KL-Regularized Policy Optimization for Critic-Free Agentic Reinforcement Learning

Yifan Zhang et al.

yifanzhangresearch@gmail.com

September 18, 2026*

Abstract

KL-Regularized Policy Optimization (KLPO) is a critic-free, single-rollout method for asynchronous off-policy agentic reinforcement learning. Its canonical objective fits a local policy mirror descent (PMD) condition by regressing trainer-to-sampler log-ratios, with no multiplicative importance weights. Our default token-regression implementation uses a per-token return coefficient and score centering to realize this gradient without a critic or reward group. Monte Carlo KL (MC-KL) estimates the conditional score mean with $M \geq 1$ independent sampler draws per prefix, recovering the full-KL gradient in expectation without extra response rollouts. Under deterministic token transitions and terminal rewards, Bellman telescoping gives an alternative sequence-regression objective. Removing its prompt value changes the population loss only by a parameter-independent constant; independent MC-KL with $M \geq 2$ and leave-one-out trajectory feedback preserves its expected gradient. Fresh auxiliary draws from the historical sampler preserve conditional unbiasedness after adaptive updates; fixed-record reuse is an empirical surrogate. Top-K Aggregated KL (TopK-KL) and Binary KL are optional approximations that need not preserve the exact equivalence. Both routes avoid estimating or learning prompt-and-version normalizers, so asynchronous replay needs no auxiliary normalization predictor or same-prompt response group.

Project Page: https://github.com/yifanzhang-pro/KLPO

1 Introduction

An agent can finish a response several optimizer steps after it began. At step t, the trainer uses $\pi_{\theta_t}^{\text{trainer}}$ while a worker may still sample from $\pi_{\theta_{t-k}}^{\text{sampler}}$. Tool latency and queued trajectories produce different lags across workers; sampling and execution differences can separate the two distributions even at identical parameters. Training on these responses is an off-policy problem.

KL-Regularized Policy Optimization (KLPO) fits the optimality condition of a local KL-regularized policy improvement step. Starting from the RPG–URKL objective (Zhang et al., 2026), we obtain the Gibbs policy and its normalizer, take log-ratios, and regress the resulting condition under the collection sampler. Profiling the local intercept gives the canonical loss $\mathcal{L}_{\text{tok}}$. The sampler probability enters the log-ratio without a multiplicative importance weight.

*Revised: September 20, 2026

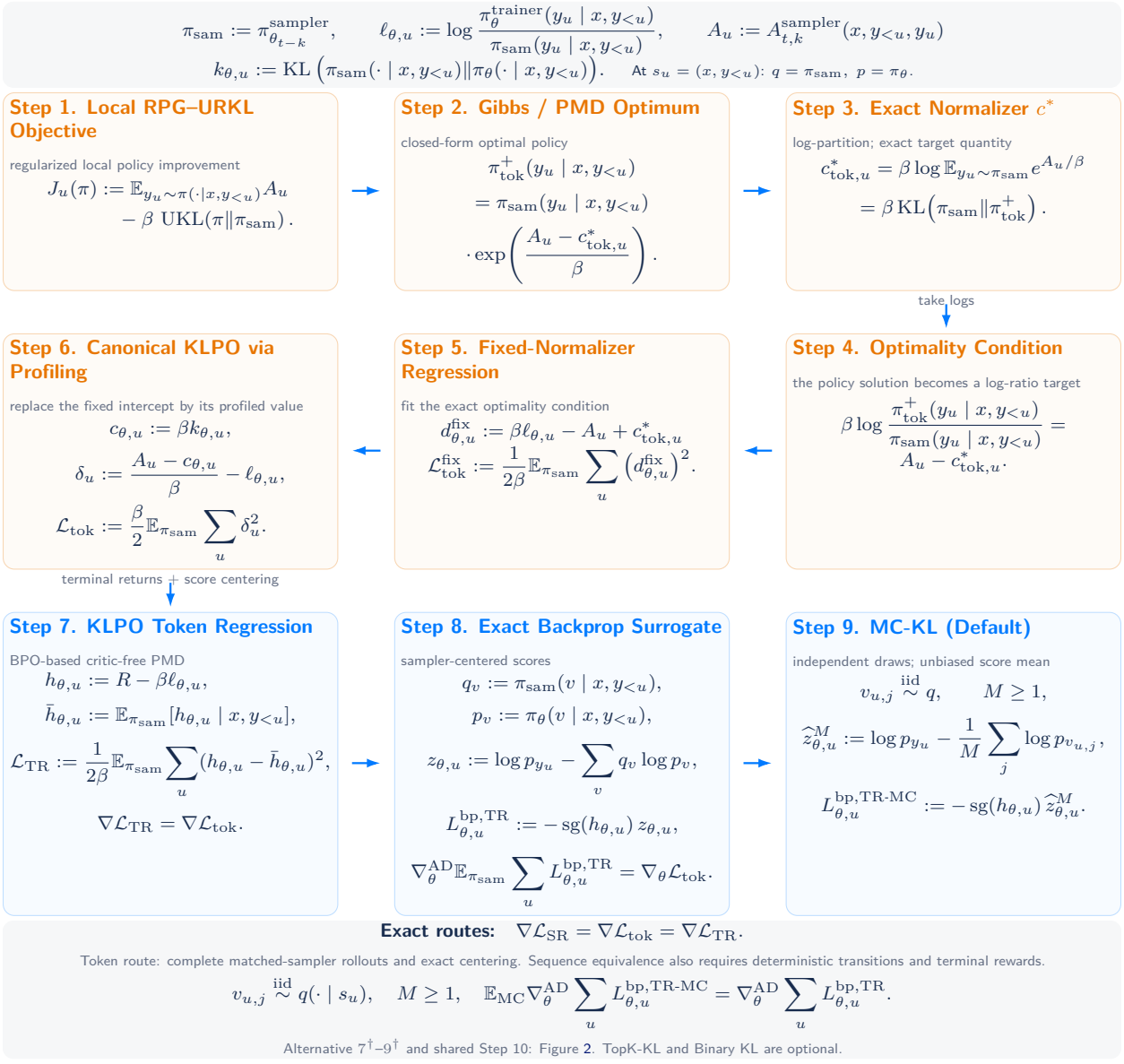

Figure 1 KLPO token regression + MC-KL (default). Steps 1–4 derive the RPG–URKL Gibbs target (Zhang et al., 2026); Steps 5–6 fit and profile its optimality condition, connecting to SPPO and GPO (Wu et al., 2024; Zhang et al., 2025). Step 7 builds on BPO's critic-free PMD reformulation (Song et al., 2026), using terminal returns for token regression with the same population gradient. Step 8 realizes this gradient with sampler-centered scores (Marek and Ryabinin, 2026). Step 9 estimates the conditional score mean with $M \geq 1$ independent auxiliary draws per prefix, recovering the full-KL gradient in expectation. TopK-KL and Binary KL are optional approximations. Figure 2 gives the sequence-regression alternative and shared optional KL budget.

The canonical loss depends on sampler advantages. Our default KLPO token-regression route replaces these advantages with terminal returns and uses score centering (Marek and Ryabinin, 2026) to preserve the exact population gradient. It uses the local coefficient $R - \beta \ell_{\theta,u}$ and an additive conditional score correction. Default MC-KL estimates this correction with $M \geq 1$ independent

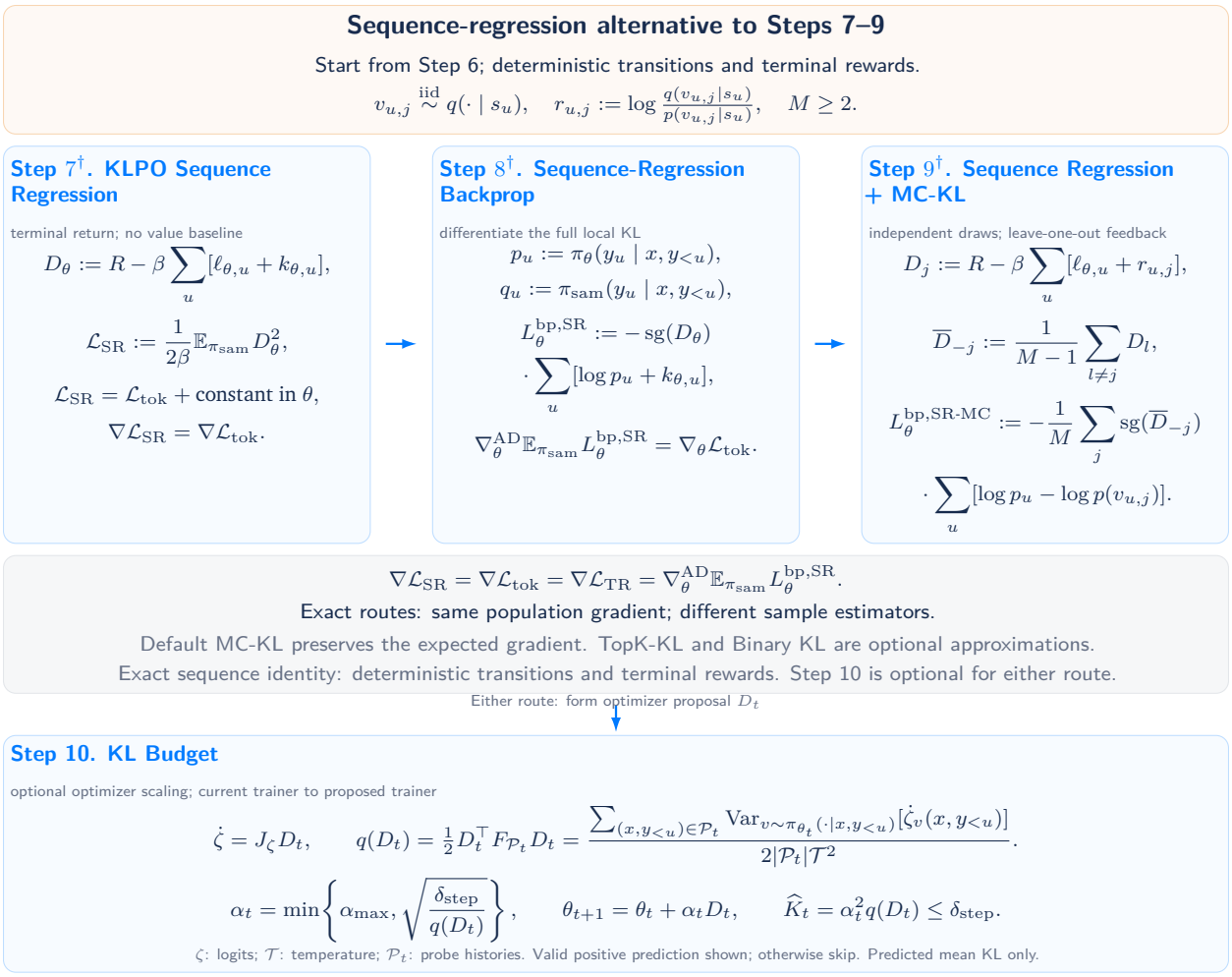

Figure 2 KLPO sequence regression + MC-KL and shared KL budget. Alternative Steps 7†–9† branch from Step 6 of Figure 1. Step 7† uses the BPO telescoping identity (Song et al., 2026) under deterministic transitions and terminal rewards, omitting the prompt baseline without changing the population gradient. Step 8† differentiates the trajectory residual through the full local KLs. Step 9† uses $M \geq 2$ independent auxiliary draws per prefix and leave-one-out trajectory feedback. Under the stated assumptions, the two MC routes share a population gradient but need not agree samplewise. TopK-KL and Binary KL are optional for either route. Step 10 budgets the predicted mean incremental KL of either optimizer proposal.

auxiliary sampler draws per prefix. It records their IDs and sampler log-probabilities, together with each rollout action's log-probability; the extra token draws need no response continuations. Optional TopK-KL stores sampler heads, and Binary KL needs only rollout-action records.

For deterministic token transitions and terminal rewards, Bellman telescoping gives the alternative KLPO sequence-regression route (Song et al., 2026). Removing its prompt value adds only a parameter-independent constant to the population loss. Sequence regression uses $M \geq 2$ independent MC-KL draws and leave-one-out trajectory feedback to avoid covariance bias. The gradient route (token by default, or sequence) and KL estimator (MC-KL by default, TopK-KL, Binary, or full) are independent choices. Figures 1 and 2 show both routes with MC-KL, which preserves their respective full-KL gradients in expectation over independent auxiliary draws (Section 3.9). Fresh draws from the

historical sampler are required for conditional unbiasedness after adaptive updates; fixed-record replay is an empirical surrogate. Either route can use an optional KL budget on the optimizer proposal.

Section 3 derives both routes, and Section 4 applies them to historical sampler versions. Section 5 compares their single-rollout requirements with Sequence KL-Regularized Policy Optimization (SKLPO), which needs prompt-and-version normalization. KLPO requires no auxiliary value or normalizer predictor and no same-prompt response group. Appendices A–C develop SKLPO's response-Gibbs target, normalization choices, and implicit regularized $Q^\star$ interpretation. Proofs and detailed comparisons with SPPO and GPO (Wu et al., 2024; Zhang et al., 2025) are also in the appendix.

2 Background

We index optimizer steps by t and token positions by u. The current trainer and a lagged sampler are

$$\pi_{\theta_t}^{\text{trainer}}, \qquad \pi_{\theta_{t-k}}^{\text{sampler}}, \qquad k \in \{1, 2, \dots, K_t\}. \tag{2.1}$$

Each response uses one sampler checkpoint and records its version v_i. Its lag is $k_i(t) = t - v_i$: reuse increases the lag while leaving $\theta_{t-k_i(t)} = \theta_{v_i}$ fixed. A batch can contain several versions; K_t is its maximum lag, rather than a fixed system limit. Synchronous collection corresponds to $k = 0$ with matched execution.

The collection policy $\pi_{\theta_{t-k}}^{\text{sampler}}$ is also the fixed KL reference. Logged probabilities must include the sampling transforms and execution used for collection. We write $\pi_\theta = \pi_\theta^{\text{trainer}}$ in local derivations and evaluate gradients at θ_t; no separate reference snapshot is needed.

Section 3 fixes one collection version (t, k); Section 4 combines versions in replay. Sampler-dependent quantities retain this conditioning even when their indices are suppressed. Appendix F treats outcome-dependent selection.

Let $x \sim \mathcal{D}$ be a prompt and $y \in \mathcal{Y}_x$ a complete response, including termination. We assume a finite vocabulary, bounded response length, a deterministic verifier reward $R(x, y)$, and normalized policies positive on a common feasible response support. The exact update optimizes over the full probability simplex; a neural parameterization may restrict the attainable policies. Section 4 extends the setting to interactive trajectories.

For $y = (y_1, \dots, y_T)$, the sequence probability is a product of token probabilities. For either $\pi = \pi_\theta$ or $\pi = \pi_{\theta_{t-k}}^{\text{sampler}}$,

$$\begin{aligned} \pi(y \mid x) &= \prod_{u=1}^{T} \pi(y_u \mid x, y_{<u}), \\ \log \pi(y \mid x) &= \sum_{u=1}^{T} \log \pi(y_u \mid x, y_{<u}). \end{aligned} \tag{2.2}$$

Complete-response log-probabilities are evaluated by summing token log-probabilities, avoiding underflow.

3 KL-Regularized Policy Optimization

We follow Figure 1: derive a local policy target, fit its log-ratio condition, and realize the profiled regression gradient through default KLPO token regression. Figure 2 gives the sequence-regression

alternative under deterministic transitions and terminal rewards. We abbreviate the two regression losses by $\mathcal{L}_{\mathrm{SR}}$ and $\mathcal{L}_{\mathrm{TR}}$; $\mathcal{L}_{\mathrm{tok}}$ remains the canonical advantage-based objective. Sequence regression shares a trajectory residual across tokens, while token regression uses a local coefficient. These names describe residual granularity; score centering is a shared correction operation. KLPO sequence regression uses pathwise local KLs and is distinct from the response-Gibbs SKLPO objective in Appendix A. Throughout this section, $x \sim \mathcal{D}$ and $Y \sim \pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot \mid x)$ are complete rollouts from one fixed collection version.

Write $s_u = (x, y_{<u})$, $a_u = y_u$, and

$$\ell_{\theta,u} = \log \frac{\pi_\theta(a_u \mid s_u)}{\pi^{\mathrm{sampler}}_{\theta_{t-k}}(a_u \mid s_u)}, \qquad k_\theta(s) = \mathrm{KL}(\pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot \mid s) \| \pi_\theta(\cdot \mid s)). \tag{3.1}$$

3.1 The Local RPG–URKL Objective

Define $Q^{\mathrm{sampler}}_{t,k}(s,a) = \mathbb{E}_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}}[R \mid s,a]$, $V^{\mathrm{sampler}}_{t,k}(s) = \mathbb{E}_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}}[R \mid s]$, and $A^{\mathrm{sampler}}_{t,k} = Q^{\mathrm{sampler}}_{t,k} - V^{\mathrm{sampler}}_{t,k}$. Step 1 applies the RPG–URKL objective (Zhang et al., 2026) to the action distribution at a visited history:

$$\begin{aligned} J_s(\pi) &= \mathbb{E}_{a\sim\pi(\cdot|s)} A^{\mathrm{sampler}}_{t,k}(s,a) - \beta\, \mathrm{UKL}\left(\pi(\cdot \mid s) \| \pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot \mid s)\right), \\ \mathrm{UKL}(p\|q) &= \sum_a \left[p_a \log \frac{p_a}{q_a} + q_a - p_a \right], \qquad \beta > 0. \end{aligned} \tag{3.2}$$

Both policies are normalized, so the mass correction vanishes and $\mathrm{UKL} = \mathrm{KL}$. Maximizing J_s is a local policy mirror descent (PMD) step with the sampler advantage and reference held fixed.

3.2 The Gibbs Optimum and Exact Normalizer

Steps 2–3 solve this constrained improvement problem and normalize the result.

Proposition 3.1 (Token PMD Target)**.** At each covered history, define

$$c^*_{\mathrm{tok}}(s) := \beta \log \mathbb{E}_{a\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot|s)} e^{A^{\mathrm{sampler}}_{t,k}(s,a)/\beta}. \tag{3.3}$$

The unique maximizer of Equation (3.2) is

$$\pi^+_{\mathrm{tok}}(a \mid s) := \pi^{\mathrm{sampler}}_{\theta_{t-k}}(a \mid s) e^{(A^{\mathrm{sampler}}_{t,k}(s,a) - c^*_{\mathrm{tok}}(s))/\beta}. \tag{3.4}$$

Its normalizer satisfies

$$c^*_{\mathrm{tok}}(s) = \beta\, \mathrm{KL}(\pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot \mid s) \| \pi^+_{\mathrm{tok}}(\cdot \mid s)). \tag{3.5}$$

Proof: Appendix I.1. Equation (3.3) defines c^*_{tok} from the fixed sampler and its advantages. Equation (3.5) is an identity at the solved policy, not a recursive definition. The zero sampler mean of $A^{\mathrm{sampler}}_{t,k}$ gives this KL form.

3.3 Optimality-Condition Regression

Step 4 takes logarithms of the Gibbs solution:

$$\beta \log \frac{\pi^{+}_{\mathrm{tok}}(a \mid s)}{\pi^{\mathrm{sampler}}_{\theta_{t-k}}(a \mid s)} = A^{\mathrm{sampler}}_{t,k}(s,a) - c^{*}_{\mathrm{tok}}(s). \tag{3.6}$$

Step 5 fits this condition using the fixed-normalizer residual

$$d^{\mathrm{fix}}_{\theta,u} := \beta \ell_{\theta,u} - A^{\mathrm{sampler}}_{t,k}(s_u,a_u) + c^{*}_{\mathrm{tok}}(s_u). \tag{3.7}$$

The loss is $(2\beta)^{-1}\mathbb{E}\sum_u (d^{\mathrm{fix}}_{\theta,u})^2$, or equivalently

$$\mathcal{L}^{\mathrm{fix}}_{\mathrm{tok}} = \frac{\beta}{2}\mathbb{E}_{x,Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}} \sum_u \left[\frac{A^{\mathrm{sampler}}_{t,k}(s_u,a_u) - c^{*}_{\mathrm{tok}}(s_u)}{\beta} - \ell_{\theta,u} \right]^2. \tag{3.8}$$

With exact targets, coverage, and realizability, zero loss identifies π^{+}_{tok} (Theorem D.2). Its gradient is $\mathbb{E}\sum_u d^{\mathrm{fix}}_{\theta,u} \nabla_\theta \log \pi_\theta(a_u \mid s_u)$, with no importance multiplier. Regression and regularized return share this exact solution but generally have different gradients and restricted-model fits.

This squared log-ratio construction follows the regression pattern in SPPO and GPO (Wu et al., 2024; Zhang et al., 2025). GPO Equation (5.3) is response-level: its local analogue replaces the preference score by the sampler advantage and retains the exact normalizer. GPO's practical loss sets its log-normalization term to zero. Canonical KLPO instead profiles the local intercept.

3.4 Profiling the Local Intercept

In Step 6, minimize the squared residual over one action-independent intercept at each history:

$$\begin{aligned} c_\theta(s) &:= \operatorname*{argmin}_{c\in\mathbb{R}} \mathbb{E}_{a\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot|s)} \left[\beta \log \frac{\pi_\theta(a \mid s)}{\pi^{\mathrm{sampler}}_{\theta_{t-k}}(a \mid s)} - A^{\mathrm{sampler}}_{t,k}(s,a) + c \right]^2 \\ &= \beta k_\theta(s). \end{aligned} \tag{3.9}$$

The equality uses $\mathbb{E}_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}}[A^{\mathrm{sampler}}_{t,k}(s,a) \mid s] = 0$ and $\mathbb{E}_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}}[\ell_\theta \mid s] = -k_\theta(s)$.

Lemma 3.2 (Canonical KLPO)**.** Profiling the advantage-regression intercept gives $c_\theta(s) = \beta k_\theta(s)$. Define

$$\delta_u = \frac{A^{\mathrm{sampler}}_{t,k}(s_u,a_u)}{\beta} - \ell_{\theta,u} - k_\theta(s_u), \qquad \mathcal{L}_{\mathrm{tok}} = \frac{\beta}{2}\mathbb{E}_{x,Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}} \sum_u \delta_u^2. \tag{3.10}$$

Then $\mathbb{E}_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}}[\delta_u \mid s_u] = 0$ and

$$\nabla_\theta \mathcal{L}_{\mathrm{tok}} = -\beta \mathbb{E}_{x,Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}} \sum_u \delta_u g_\theta(s_u,a_u), \qquad g_\theta(s,a) = \nabla_\theta \log \pi_\theta(a \mid s). \tag{3.11}$$

Under coverage and realizability, the unique zero-loss policy on covered histories is π^{+}_{tok}.

Proof: Appendix I.1.

At $\pi_\theta = \pi^+_{\mathrm{tok}}$, the profiled intercept equals c^*_{tok} by Equation (3.5). The fixed and profiled losses therefore share the realizable zero-loss policy, but their gradients generally differ away from it. Direct evaluation of $\mathcal{L}_{\mathrm{tok}}$ still requires sampler advantages and local KLs. Steps 7–9 follow the default KLPO token-regression route using terminal-return score centering. Alternative Steps $7^\dagger$–$9^\dagger$ use sequence regression via Bellman telescoping.

3.5 Step 7: KLPO Token Regression (Default)

Step 7 of Figure 1 builds on BPO's critic-free PMD reformulation (Song et al., 2026). Here we express the canonical gradient through terminal-return token regression; Step 8 realizes it with sampler-centered scores (Marek and Ryabinin, 2026).

Step 7 replaces sampler advantages by terminal returns while preserving the gradient of the profiled loss. Center the trainer score under the sampler and define the dynamic return coefficient:

$$\begin{aligned} \widetilde{g}_\theta(s,a) &= g_\theta(s,a) - \mathbb{E}_{v\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot|s)} g_\theta(s,v) = g_\theta(s,a) + \nabla_\theta k_\theta(s), \\ h_{\theta,u} &= R(x,Y) - \beta\ell_{\theta,u}, \qquad \bar{h}_\theta(s) = \mathbb{E}_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}}[h_\theta \mid s]. \end{aligned} \tag{3.12}$$

The conditional expectation in $\bar{h}_\theta$ includes both the next action and its sampler continuation. For analysis, define the profiled terminal-return token-regression loss

$$\mathcal{L}_{\mathrm{TR}}(\theta) = \frac{1}{2\beta}\mathbb{E}_{x,Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}} \sum_u \left(h_{\theta,u} - \bar{h}_\theta(s_u)\right)^2. \tag{3.13}$$

Theorem 3.3 (Critic-Free Token-Regression Gradient)**.** For complete rollouts from the fixed sampler and exact conditional score means,

$$\begin{aligned} \mathcal{L}_{\mathrm{TR}} = \mathcal{L}_{\mathrm{tok}} \\ + \frac{1}{2\beta}\mathbb{E}_{x,Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}} \sum_u \left(R - Q^{\mathrm{sampler}}_{t,k}(s_u,a_u)\right)^2 . \end{aligned} \tag{3.14}$$

The second term is independent of θ. Consequently,

$$\begin{aligned} \nabla_\theta\mathcal{L}_{\mathrm{tok}} &= \nabla_\theta\mathcal{L}_{\mathrm{TR}} \\ &= -\mathbb{E}_{x,Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}} \sum_u (R - \beta\ell_{\theta,u})\,\widetilde{g}_\theta(s_u,a_u) . \end{aligned} \tag{3.15}$$

Under state/action coverage and realizability, both losses have the same unique minimizing policy π^+_{tok} on covered histories. A fixed action-independent baseline $b(s_u)$ may be subtracted from the coefficient in Equation (3.15) without changing its expectation.

Proof: Appendix D.2. The continuation-noise term in Equation (3.14) explains the tradeoff: terminal returns remove the sampler-conditioned critic and retain Monte Carlo continuation variance. One complete response per prompt suffices when the conditional score mean is supplied independently of that sampled action. Both the action and continuation must come from the sampler in the denominator. The term $-\beta\ell_{\theta,u}$ retains the regression feedback as the trainer changes; using R alone would give a different gradient.

3.6 Step 8: Exact Token-Regression Backpropagation

Step 8 implements Equation (3.15) without evaluating the conditional return mean $\bar{h}_\theta$. For autodifferentiation, use the per-token backward surrogate

$$\begin{aligned} z_{\theta,u} &= \log \pi_\theta(a_u \mid s_u) - \sum_v \pi^{\text{sampler}}_{\theta_{t-k}}(v \mid s_u) \log \pi_\theta(v \mid s_u), \\ L^{\text{bp,TR}}_{\theta,u} &= -\operatorname{sg}(h_{\theta,u})\, z_{\theta,u}. \end{aligned} \tag{3.16}$$

Proposition 3.4 (Backward-Surrogate Equivalence)**.** Hold the sampler fixed, detach $h_{\theta,u}$ and the sampler probabilities in Equation (3.16), and differentiate both trainer log-probability terms. Then the expected autodifferentiation gradient satisfies

$$\begin{aligned} \nabla^{\text{AD}}_\theta \mathbb{E}_{x,Y\sim\pi^{\text{sampler}}_{\theta_{t-k}}} \sum_u L^{\text{bp,TR}}_{\theta,u} &= -\mathbb{E}_{x,Y\sim\pi^{\text{sampler}}_{\theta_{t-k}}} \sum_u h_{\theta,u}\, \widetilde{g}_\theta(s_u, a_u) \\ &= \nabla_\theta \mathcal{L}_{\text{TR}} = \nabla_\theta \mathcal{L}_{\text{tok}}. \end{aligned} \tag{3.17}$$

The scalar surrogate value need not equal either regression loss; the equality concerns its expected autodifferentiation gradient.

Proof: Appendix D.3. Squaring $z_{\theta,u}$ or differentiating through $h_{\theta,u}$ produces a different update.

3.7 Alternative $7^\dagger$: Bellman Telescoping

Step $7^\dagger$ sums the local residuals along a response. Their conditional zero means remove the cross terms in expectation; the Bellman identity telescopes the sampler advantages into the terminal reward. This gives the exact BPO trajectory objective (Song et al., 2026, Equations (18)–(24)).

Theorem 3.5 (Population Equivalence to Exact BPO)**.** For complete rollouts from the specified sampler, exact sampler advantages, and exact local KLs,

$$\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}} \left[\left(\sum_u \delta_u \right)^2 \right] = \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}} \sum_u \delta_u^2. \tag{3.18}$$

With deterministic token transitions and terminal reward,

$$\sum_u \delta_u = \frac{R - V^{\text{sampler}}_{t,k}(x)}{\beta} - \sum_u [\ell_{\theta,u} + k_\theta(s_u)] = \delta_{\text{BPO}}. \tag{3.19}$$

Thus

$$\mathcal{L}_{\text{tok}} = \frac{\beta}{2} \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}} [\delta^2_{\text{BPO}}]. \tag{3.20}$$

Proof: Appendix I.3. The value $V^{\text{sampler}}_{t,k}(x)$ in δ_{BPO} is not needed to implement its population gradient. Define the uncentered trajectory residual and loss

$$\begin{aligned} D_\theta(x, Y) &:= R(x, Y) - \beta \sum_u [\ell_{\theta,u} + k_\theta(s_u)], \\ \mathcal{L}_{\text{SR}}(\theta) &:= \frac{1}{2\beta} \mathbb{E}_{x,Y\sim\pi^{\text{sampler}}_{\theta_{t-k}}} D_\theta(x, Y)^2. \end{aligned} \tag{3.21}$$

Proposition 3.6 (Baseline-Free Trajectory Regression)**.** Under the assumptions of Theorem 3.5,

$$\mathcal{L}_{\mathrm{SR}} = \mathcal{L}_{\mathrm{tok}} + \frac{1}{2\beta}\mathbb{E}_x[V_{t,k}^{\mathrm{sampler}}(x)^2], \qquad \nabla_\theta \mathcal{L}_{\mathrm{SR}} = \nabla_\theta \mathcal{L}_{\mathrm{tok}}. \tag{3.22}$$

More generally, replacing R by $R - b(x)$ for a fixed prompt baseline changes the constant to $\mathbb{E}_x[(V_{t,k}^{\mathrm{sampler}}(x) - b(x))^2]/(2\beta)$. Thus $b = 0$ needs neither a value estimator nor a same-prompt response group.

Proof: Appendix I.4. The equality is in population, not on each trajectory; a baseline estimated from that same trajectory is not fixed in the required sense. The uncentered loss need not vanish at the PMD policy.

3.8 Alternative $8^\dagger$: Exact Sequence-Regression Gradient

Let $g_{\theta,u} = \nabla_\theta \log \pi_\theta(a_u \mid s_u)$. Differentiating Equation (3.21) gives

$$\nabla_\theta \mathcal{L}_{\mathrm{SR}} = -\mathbb{E}_{x,Y\sim\pi_{\theta_{t-k}}^{\mathrm{sampler}}} D_\theta \sum_u [g_{\theta,u} + \nabla_\theta k_\theta(s_u)]. \tag{3.23}$$

The trajectory coefficient D_θ retains the current regression residual. BPO Equation (25) gives this full-KL derivative before its later linearization (Song et al., 2026). One complete rollout provides a stochastic gradient when the local KLs and their derivatives are available from the actual sampler conditionals. These conditionals may be stored or recomputed with the collection execution.

An equivalent backward surrogate is

$$L_\theta^{\mathrm{bp,SR}}(x, Y) := -\operatorname{sg}(D_\theta) \sum_u [\log \pi_\theta(a_u \mid s_u) + k_\theta(s_u)]. \tag{3.24}$$

Its autodifferentiation gradient equals that of $D_\theta^2/(2\beta)$ on each trajectory. Differentiate through the KLs in the bracket and detach only the coefficient; detaching the KL correction changes the update. This route requires no multiplicative importance weights.

Under the deterministic-transition and terminal-reward assumptions above, the exact routes satisfy

$$\nabla_\theta \mathcal{L}_{\mathrm{SR}} = \nabla_\theta \mathcal{L}_{\mathrm{tok}} = \nabla_\theta \mathcal{L}_{\mathrm{TR}}. \tag{3.25}$$

The scalar losses differ by sampler-dependent constants. Their single-trajectory estimators generally differ; the identity neither orders their variances nor equates the practical approximations.

3.9 Steps 9 and $9^\dagger$: Monte Carlo KL (Default)

The default estimator, *Monte Carlo KL (MC-KL)*, estimates the full conditional KL by independent sampling. At each visited prefix s_u, write $q(v) = \pi_{\theta_{t-k}}^{\mathrm{sampler}}(v \mid s_u)$ and $p(v) = \pi_\theta(v \mid s_u)$. Draw M auxiliary tokens from the recorded sampler with replacement, independently of the complete rollout and independently across prefixes and sample indices:

$$\widehat{k}_{\theta,u}^{M} := \widehat{D}_{\mathrm{MC},M}(q\|p) = \frac{1}{M}\sum_{j=1}^{M} \log \frac{q(v_{u,j})}{p(v_{u,j})}, \qquad v_{u,j} \overset{\mathrm{iid}}{\sim} q. \tag{3.26}$$

The symbols have separate roles: K is the TopK-KL head size, and M is the MC-KL sample count. MC-KL has no head or tail bucket and does not renormalize or reweight the sampled probabilities. Duplicate token IDs are valid and must be retained. For a fixed trainer and prefix, $\mathbb{E}_{\mathrm{MC}}\widehat{k}^{M}_{\theta,u} = k_\theta(s_u)$ and its derivative is unbiased. Individual estimates may be negative; clamping them destroys this identity. The auxiliary draws need no response continuations or rewards, so either regression route still uses one complete rollout per prompt.

Step 9: token regression, $M \geq 1$. The return coefficient $h_{\theta,u} = R - \beta\ell_{\theta,u}$ is independent of the auxiliary draws conditional on the rollout. Therefore

$$L_\theta^{\mathrm{bp,TR\text{-}MC}} = -\sum_u \mathrm{sg}(h_{\theta,u}) \left[\log p(a_u \mid s_u) - \frac{1}{M}\sum_j \log p(v_{u,j} \mid s_u)\right] \tag{3.27}$$

has the full-KL token-regression gradient in expectation over these draws, including at $M = 1$. Using the rollout action itself as the sole auxiliary draw instead cancels the corrected score and is not an independent MC estimator.

Step $9^\dagger$: sequence regression, $M \geq 2$. An unbiased KL estimate alone does not justify substituting it into a squared residual: using the same draws in the residual and its derivative introduces covariance bias. To remove it, form one trajectory residual per independent sample column and use the other columns for its feedback:

$$\begin{aligned} D_j &= R - \beta\sum_u \left[\ell_{\theta,u} + \log\frac{q(v_{u,j} \mid s_u)}{p(v_{u,j} \mid s_u)}\right], \\ \overline{D}_{-j} &= \frac{1}{M-1}\sum_{l\neq j} D_l, \\ L_\theta^{\mathrm{bp,SR\text{-}MC}} &= -\frac{1}{M}\sum_j \mathrm{sg}(\overline{D}_{-j})\sum_u [\log p(a_u \mid s_u) - \log p(v_{u,j} \mid s_u)]. \end{aligned} \tag{3.28}$$

This uses the same M IID draws per prefix, not M complete responses. Computing all leave-one-out means from their sum takes linear time in M.

Proposition 3.7 (Independent MC-KL Gradients)**.** At a fixed trainer and complete rollout, averaging the gradients in Equations (3.27) and (3.28) over independent auxiliary draws recovers their respective full-KL sample gradients. The sequence surrogate also has the sample gradient of the U-statistic

$$\widehat{\mathcal{L}}_{\mathrm{SR,MC}} = \frac{1}{2\beta M(M-1)}\sum_{j\neq l} D_j D_l, \qquad \mathbb{E}_{\mathrm{MC}}\widehat{\mathcal{L}}_{\mathrm{SR,MC}} = \frac{D_\theta^2}{2\beta}. \tag{3.29}$$

Consequently, the stated full-KL population-gradient identities carry over after averaging over independent auxiliary draws and sampler rollouts. Equality between the sequence and token routes retains the deterministic-transition and terminal-reward assumptions.

Proof: Appendix D.5. The U-statistic can be negative; it is not a squared loss on an individual sample. The implementation reports it without clamping and rejects $M = 1$ for sequence regression.

Collection and replay. Store each auxiliary token ID and its actual sampler log-probability; gather current trainer log-probabilities at those same IDs. Keep the historical sampler fixed and detach

its records. A fresh conditional-unbiased gradient after an adaptive trainer update requires fresh auxiliary draws from that historical sampler. Reusing a fixed MC record is a valid empirical-surrogate update, but the trainer then depends on the records and the fresh-draw conditional identity no longer applies. Resampling requires stored full conditionals or access to the matching sampler execution. MC-KL exchanges full-vocabulary summation or a deterministic head approximation for sampling variance; finite-M updates need not be monotone or match one another samplewise.

3.10 Optional KLPO Sequence Regression + TopK-KL

The optional *Top-K Aggregated KL (TopK-KL)* approximation keeps the K highest-probability sampler actions and merges the rest into one tail category. Here K denotes the head size, distinct from the default MC-KL sample count M. At history s_u, store the actual sampler's top-K action IDs $H_u = H_K(s_u)$ and original probabilities $q_v = \pi^{\text{sampler}}_{\theta_{t-k}}(v \mid s_u)$, with $K = 128$ as the default head size when TopK-KL is selected. Gather the current trainer probabilities $p_v = \pi_\theta(v \mid s_u)$ on the same IDs. Also retain the sampled action's actual log-probability even if it lies outside H_u. Do not renormalize the head. Define

$$\begin{aligned} q_{\text{tail}} &= 1 - \sum_{v \in H_u} q_v, \qquad p_{\text{tail}} = 1 - \sum_{v \in H_u} p_v, \\ k^K_{\theta,u} &:= D_{\text{TopK},K}(q\|p) := \sum_{v \in H_u} q_v \log \frac{q_v}{p_v} + q_{\text{tail}} \log \frac{q_{\text{tail}}}{p_{\text{tail}}}. \end{aligned} \tag{3.30}$$

All omitted actions form one tail category. This is a KL between coarsened distributions, rather than a KL between renormalized heads. The dynamic trajectory residual is

$$D^K_\theta := R - \beta \sum_u [\ell_{\theta,u} + k^K_{\theta,u}]. \tag{3.31}$$

Proposition 3.8 (TopK-KL Backward Surrogate)**.** Hold the sampler records and head IDs fixed. With positive tail masses, the surrogate

$$L^{\text{bp,SRK}}_\theta := -\operatorname{sg}(D^K_\theta) \sum_u [\log \pi_\theta(a_u \mid s_u) + k^K_{\theta,u}] \tag{3.32}$$

has exactly the sample gradient of $(D^K_\theta)^2/(2\beta)$ when the KL in the bracket remains differentiable. In particular, with $g_v = \nabla_\theta \log p_v$ and $\rho = q_{\text{tail}}/p_{\text{tail}}$,

$$\nabla_\theta k^K_{\theta,u} = -\sum_{v \in H_u} (q_v - \rho p_v) g_v. \tag{3.33}$$

Proof: Appendix I.4. This combination is an optional replacement for the MC-KL estimator in Step $9^\dagger$ of Figure 2. The residual and the differentiated correction must use the same KL. Finite K generally changes the canonical objective, baseline invariance, and PMD stationary point; the proposition is an identity for the approximate squared-residual loss. A full-vocabulary record uses the exact KL directly, without a tail ratio. A singleton tail also recovers full KL before numerical flooring. Appendix D.4 specifies tail stabilization.

3.11 Optional KLPO sequence regression + Binary KL

When only sampled-action probabilities are retained, binary KL is an optional approximation for either gradient route. Following BPO's binary partition (Song et al., 2026, Equation (27)), write $q_u = \pi^{\text{sampler}}_{\theta_{t-k}}(a_u \mid s_u)$, $p_u = \pi_\theta(a_u \mid s_u)$ and use

$$\begin{aligned} k^{\text{bin}}_{\theta,u} &:= q_u \log \frac{q_u}{p_u} + (1-q_u)\log\frac{1-q_u}{1-p_u}, \\ \widehat{D}_\theta &:= R - \beta \sum_u [\ell_{\theta,u} + k^{\text{bin}}_{\theta,u}], \qquad \omega_{\theta,u} := \frac{1-q_u}{1-p_u}. \end{aligned} \tag{3.34}$$

Proposition 3.9 (Binary-KL Backward Surrogate)**.** For $0 < p_u, q_u < 1$, with the sampler held fixed,

$$\nabla_\theta[\log p_u + k^{\text{bin}}_{\theta,u}] = \omega_{\theta,u} \nabla_\theta \log p_u. \tag{3.35}$$

Consequently, the surrogate

$$L^{\text{bp,SR-bin}}_\theta := -\operatorname{sg}(\widehat{D}_\theta) \sum_u \operatorname{sg}(\omega_{\theta,u}) \log p_u \tag{3.36}$$

has exactly the sample gradient of $\widehat{D}^2_\theta/(2\beta)$.

Proof: Appendix I.4. Like the MC-KL and TopK-KL routes, this option retains the dynamic regression coefficient, unlike BPO's final implementation, which linearizes that coefficient, estimates group advantages, and applies smoothing, masking, and clipping (Song et al., 2026, Section 3.2). The weight ω is a complementary-probability correction, not the action importance ratio p_u/q_u.

Binary KL needs only the sampled action's stored probability. It nevertheless changes the objective: it depends on the sampled action and generally loses the conditional zero-mean identity of the full KL. This option therefore need not share the exact PMD stationary point or baseline invariance. It is the partition in Equation (3.30) with $H_u = \{a_u\}$, which need not be the sampler's top-1 action. Evaluate complementary log-probabilities with stable `log1mexp` arithmetic in float32 or higher precision; do not form $1 - p_u$ by subtracting a rounded probability. Any additional smoothing or clipping defines a further approximation.

3.12 Optional KLPO Token Regression + TopK-KL

As an optional replacement for MC-KL in Step 9, approximate the full conditional score mean using the top-K construction of Marek and Ryabinin (2026, Section 4.1 and Appendix A), with $K = 128$ as the default head size for this option. At each stored history, abbreviate the actual sampler probabilities by q_v and current trainer probabilities by p_v. Log the sampler's top-K set H and its original probabilities, plus the sampled action's log-probability even when it lies outside H. The head is not renormalized.

$$\begin{aligned} q_{\text{tail}} &= 1 - \sum_{v\in H} q_v, \qquad p_{\text{tail}} = 1 - \sum_{v\in H} p_v, \qquad \rho = \frac{q_{\text{tail}}}{p_{\text{tail}}}, \\ C_\theta(s) &= \sum_{v\in H} \operatorname{sg}(q_v - \rho p_v)\log p_v, \qquad L^{\text{bp,TRK}}_{\theta,u} = -\operatorname{sg}(h_{\theta,u})[\log p_{a_u} - C_\theta(s_u)]. \end{aligned} \tag{3.37}$$

The tail is modeled as $\widehat{q}_v = \rho p_v$; the complete coefficient $q_v - \rho p_v$ is detached. Before flooring, Equation (3.33) gives $\nabla_\theta C_\theta = -\nabla_\theta k^K_\theta$. Thus the same TopK-KL supplies the score correction for

either route. Sequence regression weights the summed corrected scores by the trajectory residual D_θ^K, whereas token regression weights each token by $h_{\theta,u} = R - \beta\ell_{\theta,u}$. Their approximate updates generally differ, even with identical stored heads. The approximation does not estimate c_{tok}^*, and finite K need not preserve the exact gradient identity or the PMD stationary point. Appendix D.4 gives the derivation and numerical safeguards.

For a common unfloored TopK-KL, set $\widetilde{g}_{\theta,u}^K = g_{\theta,u} + \nabla_\theta k_{\theta,u}^K$. The two single-trajectory backward gradients are

$$\begin{aligned} \nabla_\theta^{\text{AD}} L_\theta^{\text{bp,SRK}} &= -D_\theta^K \sum_u \widetilde{g}_{\theta,u}^K, \\ \nabla_\theta^{\text{AD}} \sum_u L_{\theta,u}^{\text{bp,TRK}} &= -\sum_u (R - \beta\ell_{\theta,u}) \widetilde{g}_{\theta,u}^K. \end{aligned} \tag{3.38}$$

The score correction is shared; the regression coefficients distinguish the routes. Full KL restores the stated population equivalence, not samplewise equality. Finite TopK-KL need not preserve even population equivalence.

3.13 Optional KLPO token regression + Binary KL

The gradient route and KL approximation are independent choices. Substituting the binary correction from Equation (3.35) into the token-regression route gives

$$L_{\theta,u}^{\text{bp,TR-bin}} := -\operatorname{sg}(R - \beta\ell_{\theta,u})\ \operatorname{sg}(\omega_{\theta,u}) \log p_u. \tag{3.39}$$

This uses only sampled-action probabilities. Keep the per-token coefficient $h_{\theta,u}$; replacing it with $\widehat{D}_\theta$ gives KLPO sequence regression + Binary KL instead. The binary-corrected score generally has nonzero mean under the sampler, so this is an approximation to exact score centering, without its canonical-gradient or baseline-invariance guarantee. Both sequence regression and token regression can use MC-KL (the default), full KL, TopK-KL, or Binary KL.

Recompute all trainer-dependent residuals and corrections at every update, including batch reuse. Sum over policy tokens and average over responses; exclude prompt, padding, and tool-output tokens and count termination consistently. An outer $1/T_i$ generally breaks the exact equivalence. The default MC-KL needs independently sampled IDs and their original log-probabilities; optional TopK-KL needs head IDs and original probabilities. Both retain rollout-action log-probabilities separately. Binary KL needs only the rollout-action records, and full KL needs full sampler conditionals. Resampling MC records during replay requires access to the historical conditionals; fixed-record reuse has the empirical-surrogate interpretation described above.

3.14 Step 10: KL Budget

After either Steps 7–9 or Steps $7^\dagger$–$9^\dagger$ construct the gradient, Step 10 optionally scales the optimizer's proposed update to a predicted KL budget, following the calibration rule in RPGA (Anonymous authors, 2026). Let D_t be the complete parameter displacement the optimizer would apply, including its learning rate, momentum, preconditioning, and any weight decay. Hold θ_t and D_t fixed during the probe. Scaling the raw gradient before an adaptive optimizer generally does not produce the same displacement as scaling D_t afterward.

Algorithm 1 KLPO Token Regression and Sequence Regression

Require: Completed responses with actual sampled-action probabilities and collection versions
Require: Route: token (default) or sequence; KL: MC-KL (default), TopK-KL, binary, or full
Require: MC-KL: $M \geq 1$ for token, $M \geq 2$ for sequence; optional TopK-KL: head size $K = 128$
Require: Optional KL budget $\delta_{\text{step}} \geq 0$, cap $0 < \alpha_{\max} \leq 1$, probe histories $\mathcal{P}_t$

for optimizer steps $t = 0, 1, \ldots$ **do**
  Read B responses $(x_i, y_i, T_i, R_i, v_i)$ and their required sampler records.
  Score policy tokens with $\pi_{\theta_t}^{\text{trainer}}$; compute $\ell_{i,u}$ against version v_i.
  For MC-KL, draw M fresh independent auxiliary tokens per prefix from its historical sampler; score their log-ratios.
  for responses $i = 1, \ldots, B$ **do**
    if token-regression route (default) **then**
      Set $h_{i,u} = R_i - \beta \ell_{i,u}$; with default MC-KL, use Equation (3.27).
      With TopK-KL, form $L_i = \sum_u L_{i,u}^{\text{bp,TRK}}$ by Equation (3.37).
      With full KL, use $L_i = \sum_u L_{i,u}^{\text{bp,TR}}$ from Equation (3.16).
      With binary KL, instead use $L_i = \sum_u L_{i,u}^{\text{bp,TR-bin}}$ by Equation (3.39).
    else
      With default MC-KL, use leave-one-out feedback in Equation (3.28).
      With TopK-KL, form $D_{i,\theta}^K$ and $L_i = L_{i,\theta}^{\text{bp,SRK}}$ by Equations (3.31)–(3.32).
      With binary KL, form $\widehat{D}_{i,\theta}$ and $L_i = L_{i,\theta}^{\text{bp,SR-bin}}$ by Equations (3.34)–(3.36).
      With full KL, use $D_{i,\theta}$ and $L_i = L_{i,\theta}^{\text{bp,SR}}$ by Equation (3.24).
    end if
  end for
  Compute $G_t = \nabla_\theta^{\text{AD}}[B^{-1} \sum_i L_i]$ with the specified stop-gradients.
  Form the optimizer's complete proposed displacement D_t; do not apply it yet.
  Set $\alpha_t = 1$; if the KL budget is enabled, probe $J_\zeta D_t$ and use Equation (3.43).
  Apply $\theta_{t+1} = \theta_t + \alpha_t D_t$; record realized probe KL if enabled.
  Publish checkpoints; retain original rewards, sampler probabilities, and collection versions.
end for

Choose a nonempty set $\mathcal{P}_t$ of generated-token histories from the current batch. Define the mean incremental trainer KL on these fixed histories:

$$K_{\mathcal{P}_t}(\alpha) := \frac{1}{|\mathcal{P}_t|} \sum_{s \in \mathcal{P}_t} \text{KL}(\pi_{\theta_t}(\cdot \mid s) \| \pi_{\theta_t + \alpha D_t}(\cdot \mid s)) \,. \tag{3.40}$$

The histories may come from different sampler versions; the two distributions in this KL are the current and proposed trainers. The budget δ_{step} is separate from the regression coefficient β.

Proposition 3.10 (Predicted KL Along the Optimizer Proposal)**.** For a smooth positive policy with bounded third derivatives along the proposed segment,

$$K_{\mathcal{P}_t}(\alpha) = \alpha^2 q(D_t) + O(\alpha^3 \|D_t\|^3), \qquad q(D_t) := \tfrac{1}{2} D_t^\top F_{\mathcal{P}_t} D_t, \tag{3.41}$$

where $F_{\mathcal{P}_t}$ is the current trainer's policy Fisher averaged over the probe histories. Write its logits as $\zeta_\theta(s)$, set $p_v(s) = \text{softmax}(\zeta_{\theta_t}(s)/\mathcal{T})_v$ for fixed temperature $\mathcal{T} > 0$, and compute the directional derivative

$$\dot{\zeta}(s) = J_{\zeta_{\theta_t}}(s) D_t, \qquad q(D_t) = \frac{1}{2|\mathcal{P}_t|\mathcal{T}^2} \sum_{s \in \mathcal{P}_t} \text{Var}_{v \sim p(\cdot|s)}[\dot{\zeta}_v(s)] \,. \tag{3.42}$$

For $0 < q(D_t) < \infty$, the scale below satisfies $\alpha_t^2 q(D_t) \le \delta_{\text{step}}$.

Proof: Appendix D.6.
With $\delta_{\text{step}} \ge 0$ and $0 < \alpha_{\max} \le 1$, use

$$\alpha_t = \begin{cases} \min\left\{\alpha_{\max}, \sqrt{\delta_{\text{step}}/q(D_t)}\right\}, & \delta_{\text{step}} > 0,\ 0 < q(D_t) < \infty, \\ 0, & \text{otherwise}, \end{cases} \qquad \theta_{t+1} = \theta_t + \alpha_t D_t. \tag{3.43}$$

If the probe is unavailable or numerically invalid, skip the parameter update and flag it. A zero prediction also triggers a skipped update: zero directional curvature does not certify that a finite step leaves the policy unchanged. Disabling this optional step uses $\alpha_t = 1$.

Compute $J_\zeta D_t$ by a Jacobian–vector product through all updated parameters. The vocabulary variance uses the current trainer probabilities and needs no additional sampled action or stored sampler head. It can be accumulated as $\sum_v p_v(\dot\zeta_v - \sum_w p_w \dot\zeta_w)^2$ and reduced across ranks with probe-token counts. This requires extra probe computation but no explicit Fisher matrix or Fisher inverse. Either route still requires the records selected by its KL approximation: sampled-action probabilities, plus top-K heads or full conditionals when applicable.

The constraint is on the predicted quadratic cost, not a guaranteed realized KL. Record $K_{\mathcal{P}_t}(\alpha_t)$ after applying the update; a hard acceptance rule would additionally require a remainder bound or a post-update check with rejection. This fixed-state mean is neither a maximum over histories nor a trajectory KL, and it does not bound historical-sampler-to-new-trainer divergence. The probe models a smooth temperature-scaled softmax; additional truncation, unsupported update components, and finite-precision rendering require separate treatment. Step 10 leaves the loss and its gradient identities unchanged, but a batch-dependent scale need not preserve the unbiasedness of the resulting parameter update. It does not remove binary-KL or finite-K approximation error.

4 Asynchronous Off-Policy Agentic Training

KLPO trains from a queue of completed responses using Algorithm 1. Let $\mathcal{B}_t = \{(x_i, y_i, T_i, R_i, v_i)\}_{i=1}^B$ be the batch at trainer step t, where response i was generated by $\pi^{\text{sampler}}_{\theta_{v_i}}$ and $v_i = t - k_i(t)$. Each response retains its collection version and actual sampled-action probabilities. Default token regression + MC-KL uses $M \ge 1$ independent auxiliary sampler draws per prefix. The alternative sequence-regression route requires $M \ge 2$ with leave-one-out feedback. Record their IDs and actual sampler log-probabilities. Fresh auxiliary draws at each adaptive update require access to the historical sampler conditionals; fixed-record reuse is an empirical surrogate. Optional TopK-KL needs top-K head IDs and original probabilities, full KL needs full conditionals, and Binary KL uses only the rollout-action records. Figure 3 shows KLPO collection and training without a critic, auxiliary normalizer, or same-prompt response group. For interactive trajectories, $y_{i,<u}$ includes all tool outputs observed before policy token $y_{i,u}$; T_i counts only policy tokens.

4.1 Token Updates across Sampler Versions

Let $\widetilde{g}_{i,u,\theta}$ be the score for token $y_{i,u}$ centered under its collection sampler $\pi^{\text{sampler}}_{\theta_{v_i}}$, as in Equation (3.12).

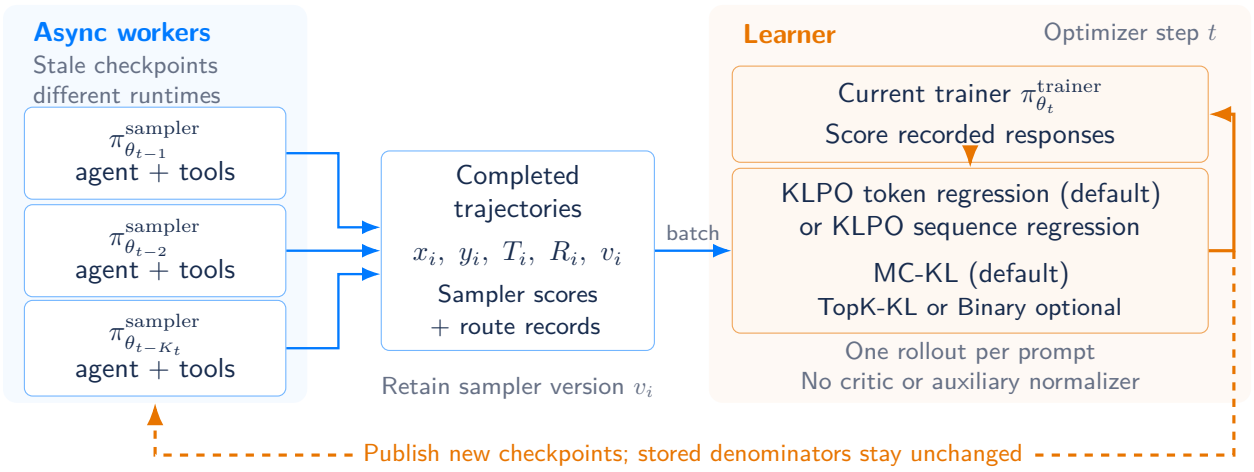

Figure 3 Asynchronous single-rollout KLPO. Workers return completed responses with their collection probabilities and the records required by the selected KL approximation. The learner uses sequence regression or token regression without a critic, auxiliary normalizer, or same-prompt response group. All sampler-dependent quantities remain tied to the recorded collection version.

Proposition 4.1 (Version-Conditioned Token Update)**.** For the token-regression backward surrogate with full conditional centering,

$$\begin{aligned} h_{i,u,\theta} &= R_i - \beta \ell_{i,u,\theta}, \\ \nabla_\theta \widehat{\mathcal{L}}^{\mathrm{bp,TR}}_{\mathrm{tok},t} &= -\frac{1}{B} \sum_i \sum_{u=1}^{T_i} h_{i,u,\theta}\, \widetilde{g}_{i,u,\theta}. \end{aligned} \tag{4.1}$$

For the full-KL sequence-regression route, let $D_{i,\theta} = R_i - \beta \sum_u [\ell_{i,u,\theta} + k_{i,\theta}(s_{i,u})]$. Its batch gradient is

$$\nabla^{\mathrm{AD}}_\theta \widehat{\mathcal{L}}^{\mathrm{bp,SR}}_t = -\frac{1}{B} \sum_i D_{i,\theta} \sum_{u=1}^{T_i} \widetilde{g}_{i,u,\theta}. \tag{4.2}$$

If each version supplies complete, unfiltered rollouts from its recorded sampler, the expectation is the gradient of the corresponding mixture of canonical $\mathcal{L}_{\mathrm{tok}}$ objectives, equivalently the mixture of $\mathcal{L}_{\mathrm{TR}}$ objectives, with version frequencies held fixed. The sequence-regression gradient has the same expectation under deterministic transitions and terminal rewards. The two batch estimators need not coincide.

Proof: Appendix D.12. The default independent MC-KL recovers each full-KL gradient in expectation using Proposition 3.7, with $M \geq 2$ for sequence regression and $M \geq 1$ for token regression. For optional KLPO sequence regression + TopK-KL, replace the trajectory residual and its differentiated KL by Equations (3.31)–(3.32). For KLPO token regression + TopK-KL, replace each conditional score mean by Equation (D.4); Equation (D.5) describes the resulting component error. Optional Binary KL uses Equation (3.36) or (3.39) for sequence or token regression, respectively. These approximations change the exact gradients and need not agree across routes.

The denominator, score-centering distribution, and continuation return must use the same collection version. Recompute the trainer-dependent trajectory residual or local $h_{i,u,\theta}$, together

with the selected correction, on every reuse. Pooling sampler heads or borrowing another version's continuations generally changes the update. Each exact component targets its own π^{+}_{tok}, so a shared trainer may have to compromise across incompatible targets.

The reduction remains a sum over policy tokens followed by a mean over responses. The conditional-centering and sequence-regression identities do not generally survive an outer $1/T_i$ weight. Appendix A.4 separately gives the replay update for SKLPO, whose normalization information must also match the collection version.

4.2 Sampler Scores and Interactive Trajectories

An error in a stored sampler probability changes the log-ratio, and errors in the conditional records also affect the local KL correction. Truncation and hard admission filters can remove required support. Removing the IS multiplier fixes neither problem; Appendix H.1 compares the local gradients and FlashREINFORCE.

For stochastic tools, use the local token-regression route when invoking the exact PMD-gradient guarantee; the sequence-regression equivalence above assumes deterministic transitions. Local PMD fits feasible action distributions using the corresponding sampler continuation values. Under unbiased collection, that sampler's terminal return estimates its ordinary $Q^{\mathrm{sampler}}_{t,k}$, not the current trainer's value or the optimal regularized soft Q-function. Pooling continuations across sampler versions changes this expectation. Appendix H.5 also explains why an unrestricted response Gibbs tilt can reweight stochastic observations outside the policy's control.

4.3 Collection and Batch Reuse

Workers adopt new checkpoints between responses and retain the collection records for each completed trajectory. Algorithm 1 trains concurrently from this queue without a collection barrier. Reuse preserves the stored sampler probabilities while recomputing the trainer scores, dynamic residual or return coefficient, and the selected KL or score correction. Neither KLPO route waits for a second response to the same prompt or fits a prompt-and-version normalizer. MC-KL resampling during reuse draws only auxiliary tokens from the historical conditionals, not additional responses; the fixed-record alternative is an empirical surrogate.

5 Why KLPO for Single-Rollout Training

Sequence KL-Regularized Policy Optimization (SKLPO) is a useful theoretical comparison, developed in Appendix A. It fits the complete-response Gibbs target $\pi^{\star}_{t,k}(Y \mid x) \propto \pi^{\mathrm{sampler}}_{\theta_{t-k}}(Y \mid x) e^{R(x,Y)/\beta}$, whereas KLPO fits local PMD improvement. Both KLPO sequence regression and SKLPO square a response residual, but their correction terms differ. Suppressing the fixed collection version and using full local KLs,

$$
\begin{aligned}
D_{\mathrm{SR}}(x,Y) &= R(x,Y) - \beta \sum_u \ell_{\theta,u} - \beta \sum_u k_\theta(s_u), \\
D_{\mathrm{SKLPO}}(x,Y) &= R(x,Y) - \beta \sum_u \ell_{\theta,u} - c^*(x), \\
c^*(x) &= \beta \log \mathbb{E}_{Y \sim \pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot|x)} e^{R(x,Y)/\beta}.
\end{aligned}
\tag{5.1}
$$

The unweighted losses are $\mathbb{E}[D^2]/(2\beta)$. KLPO differentiates the path sum of current local KLs; SKLPO uses a prompt-and-version normalizer fixed with respect to the trainer. The shared response-level regression form does not make their targets or gradients equal.

No auxiliary value or normalizer predictor. Theorem 3.3 supplies the default critic-free token-regression route. Under the assumptions of Proposition 3.6, the sequence-regression alternative omits the prompt value with only a parameter-independent change to its population loss. Consequently, either route uses a terminal reward and local probability information without fitting a value function, success predictor, or soft normalizer. SKLPO can avoid a token-value critic, but still needs normalization information: an exact supplied c^*, an independent estimate, an auxiliary predictor, or multiple responses for centering. An auxiliary prompt normalizer is not necessarily a critic; KLPO dispenses with both.

No same-prompt response group. One complete rollout per prompt suffices for KLPO's exact stochastic gradient when the required local conditional information is available and the route's assumptions hold. Independent MC-KL retains the expected gradient using extra token draws without extra response rollouts; TopK-KL and Binary KL retain the collection pattern while approximating the gradient. In SKLPO, centering the sole response against itself gives $h_1 - \bar{h} = 0$ for $h = R - \beta \sum_u \ell_{\theta,u}$, so the centered loss and gradient vanish. Estimating the fixed normalizer from that same response instead gives $\widehat{c} = R$, which yields zero initial residual at trainer–sampler equality. SKLPO still supports single-rollout collection when independent normalization information or a shared predictor is supplied; single rollout alone does not provide it (Appendix B).

No prompt-normalizer estimation during replay. KLPO retains each response's sampler probabilities and recomputes its residual and correction under the current trainer. It needs neither prompt–version reward groups nor an auxiliary normalization model to maintain across historical samplers. SKLPO's fixed normalizer depends on the prompt and collection version; fitted intercepts also track the trainer. A prompt-normalizer error can bias its off-policy regression gradient (Equation (H.9)). KLPO removes that source of error, while remaining sensitive to incorrect sampler records and approximate local KLs.

Information and target tradeoffs. The default MC-KL estimates the full-KL gradients from independent auxiliary sampler draws, with leave-one-out feedback for sequence regression. Full KL uses exact local conditionals; optional TopK-KL stores sampler heads, and optional Binary KL uses only rollout-action probabilities. SKLPO needs only sampled-action log-probabilities once its normalization information is supplied. Neither objective's target alone establishes better reward, stability, or sample efficiency under neural optimization. The advantage claimed here is single-rollout training without an auxiliary predictor or independent normalization data, under the stated probability-information requirements. SKLPO's Gibbs and implicit regularized $Q^\star$ results remain theoretical context in Appendices A–C.

6 Related Work

Implicit and inverse Q-learning. IQL (Kostrikov et al., 2022) fits expectile values and Q-functions, then extracts a policy by advantage-weighted behavioral cloning. IQ-Learn (Garg et al., 2021) learns a soft Q-function from demonstrations, implicitly representing rewards and a policy. KLPO fits policy log-ratios from scalar rewards without learning these values. Appendix C treats the implicit

$Q^\star$ interpretation of SKLPO and soft-value token regression through the regularized policy–value relationship, also studied for DPO (Rafailov et al., 2024).

Regularized policy regression. RPG–URKL optimizes the regularized return directly (Zhang et al., 2026). REBEL (Gao et al., 2024) uses relative-reward regression, matching the pairwise form of centered SKLPO. AWR (Peng et al., 2019) fits reward-weighted likelihoods. These objectives can share a realizable target while giving different gradients and restricted-model fits (Appendix J).

Preference regression. SPPO (Wu et al., 2024, Equations (4.1)–(4.6)) fits an exponential update driven by win probabilities against the previous policy; GPO (Zhang et al., 2025, Equation (5.3)) regresses log-ratios against preference scores. Their practical losses approximate the log-partition by $\eta/2$ and zero, respectively, with $\eta = 1/\beta$. Under a matched frozen reference, scalar reward-difference scores, and exact normalization, GPO's population residual equals SKLPO's up to scale. General preferences can be cyclic. Appendix J gives the comparisons and distinguishes SPPO's one-step identity from its preference-game convergence result.

Token updates and off-policy learning. BPO (Song et al., 2026) starts from token PMD and shares canonical KLPO's population loss under deterministic token transitions and terminal rewards. KLPO removes the prompt value without changing the exact population gradient. Its sequence-regression route retains the dynamic residual with default MC-KL and leave-one-out feedback, or optional TopK-KL and Binary KL; BPO Equation (25) instead linearizes this coefficient before group normalization, binary-KL approximation, smoothing, and clipping. The default token-regression route uses the additive correction of Marek and Ryabinin (2026, Equation (4)), with $R - \beta\ell_{\theta,u}$ in place of reward alone. Its optional Top-K correction (Marek and Ryabinin, 2026, Section 4.1) is the negative derivative of the same head-plus-tail KL before numerical flooring. FlashREINFORCE (Hu et al., 2026) studies critic-free, single-rollout asynchronous training with token IS, batch reward centering, sequence admission, and within-trajectory averaging.

7 Conclusion

KLPO realizes local PMD regression through sequence regression or token regression. Under the stated assumptions, full local KLs make their population gradients equal and eliminate the need for a critic, reward group, or multiplicative importance weights. The default is KLPO token regression + MC-KL, with $M \geq 1$ independent auxiliary draws per prefix for its score correction. Sequence regression is the alternative route, with $M \geq 2$ and leave-one-out trajectory feedback. MC-KL recovers the full-KL gradients in expectation under independent sampling. Fresh historical-sampler draws retain conditional unbiasedness after adaptive updates, while fixed-record reuse is an empirical surrogate. Optional TopK-KL and Binary KL are biased approximations whose updates need not agree. Optional predicted-KL calibration scales the optimizer proposal without changing the regression objective. Compared with SKLPO (Appendix A), KLPO removes prompt-normalizer estimation and same-prompt grouping from asynchronous replay. Its single-rollout updates need no auxiliary value or normalization predictor, while retaining the local probability-information requirements of the chosen KL correction.

Acknowledgement

We used large language models to improve the clarity and readability of this manuscript.

References

Anonymous authors. RPGA: Low-rank updates for efficient reinforcement learning. Manuscript, 2026. URL https://github.com/yifanzhang-pro/RPGA-Overleaf. Predicted-KL step-size calibration; accessed September 20, 2026.

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

Divyansh Garg, Shuvam Chakraborty, Chris Cundy, Jiaming Song, Matthieu Geist, and Stefano Ermon. IQ-Learn: Inverse soft-Q learning for imitation. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://arxiv.org/abs/2106.12142.

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

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 1352–1361. PMLR, 2017. URL https://proceedings.mlr.press/v70/haarnoja17a.html.

Jian Hu, Yifan Zhang, Hao Zhang, Binfeng Xu, Shaokun Zhang, Hongqing Peng, Zhiding Yu, Pavlo Molchanov, Jan Kautz, and Yi Dong. FlashREINFORCE: Critic-free single-rollout asynchronous RL for agentic language models. Manuscript, 2026. URL https://github.com/yifanzhang-pro/FlashREINFORCE-Overleaf/tree/774e4ccd79435e8251a5b1aff8366575467de4ea. Repository revision 774e4ccd, accessed September 19, 2026.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit Q-learning. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2110.06169.

Martin Marek and Max Ryabinin. Score centering stabilizes off-policy reinforcement learning. *arXiv preprint arXiv:2609.20807*, 2026. URL https://arxiv.org/abs/2609.20807v1.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019. URL https://arxiv.org/abs/1910.00177.

Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From r to Q^*: Your language model is secretly a Q-function. In *Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2404.12358.

Zhuoqing Song, Haotian Xu, Xikun Zhang, and Lidong Bing. Bellman policy optimization. *arXiv preprint arXiv:2609.15987*, 2026. URL https://arxiv.org/abs/2609.15987v1.

Yue Wu, Zhiqing Sun, Huizhuo Yuan, Kaixuan Ji, Yiming Yang, and Quanquan Gu. Self-play preference optimization for language model alignment. *arXiv preprint arXiv:2405.00675*, 2024. URL https://arxiv.org/abs/2405.00675.

Yifan Zhang, Ge Zhang, Yue Wu, Kangping Xu, and Quanquan Gu. Beyond Bradley–Terry models: A general preference model for language model alignment. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025. URL https://arxiv.org/abs/2410.02197.

Yifan Zhang, Yifeng Liu, Huizhuo Yuan, Yang Yuan, Quanquan Gu, and Andrew Chi-Chih Yao. On the design of KL-regularized policy gradient algorithms for LLM reasoning. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2505.17508.

Appendix

A Sequence KL-Regularized Policy Optimization (SKLPO)

This appendix develops SKLPO as the response-level theoretical comparison to the main-text KLPO algorithm. It fits terminal reward with one prompt-and-version normalizer, then applies the response residual to every generated token score. With supplied normalization information it admits single-response gradients, but it does not remove that information requirement as KLPO does (Section 5). The global Gibbs target, regression loss, length weighting, replay results, and update algorithm are collected here; Appendix B gives normalization estimators and Appendix C gives the soft-value interpretation. Fix a collection version and abbreviate $c^* = c^*_{t,k}$ and $Z = Z_{t,k}$.

A.1 The Sequence Target

For KL coefficient $\beta > 0$, the response-level objective at this step is

$$\begin{aligned} J(\pi_\theta) &= \mathbb{E}_{y\sim\pi_\theta}[R(y)] - \beta\,\mathrm{UKL}(\pi_\theta\|\pi^{\mathrm{sampler}}_{\theta_{t-k}}), \\ \mathrm{UKL}(\pi_\theta\|\pi^{\mathrm{sampler}}_{\theta_{t-k}}) &= \sum_y \left[\pi_\theta(y)\log\frac{\pi_\theta(y)}{\pi^{\mathrm{sampler}}_{\theta_{t-k}}(y)} + \pi^{\mathrm{sampler}}_{\theta_{t-k}}(y) - \pi_\theta(y)\right]. \end{aligned} \tag{A.1}$$

For normalized policies, the mass correction vanishes and unnormalized KL equals ordinary KL. The URKL formulation retains the policy normalization constraint.

Theorem A.1 (Sequence Gibbs Improvement)**.** Under the assumptions of Section 2, define $Z(x) = \mathbb{E}_{y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot|x)} e^{R(x,y)/\beta}$ and $c^*(x) = \beta\log Z(x)$. The unique maximizer of Equation (A.1) is

$$\pi^\star_{t,k}(y\mid x) = \frac{\pi^{\mathrm{sampler}}_{\theta_{t-k}}(y\mid x)e^{R(x,y)/\beta}}{Z(x)}. \tag{A.2}$$

For every feasible π_θ,

$$J(\pi_\theta) = c^* - \beta\,\mathrm{KL}(\pi_\theta\|\pi^\star_{t,k}). \tag{A.3}$$

Equivalently, it is the entropic mirror-ascent update

$$\pi^\star_{t,k} = \operatorname*{argmax}_{\pi\in\Delta(\mathcal{Y}_x)} \left\{\langle\pi, R\rangle - \eta^{-1}\,\mathrm{KL}(\pi\|\pi^{\mathrm{sampler}}_{\theta_{t-k}})\right\}, \qquad \eta = \beta^{-1}. \tag{A.4}$$

Subtracting a prompt-dependent baseline $b(x)$ from R changes c^* to $c^* - b$ and leaves the target unchanged.

Proof: Appendix D.8. The target is conditioned on the collection sampler; a finite trainer step need not reach it.

A.2 Fitting the Optimality Condition

Taking logarithms of the Gibbs update yields the condition that SKLPO fits. The response log-ratio decomposes into token terms:

$$\begin{aligned} \ell_{\theta,u}(x,y) &= \log\pi_\theta(y_u\mid x, y_{<u}) - \log\pi^{\mathrm{sampler}}_{\theta_{t-k}}(y_u\mid x, y_{<u}), \\ \ell_\theta(x,y) &= \log\frac{\pi_\theta(y\mid x)}{\pi^{\mathrm{sampler}}_{\theta_{t-k}}(y\mid x)} = \sum_{u=1}^{T}\ell_{\theta,u}(x,y). \end{aligned} \tag{A.5}$$

The response residual is

$$d_\theta(x, y) = \beta \ell_\theta(x, y) - R(x, y) + c^*(x). \tag{A.6}$$

The unweighted regression objective is

$$\begin{aligned} \mathcal{L}^{\text{fix}}_{\text{seq}}(\theta) &= \frac{1}{2\beta} \mathbb{E}_{x\sim\mathcal{D},\, Y\sim\pi^{\text{sampler}}_{\theta_{t-k}}(\cdot|x)}[d_\theta(x, Y)^2] \\ &= \frac{1}{2\beta} \mathbb{E}_{x,Y}\left[\left(\beta \sum_{u=1}^{|Y|} \ell_{\theta,u}(x, Y) - R(x, Y) + c^*(x)\right)^2\right]. \end{aligned} \tag{A.7}$$

Write $\kappa_\theta(x) = \mathrm{KL}(\pi^{\text{sampler}}_{\theta_{t-k}}(\cdot \mid x) \| \pi_\theta(\cdot \mid x))$ for the complete-response KL. Sum the token log-ratios *before* squaring; the reward and normalizer each enter once. The same response residual multiplies every token score. KLPO sequence regression in Section 3 also shares one residual across a response, but its correction is the path sum $\beta \sum_u k_\theta(s_u)$ rather than this prompt normalizer. Together with KLPO token regression, it realizes the local PMD gradient under the stated assumptions; SKLPO instead fits the response Gibbs target.

Theorem A.2 (SKLPO Regression Target)**.** Under the support assumptions of Section 2, $d_\theta = \beta \log(\pi_\theta/\pi^\star_{t,k})$. If $\pi^\star_{t,k}$ is realizable, Equation (A.7) vanishes exactly at $\pi_\theta = \pi^\star_{t,k}$. Within a restricted model class, regression minimizes a squared log-ratio under $\pi^{\text{sampler}}_{\theta_{t-k}}$, whereas regularized return minimizes $\mathrm{KL}(\pi_\theta \| \pi^\star_{t,k})$; their minimizers need not agree.

Proof: Appendix D.8.

Proposition A.3 (Regression and Return Gradients)**.** With responses drawn from $\pi^{\text{sampler}}_{\theta_{t-k}}$ and the sampler and normalizer fixed during differentiation,

$$\nabla_\theta \mathcal{L}^{\text{fix}}_{\text{seq}} = \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[d_\theta \nabla_\theta \log \pi_\theta(y)], \tag{A.8}$$

$$-\nabla_\theta J(\pi_\theta) = \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}\left[\frac{\pi_\theta(y)}{\pi^{\text{sampler}}_{\theta_{t-k}}(y)} d_\theta \nabla_\theta \log \pi_\theta(y)\right]. \tag{A.9}$$

The population gradients agree at $\pi_\theta = \pi^{\text{sampler}}_{\theta_{t-k}}$ and generally differ elsewhere.

Proof: Appendix E. The return gradient carries the multiplier $w_\theta = \pi_\theta/\pi^{\text{sampler}}_{\theta_{t-k}}$. Appendix H.1 gives the action-level comparison.

A.3 Outer Length Weighting

Let $T(x, y) \geq 1$ count generated policy tokens, including termination under the chosen stopping convention. Length weighting changes the relative contribution of responses while keeping the summed log-ratio inside the residual.

Proposition A.4 (Positive Outer Weights)**.** Under the support and realizability assumptions of Theorem A.2, any fixed positive response weight $a(x, y)$ gives $\mathcal{L}_a = (2\beta)^{-1} \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[a\, d^2_\theta]$ the same

unique zero-loss policy $\pi^{\star}_{t,k}$ and normalizer c^{*}. For $a = 1/T$,

$$\begin{aligned}\mathcal{L}^{\mathrm{len}}_{\mathrm{seq}}(\theta) &= \frac{1}{2\beta}\mathbb{E}_{x,Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}}\left[\frac{d_\theta(x,Y)^2}{T(x,Y)}\right],\\ \nabla_\theta\mathcal{L}^{\mathrm{len}}_{\mathrm{seq}} &= \mathbb{E}_{x,Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}}\left[d_\theta\,\frac{1}{T}\sum_{u=1}^{T}\nabla_\theta\log\pi_\theta(y_u\mid x,y_{<u})\right].\end{aligned} \tag{A.10}$$

The sampler law, recorded lengths, and normalizers are held fixed during differentiation.

Proof: Appendix D.8. Each response residual now multiplies its mean token score. Positive weights therefore change how a restricted model class balances approximation error across responses. At equal absolute residual, a response of length 100 contributes ten times as much loss as one of length 1000. The batch reduction is $B^{-1}\sum_i d_i^2/T_i$; dividing $\sum_i d_i^2$ by the total token count is a different reduction.

The accumulated log-ratio can still grow with length. Appendix L gives the gradient bound and explains why the weighted update need not equal the return gradient even at matched policies.

A.4 Sequence Replay across Sampler Versions

For the response-level companion, each example additionally retains its fixed-target normalizer or enough information to fit one.

Proposition A.5 (Version-Conditioned Sequence Update)**.** Holding these quantities fixed, the empirical length-weighted loss and its gradient are

$$\begin{aligned}d_{i,\theta} &= \beta\log\frac{\pi^{\mathrm{trainer}}_\theta(y_i\mid x_i)}{\pi^{\mathrm{sampler}}_{\theta_{t-k_i(t)}}(y_i\mid x_i)} - R_i + c^{*}_{t,k_i(t)}(x_i),\\ \widehat{\mathcal{L}}^{\mathrm{len}}_{\mathrm{seq},t}(\theta) &= \frac{1}{2\beta B}\sum_{i=1}^{B}\frac{d_{i,\theta}^2}{T_i},\\ \nabla_\theta\widehat{\mathcal{L}}^{\mathrm{len}}_{\mathrm{seq},t}(\theta) &= \frac{1}{B}\sum_{i=1}^{B}\frac{d_{i,\theta}}{T_i}\,\nabla_\theta\log\pi^{\mathrm{trainer}}_\theta(y_i\mid x_i).\end{aligned} \tag{A.11}$$

Proof: Appendix D.12. At $\theta=\theta_t$, collection frequencies and response lengths determine each sampler's contribution, without an additional version weight or IS multiplier. This gradient is generally different from on-policy return maximization.

Proposition A.6 (Compatibility across Sampler Versions)**.** For a fixed version, full response support, exact normalization, and realizability give the unique zero-loss target $\pi^{\star}_{t,k}$, with or without positive outer weights. If several versions occur with positive frequency and have full common support, their joint population loss is zero exactly when all Gibbs targets coincide, and the trainer equals that target. For a common reward and β, full-support samplers share a Gibbs target only if their distributions coincide.

Proof: Appendix D.12. Distinct sampler distributions therefore induce competing targets. Their relative frequencies can change the best fit even with unrestricted policy capacity. A finite batch alone does not ensure the population coverage assumed here.

Use each response's $c^{*}_{t,k_i(t)}(x_i)$ or fit a weighted intercept within its prompt–version group. Pooling normalizers across versions changes the residual (Appendix B.4).

A.5 Sequence Update

Algorithm 2 applies the length-weighted loss to completed responses. For response i, $\ell_{i,u}$ is Equation (A.5)'s token log-ratio evaluated with the current trainer and the stored collection version v_i. Collection follows Section 4.

The sequence normalizer may be supplied directly, learned as an intercept, or estimated by weighted centering within each prompt–sampler-version group. Fixed normalizers remain unchanged during reuse; fitted intercepts track the trainer. A one-response group has zero centered loss, so single-rollout fitting requires independent normalization information or a predictor shared across prompts. Appendix B gives the estimators and their assumptions.

Algorithm 2 SKLPO with Each Response's Historical Sampler

Require: Trainer $\pi_{\theta_0}^{\text{trainer}}$, completed-response queue, $\beta > 0$, normalization route
for optimizer steps $t = 0, 1, \ldots$ **do**
  Read B responses $(x_i, y_i, T_i, R_i, v_i)$ and their stored sampler token scores.
  Score with $\pi_{\theta_t}^{\text{trainer}}$; compute $\ell_i = \sum_{u=1}^{T_i} \ell_{i,u}$, $a_i = 1/T_i$, $h_i = R_i - \beta \ell_i$.
  if using a supplied prompt-and-version normalizer **then**
    Set $d_i = \widehat{c}_{v_i}(x_i) - h_i$; keep $\widehat{c}_{v_i}$ detached and fixed during reuse.
  else if using a learned intercept c_ϕ **then**
    Set $d_i = \mathrm{sg}(c_\phi(x_i, v_i)) - h_i$ before updating either model.
  else
    Group by (x_i, v_i) with $G_{x,v} \geq 2$.
    For each group, set $\bar{h}_a = (\sum_j a_j h_j)/(\sum_j a_j)$ and $d_i = \bar{h}_a - h_i$.
  end if
  Update θ_t using the gradient of $\frac{1}{2\beta B} \sum_{i=1}^{B} a_i d_i^2$.
  if using a learned intercept **then**
    Update ϕ by Equation (F.14), using the pre-update h_i and a_i.
  end if
  Publish checkpoints on the synchronization schedule; retain historical sampler scores.
end for

B Normalization and Policy Fitting

SKLPO uses one normalizer $c_{t,k}^*(x)$ per prompt and collection version. Canonical KLPO profiles its state intercept as $\beta k_\theta(s)$; the exact sequence- and token-regression routes realize the same population gradient. Fixed-normalizer token regression is a theoretical alternative. The fixed sequence normalizer is independent of the sampled response and its length. A fitted intercept generally changes with the trainer. All results below condition on one sampler version.

B.1 Normalization Choices

SKLPO does not require a learned value model: normalization information can come from repeated responses or independent statistics. Learning a prompt normalizer is another implementation choice. It introduces an auxiliary scalar predictor, but does not require a token-value critic at every prefix. At the exact sequence target, $c^*(x)$ is the prompt soft value; a fitted intercept $c_\theta(x)$ generally has a different value during training.

Route	Information used	Policy-update schedule
Direct estimate	Same-prompt, same-version rewards or independent success statistics	Estimate $\widehat{c}$; hold it fixed while reusing the data.
Learned normalizer	A shared predictor of success probability or exponentiated reward	Predict $\widehat{c}$ out of sample; hold that prediction fixed during reuse.
Group centering	At least two responses for the same prompt and version	Recompute the weighted group mean after each trainer update.
Learned intercept	A shared predictor of the conditional mean of $R - \beta\ell_\theta$	Alternate policy and intercept updates, using the same outer weights.

Table 1 Sequence normalization choices. The two learned routes add a prompt-conditioned predictor; the other two do not. All four condition on the collection sampler.

Single-rollout *collection* means one response per prompt per collection event; it can still use independent historical data or a predictor shared across prompts. Historical rewards estimate the same normalizer only when their collection policy matches. If a prompt has only one response and no other normalization information, the plug-in and empirical-centering constructions below cannot supply the intended update. Appendix F.7 specifies the learned routes and their update order.

B.2 Fixed Sequence Normalizers

Let $V_{t,k}^{\text{sampler}}(x) = \mathbb{E}_{\pi_{\theta_{t-k}}^{\text{sampler}}} R(x, Y)$ and $A_{t,k}^{\text{sampler}}(x, y) = R(x, y) - V_{t,k}^{\text{sampler}}(x)$.

Lemma B.1 (Sequence Normalization and Plug-In Estimation)**.** The exact prompt normalizer satisfies

$$c^*(x) = V_{t,k}^{\text{sampler}}(x) + \beta\,\mathrm{KL}(\pi_{\theta_{t-k}}^{\text{sampler}}(\cdot \mid x) \| \pi_{t,k}^{\star}(\cdot \mid x)). \tag{B.1}$$

For G independent responses to the same prompt from $\pi_{\theta_{t-k}}^{\text{sampler}}$, define

$$\widehat{c}(x) = \beta \left[\mathrm{logsumexp}_{i=1}^{G} \left(\frac{R(x, y_i)}{\beta} \right) - \log G \right]. \tag{B.2}$$

Then $e^{\widehat{c}/\beta}$ is an unbiased estimator of Z, while $\mathbb{E}\widehat{c} \le c^*$.

Proof: Appendix F.1. A mean-reward baseline omits the KL correction. Smaller β makes the exponential average more sensitive to rare high rewards. Holding $\widehat{c}$ fixed defines a sampled regression target, not an unbiased estimate of the exact-normalizer loss. Appendix F gives the binary-reward form and the effect of data selection.

B.3 Sequence Centering within Each Prompt

For unweighted, unfiltered responses from one sampler, write $h_\theta = R - \beta\ell_\theta$. Let $\kappa_\theta(x)$ denote the complete-response KL defined in Section A.

Proposition B.2 (Profiled SKLPO Regression)**.** At fixed θ, the least-squares prompt intercept is

$$c_\theta(x) = \mathbb{E}_{\pi_{\theta_{t-k}}^{\text{sampler}}}[h_\theta(x, Y)] = \mathbb{E}_{\pi_{\theta_{t-k}}^{\text{sampler}}} R + \beta\,\mathrm{KL}(\pi_{\theta_{t-k}}^{\text{sampler}} \| \pi_\theta). \tag{B.3}$$

The centered residual and loss are

$$\delta_{\mathrm{KL}}(x, y; \theta) = \frac{A^{\mathrm{sampler}}_{t,k}(x, y)}{\beta} - \ell_\theta(x, y) - \kappa_\theta(x). \tag{B.4}$$

$$\begin{aligned} \mathcal{L}_{\mathrm{ctr},\pi^{\mathrm{sampler}}_{\theta_{t-k}}}(\theta) &= \frac{\beta}{2}\mathbb{E}_x\mathbb{E}_{Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot|x)}[\delta_{\mathrm{KL}}(x, Y; \theta)^2] \\ &= \frac{1}{2\beta}\mathbb{E}_x \operatorname{Var}_{Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot|x)}[R(x, Y) - \beta\ell_\theta(x, Y)]. \end{aligned} \tag{B.5}$$

Under full support and realizability, the unique zero-loss policy is $\pi^\star_{t,k}$, and its fitted intercept equals c^*. Away from this target, c_θ generally differs from c^*.

Proof: Appendix F.3.

For comparison with the token objectives, the same residual and loss can be written as

$$\delta_{\mathrm{seq}} = \frac{R - V^{\mathrm{sampler}}_{t,k}(x)}{\beta} - \sum_u \ell_{\theta,u} - \kappa_\theta(x), \qquad \mathcal{L}_{\mathrm{seq}} = \frac{\beta}{2}\mathbb{E}_{x,Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}}[\delta^2_{\mathrm{seq}}]. \tag{B.6}$$

Thus $\delta_{\mathrm{seq}} = \delta_{\mathrm{KL}}$ and $\mathcal{L}_{\mathrm{seq}} = \mathcal{L}_{\mathrm{ctr},\pi^{\mathrm{sampler}}_{\theta_{t-k}}}$. The fixed-normalizer version replaces κ_θ by the KL to the Gibbs target, $\mathrm{KL}(\pi^{\mathrm{sampler}}_{\theta_{t-k}} \| \pi^\star_{t,k})$.

Lemma B.3 (Empirical Centering)**.** For G responses in one prompt–sampler-version group, set $h_i = R_i - \beta\ell_i$ and $\bar{h} = G^{-1}\sum_i h_i$. Then

$$\widehat{\mathcal{L}}_{\mathrm{ctr}} = \frac{1}{2\beta G}\sum_i (h_i - \bar{h})^2 = \frac{1}{4\beta G^2}\sum_{i,j}(h_i - h_j)^2. \tag{B.7}$$

Differentiating through $\bar{h}$ or detaching it gives the same gradient. For fixed θ and i.i.d. responses, the expected empirical loss is $(G-1)/G$ times the conditional population loss.

Proof: Appendix F.4. The centered form costs $O(G)$ after scoring; its pairwise form connects to REBEL (Gao et al., 2024). Recompute $\bar{h}$ at each optimizer step. Pooling sampler versions change the component residuals. Outcome-dependent selection changes the centering measure and can invalidate the KL expression in Equation (B.3); Appendix H.3 gives the corresponding gradient. The empirical KL estimate $-\bar{\ell}$ may be negative; clipping it changes the loss.

Proposition B.4 (Singleton Normalization)**.** For $G = 1$, Equation (B.2) gives $\widehat{c} = R_1$ and hence zero fixed-normalizer residual at $\pi_\theta = \pi^{\mathrm{sampler}}_{\theta_{t-k}}$. Empirical centering of a one-response group gives zero loss at every policy, including with a positive outer length weight.

Proof: Appendix F.4. Fixed-normalizer single-response fitting therefore needs normalization information supplied independently of that response. A shared learned intercept can instead estimate a conditional mean across training examples; it does not compute a singleton group mean. A cross-prompt reward mean cannot replace the prompt normalizer. Appendix H.4 gives signed-feedback and normalizer-error identities.

B.4 Centering with Length Weights

For length-weighted training, profile the intercept with the same outer weight used in the loss.

Proposition B.5 (Weighted Sequence Centering)**.** At a fixed prompt and sampler version, set $a = 1/T$ and $h_\theta = R - \beta\ell_\theta$. The population intercept and profiled loss are

$$c_{\theta,a}(x) = \frac{\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[a\, h_\theta \mid x]}{\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[a \mid x]}, \qquad \mathcal{L}_{\text{ctr},a} = \frac{1}{2\beta}\mathbb{E}_{x,Y\sim\pi^{\text{sampler}}_{\theta_{t-k}}}[a(h_\theta - c_{\theta,a}(x))^2]. \tag{B.8}$$

For a group of G responses, let $a_i = 1/T_i$, $W = \sum_i a_i$, and $\bar{h}_a = W^{-1}\sum_i a_i h_i$. Then

$$\widehat{\mathcal{L}}_{\text{ctr},a} = \frac{1}{2\beta G}\sum_i a_i(h_i - \bar{h}_a)^2 = \frac{1}{4\beta G W}\sum_{i,j} a_i a_j (h_i - h_j)^2. \tag{B.9}$$

The empirical gradient is $G^{-1}\sum_i a_i(\bar{h}_a - h_i)\nabla_\theta \log \pi_\theta(y_i)$ whether $\bar{h}_a$ is differentiated or detached. Under full support and realizability, the unique zero-loss policy remains $\pi^\star_{t,k}$, with intercept c^*.

Proof: Appendix F.6. Recompute the weighted mean at every trainer update. It generally differs from the ordinary mean and the fixed c^*. Direct estimation of c^* in Equation (B.2) still uses the original, unweighted sampler distribution. The unweighted $(G-1)/G$ bias formula does not generally apply to random-length weights.

B.5 Fixed Token Normalizers

Canonical KLPO in Section 3.4 profiles the local intercept rather than evaluating c^*_{tok}; both exact routes in Sections 3.8 and 3.5 preserve that population gradient. The fixed-target alternative uses the log-partition in Equation (3.3). Given exact sampler advantages, it can be computed over next actions or estimated from independent actions at that history; obtaining the advantages themselves requires same-sampler continuation information.

Holding the advantage and normalizer fixed, the token residual, loss, and gradient are

$$\begin{aligned} d^{\text{tok}}_{\theta,u} &= \beta\ell_{\theta,u} - A^{\text{sampler}}_{t,k}(x, y_{<u}, y_u) + c^*_{\text{tok}}(x, y_{<u}), \\ \mathcal{L}^{\text{fix}}_{\text{tok}}(\theta) &= \frac{1}{2\beta}\mathbb{E}_{x,Y\sim\pi^{\text{sampler}}_{\theta_{t-k}}}\sum_u (d^{\text{tok}}_{\theta,u})^2, \\ \nabla_\theta\mathcal{L}^{\text{fix}}_{\text{tok}} &= \mathbb{E}_{x,Y\sim\pi^{\text{sampler}}_{\theta_{t-k}}}\sum_u d^{\text{tok}}_{\theta,u}\nabla_\theta \log \pi_\theta(y_u \mid x, y_{<u}). \end{aligned} \tag{B.10}$$

Under state/action coverage and realizability, this loss vanishes exactly at $\pi_\theta = \pi^+_{\text{tok}}$ on the covered histories, by Theorem D.2.

The fixed c^*_{tok} and the profiled intercept $c_\theta(s) = \beta k_\theta(s)$ agree at the PMD target and generally differ during training. One logged action probability determines the sampled log-ratio, but not a fixed normalizer or the conditional score mean. Both KLPO sequence regression and KLPO token regression use independent auxiliary sampler draws for default MC-KL: sequence regression uses leave-one-out trajectory coefficients with $M \geq 2$, while token regression uses a mean-score correction with $M \geq 1$. Optional TopK-KL uses stored sampler heads; sequence regression needs the scalar KL and its derivative, while token regression needs only its score correction. Optional Binary KL uses only rollout-action probabilities for either route. Full conditionals give the exact identities; independent MC-KL preserves their expected gradients under Proposition 3.7.

B.6 An Equivalent KL-Centered Terminal-Reward Loss

For comparison with the token-regression surrogate, one can differentiate a squared residual with its full local KL. This formulation has the same population gradient under the assumptions below.

Proposition B.6 (Terminal-Reward Token Regression)**.** For a fixed, action-independent baseline $b(s)$ and complete rollouts from the specified sampler, define

$$\widehat{\delta}_u = \frac{R - b(x, y_{<u})}{\beta} - \ell_{\theta,u} - k_\theta(x, y_{<u}), \qquad \mathcal{L}_{\mathrm{MC}} = \frac{\beta}{2}\mathbb{E}_{x,Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}} \sum_u \widehat{\delta}_u^2. \tag{B.11}$$

With exact local KLs, $\mathcal{L}_{\mathrm{MC}} - \mathcal{L}_{\mathrm{tok}}$ is independent of θ. Thus their population gradients agree, including for $b(s) = 0$. This identity differentiates through k_θ and holds the rollout law and baseline fixed.

Proof: Appendix I.5. A prompt baseline shared by all tokens preserves this population gradient, while adding noise and losing the samplewise Bellman telescope. When $b(s) \neq V^{\mathrm{sampler}}_{t,k}(s)$, discarding the KL derivative generally changes the gradient. A baseline fitted on the same rollout requires separate analysis.

The local KL requires the full next-token distributions or a justified estimator using additional conditional samples. Using $\widehat{k}_\theta(x, y_{<u}) = -\ell_{\theta,u}$ from the same sampled action cancels the log-ratio and removes the learning signal. Appendix G specifies the distribution information needed for single-rollout training.

B.7 Scoring and Batch Reuse

Sequence scoring sums the token log-ratios in Equation (A.5), multiplies by β, and adds the reward and normalizer terms to form one response residual. The batch loss averages its square divided by T_i, as in Equation (A.10). Both the likelihood and T_i exclude prompt, padding, and external tool-output tokens and count termination consistently. The log-ratio inside the residual remains a sum; averaging it would change the target.

Algorithms 2 and 1 in Sections A.5 and 3.9 give the SKLPO and KLPO updates on completed responses. Both pair the stored $\pi^{\mathrm{sampler}}_{\theta_{v_i}}$ scores with current trainer scores, retaining each response's collection probabilities and version throughout reuse. Fixed-target SKLPO fitting retains the prompt normalizer; each KLPO route retains its required sampler records and recomputes its regression feedback and correction. Fixed-target token alternatives retain their action targets and state normalizers. Group centering recomputes weighted means within each prompt–sampler-version group after a trainer update; the learned-intercept branch instead updates its auxiliary predictor against pre-update residual targets. Correctly logged collection probabilities eliminate a separate reference-model scoring pass for the log-ratios. For either KLPO route, Binary KL requires only sampled-action scores; TopK-KL additionally requires head records, MC-KL requires independent auxiliary sampled-token records, and full KL requires full sampler conditionals. Resampling MC records for a fresh conditional-unbiased update requires access to the original sampler conditionals.

C Implicit $Q^\star$ Learning

This appendix develops the implicit action-value representation of SKLPO's Gibbs policy from Appendix A. For a fixed collection sampler $\pi^{\mathrm{sampler}}_{\theta_{t-k}}$, let $Q^\star_{\beta,t,k}$ and $V^\star_{\beta,t,k}$ denote the optimal values

regularized toward that sampler. These values generally differ from the unregularized optimum and from the ordinary sampler value $Q_{t,k}^{\text{sampler}}$ used by canonical KLPO. The results below apply to SKLPO and soft-value token regression; local PMD need not recover their global target. They provide theoretical context rather than a value-learning component of the main-text KLPO algorithm.

C.1 The Regularized Bellman Target

Consider a finite-horizon process with finite state and action spaces, bounded rewards, and a positive reference $\pi_{\theta_{t-k}}^{\text{sampler}}(a \mid s)$. Let s include time and the full observed history, and let $P(s' \mid s, a)$ be the environment transition law. We retain the terminal-reward setting: intermediate rewards are zero and $R(S_T)$ is paid at termination. Define

$$V_{\beta,t,k}^{\star}(s) = \max_{\pi} \mathbb{E}_{\pi,P}\left[R(S_T) - \beta \sum_{u<T} \log \frac{\pi(A_u \mid S_u)}{\pi_{\theta_{t-k}}^{\text{sampler}}(A_u \mid S_u)} \,\middle|\, S_0 = s \right]. \tag{C.1}$$

Proposition C.1 (Sampler-Regularized Bellman Equations)**.** In the finite-horizon setting above, the optimal values and policy satisfy

$$\begin{aligned} Q_{\beta,t,k}^{\star}(s,a) &= \mathbb{E}_{S' \sim P(\cdot \mid s,a)} V_{\beta,t,k}^{\star}(S'), \\ V_{\beta,t,k}^{\star}(s) &= \beta \log \sum_{a} \pi_{\theta_{t-k}}^{\text{sampler}}(a \mid s) e^{Q_{\beta,t,k}^{\star}(s,a)/\beta}, \\ \pi_{\beta,t,k}^{\star}(a \mid s) &= \pi_{\theta_{t-k}}^{\text{sampler}}(a \mid s) e^{(Q_{\beta,t,k}^{\star}(s,a) - V_{\beta,t,k}^{\star}(s))/\beta}, \end{aligned} \tag{C.2}$$

with $V_{\beta,t,k}^{\star}(s_T) = R(s_T)$. For a fixed positive sampler, the regularized values approach the unregularized optimal values as $\beta \downarrow 0$.

Proof: Appendix D.9. These are the regularized Bellman equations of Geist et al. (2019) for $\beta\,\mathrm{KL}(\pi \| \pi_{\theta_{t-k}}^{\text{sampler}})$ and the reference-weighted counterpart of the Boltzmann policy in soft Q-learning (Haarnoja et al., 2017). The expectation over P keeps environment outcomes conditioned on the chosen action. At finite β, future KL costs are part of the continuation value. Exponentiating ordinary $Q^{\star}$ omits those costs, whereas exponentiating $Q_{t,k}^{\text{sampler}}$ gives the local PMD update; neither generally solves this Bellman system.

C.2 Soft-Value Token Regression

In deterministic generation, soft continuation values factor the sequence Gibbs target into token conditionals. Regressing these local conditions gives a token objective with the same target as SKLPO. It requires the soft values at each prefix, which score centering with ordinary sampler returns does not supply.

Theorem C.2 (Soft-Value Token Regression)**.** Under the deterministic generation assumptions of Section 2, define

$$F(s) = \beta \log \mathbb{E}_{\pi_{\theta_{t-k}}^{\text{sampler}}}[e^{R/\beta} \mid s], \qquad \pi_{t,k}^{\star}(a \mid s) = \pi_{\theta_{t-k}}^{\text{sampler}}(a \mid s) e^{(F(sa) - F(s))/\beta}. \tag{C.3}$$

Here sa is the successor prefix, $F(x) = c^*(x)$, and terminal $F = R$. The local residuals satisfy

$$\epsilon_u = \frac{F(s_{u+1}) - F(s_u)}{\beta} - \ell_{\theta,u}, \qquad \sum_u \epsilon_u = \frac{R - c^*(x)}{\beta} - \ell_\theta(x,y), \tag{C.4}$$

For fixed exact F and sampler,

$$\mathcal{L}_{\text{soft-tok}} = \frac{\beta}{2}\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}} \sum_u \epsilon_u^2, \qquad \nabla_\theta \mathcal{L}_{\text{soft-tok}} = -\beta \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}} \sum_u \epsilon_u \nabla_\theta \log \pi_\theta(a_u \mid s_u). \tag{C.5}$$

Under coverage and realizability, the unique zero-loss policy is $\pi^\star_{t,k}$, the SKLPO target.

Proof: Appendix I.6. Because $d_\theta = -\beta \sum_u \epsilon_u$, SKLPO squares the sum of local residuals, whereas soft-value token regression sums their squares. The cross terms generally survive in expectation, so the gradients differ. Soft-value token fitting needs F at every prefix; SKLPO needs only R and $F(x) = c^*(x)$. These soft values solve Equation (C.2) in deterministic generation.

The soft-value loss sums over generated policy tokens and averages over responses. Include termination consistently; exclude prompt, padding, and external tool-output tokens. Appendix I.7 distinguishes global token means from response-length weighting.

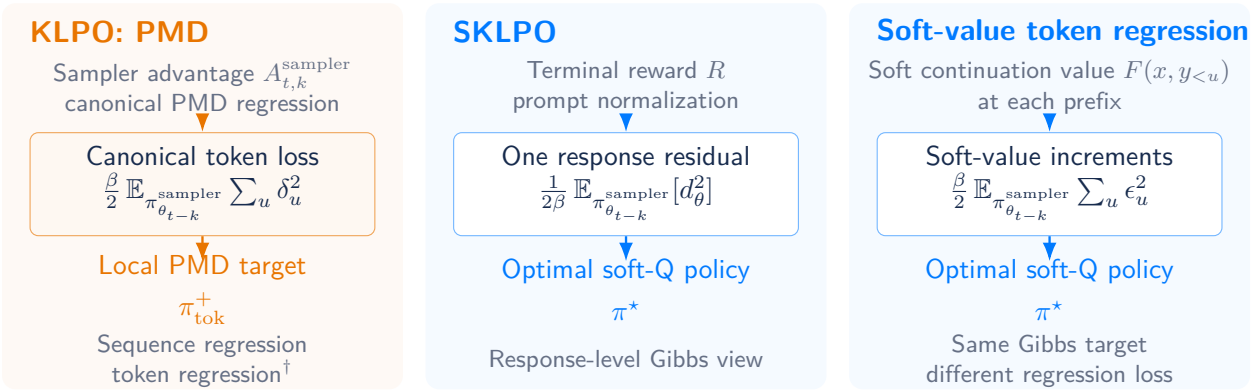

Figure 4 Sequence and token targets. SKLPO and soft-value token regression share the response Gibbs target under exact fitting in deterministic generation. Canonical KLPO is local PMD regression; its sequence- and token-regression realizations have the same exact population gradient under the stated conditions and require no sampler-conditioned critic.

C.3 The Implicit Action-Value Representation

Define the policy's implicit soft advantage by

$$A^{\text{imp}}_\theta(s,a) = \beta \log \frac{\pi_\theta(a \mid s)}{\pi^{\text{sampler}}_{\theta_{t-k}}(a \mid s)}. \tag{C.6}$$

Policy normalization gives $\sum_a \pi^{\text{sampler}}_{\theta_{t-k}}(a \mid s) e^{A^{\text{imp}}_\theta(s,a)/\beta} = 1$. The log-ratio therefore represents action-value differences; recovering absolute Q-values additionally requires a Bellman-consistent state baseline.

Theorem C.3 (Implicit Recovery of Optimal Soft Advantages)**.** Under the deterministic response assumptions of Section 2, the regularized Bellman values are

$$V^\star_{\beta,t,k}(s) = F(s), \qquad Q^\star_{\beta,t,k}(s,a) = F(sa), \qquad F(s) = \beta \log \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[e^{R/\beta} \mid s]. \tag{C.7}$$

Suppose replay covers every feasible response and the Gibbs target is realizable. With exact $c^*(x)$ for SKLPO or exact F for soft-value token regression, any zero-loss fit satisfies, at every covered prefix,

$$A_\theta^{\mathrm{imp}}(s,a) = Q^\star_{\beta,t,k}(s,a) - V^\star_{\beta,t,k}(s) = F(sa) - F(s). \tag{C.8}$$

The result also holds for fixed positive outer weights on the corresponding regression losses.

Proof: Appendix D.10. Theorem C.3 describes an exact fit, not convergence of neural optimization. An intermediate policy's log-ratio need not be Bellman-consistent, and absolute action values still require $Q^\star_{\beta,t,k} = V^\star_{\beta,t,k} + A_\theta^{\mathrm{imp}}$.

With stochastic tools, an unrestricted trajectory tilt can change outcomes outside the policy's control, so the deterministic equivalence can fail. Appendix H.6 gives a one-action counterexample. The Bellman equations remain feasible because they preserve P.

C.4 Policy Regression with an Optimal Q-Function

If optimal regularized Q-values are available, the same regression principle applies directly to action policies, including with stochastic transitions.

Proposition C.4 (Regression against Optimal Soft Q-Values)**.** Hold the reference and exact target values fixed. For a fixed replay law ζ, define

$$\begin{aligned} d_\theta^Q(s,a) &= \beta \log \frac{\pi_\theta(a \mid s)}{\pi^{\mathrm{sampler}}_{\theta_{t-k}}(a \mid s)} - Q^\star_{\beta,t,k}(s,a) + V^\star_{\beta,t,k}(s), \\ \mathcal{L}_Q(\theta) &= \frac{1}{2\beta}\mathbb{E}_{(S,A)\sim\zeta}[d_\theta^Q(S,A)^2], \\ \nabla_\theta \mathcal{L}_Q &= \mathbb{E}_\zeta[d_\theta^Q(S,A)\nabla_\theta \log \pi_\theta(A \mid S)]. \end{aligned} \tag{C.9}$$

Under state/action coverage and realizability, zero loss identifies $\pi^\star_{\beta,t,k}$. Its state normalizer satisfies

$$V^\star_{\beta,t,k}(s) = \mathbb{E}_{a\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot|s)} Q^\star_{\beta,t,k}(s,a) + \beta\,\mathrm{KL}(\pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot \mid s) \| \pi^\star_{\beta,t,k}(\cdot \mid s)). \tag{C.10}$$

In deterministic generation, $d_\theta^Q = -\beta\epsilon_u$, recovering the soft-value token objective.

Proof: Appendix D.11. Appendix H.7 describes a critic extension using sampled Bellman targets. A sampler's terminal return estimates its ordinary continuation value and generally cannot replace the optimal regularized target.

D Proofs of the Regression and Value Results

D.1 The General Optimality-Condition Regression Principle

At either granularity, let $p_0(z \mid h)$ be a fixed positive sampler conditional and $f(h,z)$ a fixed improvement signal on a finite support. For sequences, $(h,z,f) = (x,y,R)$; for token PMD, they are a history, an action, and the sampler advantage. Throughout, $\beta > 0$.

Lemma D.1 (Gibbs Optimality Condition)**.** The unique maximizer of $\mathbb{E}_p f - \beta\,\mathrm{KL}(p\|p_0)$ over normalized p is

$$p_f^\star(z \mid h) = p_0(z \mid h)e^{(f(h,z)-c_f(h))/\beta}, \qquad c_f(h) = \beta \log \mathbb{E}_{z\sim p_0(\cdot|h)} e^{f(h,z)/\beta}. \tag{D.1}$$

The proofs are in Appendix D.7.

Theorem D.2 (Optimality-Condition Regression)**.** Let $r_\theta = \beta \log(p_\theta/p_0) - f + c_f$. For a replay law ν and positive weight a, both fixed during differentiation,

$$\mathcal{L}_f = \frac{1}{2\beta}\mathbb{E}_\nu[a\, r_\theta^2], \qquad \nabla_\theta \mathcal{L}_f = \mathbb{E}_\nu[a\, r_\theta \nabla_\theta \log p_\theta]. \tag{D.2}$$

If ν covers every relevant (h, z) and $p_f^\star$ is realizable, zero loss identifies $p_\theta = p_f^\star$ on that support. The regression gradient contains no importance multiplier.

The proofs are in Appendix D.7. The theorem identifies the exact target, but not an equality of optimization landscapes: regression and regularized return generally have different gradients and may have different minimizers in a restricted model class.

D.2 Critic-Free Token-Regression Gradient

Proof of Theorem 3.3. Fix a visited history s and abbreviate its sampler by q. Let $g_a = \nabla_\theta \log \pi_\theta(a \mid s)$, $m_\theta = \mathbb{E}_q g_a$, $h = R - \beta \log(\pi_\theta(a \mid s)/q(a \mid s))$, and $\bar{h}_\theta(s) = \mathbb{E}_q[h \mid s]$. The action and continuation law is fixed, so

$$\nabla_\theta \frac{\mathrm{Var}_q(h \mid s)}{2\beta} = -\mathbb{E}_q[(h - \mathbb{E}_q[h \mid s])g_a \mid s] = -\mathbb{E}_q[h(g_a - m_\theta) \mid s].$$

Summing over the fixed sampler visitation measure proves Equation (3.15). Since $\mathbb{E}_q[g_a - m_\theta \mid s] = 0$, any fixed action-independent $b(s)$ may be subtracted from h. This baseline must not be fitted to the same sampled action or continuation without a separate argument.

Let $\xi = R - Q_{t,k}^{\text{sampler}}(s, a)$. We have $\mathbb{E}_q[\xi \mid s, a] = 0$ and $\bar{h}_\theta(s) = V_{t,k}^{\text{sampler}}(s) + \beta k_\theta(s)$. Consequently $h - \bar{h}_\theta = \beta\delta + \xi$ and

$$\mathcal{L}_{\text{TR}} - \mathcal{L}_{\text{tok}} = \frac{1}{2\beta}\mathbb{E}_{x,Y\sim\pi_{\theta_{t-k}}^{\text{sampler}}} \sum_u \mathrm{Var}_{\pi_{\theta_{t-k}}^{\text{sampler}}}(R \mid x, y_{\le u}). \tag{D.3}$$

The right side is independent of θ. Lemma 3.2 therefore gives the same minimizers; under coverage and realizability the policy is π_{tok}^+. The token-regression objective need not vanish at this policy because terminal returns have continuation noise.

Differentiating Equation (3.16) with its indicated stop-gradients gives $-h(g_a - m_\theta)$ on each sample. Hence a single complete rollout gives a stochastic estimator of the population gradient when the conditional score mean is supplied independently of its sampled action. Its surrogate value is neither the variance objective nor an empirical estimate of it. Differentiating through h adds an unwanted product-rule term. Replacing h by the terminal reward alone instead gives ordinary score-centered policy gradient and removes the regression feedback.

The fixed-normalizer loss and the profiled canonical objective generally have different gradients away from the PMD target. Score centering provides a critic-free realization of the canonical profiled objective's population gradient; it is not an evaluation of the unknown fixed c_{tok}^*. In particular, the zero-mean identity does not turn this gradient into an importance-corrected current-policy return gradient. □

D.3 Exact Backpropagation Surrogate

Proof of Proposition 3.4. Because $h_{\theta,u}$ and $\pi^{\text{sampler}}_{\theta_{t-k}}$ are detached,

$$\begin{aligned}\nabla^{\text{AD}}_{\theta} L^{\text{bp,TR}}_{\theta,u} &= -h_{\theta,u}\left[\nabla_\theta \log \pi_\theta(a_u \mid s_u) - \sum_v \pi^{\text{sampler}}_{\theta_{t-k}}(v \mid s_u)\nabla_\theta \log \pi_\theta(v \mid s_u)\right] \\ &= -h_{\theta,u}\widetilde{g}_\theta(s_u, a_u).\end{aligned}$$

Taking the sampler expectation gives Equation (3.15); Theorem 3.3 gives the remaining equalities. □

D.4 TopK-KL Score Correction and Approximation Error

At fixed s, take the head H from the stored sampler, with unrenormalized probabilities q_v. Let $T = \mathcal{V} \setminus H$, $\rho = q_{\text{tail}}/p_{\text{tail}}$, and $\widehat{q}_v = q_v$ on H, $\widehat{q}_v = \rho p_v$ on T. Since $\sum_v p_v g_v = 0$,

$$\widehat{m}_\theta(s) := \sum_v \widehat{q}_v g_v = \sum_{v \in H}(q_v - \rho p_v) g_v. \tag{D.4}$$

Detaching the entire coefficient $q_v - \rho p_v$ in Equation (3.37) gives $\nabla_\theta C_\theta = \widehat{m}_\theta = -\nabla_\theta k^K_\theta$ by Equation (3.33). Detaching ρ alone is insufficient. The token-regression route retains the original logged q_{y_u} in $h_{\theta,u}$, even for a sampled action outside H.

With actual rollouts from q, the conditional gradient error of this top-K update is

$$\mathbb{E}_q[-h(g_a - \widehat{m}_\theta) \mid s] - \mathbb{E}_q[-h(g_a - m_\theta) \mid s] = \bar{h}_\theta(s)(\widehat{m}_\theta(s) - m_\theta(s)). \tag{D.5}$$

With a fixed baseline, replace $\bar{h}_\theta$ by $\bar{h}_\theta - b$. The error vanishes for exact centering, including when the modeled and actual conditional tails agree. Finite K does not in general preserve baseline invariance or the exact PMD stationary point. These are approximation statements, not an assertion that the top-K surrogate is the gradient of the exact variance objective.

Tail stabilization. Use float32 or higher precision for log-probabilities and tail masses. With $\epsilon = 10^{-6}$, write $\bar{q}_{\text{tail}} = \max(q_{\text{tail}}, \epsilon)$ and $\bar{p}_{\text{tail}} = \max(p_{\text{tail}}, \epsilon)$. The sequence-regression implementation uses the stabilized scalar

$$k^{K,\epsilon}_\theta = \sum_{v \in H} q_v \log \frac{q_v}{p_v} + \bar{q}_{\text{tail}} \log \frac{\bar{q}_{\text{tail}}}{\bar{p}_{\text{tail}}} \tag{D.6}$$

in both the residual and the differentiable bracket. Away from the clamp boundary, the trainer-tail derivative is zero below the floor; differentiating the literal stabilized scalar preserves its own squared-residual identity. The stabilized expression need not itself be a KL between normalized distributions.

For token regression, following the practical convention of Marek and Ryabinin (2026), use $\bar{q}_{\text{tail}}/\bar{p}_{\text{tail}}$ in the detached head coefficients. When either floor is active, this correction need not equal $-\nabla k^{K,\epsilon}_\theta$. Both conventions are further approximations; their derivative correspondence in Equation (3.33) assumes inactive floors. For a full-vocabulary record, use full KL or the exact score sum directly, with no tail bucket or ratio. Mask padding before reduction and preserve the actual sampling transforms. Neither route uses IS weighting.

D.5 Independent MC-KL and Leave-One-Out Feedback

Proof of Proposition 3.7. Condition on a complete rollout Y and a trainer θ fixed independently of the auxiliary draws. All visited histories, rollout actions, rewards, and sampler conditionals are then fixed. Write $g_u(v) = \nabla_\theta \log p_\theta(v \mid s_u)$. For each auxiliary column, define

$$Z_j = \sum_u \log \frac{q(v_{u,j} \mid s_u)}{p_\theta(v_{u,j} \mid s_u)}, \qquad G_j = \sum_u [g_u(a_u) - g_u(v_{u,j})].$$

The conditional means are $\mathbb{E}_{\mathrm{MC}} Z_j = \sum_u k_\theta(s_u)$, $\mathbb{E}_{\mathrm{MC}} G_j = \sum_u \widetilde{g}_{\theta,u}$, and $\mathbb{E}_{\mathrm{MC}} D_j = D_\theta$, where $D_j = R - \beta(\sum_u \ell_{\theta,u} + Z_j)$. Token regression multiplies its independent MC score estimate by a rollout-measurable $h_{\theta,u}$, giving the full-KL sample gradient in conditional expectation directly.

For sequence regression, the auxiliary columns are independent conditional on Y. Thus $\overline{D}_{-j}$ is independent of G_j, and

$$\mathbb{E}_{\mathrm{MC}} \left[-\frac{1}{M} \sum_j \overline{D}_{-j} G_j \right] = -D_\theta \sum_u \widetilde{g}_{\theta,u}.$$

Since $\nabla_\theta D_j = -\beta G_j$, differentiating the ordered cross-product sum in Equation (3.29) gives exactly this sample estimator before taking expectations. For $j \neq l$, independence gives $\mathbb{E}_{\mathrm{MC}} D_j D_l = D_\theta^2$, proving the loss identity. Averaging over sampler rollouts completes the proof under the original full-KL assumptions. □

Why the direct plug-in square is biased. Let $\overline{D} = M^{-1} \sum_j D_j$. With independent draws across prefixes,

$$\mathbb{E}_{\mathrm{MC}} \frac{\overline{D}^2}{2\beta} = \frac{D_\theta^2}{2\beta} + \frac{\beta}{2M} \sum_u \mathrm{Var}_{v \sim q(\cdot \mid s_u)} \left[\log \frac{q(v \mid s_u)}{p_\theta(v \mid s_u)} \right]. \tag{D.7}$$

The added variance generally depends on the trainer, so differentiating the plug-in square is not an unbiased full-KL gradient. For $M \geq 2$, the implemented U-statistic subtracts an unbiased estimate of that term:

$$\widehat{\mathcal{L}}_{\mathrm{SR,MC}} = \frac{1}{2\beta} \left[\overline{D}^2 - \frac{1}{M-1} \frac{1}{M} \sum_j (D_j - \overline{D})^2 \right].$$

Neither this scalar nor an individual MC-KL estimate must be nonnegative. The leave-one-out coefficient is $\overline{D}_{-j} = \overline{D} - (D_j - \overline{D})/(M-1)$, avoiding an explicit quadratic enumeration of sample pairs. Independent groups could also estimate the product, but the leave-one-out construction uses all M draws symmetrically. At $M = 1$, this implementation has no independent residual estimate and rejects the sequence route; the token route remains valid.

Independence during reuse. All conditional identities above hold at a trainer fixed independently of the auxiliary records. If the trainer is updated using a fixed set of those records, conditioning on that updated trainer can destroy their original IID law. Recompute current log-probabilities for empirical-surrogate reuse, or draw fresh auxiliaries from the same historical sampler for the conditional-unbiased interpretation. Sampling from the current trainer, deduplicating IDs, clamping negative estimates, or reusing the rollout action changes the estimator. Probability tensors alone cannot certify this collection contract.

D.6 The Predicted KL Budget

Proof of Proposition 3.10. Condition on the realized batch, optimizer state, proposal D_t, and probe histories $\mathcal{P}_t$. For one history, let $p_\alpha(v) = \pi_{\theta_t+\alpha D_t}(v \mid s)$ and $p = p_0$. Normalization gives

$$K_s(0) = 0, \qquad K_s'(0) = -\sum_v p(v)\, \partial_\alpha \log p_\alpha(v)|_0 = 0.$$

Differentiating $\sum_v p_\alpha(v) = 1$ twice yields the Fisher identity

$$K_s''(0) = -\sum_v p(v)\, \partial_\alpha^2 \log p_\alpha(v)\Big|_0 = \sum_v p(v)\, [\partial_\alpha \log p_\alpha(v)|_0]^2 . \tag{D.8}$$

For temperature-scaled softmax logits,

$$\partial_\alpha \log p_\alpha(v)|_0 = \mathcal{T}^{-1}\left(\dot\zeta_v - \sum_w p(w)\dot\zeta_w\right).$$

The quadratic coefficient is therefore the stated vocabulary variance divided by $2\mathcal{T}^2$. Averaging over histories gives both the JVP expression and $\frac{1}{2} D_t^\top F_{\mathcal{P}_t} D_t$, where

$$F_{\mathcal{P}_t} = \frac{1}{|\mathcal{P}_t|} \sum_{s\in\mathcal{P}_t} \sum_v p_v(s) g_{\theta_t}(s,v) g_{\theta_t}(s,v)^\top.$$

Bounded third derivatives give the Taylor remainder. For positive finite $q(D_t)$, Equation (3.43) implies $\alpha_t^2 q(D_t) = \min\{\alpha_{\max}^2 q(D_t), \delta_{\text{step}}\} \le \delta_{\text{step}}$. This bounds the quadratic prediction only. □

The vocabulary expectation remains valid when the proposal depends on the sampled actions in the same batch: the proof conditions on that proposal and sums over all vocabulary entries at each probe history. Replacing this sum by the same rollout action would not in general give an unbiased estimate of the conditional quadratic form. The JVP propagates all updated parameter blocks jointly, retaining their cross terms in $D_t^\top F_{\mathcal{P}_t} D_t$.

To see why the incremental budget does not control the historical-sampler KL, let u denote the displacement from a sampler to the current trainer in a common local Fisher approximation. The combined quadratic cost is

$$\tfrac{1}{2}(u+\alpha_t D_t)^\top F (u+\alpha_t D_t) = \tfrac{1}{2} u^\top F u + \tfrac{1}{2}\alpha_t^2 D_t^\top F D_t + \alpha_t u^\top F D_t. \tag{D.9}$$

The cross term can have either sign. Subtracting an estimated sampler–trainer KL from a total budget would omit this term; it is not a KL triangle inequality. Step 10 instead budgets the incremental current-trainer KL explicitly.

D.7 The Common Regression Principle

Proof of Lemma D.1 and Theorem D.2. Fix h and omit it from the notation. Set $Z_f = \sum_z p_0(z) e^{f(z)/\beta}$ and $c_f = \beta \log Z_f$. The distribution $p_f^\star = p_0 e^{(f-c_f)/\beta}$ is positive and normalized. For any normalized p,

$$\mathbb{E}_p f - \beta\, \mathrm{KL}(p\|p_0) = c_f - \beta\, \mathrm{KL}(p\|p_f^\star).$$

Nonnegativity of KL, with equality only at $p = p_f^\star$, proves the lemma. The residual satisfies $r_\theta = \beta \log(p_\theta/p_f^\star)$. Thus $p_f^\star$ gives zero loss; positive replay mass and weights force every residual to vanish at any zero-loss fit. Finally, $\nabla_\theta r_\theta = \beta \nabla_\theta \log p_\theta$. Differentiating the squared loss with the replay law and weights fixed gives Equation (D.2). □

D.8 Sequence Targets and Positive Weights

Proof of Theorems A.1 and A.2. For normalized policies the mass correction in UKL is zero. Apply Lemma D.1 with $(p_0, f) = (\pi^{\text{sampler}}_{\theta_{t-k}}, R)$ to obtain Equations (A.2)–(A.3). Equivalently, stationarity under $\sum_y \pi(y) = 1$ gives $R(y) - \beta \log(\pi(y)/\pi^{\text{sampler}}_{\theta_{t-k}}(y)) = \lambda$, and normalization fixes $\lambda = c^*$. Equation (A.4) uses $\eta^{-1} = \beta$. Replacing R by $R - b(x)$ multiplies Z by $e^{-b(x)/\beta}$, so the normalized target is unchanged.

Taking the logarithm of Equation (A.2) gives $d_\theta = \beta \log(\pi_\theta/\pi^\star_{t,k})$. Theorem D.2 then identifies the unique zero-loss response distribution. Equation (A.3) gives the reverse-KL criterion for regularized return; Equation (A.7) gives the squared log-ratio criterion for regression. Sharing a zero in the full simplex does not equate their projections onto a restricted model class. □

Proof of Proposition A.4. At $\pi^\star_{t,k}$, every residual is zero. Conversely, positivity of the sampler and a forces every residual to vanish at zero weighted loss. The weight therefore leaves both the target and its normalizer unchanged. With $a = 1/T$ fixed, differentiation gives

$$\nabla_\theta \mathcal{L}_a = \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[a\, d_\theta \nabla_\theta \log \pi_\theta(Y)] = \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}\left[d_\theta \frac{1}{T} \sum_u \nabla_\theta \log \pi_\theta(y_u \mid x, y_{<u})\right].$$

This is Equation (A.10). □

D.9 Regularized Bellman Values

Proof of Proposition C.1. At a nonterminal state, condition on the first action and successor state in Equation (C.1). Once the optimal continuation values are fixed by backward induction, the remaining maximization is

$$\max_{\pi(\cdot|s)} \sum_a \pi(a \mid s) \left[\mathbb{E}_P V^\star_{\beta,t,k}(S') - \beta \log \frac{\pi(a \mid s)}{\pi^{\text{sampler}}_{\theta_{t-k}}(a \mid s)}\right].$$

Lemma D.1 gives the value and policy in Equation (C.2), starting from the terminal boundary $V^\star_{\beta,t,k} = R$. This maximization changes the action law and retains P.

The unregularized values satisfy $Q^\star(s,a) = \mathbb{E}_P V^\star(S')$ and $V^\star(s) = \max_a Q^\star(s,a)$, with the same terminal boundary. For a fixed positive reference, let $p_{\min} = \min_{s,a} \pi^{\text{sampler}}_{\theta_{t-k}}(a \mid s) > 0$ over feasible pairs and let H bound the remaining horizon. The unregularized optimal value bounds the regularized value from above. Evaluating a deterministic unregularized optimal policy in the regularized objective gives the lower bound

$$V^\star(s) - \beta H \log(1/p_{\min}) \le V^\star_{\beta,t,k}(s) \le V^\star(s).$$

Thus the state values converge as $\beta \downarrow 0$; taking the fixed transition expectation gives convergence of the action values. □

D.10 Implicit Soft-Advantage Recovery

Proof of Theorem C.3. For deterministic transitions, $Q^\star_{\beta,t,k}(s,a) = V^\star_{\beta,t,k}(sa)$. Starting from terminal $F = R$, backward induction gives

$$e^{F(s)/\beta} = \sum_a \pi^{\text{sampler}}_{\theta_{t-k}}(a \mid s) e^{F(sa)/\beta} = \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[e^{R/\beta} \mid s].$$

Hence $V^\star_{\beta,t,k} = F$, $Q^\star_{\beta,t,k}(s,a) = F(sa)$, and the product of the optimal token conditionals telescopes to the response Gibbs distribution.

For SKLPO, zero loss and coverage imply $\pi_\theta(y \mid x) = \pi^\star_{t,k}(y \mid x)$ on every feasible response. Their token conditionals agree at every positive-mass prefix. For soft-value token regression, each zero residual gives those conditionals directly. Substitution in the Bellman policy proves Equation (C.8). Fixed positive outer weights preserve these zero-residual arguments. □

D.11 Regression against Optimal Soft Q-Values

Proof of Proposition C.4. The Bellman policy gives $d^Q_\theta = \beta \log(\pi_\theta / \pi^\star_{\beta,t,k})$. Theorem D.2 identifies the zero-loss conditional policy and its gradient. Taking a sampler expectation of $\beta \log(\pi^\star_{\beta,t,k} / \pi^{\text{sampler}}_{\theta_{t-k}}) = Q^\star_{\beta,t,k} - V^\star_{\beta,t,k}$ yields Equation (C.10). Under deterministic transitions, Equation (C.7) gives $d^Q_\theta = \beta\ell_\theta - F(sa) + F(s) = -\beta\epsilon_u$. □

D.12 Sampler Compatibility and Batch Gradients

Proof of Proposition A.6. For each component, $d_\theta = \beta \log(\pi^{\text{trainer}}_\theta / \pi^\star_{t,k})$. Positive sampling mass and outer weights force that residual to vanish on its support at zero loss. A positive sum of nonnegative component losses is zero exactly when every component loss is zero. A shared trainer must therefore equal every component target.

If full-support samplers p_1, p_2 have the same Gibbs target for a common reward and β, then $p_1(y)e^{R(y)/\beta}/Z_1 = p_2(y)e^{R(y)/\beta}/Z_2$ for every y. Their ratio is the constant Z_1/Z_2; normalization makes it one. Thus the samplers coincide. □

Proof of Propositions A.5 and 4.1. For SKLPO regression, all collection scores, lengths, and fixed normalizers are held constant. Thus $\nabla_\theta d_{i,\theta} = \beta \nabla_\theta \log \pi^{\text{trainer}}_\theta(y_i \mid x_i)$, and differentiation gives the stated weighted batch gradient. For KLPO token regression, the stop-gradients in Equation (3.16) give $-h_{i,u,\theta}\widetilde{g}_{i,\theta}$ on each sample. Apply Theorem 3.3 separately to each collection version, then average with its fixed frequency. For KLPO sequence regression, apply Proposition 3.6 separately to each version and differentiate its stopped-gradient surrogate. This proves both expected mixture-gradient identities under their respective assumptions; arbitrary outcome-dependent replay selection need not preserve it. □

E RPG–URKL and Gradient Identities

We compare unweighted SKLPO with RPG on unfiltered responses from a fixed historical sampler $\pi^{\text{sampler}}_{\theta_{t-k}}$, off-policy relative to the trainer. Equation (A.8) gives the unweighted SKLPO gradient, and Section A.3 gives the length-weighted version. Write $w_\theta = e^{\ell_\theta} = \pi_\theta / \pi^{\text{sampler}}_{\theta_{t-k}}$. For normalized $\pi^{\text{sampler}}_{\theta_{t-k}}$, the fully differentiable URKL surrogate in Zhang et al. (2026, Proposition 3.6) is

$$\mathcal{L}_{\text{RPG}}(\theta) = \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[-w_\theta R + \beta(w_\theta \log w_\theta - w_\theta)]. \tag{E.1}$$

It equals $-J(\pi_\theta) - \beta$. No clipping is applied in the identities below.

Proof of Proposition A.3. Differentiating (A.7) gives (A.8). For RPG, differentiating (E.1) yields

$$\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[w_\theta(\beta \log w_\theta - R)\nabla_\theta \log \pi_\theta].$$

Adding c^* leaves this expectation unchanged because $\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[w_\theta \nabla_\theta \log \pi_\theta] = \nabla_\theta \sum_y \pi_\theta(y) = 0$. □

For a common response, residual, and score, $g_{\text{RPG}} = e^{\ell_\theta} g_{\text{SKLPO}}$. SKLPO's coefficient d_θ is affine in the accumulated log-ratio; RPG additionally multiplies by the product of token ratios, e^{ℓ_θ}. At $\pi_\theta = \pi^{\text{sampler}}_{\theta_{t-k}}$, the population gradients agree, although different baseline estimates can produce different finite-sample gradients.

With outer weight $1/T$, the corresponding sample identity is $g_{\text{RPG}} = w_\theta T\, g^{\text{len}}_{\text{SKLPO}}$. Thus the matched-policy equality above applies to unweighted regression. Multiplying the weighted loss by $\text{sg}(w_\theta T)$ recovers the same RPG gradient, but the training algorithm does not use that multiplier.

A stop-gradient importance weight recovers the RPG gradient:

$$\mathcal{L}_{\text{sg}}(\theta) = \frac{1}{2\beta}\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[\text{sg}(w_\theta) d_\theta^2], \qquad \nabla_\theta \mathcal{L}_{\text{sg}} = -\nabla_\theta J(\pi_\theta). \tag{E.2}$$

This is an autodifferentiation surrogate, not an equality of scalar objectives. Differentiating the weight as well would add

$$\frac{1}{2\beta}\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[w_\theta d_\theta^2 \nabla_\theta \log \pi_\theta(y)]. \tag{E.3}$$

Changing c^* to any fixed prompt baseline in (E.2) still gives the same population RPG gradient. The unweighted regression does not have this baseline invariance: its zero-loss target depends on the correct normalizer.

	RPG–URKL	Unweighted SKLPO with exact c^*
Criterion, up to constants	$\beta\,\text{KL}(\pi_\theta \Vert \pi^\star_{t,k})$	$\frac{\beta}{2}\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[\log^2(\pi_\theta/\pi^\star_{t,k})]$
Gradient weight on $\pi^{\text{sampler}}_{\theta_{t-k}}$ data	$w_\theta d_\theta$	d_θ
Unrestricted optimum	$\pi^\star_{t,k}$	$\pi^\star_{t,k}$
Neural approximation	Optimizes regularized return	Fits the optimality condition

Table 2 SKLPO removes the current-policy importance weight. The objectives share an unrestricted optimum but generally have different gradients and best approximations within a model class.

F Normalizer Identities and Estimation

F.1 The Normalizer as a KL Divergence

Proof of Lemma B.1. Taking the reference expectation of $\beta \log(\pi^\star_{t,k}/\pi^{\text{sampler}}_{\theta_{t-k}}) = R - c^*$ gives Equation (B.1). For the estimator, $e^{\widehat{c}/\beta} = G^{-1}\sum_i e^{R_i/\beta}$ is unbiased for Z. Jensen's inequality gives $\mathbb{E}\widehat{c} \le \beta \log Z = c^*$. □

For centered rewards $A^{\text{sampler}}_{t,k} = R - V^{\text{sampler}}_{t,k}$, the normalizer is purely a KL term:

$$c^*_A(x) := \beta \log \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[e^{A^{\text{sampler}}_{t,k}(x,Y)/\beta}] = c^*(x) - V^{\text{sampler}}_{t,k}(x) = \beta\,\text{KL}(\pi^{\text{sampler}}_{\theta_{t-k}} \Vert \pi^\star_{t,k}). \tag{F.1}$$

Writing $\kappa^*(x) = \mathrm{KL}(\pi^{\mathrm{sampler}}_{\theta_{t-k}} \| \pi^\star_{t,k})$, the fixed-target residual becomes

$$-\frac{d_\theta(x,y)}{\beta} = \frac{A^{\mathrm{sampler}}_{t,k}(x,y)}{\beta} - \ell_\theta(x,y) - \kappa^*(x). \tag{F.2}$$

This eliminates the response partition function in the same way that BPO eliminates the local partition function in its PMD optimality condition (Song et al., 2026, Section 3.1). The identity still depends on the unknown optimum. Substituting the current π_θ for $\pi^\star_{t,k}$ produces the profiled loss below and generally changes the intercept from c^*. For large β, $c^* = V^{\mathrm{sampler}}_{t,k} + \mathrm{Var}_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}}(R)/(2\beta) + O(\beta^{-2})$, quantifying the correction omitted by a mean-reward baseline.

F.2 Binary Rewards

For rewards taking values r_- and r_+, let $\rho_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}}(x)$ be the success probability under $\pi^{\mathrm{sampler}}_{\theta_{t-k}}$. Then

$$c^*(x) = \beta \log \left[(1 - \rho_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}}(x)) e^{r_-/\beta} + \rho_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}}(x) e^{r_+/\beta} \right]. \tag{F.3}$$

This includes $0/1$ and negative-failure rewards. A success-rate estimate gives a plug-in normalizer. The rate must correspond to the prompt and collection sampler; a global average or an EMA over changing policies generally estimates a different quantity.

F.3 A Computable KL Form via Profiling

Proof of Proposition B.2. For $h_\theta = R - \beta\ell_\theta$, least squares gives the intercept $c_\theta = \mathbb{E}_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}} h_\theta$ in Equation (B.3). Since $\kappa_\theta = -\mathbb{E}_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}} \ell_\theta$, the residual in Equation (B.4) is $(h_\theta - \mathbb{E}_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}} h_\theta)/\beta$. Substitution gives the variance loss in Equation (B.5). For sampling under the sampler policy, the fixed-normalizer and centered losses differ by

$$\mathcal{L}_{\mathrm{SKLPO},\pi^{\mathrm{sampler}}_{\theta_{t-k}}} - \mathcal{L}_{\mathrm{ctr},\pi^{\mathrm{sampler}}_{\theta_{t-k}}} = \frac{1}{2\beta} \mathbb{E}_x[(c^*(x) - c_\theta(x))^2]. \tag{F.4}$$

They share the same zero-loss policy: zero variance makes $R - \beta\ell_\theta$ constant across responses, and normalization fixes that constant to c^*. Since $\mathbb{E}_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}} \delta_{\mathrm{KL}} = 0$ for each prompt, the derivative of κ_θ cancels after expectation:

$$\nabla_\theta \mathcal{L}_{\mathrm{ctr}} = -\beta \mathbb{E}_x \mathbb{E}_{Y \sim \pi^{\mathrm{sampler}}_{\theta_{t-k}}(\cdot|x)} [\delta_{\mathrm{KL}}(x, Y; \theta) \nabla_\theta \log \pi_\theta(Y)]. \tag{F.5}$$

The KL form therefore uses no multiplicative importance weight in either its loss or gradient. □

For a group from $\pi^{\mathrm{sampler}}_{\theta_{t-k}}$, $\widehat{V} = \bar{R}$ and $\widehat{\kappa}_\theta = -\bar{\ell}$ give $\widehat{\delta}_i = (R_i - \bar{R})/\beta - (\ell_i - \bar{\ell})$. This recovers Equation (B.7). The finite-group bias statement there assumes fixed parameters and independent samples; it is not a generalization guarantee after fitting the batch.

F.4 Empirical Centering and Singletons

Proof of Lemma B.3. Expanding the pairwise square gives $\sum_{i,j}(h_i - h_j)^2 = 2G\sum_i(h_i - \bar{h})^2$, proving Equation (B.7). In differentiation, every term containing $\nabla_\theta \bar{h}$ is multiplied by $\sum_i(h_i - \bar{h}) = 0$, so detaching the mean gives the same gradient. At fixed θ, independence gives $\mathbb{E}[G^{-1}\sum_i(h_i - \bar{h})^2] = (G-1)\operatorname{Var}(h_\theta)/G$. □

Proof of Proposition B.4. For $G = 1$, the log-mean-exp estimator is $\widehat{c} = \beta \log e^{R_1/\beta} = R_1$. At matched policies $\ell_\theta = 0$, so $d_\theta = 0$. Empirical centering instead sets $\bar{h} = h_1$ at every policy; a weighted singleton also has $\bar{h}_a = h_1$. Their centered residuals and losses are therefore identically zero. □

F.5 Data Selection and the Normalizer

Fix a sampler $\pi^{\text{sampler}}_{\theta_{t-k}}$ and let ν denote its response law after outcome-dependent selection. Unfiltered collection satisfies $\nu = \pi^{\text{sampler}}_{\theta_{t-k}}$ even when $k > 0$ and the current trainer differs from the sampler. Selection may break this equality. For exact c^*, consider

$$\mathcal{L}_{\text{SKLPO},\nu}(\theta) = \frac{1}{2\beta}\mathbb{E}_{Y\sim\nu}[d_\theta(Y)^2]. \tag{F.6}$$

If ν is positive on the full support of $\pi^{\text{sampler}}_{\theta_{t-k}}$ and the target is realizable, this loss has the same unique zero-loss policy $\pi^\star_{t,k}$. It requires no importance weight to estimate *its own* gradient. Its best approximation can differ from that of the sampler-conditioned loss $\mathcal{L}_{\text{SKLPO},\pi^{\text{sampler}}_{\theta_{t-k}}}$ under model misspecification, and finite data do not ensure coverage.

Estimating c^* from ν is a separate issue. For this estimation problem, a general importance-sampling identity is

$$Z = \mathbb{E}_{Y\sim\nu}\left[\frac{\pi^{\text{sampler}}_{\theta_{t-k}}(Y)}{\nu(Y)}e^{R(Y)/\beta}\right]. \tag{F.7}$$

Equation (F.7) concerns normalizer estimation; the SKLPO loss and algorithm do not use this weight. Unfiltered data from the corresponding sampler require no correction, whereas an uncorrected exponential average under a different ν estimates a different normalizer. An alternative is to profile the intercept under ν, giving $c_{\theta,\nu} = \mathbb{E}_\nu[R - \beta\ell_\theta]$ and a ν-variance loss. Under the same support and realizability assumptions, this loss identifies $\pi^\star_{t,k}$ without a density-ratio weight, with finite-sample fitting behavior that depends on ν.

F.6 Weighted Profiling

Proof of Proposition B.5. Fix a prompt and sampler version. Let ν be the possibly selected response law, let $a(y) > 0$ be fixed, and set $Z_a = \mathbb{E}_\nu a$, $\nu_a(y) = a(y)\nu(y)/Z_a$, and $c_{\theta,a} = \mathbb{E}_{\nu_a}h_\theta$. Completing the square gives

$$\mathbb{E}_\nu[a(c - h_\theta)^2] = \mathbb{E}_\nu[a(h_\theta - c_{\theta,a})^2] + Z_a(c - c_{\theta,a})^2. \tag{F.8}$$

Consequently the profiled loss is $Z_a \operatorname{Var}_{\nu_a}(h_\theta)/(2\beta)$. The pairwise variance identity under ν_a gives Equation (B.9) for an empirical group. The reweighted law changes the regression measure; the sampler in $\ell_\theta = \log(\pi_\theta/\pi^{\text{sampler}}_{\theta_{t-k}})$ and the fixed Gibbs normalizer remain those of the original target.

For $g_\theta = \nabla_\theta \log \pi_\theta(Y)$, differentiation gives

$$\begin{aligned} \nabla_\theta \mathcal{L}_{\mathrm{ctr},a} &= -\mathbb{E}_\nu[a(h_\theta - c_{\theta,a})g_\theta] \\ &= -Z_a \mathbb{E}_{\nu_a}[h_\theta(g_\theta - \mathbb{E}_{\nu_a} g_\theta)]. \end{aligned} \tag{F.9}$$

The intercept derivative vanishes because $\mathbb{E}_\nu[a(h_\theta - c_{\theta,a})] = 0$. An ordinary, unweighted intercept generally lacks this cancellation for the weighted loss. Positive a preserves the support, so zero weighted variance still identifies the original Gibbs target under full coverage and realizability. □

F.7 Learned Normalizers for Single Rollouts

Fix a collection version $v = t-k$ and β. A predictor takes the prompt and collection-policy information as input, written (x, v); its prediction must describe that sampler rather than the current trainer. One response per prompt supplies one training example. Sharing parameters across prompts permits generalization, but does not make a single observed reward an exact conditional expectation. We describe two fixed-normalizer estimators and a learned-intercept alternative. These are implementation choices, not additional assumptions of the sequence objective.

Binary rewards: predict success probability. For $R \in \{r_-, r_+\}$, train $p_\phi(x, v) \in (0, 1)$ with the Bernoulli negative log-likelihood

$$\mathcal{L}_p(\phi) = \mathbb{E}_{x,Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}}[-z \log p_\phi(x, v) - (1 - z) \log(1 - p_\phi(x, v))], \qquad z = \mathbf{1}\{R = r_+\}. \tag{F.10}$$

The unrestricted conditional minimizer is the sampler success probability. Substitute its prediction into Equation (F.3):

$$\widehat{c}_\phi(x, v) = \beta \log \left[(1 - p_\phi(x, v))e^{r_-/\beta} + p_\phi(x, v)e^{r_+/\beta}\right]. \tag{F.11}$$

Evaluate the two-term sum in log space. A single success/failure label can update ϕ; it does not justify setting p_ϕ to that label before scoring the same response. The predictor estimates ordinary prompt success probability, from which the soft normalizer is computed.

General bounded rewards: predict an exponential moment. Let M be a known upper reward bound and set $q(Y) = e^{(R(x,Y)-M)/\beta} \in (0, 1]$. A positive predictor $m_\phi(x, v)$ can be trained by

$$\mathcal{L}_m(\phi) = \mathbb{E}_{x,Y\sim\pi^{\mathrm{sampler}}_{\theta_{t-k}}}[(m_\phi(x, v) - q(Y))^2], \qquad \widehat{c}_\phi(x, v) = M + \beta \log m_\phi(x, v). \tag{F.12}$$

Its unrestricted conditional minimizer is $\mathbb{E}_{\pi^{\mathrm{sampler}}_{\theta_{t-k}}}[q(Y) \mid x]$, which gives c^*. The reward shift bounds the regression labels without changing that normalizer; small β can still make rare high rewards decisive. Regressing c_ϕ directly against R would learn mean reward instead. Both Equations (F.10) and (F.12) use unfiltered rewards under the specified sampler, without the policy loss's $1/T$ weight. Outcome-dependent selection, length weighting, or pooling versions generally changes the conditional quantity being learned. A numerical floor on m_ϕ also introduces an approximation.

Using a fixed prediction in policy updates. Train the predictor on separate records, then freeze a snapshot before scoring the policy-training responses. Compute $\widehat{c}_\phi(x, v)$ from that snapshot, detach it from the policy computation graph, and retain the same prediction while reusing those responses. Newly observed rewards may train a later predictor snapshot. For replayed data, the independence

premise in Appendix H.4 requires separate estimation data or held-out folds: the predictor used for a response must not have been fitted on that response. Merely detaching a prediction or evaluating it before the current optimizer step does not undo earlier fitting on the same record. Estimation and generalization errors still affect the policy update. These routes learn a prompt-level value or normalizer; they avoid a token-value critic, not all auxiliary value estimation.

Learn an intercept jointly with the policy. Instead of estimating the fixed c^*, parameterize the intercept in weighted regression. With separate parameters θ and ϕ, let

$$\mathcal{J}(\theta, \phi) = \frac{1}{2\beta}\mathbb{E}_{x,Y\sim\pi^{\text{sampler}}_{\theta_{t-k}}}[a(Y)(\beta\ell_\theta - R + c_\phi(x, v))^2], \qquad a(Y) = 1/T(Y). \tag{F.13}$$

At fixed θ, unrestricted conditional minimization over c_ϕ gives $c_{\theta,a}$ in Equation (B.8). A practical stochastic update first computes $h_i = R_i - \beta\ell_i$ and $c_\phi(x_i, v_i)$ before either parameter update. The policy step treats c_ϕ as fixed; the auxiliary step minimizes

$$\widehat{\mathcal{L}}_c(\phi) = \frac{1}{2\beta B}\sum_i a_i\big(c_\phi(x_i, v_i) - \text{sg}(h_i)\big)^2 \tag{F.14}$$

using the pre-update h_i. These are the two partial gradients of $\mathcal{J}$ at the pre-update parameters. Keep auxiliary gradients out of the policy parameters; a small head on fixed prompt features is one possible parameterization. Refresh the intercept as the trainer changes, rather than freezing it for the life of the batch. One response per prompt gives a stochastic update of this joint objective, but an imperfect predictor does not give the exactly profiled policy gradient. This is joint regression, not the independent fixed-normalizer estimator above.

Under the same full-support and realizability assumptions, a zero-loss joint fit satisfies $R - \beta\ell_\theta = c_\phi(x, v)$ on the response support; normalization then gives $c_\phi = c^*$ and $\pi_\theta = \pi^\star_{t,k}$ for that sampler. Finite singleton data need not identify this solution: an overly flexible predictor can memorize each prompt's observed residual. A separate free intercept for every response cancels its residual altogether. The learned route therefore depends on conditional generalization and optimization; it is not empirical centering of one response.

G Implementation Details

Algorithms 2 and 1 in the main text give the update steps. This section specifies their normalization, scoring, and conditional-distribution requirements.

G.1 SKLPO Update

The trainer consumes completed responses without a collection barrier. In the supplied-normalizer branch, $\widehat{c}_{v_i}$ estimates $c^*_{t,k_i(t)}$ for each response's collection policy and stays fixed across trainer updates. Group means and learned intercepts instead track the current trainer. Stored denominators and lengths stay fixed in every branch. Correct collection scores avoid an additional reference-model scoring pass. Both the log-ratio sum and T_i count only generated policy tokens, with a consistent termination convention; they exclude prompt, padding, and external tool outputs.

For unfiltered responses to one prompt from one sampler, Equation (B.2) estimates the normalizer without length weights. Selection or pooling across versions requires a separate justification because

a pooled estimate generally fails to recover each component normalizer. Group centering recomputes weighted means within prompt–version groups after every trainer update. Each group supplies an intercept, and the final loss averages over all B responses rather than over groups. With one response per prompt, choose an independently supplied normalizer or the shared learned-intercept branch described in Appendix F.7; do not compute a singleton group mean.

Temperature, truncation, quantization, routing, and prefill/decode differences can make $\pi_{\theta_v}^{\text{sampler}}$ differ from $\pi_{\theta_v}^{\text{trainer}}$ even at identical parameters. The denominator uses the actual sampler probability, not a reconstructed trainer probability at the same weights. Removing importance weights does not restore support lost through truncation or admission.

G.2 KLPO Regression Updates

Algorithm 1 separates the gradient route from the KL estimator. The default is token regression + MC-KL; sequence regression is the alternative route.

For the default MC-KL, gather current trainer log-probabilities at the independent auxiliary IDs and retain their actual historical sampler log-probabilities in arrays of shape $[B, T, M]$. Repeated IDs are valid. Token regression subtracts their mean score; sequence regression uses leave-one-out trajectory coefficients from Equation (3.28), with $M \geq 2$. The reported sequence regression scalar is the U-statistic in Equation (3.29), not the square of the mean residual. It may be negative and is not clamped. Resample independently from the same historical sampler at each adaptive update for a fresh conditional-unbiased gradient; fixed-record reuse is an empirical surrogate.

Optional KLPO sequence regression + TopK-KL stores the sampler's head IDs and original probabilities, plus the sampled action's actual log-probability. Recompute k^K and D_θ^K on every reuse, and differentiate Equation (3.32) with only the residual coefficient detached. Keep gradients through both the head and tail contributions to the KL, subject to the literal stabilization in Equation (D.6). This is the sample gradient of the selected squared-residual objective.

Optional KLPO sequence regression + Binary KL uses only sampled-action records and recomputes k^{bin}, $\widehat{D}_\theta$, and ω. Differentiate Equation (3.36) with both $\widehat{D}_\theta$ and ω detached. Optional KLPO token regression + Binary KL instead detaches the per-token $h_{\theta,u}$ and ω in Equation (3.39). No group normalization or BPO-style residual linearization is applied.

For exact trajectory regression, use the full local KL

$$k_{i,\theta}(s) = \sum_{a \in \mathcal{V}} \pi_{\theta_{v_i}}^{\text{sampler}}(a \mid s) \left[\log \pi_{\theta_{v_i}}^{\text{sampler}}(a \mid s) - \log \pi_\theta^{\text{trainer}}(a \mid s) \right]. \tag{G.1}$$

Keep its derivative in Equation (3.24). For KLPO token regression + TopK-KL, recompute $h_{i,u} = R_i - \beta \ell_{i,u}$ and the correction in Equation (3.37); full conditionals instead give Equation (3.16). Token regression need not evaluate the KL scalar. Rollout-action probabilities, optional head records, and terminal rewards remain fixed during reuse; fresh MC auxiliary records use the same historical sampler. Neither route requires a prompt normalizer, local c^* estimate, value predictor, or same-prompt response group.

The full-KL sequence regression and full-conditional token-regression routes have the same population gradient under the deterministic-transition assumptions in the main text. Independent MC-KL preserves this equality in expectation over its auxiliary draws. TopK-KL and Binary KL generally lose it, including comparisons between routes using the same approximation. Using $-\ell_{i,u}$ from the rollout action as the entire full-KL estimate cancels the log-ratio and removes learning; it is neither an independent MC-KL draw nor TopK-KL or Binary KL.

The exact routes fit local PMD. Soft-value regression remains Equation (C.5); for stochastic tools its targets must satisfy Equation (C.9). The same-sampler terminal return generally cannot replace an optimal regularized soft-Q target.

H Additional Off-Policy Results

H.1 The Local Importance Multiplier

Write $g_\theta(s,a) = \nabla_\theta \log \pi_\theta(a \mid s)$. For a token from response i, the actual behavior conditional and trainer-to-sampler ratio are

$$\nu_i(a \mid s) = \pi^{\text{sampler}}_{\theta_{t-k_i(t)}}(a \mid s), \qquad \rho_{t,i}(s,a) = \frac{\pi^{\text{trainer}}_{\theta_t}(a \mid s)}{\pi^{\text{sampler}}_{\theta_{t-k_i(t)}}(a \mid s)}. \tag{H.1}$$

Proposition H.1 (Local Gradient Comparison)**.** At a fixed history s, use the exact local PMD residual $d_{\theta,s}(a)$ from Equation (B.10), with the collection sampler ν_i as reference. For $\rho_\theta = \pi_\theta/\nu_i$, the sample contributions to the negative local regularized-return gradient and the regression gradient are

$$g_{\text{IS}}(s,a) = \rho_\theta(s,a) d_{\theta,s}(a) g_\theta(s,a), \qquad g_{\text{KLPO}}(s,a) = d_{\theta,s}(a) g_\theta(s,a). \tag{H.2}$$

Their expectations under $a \sim \nu_i(\cdot \mid s)$ give the respective gradients.

Writing $\Delta = \log \rho_{t,i}$, IS multiplies the contribution by e^{Δ}. KLPO has no such multiplier or IS clipping threshold. Older checkpoints may differ further from the trainer, and longer trajectories contain more opportunities for an outlying token ratio. Outcome-dependent admission can change the action distribution required by the return-gradient identity.

FlashREINFORCE multiplies its token ratio by a batch advantage; the length-weighted SKLPO response update uses d_i/T_i on each token score, with no ratio multiplier. Both average scores within a response, but their residuals differ. Appendix H.2 gives the comparison.

Proof of Proposition H.1. At a fixed history, the conditional regression loss and gradient are

$$\begin{aligned} d_{\theta,s}(a) &= \beta \log \frac{\pi_\theta(a \mid s)}{\pi^{\text{sampler}}_{\theta_{t-k}}(a \mid s)} - A^{\text{sampler}}_{t,k}(s,a) + c^*_{\text{tok}}(s), \\ \mathcal{L}^{\text{fix}}_{\text{tok}}(\theta; s) &= \frac{1}{2\beta} \mathbb{E}_{a \sim \pi^{\text{sampler}}_{\theta_{t-k}}(\cdot \mid s)}[d_{\theta,s}(a)^2], \\ \nabla_\theta \mathcal{L}^{\text{fix}}_{\text{tok}}(\theta; s) &= \mathbb{E}_{a \sim \pi^{\text{sampler}}_{\theta_{t-k}}(\cdot \mid s)}[d_{\theta,s}(a) g_\theta(s,a)]. \end{aligned} \tag{H.3}$$

For the local return $J_s = \mathbb{E}_{\pi_\theta} A^{\text{sampler}}_{t,k} - \beta \, \text{KL}(\pi_\theta \| \nu_i)$, differentiation and the identity $\mathbb{E}_{\pi_\theta} g_\theta = 0$ give $-\nabla_\theta J_s = \mathbb{E}_{\pi_\theta}[d_{\theta,s} g_\theta]$. Changing the action measure to ν_i introduces $\rho_\theta = \pi_\theta/\nu_i$ and proves Equation (H.2). □

H.2 Comparison with FlashREINFORCE

Let $\nu_i = \pi^{\text{sampler}}_{\theta_{t-k_i(t)}}$ be the behavior policy that generated response i and $g_{i,u} = \nabla_\theta \log \pi_\theta(y_{i,u} \mid x_i, y_{i,<u})$. FlashREINFORCE's policy-loss gradient, with its batch advantage A_i, sequence mask m_i, and behavior

probabilities held fixed, is

$$\widehat{g}_{\text{Flash}} = -\frac{1}{B}\sum_i \frac{m_i A_i}{T_i} \sum_u \underbrace{\frac{\pi_\theta(y_{i,u} \mid x_i, y_{i,<u})}{\nu_i(y_{i,u} \mid x_i, y_{i,<u})}}_{\rho_{i,u}\text{: token IS}} g_{i,u}. \tag{H.4}$$

Both methods use the actual collection probability. FlashREINFORCE places its ratio in a multiplicative weight; SKLPO uses its logarithm in the response regression residual.

Equation (A.11) gives $\widehat{g}^{\text{len}}_{\text{SKLPO}} = B^{-1}\sum_i (d_i/T_i)\sum_u g_{i,u}$. Both updates average token scores within a response. The SKLPO residual is the complete-response optimality residual, not FlashREINFORCE's batch advantage.

Equations (H.3) and (H.2) isolate the multiplier at a fixed history using a local regularized residual. FlashREINFORCE's batch advantage defines a different residual.

H.3 Centering under the Actual Behavior Distribution

Let ν denote the response distribution for a specified sampler version, allowing outcome-dependent selection as in Appendix F. Unweighted profiling under ν gives an off-policy score identity; Equation (F.9) gives its weighted counterpart. At a fixed prompt and sampler version, write

$$\begin{aligned} h_\theta &= R - \beta \log(\pi_\theta/\pi^{\text{sampler}}_{\theta_{t-k}}), \quad c_{\theta,\nu} = \mathbb{E}_\nu h_\theta, \\ \mathcal{L}_{\text{ctr},\nu} &= \frac{1}{2\beta}\operatorname{Var}_\nu(h_\theta), \qquad\qquad g_\theta = \nabla_\theta \log \pi_\theta(Y). \end{aligned} \tag{H.5}$$

Its gradient can be written in either form

$$\begin{aligned} \nabla_\theta \mathcal{L}_{\text{ctr},\nu} &= -\mathbb{E}_\nu[(h_\theta - c_{\theta,\nu})g_\theta] \\ &= -\mathbb{E}_\nu[h_\theta(g_\theta - \mathbb{E}_\nu g_\theta)]. \end{aligned} \tag{H.6}$$

Differentiating the variance and using the zero mean of the centered residual gives both forms. The second connects response-level SKLPO regression to score centering (Marek and Ryabinin, 2026, Equation (4)) under the actual replay distribution. A prompt-dependent additive reward offset cancels from the gradient. The centering mean must match the sampling measure: $c_{\theta,\nu}$ is generally not $\mathbb{E}_\nu R + \beta\,\mathrm{KL}(\pi^{\text{sampler}}_{\theta_{t-k}}\|\pi_\theta)$, since that KL expression requires sampling from $\pi^{\text{sampler}}_{\theta_{t-k}}$. Neither form in (H.6) is generally an unbiased on-policy return gradient.

H.4 Single Rollouts and Signed Feedback

With a fixed exact response normalizer $c^*_{t,k}(x)$, each completed response gives an SKLPO regression gradient. For two reward levels with $0 < \rho_{\pi^{\text{sampler}}_{\theta_{t-k}}}(x) < 1$, the initial residual at $\pi_\theta = \pi^{\text{sampler}}_{\theta_{t-k}}$ satisfies

$$d_+ = c^*_{t,k} - r_+ < 0, \qquad d_- = c^*_{t,k} - r_- > 0. \tag{H.7}$$

Gradient descent therefore supplies positive likelihood feedback for successes and negative feedback for failures, without a token-value critic. Later, the log-ratio term moderates this feedback as the policy approaches its target. At the exact response Gibbs target, the success odds satisfy

$$\frac{\rho^*_{t,k}}{1-\rho^*_{t,k}} = \frac{\rho_{\pi^{\text{sampler}}_{\theta_{t-k}}}}{1-\rho_{\pi^{\text{sampler}}_{\theta_{t-k}}}} e^{(r_+ - r_-)/\beta}. \tag{H.8}$$

The signs hold at the matched-policy starting point for one component and need not hold for every stale response at the current trainer. Likewise, the odds identity characterizes the target, which a finite shared-trainer update need not attain.

An independently estimated prompt success rate, additional same-prompt, same-version data, or a learned prompt normalizer could approximate $c^*_{t,k}$, with estimation errors. A cross-prompt reward mean does not supply the required soft normalizer. Condition on a prompt-only estimate $\widehat{c} = c^*_{t,k} + \varepsilon(x)$, obtained independently of the response being scored and held fixed during policy updates. Writing ν for the response distribution at this fixed sampler version, its conditional gradient error is

$$\nabla_\theta \mathcal{L}_{\nu,\widehat{c}} - \nabla_\theta \mathcal{L}_{\nu,c^*_{t,k}} = \varepsilon(x)\mathbb{E}_\nu[g_\theta \mid x], \tag{H.9}$$

which need not vanish off-policy. For outer weight $1/T$, the error instead equals $\varepsilon(x)\mathbb{E}_\nu[g_\theta/T \mid x]$. It need not vanish even at $\nu = \pi_\theta$, since $\mathbb{E}_{\pi_\theta}[g_\theta/T] = \nabla_\theta \mathbb{E}_{\pi_\theta}[1/T]$. Empirical within-prompt centering with one sample gives zero loss. SKLPO can therefore use asynchronous single-rollout collection, provided its normalization estimator is supplied. FlashREINFORCE's batch reward baseline serves a different objective.

History coverage and trajectory length. Removing token IS leaves the distribution of stored histories unchanged. Their sampling weights can affect a restricted model's best fit, and hard filters can remove the coverage required by Proposition A.6. A fixed positive outer weight such as $a(x, y) = 1/|y|$ preserves each realizable component target for nonempty responses, while changing gradients and potentially the mixed-version best fit. Averaging the log-ratio *inside* the residual changes the target equation itself. Response-level feedback from a terminal failure also leaves the responsible intermediate tool action unidentified.

H.5 Feasible Updates in Interactive Environments

A full-history action policy can realize the response Gibbs target on the reference trajectory support when tools are deterministic and initial conditions are fixed. Tool outputs enter the conditioning history without contributing policy likelihood factors. Common stochastic transition factors also cancel from a trajectory likelihood ratio, but this cancellation alone does not ensure feasibility: the unrestricted Gibbs tilt may change observation probabilities outside the policy's control.

A statewise regression gives a separate route for stochastic environments. At a stored history s, take the local advantage $A^{\text{sampler}}_{t,k}(s, a)$ and PMD target from (3.4). For any covered behavior action distribution $\nu(\cdot \mid s)$, profiling

$$\min_{\pi,c} \ \mathbb{E}_{a\sim\nu(\cdot|s)} \left[\beta \log \frac{\pi(a \mid s)}{\pi^{\text{sampler}}_{\theta_{t-k}}(a \mid s)} - A^{\text{sampler}}_{t,k}(s, a) + c(s) \right]^2 \tag{H.10}$$

has the same realizable local PMD target, without action importance weights, by the zero-residual argument. The update fits a feasible action distribution and permits covered off-policy state sampling. It requires continuation values for the same sampler as its denominator. A complete return collected under that sampler estimates its ordinary $Q^{\text{sampler}}_{t,k}$; a return from another version generally does not. Neither such return is an optimal regularized soft-Q target in general. Behavior centering does not correct cross-version continuation mismatch or extend the deterministic BPO telescoping identity to stochastic tools.

H.6 A Stochastic-Tool Counterexample

Consider a forced action followed by a fair random terminal reward in $\{0, 1\}$. Every feasible policy has value $1/2$ and zero policy KL. The unrestricted trajectory normalizer is instead $\beta \log[(1 + e^{1/\beta})/2] > 1/2$, by strict Jensen inequality. The tilt increases the success probability even though the only action cannot alter it. The regularized Bellman equation preserves the environment law and gives value $1/2$.

H.7 Learning Soft Bellman Targets

A critic extension could fit Q_ψ to one-step targets $\widehat{q} = V_{\bar{\psi}}(S')$, with $V_{\bar{\psi}}(s) = \beta \log \sum_a \pi^{\text{sampler}}_{\theta_{t-k}}(a \mid s) e^{Q_{\bar{\psi}}(s,a)/\beta}$ at nonterminal states and $V_{\bar{\psi}}(s_T) = R(s_T)$. With $\bar{\psi}$ frozen, a transition sampled conditional on (s, a) gives an unbiased Bellman target regardless of the behavior policy, provided replay selection preserves the transition law. This extension requires action values or a justified estimator of the action log-sum-exp, as well as a separate convergence analysis under function approximation. A sampler's terminal return estimates $Q^{\text{sampler}}_{t,k}$ and generally cannot substitute for this optimal regularized continuation target.

I Token Objectives and BPO Identities

I.1 Token PMD and Conditional Centering

Proof of Proposition 3.1. Apply Lemma D.1 at each history with $f = A^{\text{sampler}}_{t,k}$ and the normalizer defined in Equation (3.3). This gives the normalized maximizing policy in Equation (3.4). Its log-ratio then yields

$$\begin{aligned}\beta \,\mathrm{KL}(\pi^{\text{sampler}}_{\theta_{t-k}}(\cdot \mid s) \| \pi^{+}_{\text{tok}}(\cdot \mid s)) &= \mathbb{E}_{a \sim \pi^{\text{sampler}}_{\theta_{t-k}}(\cdot|s)}[c^*_{\text{tok}}(s) - A^{\text{sampler}}_{t,k}(s, a)] \\ &= c^*_{\text{tok}}(s),\end{aligned}$$

where the last equality uses the zero conditional mean of the exact sampler advantage. The identity $d^{\text{tok}}_{\theta,u} = \beta \log(\pi_\theta / \pi^{+}_{\text{tok}})$ and Theorem D.2 give both the regression gradient and the unique zero-loss target on covered histories. □

Proof of Lemma 3.2. Profiling the local squared residual gives

$$c_\theta(s) = \mathbb{E}_{a \sim \pi^{\text{sampler}}_{\theta_{t-k}}}[A^{\text{sampler}}_{t,k}(s, a) - \beta \log(\pi_\theta(a \mid s)/\pi^{\text{sampler}}_{\theta_{t-k}}(a \mid s))] = \beta k_\theta(s). \tag{I.1}$$

The zero conditional means of the advantage and of $\ell_{\theta,u} + k_\theta(x, y_{<u})$ give $\mathbb{E}[\delta_u \mid x, y_{<u}] = 0$. Differentiating the centered squared loss yields $-\beta \mathbb{E} \sum_u \delta_u [\nabla_\theta \log \pi_\theta(y_u \mid x, y_{<u}) + \nabla_\theta k_\theta(x, y_{<u})]$. The term with $\nabla_\theta k_\theta(x, y_{<u})$ vanishes after conditioning on $(x, y_{<u})$, proving Equation (3.11). At zero loss, the local optimality residual is constant over actions. Normalization fixes the intercept to c^*_{tok} and the policy to π^{+}_{tok}. □

I.2 Response and Token Gradients

The centered response residual multiplies every token score:

$$\nabla_\theta \mathcal{L}_{\text{seq}} = -\beta \mathbb{E}_{x,Y\sim\pi^{\text{sampler}}_{\theta_{t-k}}} \left[\delta_{\text{seq}} \sum_u \nabla_\theta \log \pi_\theta(y_u \mid x, y_{<u}) \right]. \tag{I.2}$$

Equation (3.11) gives the token PMD gradient. Its local KL derivative cancels in population because $\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[\delta_u \mid x, y_{<u}] = 0$.

Both gradients omit multiplicative importance weights. The first uses a shared response residual; the second uses one residual per token.

I.3 Conditional Orthogonality

Proof of Theorem 3.5. For $v < u$, δ_v is determined by $(x, y_{<u})$. Hence

$$\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[\delta_v \delta_u] = \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[\delta_v \mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}(\delta_u \mid x, y_{<u})] = 0.$$

Padding with zero residuals after termination and expanding the square proves Equation (3.18). Under deterministic token transitions, $A^{\text{sampler}}_{t,k}(x, y_{<u}, y_u) = V^{\text{sampler}}_{t,k}(x, y_{\le u}) - V^{\text{sampler}}_{t,k}(x, y_{<u})$, so the advantage sum is $R - V^{\text{sampler}}_{t,k}(x)$. This proves Equation (3.19) and the exact BPO loss equality, for every smooth parameterized policy. A fixed prompt weight can be applied to both sides. The trajectory form uses only the terminal reward and prompt value; the token form uses intermediate advantages. Finite batches retain cross terms. □

I.4 KLPO Sequence Regression and Approximate KL Surrogates

Proof of Proposition 3.6. Fix a prompt and collection sampler. Write $V = V^{\text{sampler}}_{t,k}(x)$ and $B_\theta = \sum_u [\ell_{\theta,u} + k_\theta(s_u)]$. Each summand has conditional sampler mean zero, so $\mathbb{E}[B_\theta \mid x] = 0$ for bounded complete responses. Thus

$$D_\theta = \beta \delta_{\text{BPO}} + V, \qquad \mathbb{E}[\delta_{\text{BPO}} \mid x] = 0.$$

Squaring and taking expectations yields $\mathbb{E}[D^2_\theta \mid x] = \beta^2 \mathbb{E}[\delta^2_{\text{BPO}} \mid x] + V^2$. Theorem 3.5 proves Equation (3.22). Replacing V by $V - b(x)$ gives the baseline statement. All sampler quantities and b stay fixed while differentiating. Together with Theorem 3.3, this proves Equation (3.25) for every smooth parameterized trainer, without requiring realizability. Coverage and realizability are needed only for the common-target claim. □

TopK-KL.

Proof of Proposition 3.8. At a fixed history, $\nabla p_{\text{tail}} = -\sum_{v\in H} p_v g_v$. Differentiating Equation (3.30) with sampler records fixed yields

$$\nabla k^K_\theta = -\sum_{v\in H} q_v g_v - \frac{q_{\text{tail}}}{p_{\text{tail}}} \nabla p_{\text{tail}} = -\sum_{v\in H} (q_v - \rho p_v) g_v.$$

The chain rule then gives

$$\nabla_\theta \frac{(D_\theta^K)^2}{2\beta} = -D_\theta^K \sum_u [g_{\theta,u} + \nabla_\theta k_{\theta,u}^K],$$

which is exactly the autodifferentiation gradient of Equation (3.32). □

For a nonempty tail $T = \mathcal{V} \setminus H$, the KL chain rule gives

$$k_\theta - k_\theta^K = q_{\text{tail}} \, \mathrm{KL}(q(\cdot \mid T) \| p(\cdot \mid T)) \geq 0. \tag{I.3}$$

Thus $\mathbb{E}_{a\sim q}[\log(p_a/q_a) + k_\theta^K] = k_\theta^K - k_\theta$ need not vanish. The exact martingale, baseline-invariance, and sequence/token equivalence proofs do not extend to a general finite head. A singleton tail gives full KL; a singleton head consisting of the sampled action gives the binary partition. These identities concern unfloored probabilities. For KLPO token regression + Binary KL, the derivative below supplies the same corrected score, weighted by the local $h_{\theta,u}$ instead of a trajectory residual.

Binary KL.

Proof of Proposition 3.9. For one sampled action, keep q fixed. Differentiating the Bernoulli KL gives

$$\nabla_\theta k^{\text{bin}} = \left[-q + \frac{(1-q)p}{1-p}\right] \nabla_\theta \log p.$$

Adding $\nabla_\theta \log p$ gives $(1-q)/(1-p)$ times that score. Therefore

$$\nabla_\theta \frac{\widehat{D}_\theta^2}{2\beta} = -\widehat{D}_\theta \sum_u \omega_{\theta,u} \nabla_\theta \log p_u,$$

which is the stated stopped-gradient surrogate. This is an identity for the binary objective, not for the full-KL objective. In general, $\mathbb{E}_{a\sim q}[\log(p_a/q_a) + k^{\text{bin}}(a)] \neq 0$, so the baseline-invariance proof does not extend to binary KL.

For example, take $q = (1/3, 1/3, 1/3)$, $p = (1/2, 1/3, 1/6)$, and a constant reward. The full-KL log-ratio has zero sampler mean after adding its KL, but the binary-centered log-ratio has a negative mean: the binary KL is strictly below the full KL for at least one sampled action. A reward offset can therefore change the binary objective's gradient. Vocabulary size two is a special case in which the binary partition is the full vocabulary. □

I.5 Terminal-Reward Regression

Proof of Proposition B.6. Write $\xi_u = R - Q_{t,k}^{\text{sampler}}(x, y_{<u}, y_u)$ and $m_u = V_{t,k}^{\text{sampler}}(x, y_{<u}) - b(x, y_{<u})$. Then $\widehat{\delta}_u = \delta_u + (\xi_u + m_u)/\beta$. Conditional on $(x, y_{\leq u})$, ξ_u has zero mean; conditional on $(x, y_{<u})$, δ_u has zero mean. Expanding the square therefore gives

$$\mathcal{L}_{\text{MC}} - \mathcal{L}_{\text{tok}} = \frac{1}{2\beta} \mathbb{E}_{x, Y\sim\pi_{\theta_{t-k}}^{\text{sampler}}} \sum_u \left[\mathrm{Var}_{\pi_{\theta_{t-k}}^{\text{sampler}}}(R \mid x, y_{\leq u}) + (V_{t,k}^{\text{sampler}}(x, y_{<u}) - b(x, y_{<u}))^2 \right]. \tag{I.4}$$

Every term on the right depends only on the fixed sampler and baseline, so its θ derivative is zero. The identity differentiates the full residual, including k_θ. If $b \neq V_{t,k}^{\text{sampler}}$, then $\mathbb{E}[\widehat{\delta}_u \mid x, y_{<u}] = m_u/\beta$ need not vanish, and the KL derivative cannot generally be dropped. □

I.6 Soft-Value Token Factorization

Proof of Theorem C.2. Under deterministic generation, the Gibbs probability of a prefix is proportional to its sampler probability times $\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[e^{R/\beta} \mid s]$. Taking the ratio of the masses of sa and s gives Equation (C.3). Summing $F(x, y_{\le u}) - F(x, y_{<u})$ telescopes to $R - F(x) = R - c^*(x)$, proving Equation (C.4). With F fixed, $\nabla_\theta \epsilon_u = -\nabla_\theta \log \pi_\theta(y_u \mid x, y_{<u})$, giving Equation (C.5). Positive replay mass forces all local residuals to vanish at zero loss; the factorization then identifies $\pi^\star_{t,k}$. □

I.7 Token Reductions and Masking

Equations (3.10) and (3.18) use token sums followed by an average over responses. At the population level, a global token mean divides by the fixed constant $\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}|Y|$. Averaging each response's loss by its own length instead reweights trajectories and can invalidate the cross-term cancellation. Replacing a sum of log-ratios by a length average changes the residual itself. Include termination consistently and exclude prompt, padding, and external tool-output tokens from policy likelihoods.

I.8 BPO Equation (25) and Score Centering

Song et al. (2026, Equation (25)) derive the per-token gradient before approximating the full KL. Write $g_\theta(s,a) = \nabla_\theta \log \pi_\theta(a \mid s)$ and $g_{i,u,\theta} = \nabla_\theta \log \pi_\theta(y_{i,u} \mid x, y_{i,<u})$, and let $\phi(x)$ be a fixed prompt weight. With $\eta = 1/\beta$, their token contribution to the trajectory-loss gradient is

$$\begin{aligned} \nabla_\theta \mathcal{L}^{\text{BPO}}_{i,u} &= -\phi(x)\beta\delta_{\text{BPO}}(x, y_i)\big[g_{i,u,\theta} + \nabla_\theta k_\theta(x, y_{i,<u})\big] \\ &\approx -\phi(x)\big[R(x, y_i) - V^{\text{sampler}}_{t,k}(x)\big]\big[g_{i,u,\theta} + \nabla_\theta k_\theta(x, y_{i,<u})\big]. \end{aligned} \tag{I.5}$$

The first line is exact; the second freezes the residual factor at $\pi_\theta = \pi^{\text{sampler}}_{\theta_{t-k}}$. The derivative inside the brackets is a centered score, since $\pi^{\text{sampler}}_{\theta_{t-k}}$ is fixed:

$$\begin{aligned} \widetilde{g}_\theta^{\,\pi^{\text{sampler}}_{\theta_{t-k}}}(s,a) &:= g_\theta(s,a) + \nabla_\theta k_\theta(s) \\ &= g_\theta(s,a) - \mathbb{E}_{a' \sim \pi^{\text{sampler}}_{\theta_{t-k}}(\cdot|s)} g_\theta(s,a'), \qquad \mathbb{E}_{a \sim \pi^{\text{sampler}}_{\theta_{t-k}}(\cdot|s)} \widetilde{g}_\theta^{\,\pi^{\text{sampler}}_{\theta_{t-k}}}(s,a) = 0. \end{aligned} \tag{I.6}$$

Equation (I.6) is the operation in Marek and Ryabinin (2026, Equation (4)), with differentiated policy π_θ and sampling distribution ν, here equal to $\pi^{\text{sampler}}_{\theta_{t-k}}$. BPO's linearized gradient weights this centered score by reward; its exact gradient retains the current trajectory residual. The full-KL correction is additive and uses no importance ratio. Binary-KL approximation, smoothing, and clipping subsequently change the update.

For KLPO, (3.11) can equivalently use $\widetilde{g}_\theta^{\,\pi^{\text{sampler}}_{\theta_{t-k}}}$, because $\mathbb{E}_{\pi^{\text{sampler}}_{\theta_{t-k}}}[\delta_u \mid x, y_{<u}] = 0$. The corresponding response-level correction centers the complete-response score by its prompt-conditional mean. These cancellations hold in population; local KL derivatives need not vanish on an individual trajectory. If the actual behavior is $\nu \neq \pi^{\text{sampler}}_{\theta_{t-k}}$, score centering in Equation (4) uses the mean under ν. A KL correction based on $\pi^{\text{sampler}}_{\theta_{t-k}}$ alone does not enforce zero mean under ν. Even with the matching mean, centering does not change the sampling distribution into π_θ or recover its importance-weighted policy gradient in general.

J Detailed Comparisons with Prior Objectives

Self-Play Preference Optimization. Write $p = \pi_{\theta_{t-k}}^{\text{sampler}}$ for one frozen sampler and define its response win rate

$$w_p(x, y) = \mathbb{E}_{Y' \sim p(\cdot|x)}[\mathbb{P}(y \succ Y' \mid x)].$$

SPPO's exact update and regression (Wu et al., 2024, Equations (4.2)–(4.4)), with $\eta = 1/\beta$, become

$$\begin{aligned} c_w(x) &= \beta \log \mathbb{E}_{Y \sim p(\cdot|x)} e^{w_p(x,Y)/\beta}, \\ \pi_w^+(y \mid x) &= p(y \mid x) e^{(w_p(x,y) - c_w(x))/\beta}, \\ \mathcal{L}_{\text{SPPO}}^{\text{exact}} &= \mathbb{E}_{x,Y \sim p}\left[\log \frac{\pi_\theta(Y \mid x)}{p(Y \mid x)} - \frac{w_p(x,Y) - c_w(x)}{\beta}\right]^2. \end{aligned} \tag{J.1}$$

Thus, setting $R = w_p$ gives the fixed-normalizer SKLPO loss up to a positive scale, with matched sampling and outer weights. This is a one-step identity: SPPO updates the opponent in self-play, whereas a verifier reward need not depend on the sampler.

SPPO's practical Equation (4.6) replaces $\log Z$ by $\eta/2$, equivalently c_w by $1/2$, and estimates w_p from sampled opponents. For reciprocal preferences, $\mathbb{E}_p w_p = 1/2$, so our KL identity yields

$$c_w(x) = \frac{1}{2} + \beta \, \mathrm{KL}(p(\cdot \mid x) \| \pi_w^+(\cdot \mid x)).$$

The half-win baseline is therefore not generally the exact normalizer. GPO instead uses preference log-odds scores; even under Bradley–Terry preferences, their expected sigmoid win rate is generally not affine in the scalar reward. Accordingly, SPPO need not share the verifier-reward Gibbs target, and its self-play convergence guarantee does not by itself establish a guarantee for asynchronous SKLPO replay.

General Preference Optimization. GPO fits log-likelihood ratios specified by an exponential policy update (Zhang et al., 2025, Equations (5.1)–(5.3)). For a frozen opponent/reference $\pi_{\theta_{t-k}}^{\text{sampler}}$, let

$$u_{\text{pref}}(x, y) = \mathbb{E}_{Y' \sim \pi_{\theta_{t-k}}^{\text{sampler}}(\cdot|x)}[s(y \succ Y' \mid x)], \qquad c_{\text{pref}}^*(x) = \beta \log \mathbb{E}_{\pi_{\theta_{t-k}}^{\text{sampler}}}[e^{u_{\text{pref}}(x,Y)/\beta}]. \tag{J.2}$$

Here s is an antisymmetric preference score. GPO estimates this expected score with sampled opponents and fits the target $\pi_{\theta_{t-k}}^{\text{sampler}} \exp[(u_{\text{pref}} - c_{\text{pref}}^*)/\beta]$. Its practical implementation approximates the log-normalization term by zero; the exact comparison retains it.

For scalar reward-difference scores $s(y \succ y' \mid x) = R(x, y) - R(x, y')$, we have $u_{\text{pref}} = R - \mathbb{E}_{\pi_{\theta_{t-k}}^{\text{sampler}}} R$ and $c_{\text{pref}}^* = c^* - \mathbb{E}_{\pi_{\theta_{t-k}}^{\text{sampler}}} R$. Hence $u_{\text{pref}} - c_{\text{pref}}^* = R - c^*$: the Gibbs targets agree, and the exact-normalizer population regression losses agree up to an overall positive scale when their data distributions and outer weights match. This statement does not identify the sampled-opponent, zero-normalizer practical GPO loss with SKLPO. SKLPO can use the verifier directly without sampling a preference opponent. General pairwise scores can encode cyclic preferences with no scalar reward-difference representation.

Bellman Policy Optimization. BPO starts from a token-level PMD update using $A_{t,k}^{\text{sampler}}(s,a)$ (Song et al., 2026). With $\eta = 1/\beta$, its exact trajectory residual is

$$\delta_{\text{BPO}}(x,y) = \frac{R(x,y) - V_{t,k}^{\text{sampler}}(x)}{\beta} - \sum_u \left[\log \frac{\pi_\theta(y_u \mid x, y_{<u})}{\pi_{\theta_{t-k}}^{\text{sampler}}(y_u \mid x, y_{<u})} + \text{KL}(\pi_{\theta_{t-k}}^{\text{sampler}}(\cdot \mid x, y_{<u}) \| \pi_\theta(\cdot \mid x, y_{<u})) \right]. \tag{J.3}$$

Let $S_\theta(x,y) = \sum_u \text{KL}(\pi_{\theta_{t-k}}^{\text{sampler}}(\cdot \mid x, y_{<u}) \| \pi_\theta(\cdot \mid x, y_{<u}))$. The KL chain rule and (A.5) give

$$\kappa_\theta(x) = \mathbb{E}_{Y \sim \pi_{\theta_{t-k}}^{\text{sampler}}(\cdot|x)}[S_\theta(x,Y)], \qquad \delta_{\text{KL}} - \delta_{\text{BPO}} = S_\theta - \kappa_\theta. \tag{J.4}$$

Thus the sampler-conditioned centered response residual subtracts the expected sum of token KLs, while BPO subtracts the path sum. These residuals agree only when the path sum is constant on the reference support. Separately, Section 3.7 proves that the exact BPO trajectory loss equals the token PMD regression loss in population. Appendix I.8 connects BPO's Equation (25) to the score-centering operation in Marek and Ryabinin (2026, Equation (4)). BPO's practical algorithm further approximates its residual and uses a smoothed complementary-probability weight; that practical surrogate is outside the exact equality.

Reward-weighted maximum likelihood. Another way to fit (A.2) is to minimize $\text{KL}(\pi_{t,k}^\star \| \pi_\theta)$, yielding

$$\mathcal{L}_{\text{MLE}}(\theta) = -\mathbb{E}_{\pi_{\theta_{t-k}}^{\text{sampler}}} \left[\frac{e^{R/\beta}}{Z} \log \pi_\theta \right]. \tag{J.5}$$

This is the reward-weighted regression route underlying AWR (Peng et al., 2019). The exponential weight is fixed within a round, whereas the RPG weight depends on the current policy. Log-ratio regression, weighted maximum likelihood, and direct RPG optimization can share a realizable target while taking different steps toward it.

K Additional Identities

K.1 An Unnormalized Reference Measure

For this algebraic extension only, let m be a positive finite measure with mass $M = \sum_y m(y)$, while the optimized π_θ remains normalized. Set $Z = \sum_y m(y) e^{R(y)/\beta}$. Then

$$\pi_{t,k}^\star(y) = \frac{m(y) e^{R(y)/\beta}}{Z}, \qquad J(\pi_\theta) = \beta \log Z + \beta(1-M) - \beta \, \text{KL}(\pi_\theta \| \pi_{t,k}^\star). \tag{K.1}$$

Scaling m only adds a constant to the objective, leaving its normalized optimizer unchanged. Without the policy normalization constraint, the stationary measure is $\widetilde{\pi}(y) = m(y) e^{R(y)/\beta}$; its mass need not be one, so it cannot directly serve as the trainable policy π_θ.

K.2 Coverage and Uniqueness

Fix a prompt and let ν share $\pi^{\text{sampler}}_{\theta_{t-k}}$'s support. The nonnegative exact-normalizer loss vanishes at $\pi^{\star}_{t,k}$, and any zero-loss policy satisfies $\beta \log(\pi_\theta/\pi^{\text{sampler}}_{\theta_{t-k}}) - R + c^* = 0$ throughout that support, giving uniqueness. Profiling instead gives $R - \beta \log(\pi_\theta/\pi^{\text{sampler}}_{\theta_{t-k}}) = a$; normalization fixes $a = c^*$.

Full coverage is sufficient, but not always necessary. With a fixed exact normalizer, observed probabilities are fixed; normalization determines a single omitted probability, but only the total mass of two or more omitted responses. With a profiled intercept, observed probability ratios alone are fixed, so even one omitted response can allow nonuniqueness. Finite data can omit responses despite full population support.

K.3 Profiling and Relative-Reward Regression

For $h_\theta = R - \beta \log(\pi_\theta/\pi^{\text{sampler}}_{\theta_{t-k}})$,

$$\mathbb{E}_\nu[(c - h_\theta)^2] = \mathrm{Var}_\nu(h_\theta) + (c - \mathbb{E}_\nu h_\theta)^2. \tag{K.2}$$

This gives the optimal intercept. For independent $Y, Y' \sim \nu$,

$$\mathrm{Var}_\nu(h_\theta) = \frac{1}{2}\mathbb{E}[(h_\theta(Y) - h_\theta(Y'))^2]. \tag{K.3}$$

The pairwise form uses differences of scalar verifier rewards; it does not represent general pairwise preferences.

K.4 The BPO Path Normalizer

Under deterministic token transitions and terminal rewards, BPO's token target is

$$\pi^{+}_{\text{tok}}(a \mid s) = \frac{\pi^{\text{sampler}}_{\theta_{t-k}}(a \mid s) e^{A^{\text{sampler}}_{t,k}(s,a)/\beta}}{Z_s}, \qquad Z_s = \mathbb{E}_{a \sim \pi^{\text{sampler}}_{\theta_{t-k}}(\cdot|s)} e^{A^{\text{sampler}}_{t,k}(s,a)/\beta}. \tag{K.4}$$

Using $\sum_u A^{\text{sampler}}_{t,k}(x, y_{<u}, y_u) = R(x,y) - V^{\text{sampler}}_{t,k}(x)$ gives the induced response distribution

$$\pi^{+}_{\text{tok}}(y \mid x) = \pi^{\text{sampler}}_{\theta_{t-k}}(y \mid x) \exp\left(\frac{R(x,y) - V^{\text{sampler}}_{t,k}(x)}{\beta} - \sum_u \log Z_{(x, y_{<u})} \right). \tag{K.5}$$

The response Gibbs update uses a single $\log Z(x)$. The updates agree only when the accumulated local normalization is constant across supported responses to each prompt; the Bellman telescope alone does not imply this.

L Sequence Length Weighting and Gradient Scale

Gradient scale. Writing $g_u = \nabla_\theta \log \pi_\theta(y_u \mid x, y_{<u})$, each token receives coefficient d_θ/T. At matched trainer and sampler, $d_\theta = c^* - R$ has no direct length factor. If $|\ell_{\theta,u}| \le M$ and $\|g_u\| \le G$, then the per-response weighted gradient has norm at most $(\beta T M + |R - c^*|)G$; the unweighted bound has one more factor of T. Averaging the scores does not bound the accumulated residual: a nonvanishing mean token log-ratio can still make the weighted loss and this gradient bound grow linearly in T.

The return-gradient contribution is $w_\theta T$ times the weighted regression contribution, where $w_\theta = \pi_\theta / \pi_{\theta_{t-k}}^{\text{sampler}}$. Their population gradients generally differ even at matched policies when lengths vary. In particular, $\mathbb{E}_{\pi_\theta}[T^{-1}\nabla_\theta \log \pi_\theta(Y)] = \nabla_\theta \mathbb{E}_{\pi_\theta}[T^{-1}]$ need not vanish. Target preservation therefore does not imply the first-step gradient equality of the unweighted formulation.