turn, verifiable tasks.

Yet critic-free methods face challenges: group-relative advantages can introduce statistical bias, underestimating hard prompts and overestimating easy ones (Jain et al., 2025; Lin et al., 2026), and uniform trajectory-level credit assignment can struggle in long-horizon settings without early termination (de Oliveira et al., 2025; Dong et al., 2025), motivating multi-turn frameworks such as RAGEN (Wang et al., 2025) to retain a critic. These approaches typically resolve the trade-off between critic-free simplicity and critic-based credit assignment (Li et al., 2025) through a fixed baseline choice, irrespective of the critic's current predictive quality.

Adaptive Baselines and Explained Variance. Reducing policy-gradient variance is a foundational RL challenge (Williams, 1992; Sutton et al., 1999). In LLM post-training, sparse rewards and a vast state space can make critic estimates unreliable, limiting the benefits of a state-dependent baseline. Related approaches average value estimates (Anschel et al., 2017), adapt GAE's λ to critic quality (Mohammadpour et al., 2024), or filter uninformative transitions (Flet-Berliac & Preux, 2019).

However, these methods treat critic quality as a signal to tune hyperparameters or curate data while preserving the actor-critic architecture. EVPO departs from both strategies by elevating EV from a passive diagnostic to a architecture selection criterion that determines *whether* the critic is used at all, switching to critic-free estimation when the value network becomes mathematically harmful.

6 Conclusion

We presented EVPO, an adaptive RL method for LLM post-training that uses explained variance to switch between critic-based and batch-mean advantage estimation. Our Kalman analysis identifies population EV of zero as the boundary between variance-reducing and variance-inflating critic regimes (Theorem 1). Across four tasks, EVPO consistently outperforms both fixed baselines. Gating analysis shows that the adaptation tracks critic maturation, while threshold sensitivity exhibits an approximately concave profile over the tested range.

Limitations

Theorem 1 concerns population EV, whereas EVPO uses batch EV, for which we provide no finite-sample concentration or calibration bounds. Estimation error may contribute to the threshold offsets (Section 4.4), but their cause remains unverified; zero is a theoretically motivated default, not a guaranteed empirical optimum. EVPO maintains a critic throughout training, retaining PPO's memory footprint even in batch-mean mode. Our evaluation covers four tasks across three domains; generalization to dense rewards, learned reward models, and substantially different rollout lengths remains untested.

AI USE STATEMENT

Pretrained LLMs are an important component of this research: the policy π_θ trained by EVPO is itself a pretrained LLM, as described in Sections 1 and 3. All methodological uses of LLMs are disclosed in the paper. The authors reviewed and verified the research artifacts and take responsibility for the final content of this work.

ETHICS STATEMENT

In this work, we propose EVPO, an adaptive policy optimization algorithm that monitors batch-level explained variance to decide whether the learned critic should serve as the advantage baseline. By stabilizing RL post-training and reducing the variance of policy updates, EVPO lowers the compute and energy cost of fine-tuning large models with RL, making RL post-training more accessible to groups with limited resources. Our contribution is methodological and does not release any new model, dataset, or capability; it only affects how advantages are estimated from existing rollouts. Downstream models trained with EVPO may still inherit the general misuse risks of LLMs, but these risks stem from the underlying base model and reward design rather than from our optimizer, and we do not foresee any additional negative societal impacts specific to this work.

REPRODUCIBILITY STATEMENT

We provide the algorithmic details and theoretical derivations in Sections 3 and Appendix A, and report the full experimental setup and hyperparameters in Section 4 and Appendix B. An anonymized implementation is available at `https://anonymous.4open.science/r/EVPO-9785`.

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, February 2024. URL http://arxiv.org/abs/2402.14740.

Oron Anschel, Nir Baram, and Nahum Shimkin. Averaged-dqn: Variance reduction and stabilization for deep reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 176–185. PMLR, July 2017. URL https://proceedings.mlr.press/v70/anschel17a.html.

Cong Chen, Kaixiang Ji, Hao Zhong, Muzhi Zhu, Anzhou Li, Guo Gan, Ziyuan Huang, Cheng Zou, Jiajia Liu, Jingdong Chen, Hao Chen, and Chunhua Shen. Gui-shepherd: Reliable process reward and verification for long-sequence gui tasks, September 2025. URL http://arxiv.org/abs/2509.23738.

Bryan L. M. de Oliveira, Felipe V. Frujeri, Marcos P. C. M. Queiroz, Luana G. B. Martins, Telma W. de L. Soares, and Luckeciano C. Melo. Learning without critics? revisiting grpo in classical reinforcement learning environments, November 2025. URL http://arxiv.org/abs/2511.03527.

Paolo Dell'Aversana. The frozen lake problem. an example of optimization policy, December 2021. URL https://www.researchgate.net/publication/357420847_The_Frozen_Lake_Problem_An_example_of_optimization_policy.

Deming Ding, Shichun Liu, Enhui Yang, Jiahang Lin, Ziying Chen, Shihan Dou, Honglin Guo, Weiyu Cheng, Pengyu Zhao, Chengjun Xiao, Qunhong Zeng, Qi Zhang, Xuanjing Huang, Qidi Xu, and Tao Gui. Octobench: Benchmarking scaffold-aware instruction following in repository-grounded agentic coding, January 2026. URL http://arxiv.org/abs/2601.10343.

Guanting Dong, Hangyu Mao, Kai Ma, Licheng Bao, Yifei Chen, Zhongyuan Wang, Zhongxia Chen, Jiazhen Du, Huiyang Wang, Fuzheng Zhang, Guorui Zhou, Yutao Zhu, Ji-Rong Wen, and Zhicheng Dou. Agentic reinforced policy optimization. In *The Fourteenth International Conference on Learning Representations*, October 2025. URL https://openreview.net/forum?id=TX4k7BF6aO.

Shihan Dou, Ming Zhang, Zhangyue Yin, Chenhao Huang, Yujiong Shen, Junzhe Wang, Jiayi Chen, Yuchen Ni, Junjie Ye, Cheng Zhang, Huaibing Xie, Jianglu Hu, Shaolei Wang, Weichao Wang, Yanling Xiao, Yiting Liu, Zenan Xu, Zhen Guo, Pluto Zhou, Tao Gui, Zuxuan Wu, Xipeng Qiu, Qi Zhang, Xuanjing Huang, Yu-Gang Jiang, Di Wang, and Shunyu Yao. Cl-bench: A benchmark for context learning, February 2026. URL http://arxiv.org/abs/2602.03587.

Yannis Flet-Berliac and Philippe Preux. Only relevant information matters: Filtering out noisy samples to boost rl, April 2019. URL https://arxiv.org/abs/1904.04025v5.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645 (8081):633–638, September 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL https://www.nature.com/articles/s41586-025-09422-z.

Jian Hu, Jason Klein Liu, Haotian Xu, and Wei Shen. Reinforce++: Stabilizing critic-free policy optimization with global advantage normalization, November 2025. URL http://arxiv.org/abs/2501.03262.

Samyak Jain, Ayush Agrawal, and Navin Goyal. Towards understanding the optimization landscape of grpo and its variants. In *First Workshop on Foundations of Reasoning in Language Models*, December 2025. URL https://openreview.net/forum?id=DSX0b1Ds0j.

Andreas Junghanns and Jonathan Schaeffer. Sokoban: Enhancing general single-agent search methods using domain knowledge. *Artificial Intelligence*, 129(1):219–251, June 2001. ISSN 0004-3702. doi: 10.1016/S0004-3702(01)00109-6. URL https://www.sciencedirect.com/science/article/pii/S0004370201001096.

R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, March 1960. ISSN 0021-9223. doi: 10.1115/1.3662552. URL https://doi.org/10.1115/1.3662552.

Lijun Li, Xiaoyu Wang, Jianan Liu, Haoran Chen, Minghao Zhang, Zixuan Yang, and Haoran Zhang. Unifying ppo, dpo, and grpo: A theoretical and empirical study on llm post-training, November 2025. URL https://www.researchgate.net/publication/397211705_Unifying_PPO_DPO_and_GRPO_A_Theoretical_and_Empirical_Study_on_LLM_Post-Training.

Jiahang Lin, Kai Hu, Binghai Wang, Yuhao Zhou, Zhiheng Xi, Honglin Guo, Shichun Liu, Junzhe Wang, Shihan Dou, Enyu Zhou, Hang Yan, Zhenhua Han, Tao Gui, Qi Zhang, and Xuanjing Huang. Mm-doc-r1: Training agents for long document visual question answering through multi-turn reinforcement learning, April 2026. URL http://arxiv.org/abs/2604.13579.

Naemeh Mohammadpour, Meysam Fozi, Mohammad Mehdi Ebadzadeh, Ali Azimi, and Ali Kamali. Proximal policy optimization with adaptive generalized advantage estimate: Critic-aware refinements, 2024. URL https://github.com/naempr/PPO-with-adaptive-GAE.

OpenAI, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrew Duberstein, Andrew Kondrich, Andrey Mishchenko, Andy Applebaum, Angela Jiang, Ashvin Nair, Barret Zoph, Behrooz Ghorbani, Ben Rossen, Benjamin Sokolowsky, Boaz Barak, Bob McGrew, Borys Minaiev, Botao Hao, Bowen Baker, Brandon Houghton, Brandon McKinzie, Brydon Eastman, Camillo Lugaresi, Cary Bassin, Cary Hudson, Chak Ming Li, Charles de Bourcy, Chelsea Voss, Chen Shen, Chong Zhang, Chris Koch, Chris Orsinger, Christopher Hesse, Claudia Fischer, Clive Chan, Dan Roberts, Daniel Kappler, Daniel Levy, Daniel Selsam, David Dohan, David Farhi, David Mely, David Robinson, Dimitris Tsipras, Doug Li, Dragos Oprica, Eben Freeman, Eddie Zhang, Edmund Wong, Elizabeth Proehl, Enoch Cheung, Eric Mitchell, Eric Wallace, Erik Ritter, Evan Mays, Fan Wang, Felipe Petroski Such, Filippo Raso, Florencia Leoni, Foivos Tsimpourlas, Francis Song, Fred von Lohmann, Freddie Sulit, Geoff Salmon, Giambattista Parascandolo, Gildas Chabot, Grace Zhao, Greg Brockman, Guillaume Leclerc, Hadi Salman, Haiming Bao, Hao Sheng, Hart Andrin, Hessam Bagherinezhad, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian Osband, Ignasi Clavera Gilaberte, Ilge Akkaya, Ilya Kostrikov, Ilya Sutskever, Irina Kofman, Jakub Pachocki, James Lennon, Jason Wei, Jean Harb, Jerry Twore, Jiacheng Feng, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joaquin Quiñonero Candela, Joe Palermo, Joel Parish, Johannes Heidecke, John Hallman, John Rizzo, Jonathan Gordon, Jonathan Uesato, Jonathan Ward, Joost Huizinga, Julie Wang, Kai Chen, Kai Xiao, Karan Singhal, Karina Nguyen, Karl Cobbe, Katy Shi, Kayla Wood, Kendra Rimbach, Keren Gu-Lemberg, Kevin Liu, Kevin Lu, Kevin Stone, Kevin Yu, Lama Ahmad, Lauren Yang, Leo Liu, Leon Maksin, Leyton Ho, Liam Fedus, Lilian Weng, Linden Li, Lindsay McCallum, Lindsey Held, Lorenz Kuhn, Lukas Kondraciuk, Lukasz Kaiser, Luke Metz, Madelaine Boyd, Maja Trebacz, Manas Joglekar, Mark Chen, Marko Tintor, Mason Meyer, Matt Jones, Matt Kaufer, Max Schwarzer, Meghan Shah, Mehmet Yatbaz, Melody Y. Guan, Mengyuan Xu, Mengyuan Yan, Mia Glaese, Mianna Chen, Michael Lampe, Michael Malek, Michele Wang, Michelle Fradin, Mike McClay, Mikhail Pavlov, Miles Wang, Mingxuan Wang, Mira Murati, Mo Bavarian, Mostafa Rohaninejad, Nat McAleese, Neil Chowdhury, Neil Chowdhury, Nick Ryder, Nikolas Tezak, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Patrick Chao, Paul Ashbourne, Pavel Izmailov, Peter Zhokhov, Rachel Dias, Rahul Arora, Randall Lin, Rapha Gontijo Lopes, Raz Gaon, Reah Miyara, Reimar Leike, Renny Hwang, Rhythm Garg, Robin Brown, Roshan James, Rui Shu, Ryan Cheu, Ryan Greene, Saachi Jain, Sam Altman, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Santiago Hernandez, Sasha Baker, Scott McKinney, Scottie Yan, Shengjia Zhao, Shengli Hu, Shibani Santurkar, Shraman Ray Chaudhuri, Shuyuan Zhang, Siyuan Fu, Spencer Papay, Steph Lin, Suchir Balaji, Suvansh Sanjeev, Szymon Sidor, Tal Broda, Aidan Clark, Tao Wang, Taylor Gordon, Ted Sanders, Tejal Patwardhan, Thibault Sottiaux, Thomas Degry, Thomas Dimson, Tianhao Zheng, Timur Garipov, Tom Stasi, Trapit Bansal, Trevor Creech, Troy Peterson, Tyna Eloundou, Valerie Qi, Vineet Kosaraju, Vinnie Monaco, Vitchyr Pong, Vlad Fomenko, Weiyi Zheng, Wenda Zhou, Wes McCabe, Wojciech Zaremba, Yann Dubois, Yinghai Lu, Yining Chen, Young Cha, Yu Bai, Yuchen He, Yuchen Zhang, Yunyun Wang, Zheng Shao, and Zhuohan Li. Openai o1 system card, December 2024. URL http://arxiv.org/abs/2412.16720.

Yan Pei, Swarnendu Biswas, Donald S. Fussell, and Keshav Pingali. An elementary introduction to kalman filtering, June 2019. URL http://arxiv.org/abs/1710.04055.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi

Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, January 2025. URL http://arxiv.org/abs/2412.15115.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

John Schulman. The nuts and bolts of deep rl research, December 2016. URL http://joschu.net/docs/nuts-and-bolts.pdf.

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

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation, October 2018. URL http://arxiv.org/abs/1506.02438.

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, April 2024. URL http://arxiv.org/abs/2402.03300.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. Alfworld: Aligning text and embodied environments for interactive learning, March 2021. URL http://arxiv.org/abs/2010.03768.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, A. J. Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, Alex Makelov, Alex Neitz, Alex Wei, Alexandra Barr, Alexandre Kirchmeyer, Alexey Ivanov, Alexi Christakis, Alistair Gillespie, Allison Tam, Ally Bennett, Alvin Wan, Alyssa Huang, Amy McDonald Sandjideh, Amy Yang, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrei Gheorghe, Andres Garcia Garcia, Andrew Braunstein, Andrew Liu, Andrew Schmidt, Andrey Mereskin, Andrey Mishchenko, Andy Applebaum, Andy Rogerson, Ann Rajan, Annie Wei, Anoop Kotha, Anubha Srivastava, Anushree Agrawal, Arun Vijayvergiya, Ashley Tyra, Ashvin Nair, Avi Nayak, Ben Eggers, Bessie Ji, Beth Hoover, Bill Chen, Blair Chen, Boaz Barak, Borys Minaiev, Botao Hao, Bowen Baker, Brad Lightcap, Brandon McKinzie, Brandon Wang, Brendan Quinn, Brian Fioca, Brian Hsu, Brian Yang, Brian Yu, Brian Zhang, Brittany Brenner, Callie Riggins Zetino, Cameron Raymond, Camillo Lugaresi, Carolina Paz, Cary Hudson, Cedric Whitney, Chak Li, Charles Chen, Charlotte Cole, Chelsea Voss, Chen Ding, Chen Shen, Chengdu Huang, Chris Colby, Chris Hallacy, Chris Koch, Chris Lu, Christina Kaplan, Christina Kim, C. J. Minott-Henriques, Cliff Frey, Cody Yu, Coley Czarnecki, Colin Reid, Colin Wei, Cory Decareaux, Cristina Scheau, Cyril Zhang, Cyrus Forbes, Da Tang, Dakota Goldberg, Dan Roberts, Dana Palmie, Daniel Kappler, Daniel Levine, Daniel Wright, Dave Leo, David Lin, David Robinson, Declan Grabb, Derek Chen, Derek Lim, Derek Salama, Dibya Bhattacharjee, Dimitris Tsipras, Dinghua Li, Dingli Yu, D. J. Strouse, Drew Williams, Dylan Hunn, Ed Bayes, Edwin Arbus, Ekin Akyurek, Elaine Ya Le, Elana Widmann, Eli Yani, Elizabeth Proehl, Enis Sert, Enoch Cheung, Eri Schwartz, Eric Han, Eric Jiang, Eric Mitchell, Eric Sigler, Eric Wallace, Erik Ritter, Erin Kavanaugh, Evan Mays, Evgenii Nikishin, Fangyuan Li, Felipe Petroski Such, Filipe de Avila Belbute Peres, Filippo Raso, Florent Bekerman, Foivos Tsimpourlas, Fotis Chantzis, Francis Song, Francis Zhang, Gaby Raila, Garrett McGrath, Gary Briggs, Gary Yang, Giambattista Parascandolo, Gildas Chabot, Grace Kim, Grace Zhao, Gregory Valiant, Guillaume Leclerc, Hadi Salman, Hanson Wang, Hao Sheng, Haoming Jiang, Haoyu Wang, Haozhun Jin, Harshit Sikchi, Heather Schmidt, Henry Aspegren, Honglin Chen, Huida Qiu, Hunter Lightman, Ian Covert, Ian Kivlichan, Ian Silber, Ian Sohl, Ibrahim Hammoud, Ignasi Clavera, Ikai Lan, Ilge Akkaya, Ilya Kostrikov, Irina Kofman, Isak Etinger, Ishaan Singal, Jackie Hehir, Jacob Huh, Jacqueline Pan, Jake Wilczynski, Jakub Pachocki, James Lee, James Quinn, Jamie Kiros, Janvi Kalra, Jasmyn Samaroo, Jason Wang, Jason Wolfe, Jay Chen, Jay Wang, Jean Harb, Jeffrey Han, Jeffrey Wang, Jennifer Zhao, Jeremy Chen, Jerene Yang, Jerry Tworek, Jesse Chand, Jessica Landon, Jessica Liang, Ji Lin, Jiancheng Liu, Jianfeng Wang, Jie Tang, Jihan Yin, Joanne Jang, Joel Morris, Joey Flynn, Johannes Ferstad, Johannes Heidecke, John Fishbein, John Hallman, Jonah Grant, Jonathan Chien, Jonathan

Gordon, Jongsoo Park, Jordan Liss, Jos Kraaijeveld, Joseph Guay, Joseph Mo, Josh Lawson, Josh McGrath, Joshua Vendrow, Joy Jiao, Julian Lee, Julie Steele, Julie Wang, Junhua Mao, Kai Chen, Kai Hayashi, Kai Xiao, Kamyar Salahi, Kan Wu, Karan Sekhri, Karan Sharma, Karan Singhal, Karen Li, Kenny Nguyen, Keren Gu-Lemberg, Kevin King, Kevin Liu, Kevin Stone, Kevin Yu, Kristen Ying, Kristian Georgiev, Kristie Lim, Kushal Tirumala, Kyle Miller, Lama Ahmad, Larry Lv, Laura Clare, Laurance Fauconnet, Lauren Itow, Lauren Yang, Laurentia Romaniuk, Leah Anise, Lee Byron, Leher Pathak, Leon Maksin, Leyan Lo, Leyton Ho, Li Jing, Liang Wu, Liang Xiong, Lien Mamitsuka, Lin Yang, Lindsay McCallum, Lindsey Held, Liz Bourgeois, Logan Engstrom, Lorenz Kuhn, Louis Feuvrier, Lu Zhang, Lucas Switzer, Lukas Kondraciuk, Lukasz Kaiser, Manas Joglekar, Mandeep Singh, Mandip Shah, Manuka Stratta, Marcus Williams, Mark Chen, Mark Sun, Marselus Cayton, Martin Li, Marvin Zhang, Marwan Aljubeh, Matt Nichols, Matthew Haines, Max Schwarzer, Mayank Gupta, Meghan Shah, Melody Huang, Meng Dong, Mengqing Wang, Mia Glaese, Micah Carroll, Michael Lampe, Michael Malek, Michael Sharman, Michael Zhang, Michele Wang, Michelle Pokrass, Mihai Florian, Mikhail Pavlov, Miles Wang, Ming Chen, Mingxuan Wang, Minnia Feng, Mo Bavarian, Molly Lin, Moose Abdool, Mostafa Rohaninejad, Nacho Soto, Natalie Staudacher, Natan LaFontaine, Nathan Marwell, Nelson Liu, Nick Preston, Nick Turley, Nicklas Ansman, Nicole Blades, Nikil Pancha, Nikita Mikhaylin, Niko Felix, Nikunj Handa, Nishant Rai, Nitish Keskar, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Oona Gleeson, Pamela Mishkin, Patryk Lesiewicz, Paul Baltescu, Pavel Belov, Peter Zhokhov, Philip Pronin, Phillip Guo, Phoebe Thacker, Qi Liu, Qiming Yuan, Qinghua Liu, Rachel Dias, Rachel Puckett, Rahul Arora, Ravi Teja Mullapudi, Raz Gaon, Reah Miyara, Rennie Song, Rishabh Aggarwal, R. J. Marsan, Robel Yemiru, Robert Xiong, Rohan Kshirsagar, Rohan Nuttall, Roman Tsiupa, Ronen Eldan, Rose Wang, Roshan James, Roy Ziv, Rui Shu, Ruslan Nigmatullin, Saachi Jain, Saam Talaie, Sam Altman, Sam Arnesen, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Sarah Yoo, Savannah Heon, Scott Ethersmith, Sean Grove, Sean Taylor, Sebastien Bubeck, Sever Banesiu, Shaokyi Amdo, Shengjia Zhao, Sherwin Wu, Shibani Santurkar, Shiyu Zhao, Shraman Ray Chaudhuri, Shreyas Krishnaswamy, Shuaiqi, Xia, Shuyang Cheng, Shyamal Anadkat, Simón Posada Fishman, Simon Tobin, Siyuan Fu, Somay Jain, Song Mei, Sonya Egoian, Spencer Kim, Spug Golden, S. Q. Mah, Steph Lin, Stephen Imm, Steve Sharpe, Steve Yadlowsky, Sulman Choudhry, Sungwon Eum, Suvansh Sanjeev, Tabarak Khan, Tal Stramer, Tao Wang, Tao Xin, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Degry, Thomas Shadwell, Tianfu Fu, Tianshi Gao, Timur Garipov, Tina Sriskandarajah, Toki Sherbakov, Tomer Kaftan, Tomo Hiratsuka, Tongzhou Wang, Tony Song, Tony Zhao, Troy Peterson, Val Kharitonov, Victoria Chernova, Vineet Kosaraju, Vishal Kuo, Vitchyr Pong, Vivek Verma, Vlad Petrov, Wanning Jiang, Weixing Zhang, Wenda Zhou, Wenlei Xie, Wenting Zhan, Wes McCabe, Will DePue, Will Ellsworth, Wulfie Bain, Wyatt Thompson, Xiangning Chen, Xiangyu Qi, Xin Xiang, Xinwei Shi, Yann Dubois, Yaodong Yu, Yara Khakbaz, Yifan Wu, Yilei Qian, Yin Tat Lee, Yinbo Chen, Yizhen Zhang, Yizhong Xiong, Yonglong Tian, Young Cha, Yu Bai, Yu Yang, Yuan Yuan, Yuanzhi Li, Yufeng Zhang, Yuguang Yang, Yujia Jin, Yun Jiang, Yunyun Wang, Yushi Wang, Yutian Liu, Zach Stubenvoll, Zehao Dou, Zheng Wu, and Zhigang Wang. Openai gpt-5 system card, December 2025. URL http://arxiv.org/abs/2601.03267.

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems*, volume 12. MIT Press, 1999. URL https://papers.nips.cc/paper_files/paper/1999/hash/464d828b85b0bed98e80ade0a5c43b0f-Abstract.html.

Luong Trung, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. Reft: Reasoning with reinforced fine-tuning. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7601–7614, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.410. URL https://aclanthology.org/2024.acl-long.410/.

Junzhe Wang, Zhiheng Xi, yajie yang, Hao Luo, Shihan Dou, Tao Gui, and Qi Zhang. Enhancing llm-based search agents via contribution weighted group relative policy optimization, April 2026. URL http://arxiv.org/abs/2604.14267.

Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, Eli Gottlieb, Yiping Lu, Kyunghyun Cho, Jiajun Wu,

Li Fei-Fei, Lijuan Wang, Yejin Choi, and Manling Li. Ragen: Understanding self-evolution in llm agents via multi-turn reinforcement learning, May 2025. URL http://arxiv.org/abs/2504.20073.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3):229–256, May 1992. ISSN 1573-0565. doi: 10.1007/BF00992696. URL https://link.springer.com/article/10.1007/BF00992696.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, May 2025. URL http://arxiv.org/abs/2505.09388.

Shunyu Yao, Howard Chen, John Yang, and Karthik R. Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. In *Advances in Neural Information Processing Systems*, October 2022. URL https://openreview.net/forum?id=R9KnuFlvnU.

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

Rui Zheng, Shihan Dou, Songyang Gao, Yuan Hua, Wei Shen, Binghai Wang, Yan Liu, Senjie Jin, Qin Liu, Yuhao Zhou, Limao Xiong, Lu Chen, Zhiheng Xi, Nuo Xu, Wenbin Lai, Minghao Zhu, Cheng Chang, Zhangyue Yin, Rongxiang Weng, Wensen Cheng, Haoran Huang, Tianxiang Sun, Hang Yan, Tao Gui, Qi Zhang, Xipeng Qiu, and Xuanjing Huang. Secrets of rlhf in large language models part i: Ppo, July 2023. URL http://arxiv.org/abs/2307.04964.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models, November 2023. URL http://arxiv.org/abs/2311.07911.

A Extended Proofs and Derivations

Detailed proofs and derivations are provided in this appendix.

A.1 Detailed Kalman Filtering Analogy

We provide here the full state estimation analogy and derivation that motivates the EV-based switching criterion introduced in Section 2.3.

State estimation analogy. We cast baseline construction as a scalar state estimation problem. The hidden state to be estimated is the true value function $x = V^\pi(s)$. For each sample in a training batch, the observed return decomposes as

$$G = V^\pi(s) + \epsilon, \quad \epsilon \sim (0, R), \quad R \triangleq \mathbb{E}[\mathrm{Var}(G \mid s)], \tag{10}$$

where R is the irreducible *sampling noise* arising from the stochasticity of future actions under π. Two competing estimators of $V^\pi(s)$ are available at each training step:

- **Model-driven prediction (Critic).** $\hat{x}_A = \hat{V}_\phi(s) = V^\pi(s) + \delta(s)$, where $\delta(s)$ denotes the critic's estimation error with variance $P_A \triangleq \mathrm{Var}(\delta)$. This parallels the *state prediction* in a Kalman filter: it exploits learned state features but carries modelling noise whose magnitude depends on the critic's current training quality.
- **Evidence-driven observation (Batch mean).** $\hat{x}_B = \bar{G}_\mathcal{B}$, the empirical mean return over the batch, which approximates $\mathbb{E}[V^\pi(s)]$. Its error relative to any specific state s has variance $P_B \triangleq \mathrm{Var}_s(V^\pi(s))$, paralleling the *observation noise* in a Kalman filter: it reflects how much true state values vary within the batch, and is large when the batch covers heterogeneous states.

The sampling noise R affects both estimators symmetrically and controls the *sensitivity* of the EV diagnostic without altering its *sign*.

Optimal fusion and the Kalman gain. The Kalman filter combines the two estimators via a linear combination $b(s) = (1-K)\,\hat{x}_A + K\,\hat{x}_B$ that minimizes the mean squared error $\mathbb{E}[(b(s) - V^\pi(s))^2]$. The optimal *Kalman gain* is:

$$K^* = \frac{P_A}{P_A + P_B}. \tag{11}$$

When $K^* < \frac{1}{2}$ (i.e., $P_A < P_B$), the critic is more reliable than the batch mean and should receive higher weight; when $K^* \geq \frac{1}{2}$ (i.e., $P_A \geq P_B$), the critic's estimation noise dominates the state signal it provides, and the baseline should retreat toward the batch mean.

The Cont. K method. Since both P_A and P_B in the K^* formula depend on the unknown $V^\pi(s)$ and are therefore unobservable, we replace them with samplable surrogates $\tilde{P}_A = \mathrm{Var}(\hat{V}_\phi(s) - G)$ and $\tilde{P}_B = \mathrm{Var}(G)$, yielding the estimate:

$$\tilde{K} = \frac{\tilde{P}_A}{\tilde{P}_A + \tilde{P}_B} = \frac{\mathrm{Var}(\hat{V}_\phi(s) - V^\pi(s) - \epsilon)}{\mathrm{Var}(V^\pi(s) + \epsilon) + \mathrm{Var}(\hat{V}_\phi(s) - V^\pi(s) - \epsilon)} = \frac{P_A + R}{P_A + P_B + 2R}. \tag{12}$$

Using $\tilde{K}$ as the linear combination coefficient at each step, i.e., $b(s) = (1-\tilde{K})\,\hat{V}_\phi(s) + \tilde{K}\,\bar{G}_\mathcal{B}$, constitutes the Cont. K method. Because the return variance R enters both the numerator and denominator of $\tilde{K}$, the estimation target shifts from $\frac{P_A}{P_A+P_B}$ to $\frac{P_A+R}{P_A+P_B+2R}$. While $\tilde{K}$ still lies in $[0,1]$, it is systematically pulled toward $\frac{1}{2}$.

Connection to explained variance. Substituting the variance decompositions from Eq. (10) into the definition of EV yields:

$$\mathrm{ev} = \frac{P_B - P_A}{P_B + R}. \tag{13}$$

Since the denominator is always positive, the sign of EV is determined solely by $P_B - P_A$:

- $\mathrm{ev} > 0 \iff P_A < P_B \iff K^* < \frac{1}{2}$: the critic outperforms the batch mean, corresponding to the regime where PPO's state-dependent baseline is advantageous;
- $\mathrm{ev} \leq 0 \iff P_A \geq P_B \iff K^* \geq \frac{1}{2}$: the critic's noise exceeds its signal, corresponding to the regime where GRPO's constant baseline is preferable.

A.2 GAE Derivation under RLVR Conditions

Proximal Policy Optimization (Schulman et al., 2017) estimates advantages via Generalized Advantage Estimation (GAE) (Schulman et al., 2018):

$$\hat{A}_t^{\mathrm{GAE}} = \sum_{l=0}^{T-1-t} (\gamma\lambda)^l \delta_{t+l}, \quad \delta_t = r_t + \gamma\hat{V}_\phi(s_{t+1}) - \hat{V}_\phi(s_t), \tag{14}$$

where $\hat{V}_\phi$ is a learned critic and $\lambda \in [0,1]$ controls the bias–variance trade-off. The policy is updated by maximizing a clipped surrogate objective that constrains the ratio $\pi_\theta(a_t|s_t)/\pi_{\theta_{\text{old}}}(a_t|s_t)$.

Under RLVR conditions ($\gamma = \lambda = 1$, terminal-only reward $r_T = \mathcal{R}(x, y)$, $r_t = 0$ for $t < T$), GAE reduces to:

$$\hat{A}_t^{\text{PPO}} = \sum_{l=t}^{T-1} \big(r_l + \hat{V}_\phi(s_{l+1}) - \hat{V}_\phi(s_l)\big) = G - \hat{V}_\phi(s_t), \tag{15}$$

where the telescoping sum cancels all intermediate critic terms (with the terminal state value defined as 0), leaving the advantage as the deviation of the observed return from the critic's state-dependent prediction.

B TASKS

We evaluate on four tasks spanning multi-turn grid-world control, interactive web agents, and single-turn mathematical reasoning. All training code is based on the `RAGEN2` framework (Wang et al., 2025), which wraps `verl` for PPO-style LLM post-training and provides multi-turn agent environments.

Sokoban. We use the `CoordSokoban` variant of the RAGEN2 Sokoban environment (Junghanns & Schaeffer, 2001). Each episode instantiates a 6×6 grid with one box and one target, procedurally generated from an environment seed. The observation at each step is a symbol grid together with zero-indexed coordinates of the player, the box, and the target. The action space is $\{\text{UP}, \text{DOWN}, \text{LEFT}, \text{RIGHT}\}$; the policy is prompted to emit an action sequence of length up to two per turn, separated by "`||`" and enclosed in `<answer>...</answer>` tags. A trajectory consists of up to 10 executed actions across up to 5 turns. The reward is terminal and binary: $+1$ if all boxes end on targets, 0 otherwise, with an additional format penalty of -0.1 for malformed outputs.

FrozenLake. We use the `CoordFrozenLake` variant wrapping the Gymnasium `FrozenLake-v1` environment (Dell'Aversana, 2021). The map is a 4×4 grid with randomly placed holes, the start tile, and the goal tile; the map layout is resampled per environment seed. Transitions are stochastic (`is_slippery=True`): the commanded action succeeds with probability $1/3$, and with probability $2/3$ the agent slips perpendicular to the intended direction. The observation is again a symbol grid plus coordinates of start, goal, player, and all holes. The action space is $\{\text{LEFT}, \text{DOWN}, \text{RIGHT}, \text{UP}\}$, with up to 10 actions per trajectory across up to 5 turns. The reward is terminal and binary: $+1$ if the player reaches the goal, 0 if the player falls into a hole or exhausts the action budget; a format penalty of -0.1 is applied for malformed outputs.

WebShop. WebShop (Yao et al., 2022) is an interactive web-navigation task in which the agent must locate and purchase a product matching a natural-language instruction. We use the `webshop-minimal` fork with the *small* product catalogue shipped with RAGEN2. At each turn the agent observes the current page (search box, product list, or product detail) together with the list of clickable elements, and produces exactly one action drawn from `search[`*`keywords`*`]` or `click[`*`element`*`]`, again wrapped in `<answer>...</answer>`. A trajectory consists of up to 9 actions across up to 9 turns. The reward is the built-in WebShop matching score in $[0, 1]$, delivered only at the terminal `click[buy now]`.

MATH. We use DAPO-Math-17k (Yu et al., 2025) (17,267 competition-level problems) for *both* training and validation: the full training set is used for RL updates, and a *fixed subset of 32 problems*, held out from training, is used for evaluation. Each problem is a single-turn reasoning task (`max_turn=1`): the policy is prompted with the problem statement and must emit a `<think>...</think>` block followed by the final numerical answer inside `<answer>...</answer>`. The reward is terminal and binary: $+1$ if the final answer matches the ground truth (after normalization), 0 otherwise.

C BASELINES AND HYPERPARAMETERS

We compare EVPO against a Base-LLM reference and four RL baselines that span both critic-based and critic-free families.

- **Base LLM.** Instruction-tuned Qwen2.5 without any RL post-training, prompted zero-shot with the same task template as the RL runs.
- **PPO** (Schulman et al., 2017). Standard clipped-surrogate PPO with GAE advantages. Under our RLVR setting (γ=λ=1, terminal-only reward) this reduces to the PPO advantage of Eq. (1) with a learned, state-dependent critic.
- **StarPO-S** (Wang et al., 2025). A multi-turn PPO variant introduced by RAGEN that filters the lowest-variance rollouts before each update (`rollout_filter_ratio`=0.25, `rollout_filter_metric`=reward_variance) to stabilize long-horizon agentic training.
- **GRPO** (Shao et al., 2024). Critic-free; the baseline is the response-level batch mean, normalized by the batch standard deviation.
- **DAPO** (Yu et al., 2025). A modern critic-free variant built on GRPO with (i) dynamic sampling that discards groups whose rewards are all identical and resamples up to `max_num_gen_batches`=5 times until enough informative groups are collected; (ii) *clip-higher* asymmetry $(\varepsilon_{\text{low}}, \varepsilon_{\text{high}}) = (0.2, 0.28)$; (iii) token-level loss aggregation (`loss_agg_mode`=token-mean); and (iv) no std normalization of advantages and no in-reward KL term.

All five methods (PPO, StarPO-S, GRPO, DAPO, EVPO) share the same rollout stack, tokenizer, environment code and 8×H100 GPU server, and differ only in the advantage estimator and the listed hyperparameters. EVPO itself uses PPO-style training (critic and actor with clipped surrogate) and only changes the baseline subtracted from the return, as prescribed by the EV-gating rule (Eq. (8)).

Table 2 lists the hyperparameters that differ across methods or tasks. Learning rates are tuned per task on the Base-LLM vs. PPO gap on a single seed and then *kept identical across methods on the same task* for fair comparison, except where a differing value is itself a defining feature of the algorithm (see the caption of Table 2).

Table 2: Per-task / per-method hyperparameter overrides. $\varepsilon_{\text{lo}}/\varepsilon_{\text{hi}}$ are the PPO-style clip thresholds. Where DAPO and StarPO-S show different values from the other methods on the same task (e.g., DAPO's clip-higher asymmetry, or the StarPO-S Sokoban configuration that preserves the original RAGEN recipe), those settings are defining features of the respective algorithms rather than tuning choices, and we therefore do not force them to match the other methods.

Task	Method	lr_{actor}	$\text{lr}_{\text{critic}}$	$\varepsilon_{\text{lo}}/\varepsilon_{\text{hi}}$	KL coef.	Steps
Sokoban (6×6)	PPO	4e−6	1e−5	0.20/0.28	0	400
	StarPO-S	1e−6	1e−5	0.20/0.28	0	400
	GRPO	4e−6	–	0.20/0.28	0	400
	DAPO	4e−6	–	0.20/0.28	0	400
	EVPO	4e−6	1e−5	0.20/0.28	0	400
FrozenLake (4×4)	PPO	1e−6	1e−5	0.20/0.28	0	200
	StarPO-S	1e−6	1e−5	0.20/0.28	0	200
	GRPO	1e−6	–	0.20/0.28	0	200
	DAPO	1e−6	–	0.20/0.28	0	200
	EVPO	1e−6	1e−5	0.20/0.28	0	200
WebShop	PPO	5e−7	5e−6	0.20/0.20	10^{-2}	100
	StarPO-S	1e−6	1e−5	0.20/0.28	0	100
	GRPO	5e−7	–	0.20/0.20	10^{-2}	100
	DAPO	5e−7	–	0.20/0.28	10^{-2}	100
	EVPO	5e−7	5e−6	0.20/0.20	10^{-2}	100
MATH	PPO	2e−6	1e−5	0.20/0.20	0	200
	StarPO-S	2e−6	1e−5	0.20/0.28	0	200
	GRPO	2e−6	–	0.20/0.20	0	200
	DAPO	2e−6	–	0.20/0.28	0	200
	EVPO	2e−6	1e−5	0.20/0.20	0	200

D Details of the Preliminary Experiments

This section gives the exact protocols for the two preliminary studies that motivate EVPO in Section 2.

D.1 PPO on Sokoban: EV, Gradient Norm, Success Rate (Figure 2)

We train a PPO agent on Sokoban using the same setting as the Sokoban PPO main experiment (Sokoban PPO row of Table 2). At every step we record, over the same batch of $M = 16 \times 8 = 128$ rollouts used for the update: (i) the batch-level EV $\widehat{\mathrm{ev}}_{\mathcal{B}}$ defined in Eq. (6); (ii) the pre-clipping gradient norm of the actor; (iii) the training success rate (fraction of rollouts with terminal reward $= 1$). Shaded regions in Figure 2(a–c) highlight the first ≈ 150 steps where $\widehat{\mathrm{ev}}_{\mathcal{B}}$ remains near or below 0.

Noise-injection experiment (Figure 2d). We first train a PPO agent on FrozenLake to convergence under the same hyperparameters as the FrozenLake row of Table 2 for 200 steps, then continue training with i.i.d. Gaussian noise added to every critic output: $\hat{V}_\phi(s) \mapsto \hat{V}_\phi(s) + \xi$ with $\xi \sim \mathcal{N}(0, \sigma_\delta^2)$. We sweep $\sigma_\delta \in \{0\,, 0.3\,, 0.5\,, 1\,, 10\}$, corresponding to noise magnitudes from well below to well above the mean absolute state value (≈ 0.5). We report the smoothed validation success rate; Fig. 2d shows monotone degradation with complete collapse at σ_δ=10, precisely the regime where $\widehat{\mathrm{ev}}_{\mathcal{B}}$ is driven far below zero.

D.2 Cold-start and Critic Warmup (Figure 3)

We run two interventions on top of the Sokoban and FrozenLake PPO setups.

Cold-start. During the first K_{cold} training steps we replace every critic output in the advantage computation by the batch mean, $\hat{A}_t^{\text{cold-start}} = G - \bar{G}_{\mathcal{B}}$; after step K_{cold} we resume standard PPO with the original critic. We set $K_{\text{cold}} = 50$ on Sokoban and $K_{\text{cold}} = 25$ on FrozenLake, chosen to cover the EV-negative phase of Fig. 2a. The critic itself is always trained with its usual MSE loss so that by step K_{cold} it has had time to warm up.

Critic warmup. Before the actor receives any gradient, we train the critic alone for $K_{\text{warm}} = 200$ steps using the same rollouts as full PPO but with $\nabla_\theta \mathcal{L}_{\text{actor}}$ set to zero. After step K_{warm} we switch on actor updates and continue with standard PPO.

In both interventions all other hyperparameters match the PPO rows of Table 2. Figure 3 compares each intervention against the unmodified PPO baseline under the same seed and hardware; the consistent improvements confirm that it is specifically the suppression of critic-driven variance (rather than any secondary effect of the interventions) that benefits the policy.

E Additional Experiments

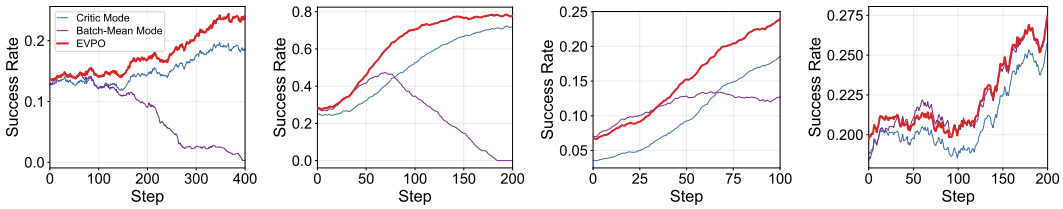
To further validate the effectiveness of EVPO, we present additional ablation and robustness analyses.

E.1 Gating Mechanism Ablation

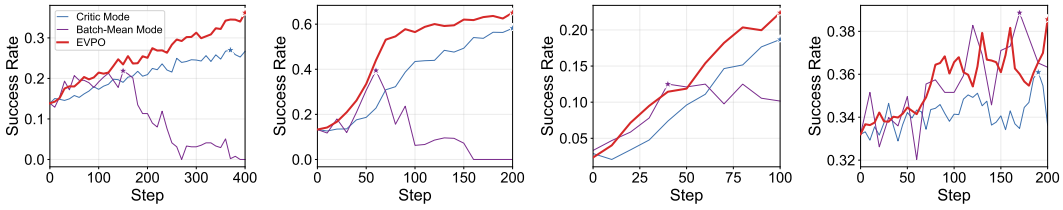
Although EVPO is a simple algorithm, the batch-mean mode it gates into differs subtly from the GRPO baseline: it uses the unnormalized batch mean $\bar{G}_{\mathcal{B}}$ rather than GRPO's std-normalized advantages, and the critic continues to be trained even when its output is not used for the actor. To isolate the contribution of the gating mechanism itself, we compare EVPO against its two constituent modes run in isolation across all four tasks:

- **Critic Mode**: the actor always uses the critic baseline $b(s_t) = \hat{V}_\phi(s_t)$, equivalent to standard PPO under our RLVR setting.
- **Batch-Mean Mode**: the actor always uses $b(s_t) = \bar{G}_{\mathcal{B}}$.

As shown in Figure 6, neither fixed mode dominates throughout training. Batch-Mean Mode provides faster initial progress (when the critic is still unreliable), while Critic Mode eventually surpasses



(a) Sokoban (Train) (b) FrozenLake (Train) (c) WebShop (Train) (d) MATH (Train)



(e) Sokoban (Val) (f) FrozenLake (Val) (g) WebShop (Val) (h) MATH (Val)

Figure 6: Gating mechanism ablation. Top row: training success rate; bottom row: validation success rate. EVPO inherits the fast early growth of Batch-Mean Mode and the sustained later improvement of Critic Mode.

it once the critic matures. EVPO inherits the best of both: it matches Batch-Mean Mode's early acceleration and transitions to Critic Mode's sustained improvement, confirming that the EV-gating mechanism successfully identifies and exploits the crossover point between the two regimes.

E.2 Robustness to Group Size

To test whether EVPO remains effective under different batch configurations, we reduce the training group size from $M{=}16$ to $M{=}8$ (i.e., each prompt is sampled 8 times per step instead of 16) and re-run PPO, GRPO, and EVPO on Sokoban and FrozenLake. All other hyperparameters are kept identical to the main experiments (Table 2).

As shown in Figure 7, EVPO retains its advantage over PPO and GRPO even with halved group size, confirming that the binary EV diagnostic remains a reliable switching signal under noisier variance estimates.

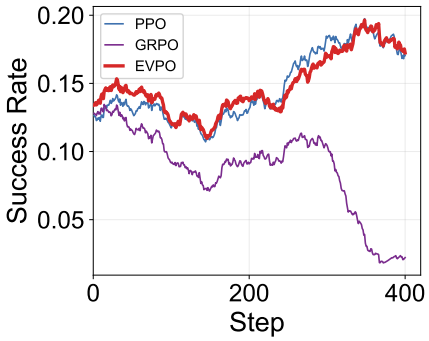
Furthermore, we compare the gating frequency (proportion of steps using the critic) between $M{=}8$ and $M{=}16$ in Figure 8. Under the smaller group size, the critic-usage rate decreases consistently across tasks. This is expected from our theoretical framework: with fewer samples per group, the batch-level EV estimate $\widehat{\mathrm{ev}}_{\mathcal{B}}$ has higher variance and is more likely to fall below zero, causing the gate to select the batch-mean mode more often. Notably, the anomalous gating-frequency pattern previously observed on FrozenLake under $M{=}16$ also disappears at $M{=}8$, suggesting that it was an artifact of the larger batch rather than a failure of the gating mechanism. The fact that EVPO adapts its gating frequency in a theoretically predictable manner—rather than breaking down—provides additional evidence that the EV-based switching criterion is well-calibrated.

E.3 Continual K Baseline

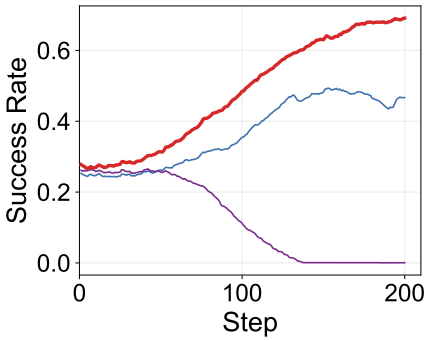
The Cont. K baseline continuously interpolates between the critic and the batch-mean baseline instead of making EVPO's binary decision. At each training step, it estimates the Kalman gain from the observed return and critic output:

$$\tilde{K} = \frac{\mathrm{Var}(\hat{V}_\phi(s) - G)}{\mathrm{Var}(G) + \mathrm{Var}(\hat{V}_\phi(s) - G)}, \qquad b_{\text{Cont. }K}(s) = (1 - \tilde{K})\hat{V}_\phi(s) + \tilde{K}\bar{G}_{\mathcal{B}}. \tag{16}$$

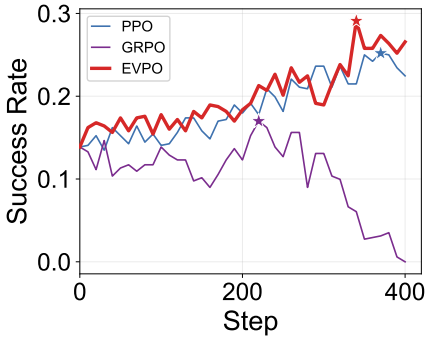
This estimator is computable from a single batch, but the return noise appears in both variance terms and pulls $\tilde{K}$ toward $\frac{1}{2}$, as derived in Appendix A.1.



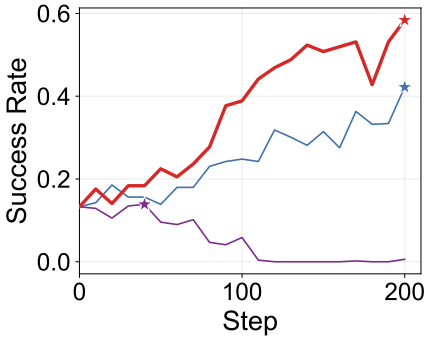
(a) Sokoban (Train)



(b) FrozenLake (Train)

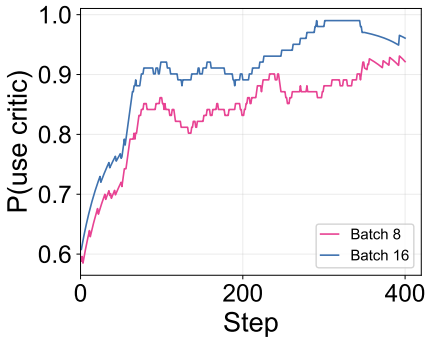


(c) Sokoban (Val)

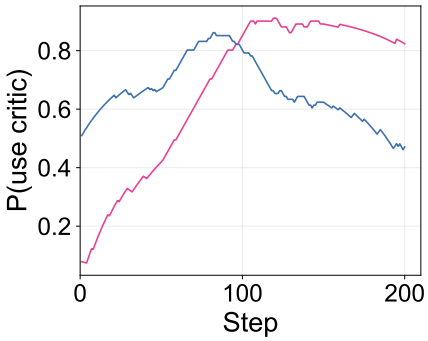


(d) FrozenLake (Val)

Figure 7: Robustness to group size (M=8). Top row: training success rate; bottom row: validation success rate. EVPO consistently outperforms both PPO and GRPO under the reduced group size, demonstrating that the EV-gating mechanism does not rely on large-batch variance estimates to function effectively.



(a) Sokoban



(b) FrozenLake

Figure 8: Gating frequency comparison between group sizes. Batch 8 vs. Batch 16. Under the smaller group size, EVPO gates into critic mode less frequently, consistent with the higher estimation noise in $\widehat{\mathrm{ev}}_{\mathcal{B}}$ when fewer samples are available.

We evaluate Cont. K on the same four tasks and with the same evaluation protocol as the main experiments. We report the existing single-run Cont. K reference evaluations together with the avg5 EVPO results from Table 1. Cont. K was not included in the avg5 run aggregation used for Figure 4.

Table 3: Validation success rate of Cont. K and EVPO across the four tasks. Cont. K values are existing single-run references, whereas EVPO values are the peaks of the avg5 mean validation curves reported in Table 1. **Bold** indicates the higher value in each column.

Method	Sokoban	FrozenLake	WebShop	MATH
Cont. K	0.244	0.490	**0.242**	0.371
EVPO (ours)	**0.362**	**0.651**	0.224	**0.386**

Compared with the avg5 EVPO results, the existing Cont. K reference is lower on Sokoban, FrozenLake, and MATH, while its WebShop value is slightly higher. Since Cont. K was not evaluated with the avg5 protocol, this comparison is descriptive rather than a claim of statistical superiority. Nevertheless, continuous interpolation does not consistently match the binary choices used by EVPO, where the sign of batch EV determines which extreme to select.

E.4 EMA Smoothing of the Gating Signal

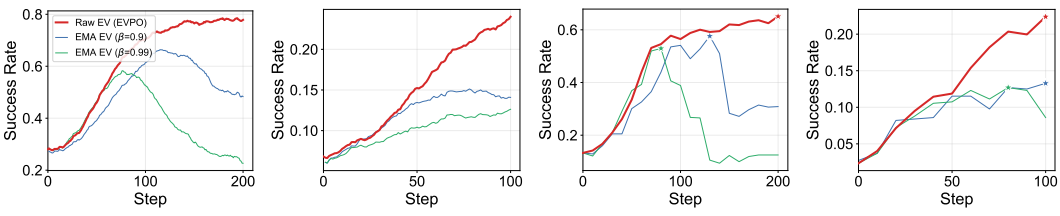
A natural alternative to the raw per-batch EV used by EVPO is to smooth the gating signal over time. Given the batch estimate $\widehat{\mathrm{ev}}_{\mathcal{B},t}$ at step t, we consider the exponential moving average

$$m_t = \beta m_{t-1} + (1-\beta)\widehat{\mathrm{ev}}_{\mathcal{B},t}, \qquad m_0 = \widehat{\mathrm{ev}}_{\mathcal{B},0}, \tag{17}$$

and apply the same binary gate as EVPO using the sign of m_t instead of the current per-batch estimate. We evaluate $\beta \in \{0.9, 0.99\}$ on FrozenLake and WebShop while keeping the remaining task-specific settings unchanged.

Table 4: Best validation success rate for raw and EMA-smoothed EV. Raw-EV entries are avg5 mean-curve peaks from Table 1; EMA entries are controlled single runs. **Bold** indicates the best result per task.

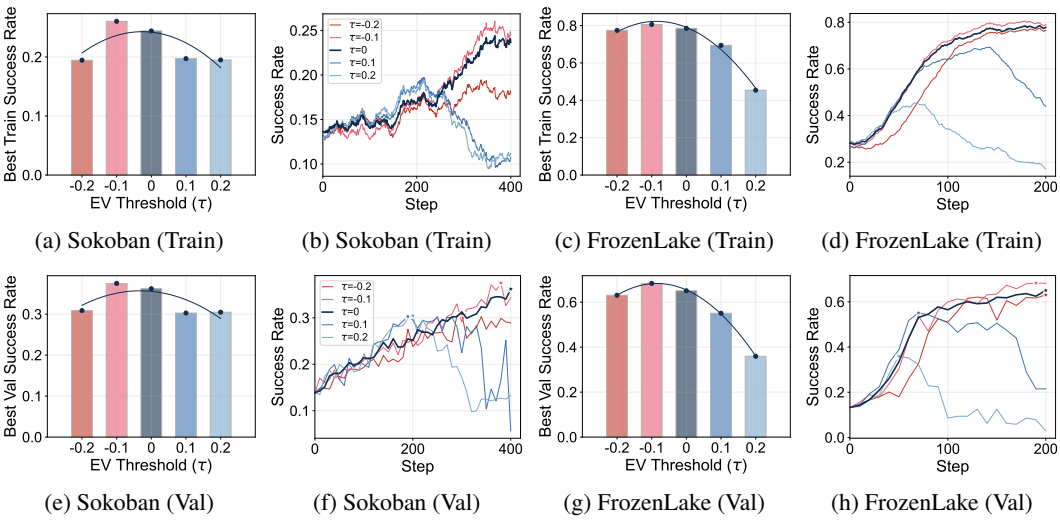
Gating signal	FrozenLake	WebShop
Raw per-batch (EVPO, avg5)	**0.651**	**0.224**
EMA ($\beta = 0.9$)	0.576	0.133
EMA ($\beta = 0.99$)	0.529	0.127



(a) FrozenLake (Train) (b) WebShop (Train) (c) FrozenLake (Val) (d) WebShop (Val)

Figure 9: EMA smoothing ablation on FrozenLake and WebShop. Left two: training; right two: validation. Training uses the Figure 4 smoothing window, while validation is unsmoothed.

Raw per-batch EV performs best on both tasks (Table 4). Increasing β from 0.9 to 0.99 degrades performance further, and the EMA curves also decline late in training (Figure 9). This suggests that smoothing delays responses to changing critic quality and preserves stale gating decisions; we therefore retain per-batch EV in EVPO.



(a) Sokoban (Train) (b) Sokoban (Train) (c) FrozenLake (Train) (d) FrozenLake (Train)

(e) Sokoban (Val) (f) Sokoban (Val) (g) FrozenLake (Val) (h) FrozenLake (Val)

Figure 10: Threshold sensitivity analysis. **Top row:** training metrics (smoothed); **bottom row:** validation metrics (unsmoothed). **(a, c, e, g)** Best success rates at the evaluated EV switching thresholds; bars and markers denote observed values, and solid curves show quadratic least-squares fits. **(b, d, f, h)** Full curves under each threshold setting. The fitted profiles are approximately concave near zero, with peaks shifted toward slightly negative thresholds.

E.5 EV Threshold Sensitivity

We vary the switching threshold across $\{-0.2, -0.1, 0, 0.1, 0.2\}$ while keeping all other configurations fixed, selecting the critic when $\widehat{\mathrm{ev}}_{\mathcal{B}} > \tau$. As shown in Figure 10(a, c, e, g), the performance profiles on Sokoban and FrozenLake are broadly single-peaked, with quadratic fits suggesting approximate concavity near zero for both training and validation metrics. Among the tested thresholds, $\tau = -0.1$ gives the highest observed values on both tasks, while $\tau = 0$ remains competitive. These fits summarize the measured points rather than establish a universal optimum.

The displacement may arise from the gap between population and sample EV. Theorem 1 characterizes the variance boundary using population quantities, whereas the practical switching rule uses $\widehat{\mathrm{ev}}_{\mathcal{B}}$ computed from finite rollouts. The best empirical threshold need not coincide with the population boundary, and the sweep does not isolate the cause of the offset. Zero therefore serves as a theoretically motivated reference rather than a guaranteed performance optimum.

The full curves in Figure 10(b, d, f, h) reveal asymmetric degradation away from this near-zero region. Positive thresholds invoke the batch-mean mode more frequently and can discard useful critic information, with pronounced late-stage deterioration in the plotted runs. More negative thresholds retain the critic under lower measured EV and can expose training to unreliable value estimates for longer. The observed sensitivity is consistent with a trade-off between these two effects.