ANO: ROBUST POLICY OPTIMIZATION VIA BOUNDED, REDESCENDING GAIN FIELDS

Anonymous authors
Paper under double-blind review

ABSTRACT

Proximal Policy Optimization (PPO) dominates reinforcement learning and LLM alignment, yet its hard-clipping mechanism and unconstrained alternatives (e.g., SPO) sit at two extremes of a stability-efficiency dilemma. We argue that this dilemma is best understood *dynamically*: a surrogate objective is a feedback law on the probability ratio, and its clipping/penalty shape defines a *gain field* that drives the update dynamics. PPO's clip induces a *dead zone* (zero feedback outside the trust region), leaving the policy to drift open-loop under momentum; SPO's quadratic penalty induces an *unbounded, linearly growing gain* that stiffens the dynamics and destabilizes under aggressive step sizes. Guided by this view, we derive **Anchored Neighborhood Optimization (ANO)**, which designs the gain field directly: a C^∞ shaping kernel that anchors the identity map at $r{=}1$, peaks exactly at a prescribed trust-region boundary $1{+}\epsilon$, bounds the push on severely off-policy samples by a tunable κ_+, and exerts a bounded, *redescending pull* of tunable depth κ_- on extreme outliers. The three hyperparameters have decoupled roles, and all internal constants are solved in *closed form*. Empirically, ANO ranks *first on both* Atari (40 games) and MuJoCo in IQM and Median of normalized scores. While the runner-up differs across domains (PAPO on Atari, SPO on MuJoCo), ANO is the only method consistently at the top. Under a learning-rate stress test ($3\times10^{-4}\to10^{-3}$), ANO degrades by only 0.9% whereas PPO collapses by 54.5%, and the stressed ANO still outperforms PPO and PAPO at their best-tuned learning rates. Our code is available at https://anonymous.4open.science/r/ano-F818.

1 INTRODUCTION

Deep Reinforcement Learning (DRL) has achieved remarkable success in areas such as gaming, robotic control, and language model alignment (Mnih et al., 2015; Silver et al., 2016; 2018; Vinyals et al., 2019; Ye et al., 2020; Makoviychuk et al., 2021; Rudin et al., 2022; Heess et al., 2017; Schulman et al., 2015b; Ouyang et al., 2022; Black et al., 2023). However, Policy Gradient (PG) methods face a fundamental trade-off between update stability and learning efficiency. Trust Region Policy Optimization (TRPO) enforces stability through second-order constraints but is computationally expensive (Schulman et al., 2015a). Proximal Policy Optimization (PPO) addressed this by introducing a first-order clipping mechanism, achieving wide adoption due to its simplicity (Schulman et al., 2017).

Despite its popularity, PPO's "hard clipping" frequently fails to enforce the true trust region (Ilyas et al., 2018; Wang et al., 2020) and depends on code-level optimizations for stability (Engstrom et al., 2020). We argue that these pathologies are most clearly understood, and most directly fixed, through a **dynamical lens**. A clipped or penalized surrogate objective induces, for every sample, an effective feedback law on the probability ratio $r(s,a) = \tilde{\pi}(a|s)/\pi(a|s)$: the derivative of the shaping function acts as a *gain field* that accelerates, brakes, or reverses the movement of r. From this viewpoint, existing methods instantiate two extreme and equally undesirable feedback laws:

- **PPO: a dead zone.** Beyond the clipping interval the gain is exactly zero: once a sample's ratio crosses $1+\epsilon$, the objective provides *no feedback at all*, and under momentum-based optimizers the ratio keeps drifting open-loop while the gradient information carried by outlier samples is discarded.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

- **SPO: unbounded linear gain.** Quadratic penalties (Xie et al., 2025) restore feedback, but with a gain that grows *linearly* in the violation: for extreme outliers a single sample can dominate the batch gradient (interference), and on the probability simplex forcefully suppressing one action unpredictably redistributes mass onto the others.

This perspective suggests that the remedy is not a better constraint enforcement mechanism, but a better *feedback law*: the gain field should (i) coincide with the identity at $r = 1$, where the updated and reference policies agree, to preserve the on-policy gradient, (ii) turn negative beyond a prescribed trust-region boundary so that violations are *actively* corrected, (iii) remain *bounded* everywhere so that no single sample can dominate a batch update, and (iv) *redescend* to zero for extreme outliers, whose information content is unreliable. Requirements (i)–(iv) echo redescending M-estimators in robust statistics (Huber, 2011; 1992), where bounded influence and outlier rejection are the classical design goals.

Guided by these principles, we propose **Anchored Neighborhood Optimization (ANO)**. Rather than tuning a fixed accidental shape, ANO constructs the shaping function *from its desired gain profile*: we parameterize the derivative directly as a rational function of an exponential, impose the anchors $G(1) = 1$, $G'(1) = 1$, $G'(1 + \epsilon) = 0$, and prescribe the two bounds of the gain field: a push bound κ_+ (saturated as $r \to -\infty$) and a pull floor κ_- as the global minimum of the gain. The three hyperparameters $(\epsilon, \kappa_+, \kappa_-)$ have decoupled roles, and every internal constant of the kernel is recovered in *closed form*: no root-finding, no iterative solve, and no feasibility gap within the admissible parameter domain. The resulting kernel is C^∞, has exactly one inflection point in its gain profile by construction, and evaluates in $O(1)$ per sample.

Empirically, we evaluate ANO on the full Atari benchmark (40 games) and 6 MuJoCo continuous-control tasks with 5 seeds, reporting *rliable* aggregate metrics (Agarwal et al., 2021). ANO ranks *first on both domains* in both IQM and Median of normalized scores. Notably, the runner-up is domain-dependent (PAPO on Atari, SPO on MuJoCo), making ANO the *only* method consistently at the top. A learning-rate stress test further confirms the dynamical account (Section 5.3).

We summarize our main contributions as follows:

- **A dynamical perspective on surrogate design.** We reinterpret clipped/penalized policy objectives as feedback laws on the probability ratio, identify the dead-zone (PPO) and unbounded-gain (SPO) pathologies, and distill the ingredients that suffice for a robust gain field, separating them from dispensable machinery such as second-order enforcement and hard constraints.
- **A principled, closed-form kernel.** We derive ANO's shaping kernel from a prescribed gain profile with three interpretable hyperparameters with decoupled roles $(\epsilon, \kappa_+, \kappa_-)$, and prove its structural properties (unique inflection point, bijective depth control, unconditional solvability) that hold uniformly over the whole parameter domain.
- **Cross-domain empirical consistency and structural robustness.** ANO ranks first on both Atari and MuJoCo in IQM and Median, and exhibits an order-of-magnitude smaller degradation than PPO under aggressive learning rates, corroborating the dynamical stability analysis.

2 PRELIMINARIES

We consider a Markov Decision Process with expected discounted return $\eta(\pi) = \mathbb{E}_{\tau\sim\pi}[\sum_{t=0}^{\infty} \gamma^t r_t]$, advantage function $A_\pi(s, a) = Q_\pi(s, a) - V_\pi(s)$, and importance ratio $r(s, a) = \frac{\tilde{\pi}(a|s)}{\pi(a|s)}$ between a candidate policy $\tilde{\pi}$ and the current policy π. Policy search methods optimize a local surrogate obtained from the performance difference lemma (Kakade & Langford, 2002; Schulman et al., 2015a):

$$S(\tilde{\pi}) = \eta(\pi) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim\rho_\pi, a\sim\pi}\left[r(s, a)A_\pi(s, a)\right], \tag{1}$$

which lower-bounds the true return $\eta(\tilde{\pi})$ up to a divergence penalty; Schulman et al. (2015a) use $\beta D_{\mathrm{KL}}^{\max}(\pi, \tilde{\pi})$ via Pinsker's inequality and enforce the restriction $\mathbb{E}_s\left[D_{\mathrm{KL}}(\pi\|\tilde{\pi})\right] \le \epsilon$ (TRPO). Proximal Policy Optimization (PPO) (Schulman et al., 2017) replaces the constraint with a first-order approximation that clips the ratio:

$$\mathcal{L}^{\mathrm{PPO}}(\tilde{\pi}) = \mathbb{E}_{\substack{s\sim\rho_\pi(\cdot)\\ a\sim\pi(\cdot|s)}}\left[\min\left(r(s, a)A_\pi, \mathrm{clip}(r(s, a), 1-\epsilon, 1+\epsilon)A_\pi\right)\right]. \tag{2}$$

In Section 3 we reinterpret objectives of this form as feedback laws on the probability ratio, and show that the clipping shape, rather than the constraint it proxies, is what determines the stability of the induced update dynamics.

3 What Matters in Surrogate Design: A Dynamical Perspective

Before deriving our algorithm, we revisit the design of surrogate objectives and ask: *which ingredients of PPO-style objectives are load-bearing, and which are accidental?* First, we prove a worst-case lower bound on the true performance: the expected return $\eta(\tilde{\pi})$ introduced in Section 2 (Section 3.1). It shows that safety is guaranteed by controlling the probability ratio of every sample on both sides, keeping all ratios within a bounded neighborhood of 1. Second, we argue that the *shape* of the ratio mapping, equivalently the gain field it induces on the update dynamics, is what separates stable methods from brittle ones (Section 3.2).

3.1 Two-Sided Ratio Control: A Sufficient Safety Ingredient

We begin from the classical pattern of Section 2: a surrogate is trusted only within a divergence neighborhood of the current policy. The following worst-case bound reformulates this guarantee in terms of *individual samples, in both directions*:

Theorem 3.1 (The Dual-Ratio Lower Bound)**.** *Let $\eta(\tilde{\pi}) = \mathbb{E}_{\tau\sim\tilde{\pi}}[\sum_{t=0}^{\infty}\gamma^t r_t]$ denote the expected return of policy $\tilde{\pi}$, and let $S(\tilde{\pi})$ be the surrogate objective defined in Eq. 1. The following bound holds:*

$$\eta(\tilde{\pi}) \geq S(\tilde{\pi}) - \frac{\beta\alpha}{2}\max_{s,a}\left[\ln\frac{\tilde{\pi}(a|s)}{\pi(a|s)}\right] - \frac{\beta(1-\alpha)}{2}\max_{s,a}\left[\ln\frac{\pi(a|s)}{\tilde{\pi}(a|s)}\right],$$

where $\alpha \in [0,1]$ is a hyperparameter, and equality holds when $\tilde{\pi} = \pi$.

Proof. See Appendix D. □

Theorem 3.1 recasts the classical TV-based bound of Schulman et al. (2015a) in terms of *worst-case log-ratios in both directions*: the performance penalty is governed by the most displaced sample, so safety requires bounding $r(s,a)$ element-wise on both sides (forward and reverse ratio). This motivates an objective family that enforces two-sided control geometrically.

Definition 3.2 (The Generalized Surrogate Objective)**.** Let $r(s,a) = \frac{\tilde{\pi}(a|s)}{\pi(a|s)}$. Let $f, g : \mathbb{R} \to \overline{\mathbb{R}}$ be extended real-valued shaping functions satisfying two **geometric constraints**:

1. **Identity Anchoring (Fixed Point Condition):** $f(1) = g(1) = 1$.
2. **Trust Region Bounding (Geometric Enclosure):** $g(x) \geq x \geq f(x)$ for all $x \in \mathbb{R}$.

The **Generalized Surrogate Objective** is

$$M(\tilde{\pi}; f, g) = \mathbb{E}_{\substack{s\sim\rho_\pi(\cdot)\\ a\sim\pi(\cdot|s)}}\Big[\min\big(g(r)A_\pi(s,a),\ f(r)A_\pi(s,a)\big)\Big]. \tag{3}$$

When $g(r) = 2 - f(2-r)$ (point symmetry about $(1,1)$), reinforcement and suppression share one shape; we term this the **Symmetric Objective** $M_s(\tilde{\pi}; f)$, the form used throughout this paper.

Corollary 3.3. *For the current policy $\tilde{\pi} = \pi$, we have $M(\pi; f, g) = 0$.*

Proof. When $\tilde{\pi} = \pi$, $r \equiv 1$ and $\min(g(1)A, f(1)A) = A$, so $M(\pi; f, g) = \mathbb{E}_{s\sim\rho_\pi, a\sim\pi}[A_\pi(s,a)] = 0$ by definition of the advantage function. □

The safety value of this construction is certified by hard-constraining the shaped objective to a worst-case ratio interval, which provably yields monotonic improvement:

Theorem 3.4 (The Safe Policy Update Condition)**.** *Let $\delta_C(\cdot)$ be the extended real-valued indicator function (0 if $x \in C$, ∞ otherwise). For any shaping functions f and g satisfying the geometric*

constraints in Definition 3.2, there exist non-negative constants ϵ_l, ϵ_u defining a feasible interval $I = [1 - \epsilon_l, 1 + \epsilon_u]$. If π^* maximizes the constrained objective

$$M(\pi^*; f_\delta, g_\delta) = \max_{\tilde{\pi}} M(\tilde{\pi}; f_\delta, g_\delta), \tag{4}$$

where $f_\delta = f - \delta_I$ and $g_\delta = g + \delta_I$, then monotonic improvement is guaranteed, i.e., $\eta(\pi^*) \geq \eta(\pi)$.

Proof. See Appendix E, which builds on Theorem 3.1. □

From hard constraints to shaped feedback. Directly enforcing the interval constraint of Theorem 3.4 at every step is computationally intractable and yields prohibitively small updates (Schulman et al., 2015a). Following the established relaxation paradigm (TRPO's local approximation, PPO's clipping, SPO's quadratic penalty), we treat the boundary as a *tunable* ϵ and replace the hard indicator with a smooth shape. The design question reduces to: *what should the shape of f be?* Section 3.2 answers from the dynamical viewpoint.

PPO and SPO as instances. Within the symmetric objective, PPO-Clip corresponds to softening the hard indicator into a saturation on a linear base, $f_{\text{PPO}}(r) = r - \mathbb{I}_{r>1+\epsilon}(r-(1+\epsilon)) = \min(r, 1+\epsilon)$:

$$f_{\text{PPO}}(r) = \min(r, 1 + \epsilon), \tag{5}$$

and SPO corresponds to the unique quadratic base function with vertex at $1 + \epsilon$ satisfying the geometric constraints ($f(1) = 1$, $f(r) \leq r$):

$$f_{\text{SPO}}(r) = -\frac{1}{2\epsilon}(r - 1 - \epsilon)^2 + \frac{\epsilon}{2} + 1. \tag{6}$$

Its induced gain $\kappa(r) = (1 + \epsilon - r)/\epsilon$ coincides with the hinge gain of SPO (Xie et al., 2025) wherever the latter is active, and grows linearly without bound; both are points in a common design space, and their failure modes can be compared on equal footing.

3.2 The Gain Field: How Shaping Functions Drive the Update Dynamics

For a single sample with advantage A, the shaped term satisfies $\nabla_\theta \big[f(r)\, A\big] = A\, f'(r)\, \nabla_\theta r$: the derivative $f'(r)$ acts as a state-dependent *gain* on the vanilla policy-gradient direction. Isolating this effect in a scalar idealization, the ratio evolves under the velocity field

$$\dot{r} \;\propto\; A \cdot \kappa(r), \qquad \kappa(r) := f'(r), \tag{7}$$

which we call the **gain field**. Eq. 7 exposes the feedback law that each method implicitly implements (Figure 1b–c):

- **PPO: a dead zone.** $\kappa(r) = 1$ inside $[0, 1 + \epsilon]$, then 0: beyond the boundary the system runs *open-loop*: no restoring force, drift under momentum, and outlier information discarded. The trust region is enforced in the objective's *value*, not its *dynamics*.
- **SPO: unbounded linear gain.** $\kappa(r) = (1 + \epsilon - r)/\epsilon$ grows with the violation: the restoring force is *stiff*, dominates the batch for extreme outliers (gradient interference), redistributes probability mass uncontrollably, and overshoots under aggressive step sizes.
- **What is needed.** A *bounded, redescending restoring force*: unit gain at $r = 1$, sign reversal at $1 + \epsilon$ (a stable fixed point of the velocity field), bounded magnitude everywhere (no sample dominates an update), and gain decaying to zero as $r \to +\infty$ (extreme outliers are too unreliable to act upon; the redescending principle of robust M-estimators (Hampel et al., 1986; Huber, 2011)). We use this scalar flow as a design lens; it does not by itself imply parameter-space convergence; adaptive optimizers further rescale the effective gain.

This analysis separates a sufficient core from the accidental in PPO-style design. **Sufficient**: (i) identity anchoring with unit gain, which preserves the on-policy gradient; (ii) the two-sided enclosure of Definition 3.2 ($g \geq r \geq f$), which Theorem 3.4 ties to monotonic improvement; (iii) a bounded gain field; (iv) smoothness. **Parsimonious**: the point symmetry $g(r) = 2 - f(2 - r)$, which halves the design space; asymmetric extensions are promising future work (Section 6). **Accidental**: hard clipping plateaus, second-order enforcement, and the exact bound coefficients. Section 4 constructs a kernel with exactly this core; Section 5 verifies the predicted robustness.

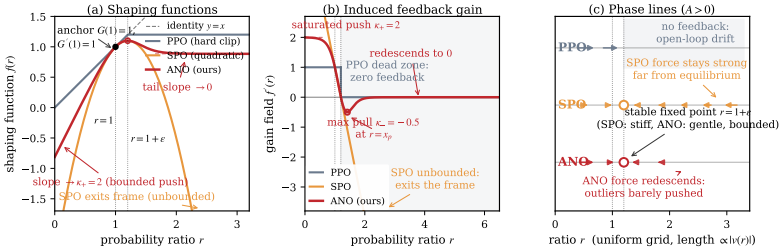


Figure 1: **Geometry and induced dynamics of surrogate objectives** (display config $\epsilon = 0.2$, $\kappa_+ = 2$, $\kappa_- = -0.5$). (a) Shaping functions $f(r)$: all three are tangent to the identity $y = x$ at $r = 1$ and peak at $1 + \epsilon$; ANO's tails have bounded slope (saturated push) and flatten toward $G(+\infty)$ (redescending pull), while SPO is unbounded and exits the frame. (b) Induced feedback gain $\kappa(r) = f'(r)$: PPO flatlines beyond $1 + \epsilon$ (dead zone, shaded); SPO grows without bound; ANO stays in $[\kappa_-, \kappa_+)$ and redescends to zero. (c) Phase lines for $A > 0$ (arrows: direction = sign of $\dot{r} \propto A\kappa(r)$, length $\propto |\dot{r}|$, normalized; panel b gives the true scale): each row has a stable fixed point at $r = 1 + \epsilon$, but only ANO's restoring force is bounded and redescending, so extreme outliers are barely pushed; PPO stalls into open-loop drift.

Table 1: **The three knobs of ANO.** Each hyperparameter controls one interpretable feature of the gain field $G'(r)$; all internal constants are then recovered in closed form (Eq. 19).

Knob	Definition	Dynamical role
ϵ	$G'(1+\epsilon) = 0$	trust-region boundary (location of the stable fixed point)
κ_+	$\lim_{r\to-\infty} G'(r)$	caps the push on severely off-policy samples
κ_-	$\min_r G'(r)$	maximal pull on outliers (bounded restoring force)

4 Anchored Neighborhood Optimization

We now construct **Anchored Neighborhood Optimization (ANO)**: a shaping kernel whose gain field realizes exactly the feedback law identified in Section 3.2. Rather than perturbing an existing clip, we design the kernel *in the derivative domain* and integrate. See Appendix C for pseudocode.

4.1 Design Requirements

We seek a globally C^∞, non-piecewise shaping function $G : \mathbb{R} \to \mathbb{R}$ whose derivative realizes a prescribed "push–pull" gain profile: the gain field $\kappa = G'$ is enclosed in $[\kappa_-, \kappa_+)$. Given hyperparameters $(\epsilon, \kappa_+, \kappa_-)$ with

$$\epsilon > 0, \qquad \kappa_+ > 1, \qquad -\kappa_+ < \kappa_- < 0, \tag{8}$$

the requirements are:

(R1) Anchoring: $G(1) = 1$ and $G'(1) = 1$

(R2) Sign reversal: $G'(1 + \epsilon) = 0$

(R3) Asymptotics: $\lim_{x\to-\infty} G'(x) = \kappa_+$ and $\lim_{x\to+\infty} G'(x) = 0$

(R4) Extremum: $\min_{x\in\mathbb{R}} G'(x) = \kappa_-$

(R5) Parsimony: G'' vanishes and changes sign at exactly one point x_p

The three knobs have direct dynamical meanings (Table 1): ϵ places the trust-region boundary, κ_+ caps the *push* on severely off-policy samples (saturating as $r \to -\infty$), and κ_- caps the *pull* on outliers. The left tail is not an artifact: the symmetric dual $g(r) = 2 - G(2 - r)$ routes overshooting samples ($r > 2$, negative advantages) through it, and its asymptote κ_+ bounds that restoring force.

4.2 The Gain Profile and Its Closed-Form Integration

Let $x_0 = 1 + \epsilon$ and $E(x) = e^{a(x-x_0)}$ with scale $a > 0$. We parameterize the gain field directly as a rational function of E, with an internal shape parameter $\nu > 0$:

$$G'(x) = \frac{\kappa_+ \, \nu \, \big(1 - E(x)\big)}{\big(1 + E(x)\big)\,\big(E(x) + \nu\big)}. \tag{9}$$

Eq. 9 is the rational form of a difference of two logistic sigmoids sharing slope a but centered at different points; since $E > 0$ and $\nu > 0$, it is manifestly C^∞ on $\mathbb{R}$. The choice of this family is not arbitrary: three of the five requirements hold *identically*, for every (a, ν):

$$\underbrace{E(x_0) = 1 \Rightarrow G'(x_0) = 0,}_{\text{(R2)}} \qquad \underbrace{E \to 0 \Rightarrow G' \to \kappa_+,}_{\text{(R3) left}} \qquad \underbrace{E \to \infty \Rightarrow G' \sim -\kappa_+ \nu / E \to 0}_{\text{(R3) right}}. \tag{10}$$

No equation needs solving to enforce them, so (R2)–(R3) impose no feasibility restriction on $(\epsilon, \kappa_+, \kappa_-)$; the remaining requirements are discharged by the following structural results.

Proposition 4.1 (Structural uniqueness; discharges (R5))**.** *For every $a > 0$, $\nu > 0$, $\kappa_+ > 0$, the second derivative*

$$G''(x) = \frac{\kappa_+ \nu a \, E \, \big(E^2 - 2E - (1 + 2\nu)\big)}{\big[(1 + E)(E + \nu)\big]^2} \tag{11}$$

vanishes at exactly one point $x_p \in \mathbb{R}$, changes sign there, and G' attains its global minimum at x_p:

$$x_p = x_0 + \frac{\ln(1 + s)}{a}, \qquad s := \sqrt{2 + 2\nu}. \tag{12}$$

Lemma 4.2 (Depth control; discharges (R4))**.** *At the inflection point, the minimal gain collapses to $G'(x_p) = -\kappa_+ \, d(\nu)$ with the* normalized depth

$$d(\nu) = \frac{s^2 - 2}{(s + 2)^2}, \qquad s = \sqrt{2 + 2\nu}, \tag{13}$$

which depends on ν alone, not on ϵ or κ_+. Moreover, $d : (0, \infty) \to (0, 1)$ is a strictly increasing C^∞ bijection, so $\kappa_- = -\kappa_+ d(\nu)$ sweeps $(-\kappa_+, 0)$ bijectively, and the parameterization inverts in closed form:

$$d = \frac{|\kappa_-|}{\kappa_+}, \qquad s = \frac{2d + \sqrt{2(d + 1)}}{1 - d}, \qquad \nu = \frac{s^2 - 2}{2}. \tag{14}$$

The restriction $|\kappa_-| < \kappa_+$ in Eq. 8 is intrinsic to the family 9: the pull-to-push ratio satisfies $|\kappa_-|/\kappa_+ = d(\nu) < 1$ for all $\nu > 0$.

Proposition 4.3 (Scale calibration; discharges (R1))**.** *Write $q := E(1) = e^{-a\epsilon} \in (0, 1)$. The anchor condition $G'(1) = 1$ reads*

$$\frac{\kappa_+ \nu (1 - q)}{(1 + q)(q + \nu)} = 1 \iff q^2 + Bq - C = 0, \quad B := \nu(\kappa_+ + 1) + 1 > 0, \;\; C := \nu(\kappa_+ - 1) > 0, \tag{15}$$

which has exactly one admissible root, giving the scale in closed form:

$$q = \frac{2C}{B + \sqrt{B^2 + 4C}} \in (0, 1), \qquad a = -\frac{\ln q}{\epsilon} > 0. \tag{16}$$

The conjugate form in Eq. 16 avoids catastrophic cancellation when $C \ll B^2$. For every $(\epsilon, \kappa_+, \kappa_-)$ in the domain 8, a unique a exists (unconditional solvability).

Lemma 4.4 (Two-sided enclosure)**.** *For every feasible $(\epsilon, \kappa_+, \kappa_-)$, $G(x) \le x$ for all $x \in \mathbb{R}$; hence the Symmetric Objective satisfies the enclosure $g(r) \ge r \ge f(r)$ of Definition 3.2, closing the loop with Theorem 3.4.*

Proof. See Appendix H. □

Finally, integrating Eq. 9 in partial fractions yields an elementary antiderivative. With $E = e^{at}$, $t = x - x_0$:

$$\Phi(t) = \frac{\kappa_+}{a}\left[at - \frac{2\nu}{\nu-1}\ln(1+E) + \frac{\nu+1}{\nu-1}\ln(E+\nu)\right] \quad (\nu \neq 1), \tag{17}$$

and $\Phi(t) = \frac{\kappa_+}{a}\left[at - \ln(1+E) + \frac{2}{1+E}\right]$ at the removable singularity $\nu = 1$. Anchoring at $x = 1$ to enforce $G(1) = 1$:

$$G(x) = 1 + \Phi\big(x - x_0\big) - \Phi\big(1 - x_0\big). \tag{18}$$

Every ingredient of Eq. 18 is elementary: no special functions, no numerical quadrature, no iterative solve. The complete evaluation chain is

$$(\epsilon, \kappa_+, \kappa_-) \longrightarrow d \longrightarrow s \longrightarrow \nu \longrightarrow (B, C) \longrightarrow q \longrightarrow a \longrightarrow \big(x_0,\, E(\cdot),\, x_p\big), \tag{19}$$

computed once before training; during training, evaluating G, G', or G'' costs $O(1)$ per sample. Proofs of Proposition 4.1, Lemma 4.2, and Proposition 4.3 are given in Appendix G.

4.3 Geometry and Induced Dynamics

Figure 1 visualizes the construction: G is tangent to the identity at $(1, 1)$ and peaks exactly at $1+\epsilon$, so maximizing the objective drives the ratio toward the peak: an **implicit trust region**. Panel (b) exposes the gain fields (PPO's dead zone, SPO's unbounded linear gain, ANO's bounded redescending pull) and panel (c) the resulting ratio-space flow (Eq. 7): ANO's velocity changes sign at $r = 1 + \epsilon$ (a *stable fixed point*), PPO's flow stops (neutral drift), and SPO's flow stiffens away from equilibrium.

Two features deserve emphasis. **Bounded stiffness.** Since G'' is globally bounded, the per-sample velocity field has a bounded Lipschitz constant: the discrete-time update remains non-oscillatory under a wide range of step sizes, the property stress-tested in Section 5.3. **Redescending influence.** As $r \to +\infty$, the pull decays like $\kappa_+\nu/E$: extreme outliers, whose advantage estimates are least reliable, exert vanishing influence, in the spirit of redescending M-estimators (Huber, 2011).

The objective. ANO instantiates the symmetric objective of Definition 3.2 with $f = G$:

$$\mathcal{L}^{\mathrm{ANO}}(\tilde{\pi}) = \mathbb{E}_{\substack{s\sim\rho_\pi \\ a\sim\pi}}\Big[\min\big(G(r)\,A_\pi(s,a),\ \big[2 - G(2-r)\big]\,A_\pi(s,a)\big)\Big]. \tag{20}$$

Implementation requires only replacing the clipped ratio in a PPO codebase with the element-wise evaluation of Eq. 18 (Appendix C); all constants in the chain 19 are precomputed once.

5 Experiments

We empirically evaluate ANO by addressing three Research Questions: **RQ 1 (Performance & consistency):** Does ANO deliver top-tier performance *consistently* across domains and metrics? **RQ 2 (Structural robustness):** Does the bounded, redescending gain field tolerate aggressive hyperparameters, as predicted? **RQ 3 (Sensitivity):** How do the three knobs affect performance?

5.1 Experimental Setup

Benchmarks. We evaluate on the full **Atari** benchmark (40 games) (Bellemare et al., 2013) and 6 **MuJoCo** continuous-control tasks (Todorov et al., 2012) (HalfCheetah, Ant, Hopper, Walker2d, Humanoid, Swimmer). Scores are normalized per environment (Human-Normalized Score for Atari; Expert-Normalized Score with a TD3 expert (Fujimoto et al., 2018) for MuJoCo); a run's final score is the mean over its last 2% of training steps, and scores are aggregated across 5 seeds using *rliable* (Agarwal et al., 2021): we report the Interquartile Mean (IQM), Mean, Median, and standard deviation of the normalized scores.

Baselines and implementations. We compare against **PPO** (Schulman et al., 2017), **TRPO** (Schulman et al., 2015a), **SPO** (Xie et al., 2025), and **PAPO** (Zhao et al., 2024), all implemented in a single shared CleanRL-style codebase (Huang et al., 2022) with `EnvPool` (Weng et al., 2022b) for parallel simulation, so all methods share identical networks, data pipelines, and evaluation protocols. Each baseline is run at several hyperparameter settings (clip/KL/ω grids in Appendix A.4); for ANO we fix $(\kappa_+, \kappa_-) = (10, -1.5)$ and sweep $\epsilon \in \{0.1, 0.2, 0.3\}$. Full hyperparameters are deferred to Appendix A.

Table 2: **Atari (40 games, 5 seeds).** Aggregate Human-Normalized Score (HNS) statistics, best configuration per method (full sweeps in Table 7); ANO ranks first in IQM and Median.

Method	IQM ↑	Mean HNS ↑	Median HNS ↑	Std HNS
ANO (ϵ=0.2, κ_+=10, κ_-=−1.5)	**0.9350**	3.8600	**0.8084**	9.2332
PAPO (ω_1=ω_2=0.005)	0.9290	4.0044	0.7239	9.8621
SPO (ϵ=0.2)	0.9012	3.9109	0.7551	9.4342
PPO (ϵ=0.1)	0.8873	3.8874	0.7199	10.0076
TRPO (KL=0.01)	0.3462	2.5543	0.2405	9.3490
ANO (ϵ=0.1 / 0.3, same κ)	0.8698 / 0.8252	3.9211 / **5.5272**	0.7665 / 0.6132	10.8456 / 18.1627

Table 3: **MuJoCo (6 environments, 5 seeds).** Aggregate Expert-Normalized Score (ENS) statistics at 3×10^{-4}, best configuration per method (full sweeps in Table 8); ANO ranks first in IQM, Mean, and Median. Bootstrap 95% IQM CIs (per-environment stratified): ANO $[0.88, 0.99]$, SPO $[0.85, 0.98]$, PPO $[0.84, 0.98]$, TRPO $[0.78, 0.88]$, PAPO $[0.76, 0.83]$; ANO's interval has the highest lower bound.

Method	IQM ↑	Mean ENS ↑	Median ENS ↑	Std ENS
ANO (ϵ=0.3, κ_+=10, κ_-=−1.5)	**0.9304**	**0.9605**	**1.0079**	0.5121
SPO (ϵ=0.2)	0.9249	0.9139	0.9780	0.4849
PPO (ϵ=0.2)	0.9185	0.8662	0.9222	0.4089
TRPO (KL=0.01)	0.8501	0.8204	0.8249	0.4060
PAPO (ω_1=ω_2=0.005)	0.7925	0.8112	0.8723	0.4504
ANO (ϵ=0.1 / 0.2, same κ)	0.8287 / 0.8764	0.8261 / 0.9057	0.7377 / 0.8791	0.3626 / 0.5225

5.2 Main Results: Consistent Top-Rank Performance Across Domains

Tables 2 and 3 report the aggregate results on Atari and MuJoCo, respectively.

Answering RQ 1. ANO attains the *best IQM and best Median on both domains* (the two robust aggregate statistics recommended by Agarwal et al. (2021)) and additionally the best Mean on MuJoCo. The runner-up is *domain-dependent* (PAPO on Atari, SPO on MuJoCo), so no single baseline matches ANO on both domains; ANO's only non-top aggregate is the Atari Mean, dominated by a few heavy-tailed games, the very reason Agarwal et al. (2021) recommend IQM and Median over the raw average. The ANO–PAPO Atari IQM gap is statistically indistinguishable from zero (paired bootstrap 95% CI $[-0.10, 0.11]$); ANO ranks first or second at every evaluation window (Table 4), and averaged over the *entire* run ANO's IQM exceeds PAPO's by 0.072 ($P > 0.999$). TRPO, the only explicitly KL-constrained method, ranks last on Atari and mid-pack on MuJoCo, further evidence that heavy constraint machinery is unnecessary.

5.3 Structural Robustness: Learning-Rate Stress Test

The dynamical analysis of Section 3.2 predicts that the bounded gain field of ANO yields a larger stability region under aggressive step sizes, whereas PPO's dead zone (drift) and SPO's linearly stiffening gain (overshoot) should degrade. We test this prediction on MuJoCo by raising the learning rate from 3×10^{-4} to 1×10^{-3}, keeping all else fixed.

Answering RQ 2. Table 10 (Appendix B) confirms the prediction quantitatively: ANO's degradation (-0.9%) is $7\times$ smaller than SPO's (-6.6%) and $60\times$ smaller than PPO's collapse (-54.5%). The stressed ANO (IQM 0.9225 at lr 10^{-3}) still exceeds the best-tuned PPO (0.9185) and PAPO (0.8160) at their nominal learning rate. This is the empirical signature of a bounded-stiffness feedback law: the dynamics remains well-conditioned far outside the tuned hyperparameter regime.

5.4 Dynamics Diagnostics: How Much Data Lives Outside the Trust Region?

Section 3.2 makes a directly testable prediction: samples whose ratio leaves the boundary receive zero gradient under PPO but bounded corrections under ANO. We monitor the resulting *out-of-boundary fraction* (share of samples with $|r - 1| > \epsilon$) at every update; Figures 5 and 6 (Appendix B) show the Asterix trace and the cross-game distributions.

Three observations stand out from Figure 6 and Table 9 (Appendix B). **(i) The dead zone is heavily populated under PPO.** At matched ϵ=0.1, PPO leaves 7.88% of samples outside the boundary with zero gradient on average; ANO keeps 1.21% (6.5× smaller, statistically matching SPO's 1.22%); at a *wider* boundary (ϵ=0.2) ANO keeps 3.92%, the restoring force pulling ratios back while PPO's dead zone lets them diffuse. **(ii) The fraction moves with ϵ in the direction predicted by the gain-field picture.** ANO's share grows monotonically with ϵ at a fixed threshold ($1.21\% \to 3.92\% \to 5.62\%$), since a later zero-crossing delays the onset of the restoring force. **(iii) SPO attains an even smaller fraction, but through an *unbounded* restoring gain**: the diagnostic measures where samples sit, not what force acts on them; force boundedness, not the fraction itself, determines robustness under aggressive hyperparameters (Section 5.3).

5.5 Sensitivity to the Trust-Region Boundary

Answering RQ 3. With (κ_+, κ_-) fixed at $(10, -1.5)$, sweeping $\epsilon \in \{0.1, 0.2, 0.3\}$ keeps ANO competitive: Atari IQM 0.87–0.94, MuJoCo 0.83–0.93, with ϵ=0.2 preferred on Atari and ϵ=0.3 on MuJoCo. The knobs admit direct interpretation (Table 1); a fuller (κ_+, κ_-) study is future work.

5.6 Extension to LLM Alignment

Finally, we test the gain-field advantage in RLHF: fine-tuning a language model against a learned reward model. We use Reddit TL;DR summarization (Stiennon et al., 2020) with a Pythia-1B policy and a Pythia-1B reward model ($\beta = 0.05$), trained for 1M episodes under the `TRL` pipeline (von Werra et al., 2020); ANO replaces the clipped objective by Eq. 20 without any other modification (one run per method; we therefore treat these results as *suggestive*; full hyperparameters in Appendix A).

KL dynamics. The same signature appears in classical RL: PPO's KL from the SFT reference never saturates, whereas ANO's restoring force contains the drift at the lowest final KL (39.5 vs. 54.0 for PPO; SPO: 40.9); ANO also preserves the most generation diversity (final entropy 23.7 vs. 21.4); bounded updates do not collapse the policy (Appendix Figure 7).

Quality evaluation without the training reward model. The training reward model is a compromised evaluator for exactly the methods it trained, so we score generations with an independent LLM judge instead: each summary is rated on a 1–10 rubric (relevance, coherence, faithfulness, conciseness) in isolation to eliminate position bias, on 300 held-out test prompts, reporting paired per-prompt differences with bootstrap CIs and a TOST equivalence test (δ=0.5); greedy decoding is primary, with sampling at $T \in \{0.7, 1.0\}$ as a robustness check.

Results. Figure 4 and Table 11 (both Appendix B) give three findings. **(i) ANO beats PPO on quality and stability simultaneously:** at T=0 the paired overall difference is $+0.83$ $[+0.63, +1.02]$ on the 10-point scale (sign test, 182 vs. 78 of 260 non-tied pairs, $p < 10^{-9}$), driven by faithfulness ($+1.36$) and relevance ($+1.15$); equal output lengths (192 vs. 188 characters) rule out a verbosity artifact. **(ii) ANO matches the strongest baseline:** against SPO it is $+0.14$ $[-0.01, +0.29]$, TOST-equivalent at all three temperatures, achieved with a *bounded* gain where SPO's unbounded force degrades (Section 5.3). **(iii) The advantage is robust to the push/pull knobs:** halving the gain magnitudes ($(\kappa_+, \kappa_-) = (2, -0.5)$) preserves both conclusions (vs. PPO: $+0.72$ $[+0.54, +0.90]$; vs. SPO: equivalent), the final KL shifting toward SPO's level as predicted (Table 1).

6 Conclusion

We presented a dynamical view: surrogate objectives are feedback laws on the probability ratio whose stability is governed by the induced gain field, separating sufficient ingredients (identity anchoring, two-sided enclosure, bounded smooth gains) from accidental machinery (hard clipping, second-order constraints). This yielded Anchored Neighborhood Optimization (ANO): a derivative-domain shaping kernel with three decoupled knobs and a closed-form parameterization. ANO ranks first on both domains (IQM, Median), degrading only 0.9% at 3.3× the learning rate; future work includes dynamic trust regions, asymmetric extensions, and GRPO-style integration (Shao et al., 2024).

AI USE STATEMENT

In this work, we used generative AI tools in the following ways. (1) **Writing:** an AI assistant (Kimi) helped polish the language of the manuscript. (2) **Retrieval and discovery:** AI-assisted search was used to locate related work and reference data, including the provenance of the normalization constants reported in Appendix A. (3) **Code:** AI assistants helped develop, debug, and refactor the experimental codebase and the analysis scripts. In addition, an LLM (DeepSeek-V4.1-Flash, API model id `deepseek-chat`, accessed via its public API) served as the independent judge for the RLHF evaluation in Section 5.6. We have not used generative AI tools for running experiments, collecting results, or fitting data. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

A complete proof of the theoretical claims is provided in the appendix. Experimental details, including hyperparameters and evaluation protocols, are described in Section 5 and Appendix A. All experiments are conducted with publicly available benchmarks and libraries.

REFERENCES

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C Courville, and Marc Bellemare. Deep reinforcement learning at the edge of the statistical precipice. *Advances in neural information processing systems*, 34:29304–29320, 2021.

Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of artificial intelligence research*, 47: 253–279, 2013.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023.

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Firdaus Janoos, Larry Rudolph, and Aleksander Madry. Implementation matters in deep policy gradients: A case study on ppo and trpo. *arXiv preprint arXiv:2005.12729*, 2020.

Scott Fujimoto, Herke van Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1587–1596. PMLR, 10–15 Jul 2018. URL `https://proceedings.mlr.press/v80/fujimoto18a.html`.

Sylvain Gugger, Lysandre Debut, Thomas Wolf, Philipp Schmid, Zachary Mueller, Sourab Mangrulkar, Marc Sun, and Benjamin Bossan. Accelerate: Training and inference at scale made simple, efficient and adaptable. `https://github.com/huggingface/accelerate`, 2022.

Frank R Hampel, Elvezio M Ronchetti, Peter J Rousseeuw, and Werner A Stahel. *Robust Statistics: The Approach Based on Influence Functions*. John Wiley & Sons, New York, 1986.

Nicolas Heess, Dhruva Tb, Srinivasan Sriram, Jay Lemmon, Josh Merel, Greg Wayne, Yuval Tassa, Tom Erez, Ziyu Wang, SM Eslami, et al. Emergence of locomotion behaviours in rich environments. *arXiv preprint arXiv:1707.02286*, 2017.

Shengyi Huang, Rousslan Fernand Julien Dossa, Chang Ye, Jeff Braga, Dipam Chakraborty, Kinal Mehta, and JoÃGo GM AraÃšjo. Cleanrl: High-quality single-file implementations of deep reinforcement learning algorithms. *Journal of Machine Learning Research*, 23(274):1–18, 2022.

Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics: Methodology and distribution*, pp. 492–518. Springer, 1992.

Peter J Huber. Robust statistics. In *International encyclopedia of statistical science*, pp. 1248–1251. Springer, 2011.

Andrew Ilyas, Logan Engstrom, Shibani Santurkar, Dimitris Tsipras, Firdaus Janoos, Larry Rudolph, and Aleksander Madry. Are deep policy gradient algorithms truly policy gradient algorithms. *arXiv preprint arXiv:1811.02553*, 2018.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of the nineteenth international conference on machine learning*, pp. 267–274, 2002.

Hojoon Lee, Youngdo Lee, Takuma Seno, Donghu Kim, Peter Stone, and Jaegul Choo. Hyperspherical normalization for scalable deep reinforcement learning. 2025.

Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, et al. Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 3505–3506, 2020.

Nikita Rudin, David Hoeller, Philipp Reist, and Marco Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning. In *Conference on robot learning*, pp. 91–100. PMLR, 2022.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pp. 1889–1897. PMLR, 2015a.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015b.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419): 1140–1144, 2018.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021, 2020.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pp. 5026–5033. IEEE, 2012.

Mark Towers, Ariel Kwiatkowski, Jordan Terry, John U Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Markus Krimmel, Arjun KG, et al. Gymnasium: A standard interface for reinforcement learning environments. *arXiv preprint arXiv:2407.17032*, 2024.

Nino Vieillard, Olivier Pietquin, and Matthieu Geist. Munchausen reinforcement learning. *Advances in Neural Information Processing Systems*, 33:4235–4246, 2020.

Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *nature*, 575(7782):350–354, 2019.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. TRL: Transformers Reinforcement Learning, 2020. URL `https://github.com/huggingface/trl`.

Yuhui Wang, Hao He, and Xiaoyang Tan. Truly proximal policy optimization. In *Uncertainty in artificial intelligence*, pp. 113–122. PMLR, 2020.

Jiayi Weng, Huayu Chen, Dong Yan, Kaichao You, Alexis Duburcq, Minghao Zhang, Yi Su, Hang Su, and Jun Zhu. Tianshou: A highly modularized deep reinforcement learning library. *Journal of Machine Learning Research*, 23(267):1–6, 2022a. URL `http://jmlr.org/papers/v23/21-1127.html`.

Jiayi Weng, Min Lin, Shengyi Huang, Bo Liu, Denys Makoviichuk, Viktor Makoviychuk, Zichen Liu, Yufan Song, Ting Luo, Yukun Jiang, et al. Envpool: A highly parallel reinforcement learning environment execution engine. *Advances in Neural Information Processing Systems*, 35:22409–22421, 2022b.

Zhengpeng Xie, Qiang Zhang, Fan Yang, Marco Hutter, and Renjing Xu. Simple policy optimization. In *Forty-second International Conference on Machine Learning*, 2025. URL `https://openreview.net/forum?id=SG8Yx1FyeU`.

Deheng Ye, Zhao Liu, Mingfei Sun, Bei Shi, Peilin Zhao, Hao Wu, Hongsheng Yu, Shaojie Yang, Xipeng Wu, Qingwei Guo, et al. Mastering complex control in moba games with deep reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 6672–6679, 2020.

Weiye Zhao, Feihan Li, Yifan Sun, Rui Chen, Tianhao Wei, and Changliu Liu. Absolute policy optimization: Enhancing lower probability bound of performance with high confidence. In *Forty-first International Conference on Machine Learning*, 2024.

A Experimental Details

In this section, we provide the comprehensive implementation details, hyperparameter settings, and computational resources used in our experiments to ensure reproducibility.

A.1 Implementation Frameworks

Our experiments are built upon established open-source libraries to ensure reproducibility and high efficiency:

- **Traditional RL (MuJoCo & Atari):** All algorithms (PPO, TRPO, SPO, PAPO, ANO) are implemented in a *single shared* CleanRL-style codebase (Huang et al., 2022), so that every method uses identical network architectures, data pipelines, advantage estimators, and evaluation protocols. We utilize `EnvPool` Weng et al. (2022b) for massive parallel environment simulation, which significantly accelerates the wall-clock training time compared to standard `Gymnasium` Towers et al. (2024) wrappers.
- **LLM Fine-tuning:** We use the `TRL` library (von Werra et al., 2020) integrated with `Accelerate` Gugger et al. (2022) and `DeepSpeed` Rasley et al. (2020). The reward model is a fine-tuned Pythia-1B scoring model, consistent with the standard CleanRL setup for efficient 1B-scale experiments. Table 6 lists the hyperparameters; generation quality is evaluated by an independent LLM judge (DeepSeek-V4.1-Flash, JSON mode, temperature 0) scoring each summary on a 1–10 rubric (relevance, coherence, faithfulness, conciseness) over 300 held-out TL;DR test prompts, at decoding temperatures $\{0, 0.7, 1.0\}$ with 53 new tokens.

A.2 Normalization and Aggregation

For Atari, we apply Human Normalized Score (HNS):

$$\text{HNS} = \frac{\text{Agent Score} - \text{Random Score}}{\text{Human Score} - \text{Random Score}}, \tag{21}$$

where the per-game random and human reference scores follow Mnih et al. (2015); ElevatorAction, Carnival, and AirRaid use the reference scores of Vieillard et al. (2020) (their Table 3): $(0, 3000)$, $(380, 4000)$, and $(400, 3000)$ in $(\text{random}, \text{human})$ form, respectively.

For MuJoCo, we apply Expert Normalized Score (ENS):

$$\text{ENS} = \frac{\text{Agent Score} - \text{Random Score}}{\text{Expert Score} - \text{Random Score}}, \tag{22}$$

where the expert we used is TD3 (Fujimoto et al., 2018) just like Lee et al. (2025) and the TD3 scores are from Weng et al. (2022a) and random scores are from Lee et al. (2025) (the Swimmer random score is set to 0, as its random performance is negligible).

Aggregate statistics (IQM, Mean, Median, Std) are computed over the per-environment normalized scores of 5 seeds using the *rliable* library (Agarwal et al., 2021); a run's final score is the mean over its last 2% of training steps.

A.3 Evaluation Protocol and Environment Preprocessing

Atari. We use the ALE `v5` environments via `EnvPool` with the standard evaluation preprocessing: 84×84 grayscale frames, frame skip 4, frame stack 4, noop-max 30, and episodic life. Reward *clipping* (sign) is applied only to the training rewards; raw (unclipped) episode returns are logged for evaluation. Episodic returns are recorded only upon true game over (all lives exhausted). For `Atlantis-v5`, whose default episode length is unbounded under good play, we additionally cap the episode at $27{,}000$ steps to guarantee termination (a different purpose from the 30k evaluation cap discussed in Appendix B, which makes non-terminating TRPO episodes measurable rather than bounding unbounded ones). **MuJoCo.** We use the `v4` environments (HalfCheetah, Ant, Hopper, Walker2d, Humanoid, Swimmer) with default termination conditions.

Table 4: **Evaluation-window sweep** (best configuration per method): a run's final score is the mean over the last $x\%$ of training. The Atari ANO–PAPO IQM gap stays within ± 0.02 through the 50% window and the two methods are statistically indistinguishable (paired bootstrap, 95% CI $[-0.10, 0.11]$ at the 2% window); averaged over the entire run (100%), ANO's Atari IQM exceeds PAPO's by 0.072, where P is the paired-bootstrap probability that ANO's IQM exceeds PAPO's ($P > 0.999$).

	Atari (HNS)					MuJoCo (ENS)				
Window	**ANO**	**PAPO**	**SPO**	**PPO**	**TRPO**	**ANO**	**SPO**	**PPO**	**TRPO**	**PAPO**
			IQM					*IQM*		
2%	**0.9350**	0.9290	0.9012	0.8873	0.3462	**0.9304**	0.9249	0.9185	0.8501	0.7925
10%	0.9137	**0.9188**	0.8861	0.9003	0.3527	**0.9413**	0.9312	0.9082	0.8476	0.7805
50%	**0.8100**	0.7909	0.7827	0.7721	0.3382	0.8991	**0.9090**	0.8716	0.7355	0.7477
100%	**0.5610**	0.4889	0.5288	0.4879	0.2931	0.5334	0.5173	**0.5645**	0.4364	0.4124
			Mean					*Mean*		
2%	3.8600	**4.0044**	3.9109	3.8874	2.5543	**0.9605**	0.9139	0.8662	0.8204	0.8112
10%	3.8613	**3.8698**	3.7543	3.6812	2.5669	**0.9652**	0.9142	0.8571	0.8177	0.7936
50%	**3.4650**	3.1869	3.2575	3.0810	2.2840	**0.8999**	0.8579	0.8270	0.7468	0.7370
100%	**1.7413**	1.6515	1.7190	1.6424	1.1040	0.5276	0.5014	**0.5686**	0.4737	0.4339
			Median					*Median*		
2%	**0.8084**	0.7239	0.7551	0.7199	0.2405	**1.0079**	0.9780	0.9222	0.8249	0.8723
10%	**0.8093**	0.6854	0.7341	0.6926	0.2298	**1.0098**	1.0024	0.8845	0.8200	0.8420
50%	**0.6745**	0.5899	0.6539	0.5692	0.2355	0.9716	**0.9746**	0.8272	0.7507	0.8086
100%	**0.4776**	0.4184	0.4514	0.3904	0.2199	0.5545	0.5152	**0.5815**	0.4559	0.3600

A.4 Hyperparameter Settings

All methods share the common training configuration within each domain; algorithm-specific hyperparameters are listed below.

Table 5: **Shared training configuration for traditional RL experiments.**

Hyperparameter	**Atari**	**MuJoCo**
Total timesteps	6×10^6	1×10^7
Learning rate (annealed)	2.5×10^{-4}	3×10^{-4}
Num. parallel envs	16	128
Rollout steps per env	128	32
Update epochs	4	10
Num. minibatches	4	4
Discount γ	0.99	0.99
GAE λ	0.95	0.95
Entropy coef.	0.01	0.0
Value coef.	0.5	0.5
Max grad norm	0.5	0.5
Optimizer	Adam ($\varepsilon = 10^{-5}$)	Adam ($\varepsilon = 10^{-5}$)
Seeds	$\{1, 2, 3, 4, 5\}$	$\{1, 2, 3, 4, 5\}$

Algorithm-specific hyperparameters. PPO and SPO use clip range $\epsilon \in \{0.1, 0.2, 0.3\}$ (grid search). TRPO uses maximal KL $\delta \in \{0.01, 0.02, 0.03\}$ (grid search), damping 0.1, 10 conjugate-gradient iterations, and 10 backtracking line-search iterations with ratio 0.5; its actor and critic use separate feature extractors so that value updates cannot violate the accepted KL step. PAPO uses $k = 7$, target KL 0.02, the detailed bias-correction variant, and $(\omega_1, \omega_2) \in \{0.001, 0.005, 0.01\}^2$ (diagonal grid search). ANO fixes $(\kappa_+, \kappa_-) = (10, -1.5)$ (the best of a two-point grid $\{(2, -0.5), (10, -1.5)\}$ over push/pull magnitudes, i.e., halving or doubling the gain scale; a denser (κ_+, κ_-) grid is prohibitive, as each candidate pair alone costs 3 boundary values $\times$ 5 seeds $\times$ 40 games $= 600$ Atari runs, and a single RLHF training run takes $\sim$60 hours) and sweeps the trust-region boundary $\epsilon \in \{0.1, 0.2, 0.3\}$. For the learning-rate stress test (Table 10), the learning rate is raised to 1×10^{-3} with every other hyperparameter unchanged.

Table 6: **Hyperparameters for LLM fine-tuning (TL;DR).** Global batch size and total episodes are aligned across methods; settings follow the standard `TRL` TL;DR setup (von Werra et al., 2020). ANO uses $\epsilon = 0.2$; the two (κ_+, κ_-) configurations correspond to the rows of Table 11.

Hyperparameter	**PPO/SPO**	**ANO (Ours)**
Base model / reward model	Pythia-1B / Pythia-1B (fine-tuned)	
Dataset	`trl-lib/tldr`	
Optimizer	AdamW	
Learning rate	3×10^{-6} (cosine decay)	
KL coefficient β	0.05	
Per-device batch size / grad. accum.	8 / 8 (global 64)	
Update epochs	4	
Total steps / episodes	15,625 / 1,000,000	
Clip range / boundary ϵ	0.2	0.2
(κ_+, κ_-)		$(10, -1.5)$, $(2, -0.5)$

A.5 Computational Resources

Experiments were conducted on NVIDIA H100 GPUs. ANO's element-wise kernel evaluation adds negligible overhead relative to the neural network forward-backward passes; all first-order methods (PPO, SPO, ANO) share the same $\mathcal{O}(N)$ time complexity per update.

B Full Experimental Results

Tables 7 and 8 give the complete hyperparameter sweeps behind the main-text summaries (Tables 2 and 3).

Table 7: **Full Atari sweep (40 games, 5 seeds).** Aggregate Human-Normalized Score statistics for every hyperparameter configuration.

Method	IQM ↑	Mean HNS ↑	Median HNS ↑	Std HNS
ANO (ϵ=0.2, κ_+=10, κ_-=−1.5)	**0.9350**	3.8600	**0.8084**	9.2332
PAPO (ω_1=ω_2=0.005)	0.9290	4.0044	0.7239	9.8621
SPO (ϵ=0.2)	0.9012	3.9109	0.7551	9.4342
PPO (ϵ=0.1)	0.8873	3.8874	0.7199	10.0076
PAPO (ω_1=ω_2=0.01)	0.8791	4.0994	0.6303	10.7148
PAPO (ω_1=ω_2=0.001)	0.9046	4.4863	0.6501	12.8591
SPO (ϵ=0.1)	0.8748	3.6115	0.7358	9.0909
ANO (ϵ=0.1, κ_+=10, κ_-=−1.5)	0.8698	3.9211	0.7665	10.8456
ANO (ϵ=0.3, κ_+=10, κ_-=−1.5)	0.8252	**5.5272**	0.6132	18.1627
PPO (ϵ=0.2)	0.8164	3.9020	0.6626	11.6492
SPO (ϵ=0.3)	0.7674	4.9649	0.6198	17.0069
PPO (ϵ=0.3)	0.7569	3.6087	0.6329	10.2429
TRPO (KL=0.01)	0.3462	2.5543	0.2405	9.3490
TRPO (KL=0.02)	0.2609	2.0484	0.1900	7.9825
TRPO (KL=0.03)	0.1969	1.9433	0.1378	8.1475

Table 8: **Full MuJoCo sweep (6 environments, 5 seeds).** Aggregate Expert-Normalized Score statistics at the default learning rate 3×10^{-4} for every hyperparameter configuration.

Method	IQM ↑	Mean ENS ↑	Median ENS ↑	Std ENS
ANO (ϵ=0.3, κ_+=10, κ_-=−1.5)	**0.9304**	**0.9605**	1.0079	0.5121
SPO (ϵ=0.2)	0.9249	0.9139	0.9780	0.4849
PPO (ϵ=0.2)	0.9185	0.8662	0.9222	0.4089
PPO (ϵ=0.1)	0.9132	0.8976	0.9197	0.5002
ANO (ϵ=0.2, κ_+=10, κ_-=−1.5)	0.8764	0.9057	0.8791	0.5225
PPO (ϵ=0.3)	0.8683	0.8308	0.8320	0.3695
SPO (ϵ=0.3)	0.8588	0.8900	0.9495	0.5603
SPO (ϵ=0.1)	0.8610	0.8529	0.8167	0.3838
TRPO (KL=0.01)	0.8501	0.8204	0.8249	0.4060
ANO (ϵ=0.1, κ_+=10, κ_-=−1.5)	0.8287	0.8261	0.7377	0.3626
PAPO (ω_1=ω_2=0.001)	0.8160	0.8166	**1.0112**	0.4595
TRPO (KL=0.02)	0.8053	0.7531	0.8319	0.3408
PAPO (ω_1=ω_2=0.005)	0.7925	0.8112	0.8723	0.4504
PAPO (ω_1=ω_2=0.01)	0.7508	0.7766	0.6144	0.4318

Per-environment training curves (mean $\pm$ std over 5 seeds, binned every 100k steps) are shown in Figures 2–3: all 40 Atari games and all 6 MuJoCo tasks, with the best-tuned configuration per method (ANO $(\epsilon{=}0.2, \kappa_+{=}10, \kappa_-{=}{-}1.5)$ on Atari and $(\epsilon{=}0.3, \kappa_+{=}10, \kappa_-{=}{-}1.5)$ on MuJoCo; PPO $\epsilon{=}0.1$; SPO $\epsilon{=}0.2$; PAPO $(0.005, 0.005)$; TRPO KL=0.01). The top-group comparison is statistically robust: the ANO–PAPO Atari IQM difference is indistinguishable from zero (paired bootstrap 95% CI $[-0.10, 0.11]$; ANO's lower bound is the highest) and the ranking is unchanged across evaluation windows (Table 4).

Two reporting caveats for the Atari curves. (i) In ElevatorAction, episode terminations become rare once a policy learns to survive, so the episodic-return curves of *all* methods are sparse and effectively shorter than the 6M-step horizon for most seeds. (ii) All TRPO runs were re-trained to the full 6M-step horizon after we found that TRPO policies on ElevatorAction and Tutankham can stop terminating episodes altogether (their episodic return then becomes undefined rather than low); a uniform episode cap of 30k steps makes such runs measurable. The re-run TRPO curves are complete, and the cap only ever truncates TRPO's own ElevatorAction episodes (often at a near-zero return, as the surviving policy stops scoring. These issues only deflate the TRPO baseline; Table 2 evaluates the final policy of every method under an identical protocol, and all of our conclusions against TRPO are therefore conservative.

Table 9: **Out-of-boundary data fraction on Atari (40 games $\times$ 5 seeds).** Time-averaged fraction of samples with $|r-1|>0.1$ (common threshold; verified from run configurations), mean $\pm$ std across games and seeds. PPO discards gradients on these samples (dead zone); ANO/SPO apply restoring forces to them. †: ϵ=0.2 and ϵ=0.3 configurations, fraction measured at the common threshold 0.1.

Method	Atari
PPO (ϵ=0.1)	7.88 ± 3.34
SPO (ϵ=0.1)	1.22 ± 0.85
ANO (ϵ=0.1, κ_+=10, κ_-=−1.5)	1.21 ± 0.90
ANO (ϵ=0.2, κ_+=10, κ_-=−1.5)†	3.92 ± 2.45
ANO (ϵ=0.3, κ_+=10, κ_-=−1.5)†	5.62 ± 2.93

Table 10: **Learning-rate stress test on MuJoCo (lr $3 \times 10^{-4} \rightarrow 10^{-3}$).** ANO degrades by only 0.9% in IQM (7$\times$ less than SPO), while PPO collapses by 54.5%. Notably, the stressed ANO still outperforms PPO and PAPO at their best-tuned learning rates (cf. Table 8).

Method	IQM @ 3×10^{-4}	IQM @ 10^{-3}	Degradation
ANO (ϵ=0.3, κ_+=10, κ_-=−1.5)	0.9304	**0.9225**	−0.9%
SPO (ϵ=0.2)	0.9249	0.8643	−6.6%
PPO (ϵ=0.2)	0.9185	0.4177	−54.5%

C Algorithm Implementation

Algorithm 1 Training Procedure for Anchored Neighborhood Optimization (ANO)

Require: Initial parameters θ_0, ϕ_0; ANO knobs $(\epsilon, \kappa_+, \kappa_-)$; coefficients $\lambda_{val}, \lambda_{ent}$
Hyperparameters: Learning rates $\alpha_\theta, \alpha_\phi$; Discount γ, GAE parameter λ
 # 0. One-time closed-form setup
Solve (ν, a, x_0) from $(\epsilon, \kappa_+, \kappa_-)$ via the chain in Eq. (19) {no iterative solve}
for iteration $k = 0, 1, 2, \ldots$ **do**
  # 1. Interaction & Data Collection
  Sample trajectories $\mathcal{T} = \{(s_t, a_t, r_t)\}$ by running policy π_{θ_k} in the environment
  # 2. Advantage Estimation
  Compute generalized advantages $\hat{A}_t$ and value targets $\hat{R}_t$ using GAE (γ, λ) based on V_{ϕ_k}
  # 3. Policy Snapshot
  $\theta_{\text{old}} \leftarrow \theta_k$
  # 4. Optimization Epochs
  for epoch $m = 1, \ldots, M$ **do**
    Resample mini-batches from $\mathcal{T}$
    for each mini-batch B **do**
      Calculate probability ratio $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}$
      # Compute Unified Ratio Objective Loss
      Evaluate shaping function $G(r_t)$ via Eq. (18) $\{O(1)$ per sample$\}$
      Derive symmetric dual $g(r_t) \leftarrow 2 - G(2 - r_t)$
      $\mathcal{L}_{\text{policy}} \leftarrow -\frac{1}{|B|} \sum_{t \in B} \min \left(g(r_t)\hat{A}_t,\ G(r_t)\hat{A}_t \right)$
      # Auxiliary Losses
      $\mathcal{L}_{\text{ent}} \leftarrow \frac{1}{|B|} \sum_{t \in B} \mathcal{H}(\pi_\theta(\cdot|s_t))$
      $\mathcal{L}_{\text{val}} \leftarrow \frac{1}{2|B|} \sum_{t \in B} (V_\phi(s_t) - \hat{R}_t)^2$
      # Gradient Step
      $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{policy}} + \lambda_{\text{val}}\mathcal{L}_{\text{val}} - \lambda_{\text{ent}}\mathcal{L}_{\text{ent}}$
      Update θ, ϕ w.r.t. $\nabla \mathcal{L}_{\text{total}}$ via optimizer
    end for
  end for
  $\theta_{k+1} \leftarrow \theta, \ \ \phi_{k+1} \leftarrow \phi$
end for

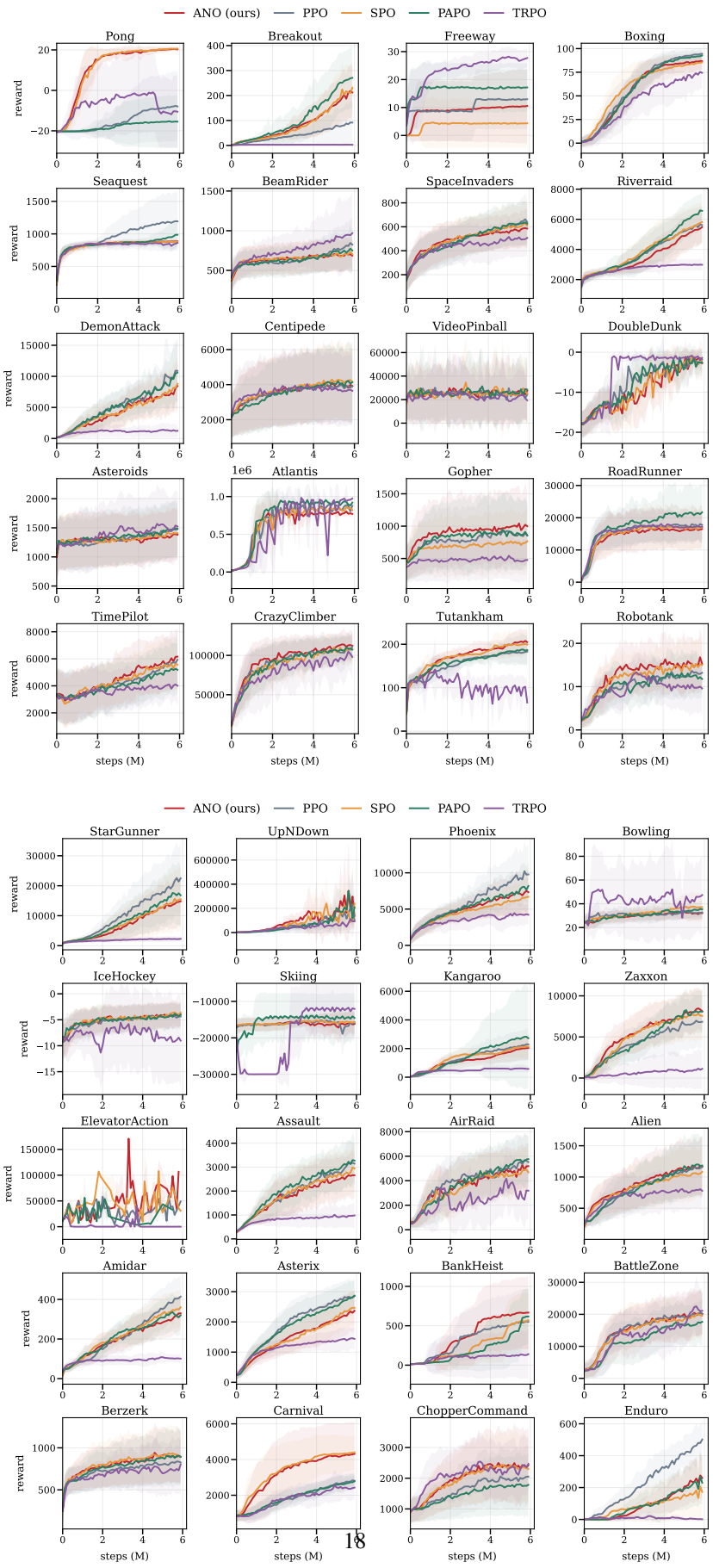


Figure 2: **Training curves on Atari (40 games).** Episode return (mean ± std, 5 seeds) vs. environment

Table 11: **RLHF results (TL;DR, Pythia-1B).** Rubric: mean of four 1–10 dimensions by an independent LLM judge, N=300 test prompts, greedy decoding. KL and entropy: final-window training means. Paired tests at T=0: ANO vs. PPO $\Delta = +0.83$ $[+0.63, +1.02]$, $p < 10^{-9}$; ANO vs. SPO $\Delta = +0.14$ $[-0.01, +0.29]$, TOST-equivalent (δ=0.5).

Method	Rubric ↑	KL ↓	Entropy ↑
PPO (ϵ=0.2)	6.76	54.0	21.4
SPO (ϵ=0.2)	7.43	40.9	23.0
ANO (ϵ=0.2, κ_+=10, κ_-=−1.5)	**7.58**	**39.5**	**23.7**
ANO (ϵ=0.2, κ_+=2, κ_-=−0.5)	7.49	42.0	22.6

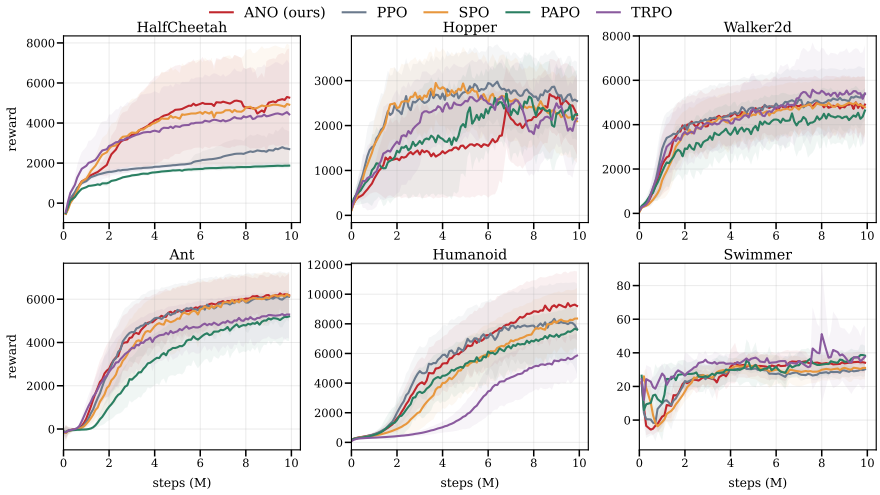


Figure 3: **Training curves on MuJoCo (6 environments).** Episode return (mean ± std, 5 seeds) vs. environment steps for the best-tuned configuration per method.

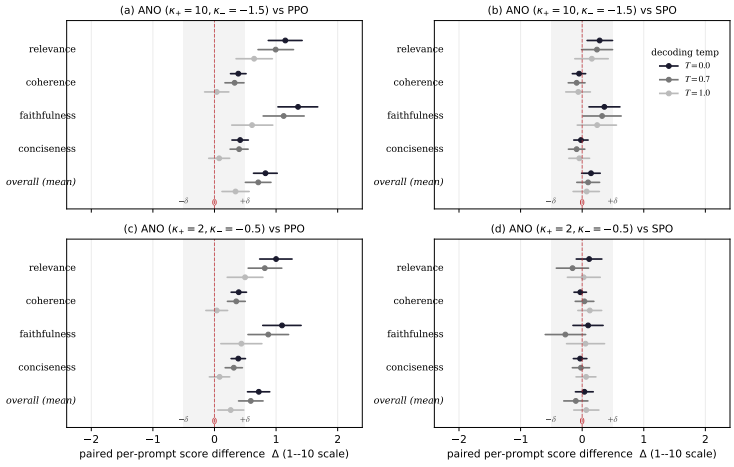


Figure 4: **Paired per-prompt rubric differences (ANO − baseline),** N=300**.** Grey band: equivalence region $(-\delta, +\delta)$ of the TOST test; darker points mark lower decoding temperatures. (a,b) ANO (κ_+=10, κ_-=−1.5) significantly outperforms PPO at all temperatures (faithfulness and relevance drive the gap) and is statistically equivalent to SPO. (c,d) The weaker configuration (κ_+=2, κ_-=−0.5) preserves both conclusions, indicating that the quality advantage is not an artifact of knife-edge tuning.

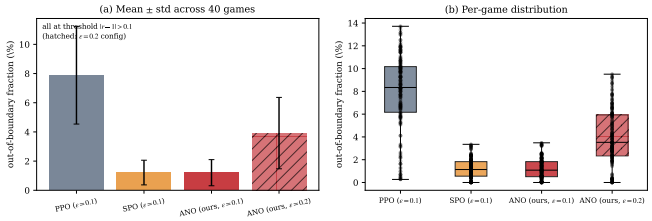


Figure 5: **Out-of-boundary fraction across all 40 Atari games (5 seeds each).** Mean ± std (left) and per-game distributions (right) of the time-averaged out-of-boundary fraction at the common threshold $|r - 1| > 0.1$; companion to Table 9 and Figure 6.

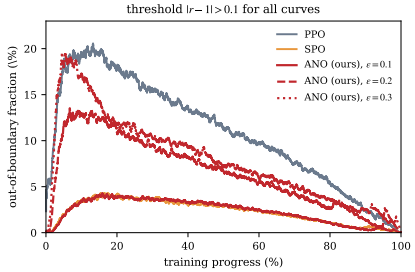


Figure 6: **Out-of-boundary fraction during training (Atari Asterix, 5 seeds); every curve is logged at the common threshold** $|r-1| > 0.1$ (PPO/SPO at ϵ=0.1; ANO's historical logging records the fraction at threshold 0.1 for all ϵ, verified from run configurations). PPO's dead zone is heavily populated (up to $\sim$20% of samples receive zero gradient), while ANO's restoring force keeps the fraction 3–5$\times$ lower; among ANO configurations the fraction grows with ϵ (a later zero-crossing delays the restoring force), as the gain-field picture predicts.

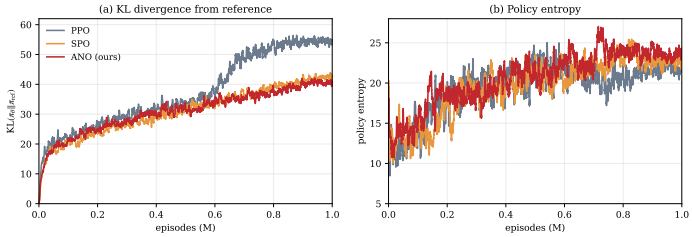


Figure 7: **RLHF training dynamics (TL;DR, Pythia-1B).** (a) PPO's dead zone lets the policy drift from the reference: the late-phase KL growth rate is $\sim$2.4$\times$ that of ANO (+4.8 vs. +2.0 per 100k episodes), and the final KL is 54.0 vs. 39.5. ANO's bounded restoring force keeps the policy closest to the reference, and (b) retains the highest policy entropy.

D PROOF OF THEOREM 3.1

Proof. From Theorem 1 of Schulman et al. (2015a), we have

$$\eta(\tilde{\pi}) \geq S(\tilde{\pi}) - \beta \left[D_{\mathrm{TV}}^{max}(\pi, \tilde{\pi})\right]^2. \tag{23}$$

Suppose $\alpha \in [0, 1]$, then we have

$$\begin{aligned}
\eta(\tilde{\pi}) &\geq S(\tilde{\pi}) - \alpha\beta \left[D_{\mathrm{TV}}^{max}(\pi, \tilde{\pi})\right]^2 - (1-\alpha)\beta \left[D_{\mathrm{TV}}^{max}(\pi, \tilde{\pi})\right]^2 \\
&= S(\tilde{\pi}) - \alpha\beta \left[D_{\mathrm{TV}}^{max}(\tilde{\pi}, \pi)\right]^2 - (1-\alpha)\beta \left[D_{\mathrm{TV}}^{max}(\pi, \tilde{\pi})\right]^2 \\
&\geq S(\tilde{\pi}) - \frac{\alpha\beta}{2}\left[D_{\mathrm{KL}}^{max}(\tilde{\pi}, \pi)\right] - \frac{(1-\alpha)\beta}{2}\left[D_{\mathrm{KL}}^{max}(\pi, \tilde{\pi})\right] \\
&\quad \text{(By Pinsker's inequality)} \\
&= S(\tilde{\pi}) - \frac{\beta}{2}\Big(\alpha \max_s \mathbb{E}_{a\sim\tilde{\pi}(\cdot|s)}\big[\ln \frac{\tilde{\pi}(a|s)}{\pi(a|s)}\big] + (1-\alpha)\max_s \mathbb{E}_{a\sim\pi(\cdot|s)}\big[\ln \frac{\pi(a|s)}{\tilde{\pi}(a|s)}\big]\Big) \\
&\geq S(\tilde{\pi}) - \frac{\beta}{2}\Big(\alpha \max_{s,a} \big[\ln \frac{\tilde{\pi}(a|s)}{\pi(a|s)}\big] + (1-\alpha)\max_{s,a} \big[\ln \frac{\pi(a|s)}{\tilde{\pi}(a|s)}\big]\Big)
\end{aligned} \tag{24}$$

□

E PROOF OF THEOREM 3.4

Proof. From Theorem 3.1, we have

$$\begin{aligned}
\eta(\tilde{\pi}) &\geq S(\tilde{\pi}) - \frac{\beta}{2}\Big(\alpha \max_{s,a} \big[\ln \frac{\tilde{\pi}(a|s)}{\pi(a|s)}\big] + (1-\alpha)\max_{s,a} \big[\ln \frac{\pi(a|s)}{\tilde{\pi}(a|s)}\big]\Big) \\
&= \eta(\pi) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim\rho_\pi,\, a\sim\pi}\Big[rA_\pi(s,a)\Big] - \frac{\beta}{2}\Big(\alpha \max_{s,a} \big[\ln \frac{\tilde{\pi}(a|s)}{\pi(a|s)}\big] + (1-\alpha)\max_{s,a} \big[\ln \frac{\pi(a|s)}{\tilde{\pi}(a|s)}\big]\Big) \\
&= \eta(\pi) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim\rho_\pi,\, a\sim\pi}\Big[\mathbb{I}_{A_\pi(s,a)<0} rA_\pi(s,a) + \mathbb{I}_{A_\pi(s,a)>0} rA_\pi(s,a)\Big] \\
&\quad - \frac{\beta}{2}\Big(\alpha \max_{s,a} \big[\ln \frac{\tilde{\pi}(a|s)}{\pi(a|s)}\big] + (1-\alpha)\max_{s,a} \big[\ln \frac{\pi(a|s)}{\tilde{\pi}(a|s)}\big]\Big) \\
&\geq \eta(\pi) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim\rho_\pi,\, a\sim\pi}\Big[\mathbb{I}_{A_\pi(s,a)<0} g(r)A_\pi(s,a) + \mathbb{I}_{A_\pi(s,a)>0} f(r)A_\pi(s,a)\Big] \\
&\quad - \frac{\beta}{2}\Big(\alpha \max_{s,a} \big[\ln \frac{\tilde{\pi}(a|s)}{\pi(a|s)}\big] + (1-\alpha)\max_{s,a} \big[\ln \frac{\pi(a|s)}{\tilde{\pi}(a|s)}\big]\Big) \\
&\quad \Big(\text{Because } g(r) \geq r \geq f(r)\Big) \\
&\geq \eta(\pi) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim\rho_\pi,\, a\sim\pi}\Big[\min\big(g(r)A_\pi(s,a),\, f(r)A_\pi(s,a)\big)\Big] \\
&\quad - \frac{\beta}{2}\Big(\alpha \max_{s,a} \big[\ln \frac{\tilde{\pi}(a|s)}{\pi(a|s)}\big] + (1-\alpha)\max_{s,a} \big[\ln \frac{\pi(a|s)}{\tilde{\pi}(a|s)}\big]\Big) \\
&= \underbrace{\eta(\pi) + \frac{1}{1-\gamma}M(\tilde{\pi}; f, g) - \frac{\beta}{2}\Big(\alpha \max_{s,a} \big[\ln \frac{\tilde{\pi}(a|s)}{\pi(a|s)}\big] + (1-\alpha)\max_{s,a} \big[\ln \frac{\pi(a|s)}{\tilde{\pi}(a|s)}\big]\Big)}_{M_F(\tilde{\pi})}.
\end{aligned} \tag{25}$$

According to Corollary 3.3, we have that $M_F(\pi) = \eta(\pi)$.

It remains to relate the unconstrained maximizer of M_F to the constrained problem. This is the classical exact-penalty fact: for constants $c_1, c_2 > 0$, a maximizer of $H(x) - c_1\delta_1(x) - c_2\delta_2(x)$ over any set that contains the feasible region $\{x : \delta_1(x) = \delta_2(x) = 0\}$ is a maximizer of $H(x)$ subject to $\delta_1(x) = \delta_2(x) = 0$, because on the feasible set the penalty terms vanish while off the feasible set the objective is $-\infty$. Applying this to $H(\tilde{\pi}) = \eta(\pi) + \frac{1}{1-\gamma}M(\tilde{\pi}; f, g)$ with $\delta_1(\tilde{\pi}) = \max_{s,a}[\ln \frac{\tilde{\pi}(a|s)}{\pi(a|s)}] - \epsilon_1$ and $\delta_2(\tilde{\pi}) = \max_{s,a}[\ln \frac{\pi(a|s)}{\tilde{\pi}(a|s)}] - \epsilon_2$ (penalized via the extended indicator δ_I,

with the constants $\frac{\beta}{2}\alpha$ and $\frac{\beta}{2}(1-\alpha)$ absorbed into it), the unconstrained problem $\max_{\tilde{\pi}} M_F(\tilde{\pi})$ is equivalent to the constrained problem:

$$\max_{\tilde{\pi}} M(\tilde{\pi}; f, g) \tag{26}$$

$$\text{s.t.} \begin{cases} \max_{s,a} \left[\ln \frac{\tilde{\pi}(a|s)}{\pi(a|s)}\right] \leq \epsilon_1, \\ \max_{s,a} \left[\ln \frac{\pi(a|s)}{\tilde{\pi}(a|s)}\right] \leq \epsilon_2. \end{cases} \tag{27}$$

Obviously, both ϵ_l and ϵ_u are non-negative. And when $\alpha = 0$, $\epsilon_1 = \infty$ while $\alpha = 1$, $\epsilon_2 = \infty$.

And the optimization can be rewritten as follows:

$$\max_{\tilde{\pi}} M(\tilde{\pi}; f, g) \tag{28}$$

$$\text{s.t.} \begin{cases} \max_{s,a} \frac{\tilde{\pi}(a|s)}{\pi(a|s)} \leq 1 + \epsilon_u, \\ \min_{s,a} \frac{\tilde{\pi}(a|s)}{\pi(a|s)} \geq 1 - \epsilon_l, \end{cases} \tag{29}$$

where $\epsilon_u = e^{\epsilon_1} - 1$ and $\epsilon_l = 1 - e^{-\epsilon_2}$.

Let π^* be the optimal solution to this optimization problem. We have

$$\begin{aligned}
\pi^* &= \operatorname*{argmax}_{\tilde{\pi}} M(\tilde{\pi}; f, g) - \delta_I(\max_r r) - \delta_I(\min_r r) \\
&= \operatorname*{argmax}_{\tilde{\pi}} M(\tilde{\pi}; f, g) - \sum_s \sum_a \rho_\pi(s) p_\pi(a|s) \big(\delta_I(\max_r r) + \delta_I(\min_r r)\big) \\
&\quad \Big(\text{Note that } \beta\delta_C(\cdot) \equiv \delta_C(\cdot) \text{ for any } \beta > 0\Big) \\
&= \operatorname*{argmax}_{\tilde{\pi}} \Big[\mathbb{E}_{s\sim\rho_\pi,\, a\sim\pi} \min\big(g(r)A_\pi(s,a),\, f(r)A_\pi(s,a)\big) - \mathbb{E}_{s\sim\rho_\pi,\, a\sim\pi}\big(|A_\pi(s,a)|\, \delta_I(r)\big)\Big] \\
&= \operatorname*{argmax}_{\tilde{\pi}} \Big[\mathbb{E}_{s\sim\rho_\pi,\, a\sim\pi} \min\big(g(r)A_\pi(s,a),\, f(r)A_\pi(s,a)\big) \\
&\quad - \mathbb{E}_{s\sim\rho_\pi,\, a\sim\pi} A_\pi(a|s)\mathbb{I}_{A_\pi(a|s)>0}\big(\delta_I(r)\big) + \mathbb{E}_{s\sim\rho_\pi,\, a\sim\pi} A_\pi(a|s)\mathbb{I}_{A_\pi(a|s)<0}\big(\delta_I(r)\big)\Big] \\
&= \operatorname*{argmax}_{\tilde{\pi}} \Big[\mathbb{E}_{s\sim\rho_\pi,\, a\sim\pi} \min\Big(\left(g(r) + \delta_I(r)\right) A_\pi(s,a),\, \left(f(r) - \delta_I(r)\right) A_\pi(s,a)\Big)\Big] \\
&= \operatorname*{argmax}_{\tilde{\pi}} \Big[\mathbb{E}_{s\sim\rho_\pi,\, a\sim\pi} \min\Big(g_\delta(r)A_\pi(s,a),\, f_\delta(r)A_\pi(s,a)\Big)\Big] \\
&= \operatorname*{argmax}_{\tilde{\pi}} M(\tilde{\pi}; f_\delta, g_\delta).
\end{aligned} \tag{30}$$

Degenerate case: It is easy to show that when $f(r) = 1 - \delta_{\{1\}}$ and $g(r) = 1 + \delta_{\{1\}}$, $\epsilon_u = 0$ and $\epsilon_l = 0$. □

F Symmetric Bounds via α-Adjustment (Remark F.1)

Remark F.1 (Symmetric Bounds via α-Adjustment). Consider a single-state MDP with $\mathcal{A} = \{a_1, a_2, a_3\}$, policy $\pi = [0.2, 0.7, 0.1]$, and advantages $A = [10, -2, -6]$. Let $\beta = 8$ and the base shaping functions be linear, $f(r) = g(r) = r$. By setting the convex combination coefficient $\alpha = 0.96$ in Theorem 3.1, the optimal constraints derived for maximizing the objective become perfectly symmetric with $\epsilon_u = \epsilon_l = 0.6$.

Proof. The optimization problem with the penalty term is:

$$\max_{\tilde{\pi}} \sum_a \tilde{\pi}(a|s)A(a|s) - \frac{\beta\alpha}{2} \max_a \ln \frac{\tilde{\pi}(a)}{\pi(a)} - \frac{\beta(1-\alpha)}{2} \max_a \ln \frac{\pi(a)}{\tilde{\pi}(a)} - \lambda(\sum \tilde{\pi} - 1) \tag{31}$$

Assuming the optimal policy $\tilde{\pi}$ hits the upper bound at a_1 and the lower bound at a_3, while a_2 remains in the interior (unconstrained relative to the trust region bounds but satisfying the probability simplex).

The first-order optimality condition (setting gradients to zero) gives:

$$\frac{\partial \mathcal{L}}{\partial \tilde{\pi}_1} = 10 - \frac{4\alpha}{\tilde{\pi}_1} - \lambda = 0 \tag{32}$$

$$\frac{\partial \mathcal{L}}{\partial \tilde{\pi}_2} = -2 - 0 - \lambda = 0 \implies \lambda = -\mathbf{2} \tag{33}$$

$$\frac{\partial \mathcal{L}}{\partial \tilde{\pi}_3} = -6 + \frac{4(1-\alpha)}{\tilde{\pi}_3} - \lambda = 0 \tag{34}$$

Substituting $\lambda = -2$ into the equations for a_1 and a_3: 1. For a_1 (Upper Bound: $\tilde{\pi}_1 = \pi_1(1+\epsilon)$):

$$10 - \frac{4\alpha}{0.2(1+0.6)} + 2 = 0 \implies 12 = \frac{4\alpha}{0.32} \implies \alpha = 0.96$$

2. For a_3 (Lower Bound: $\tilde{\pi}_3 = \pi_3(1-\epsilon)$):

$$-6 + \frac{4(1-\alpha)}{0.1(1-0.6)} + 2 = 0 \implies -4 + \frac{4(1-\alpha)}{0.04} = 0 \implies 1 - \alpha = 0.04 \implies \alpha = 0.96$$

Both conditions consistently yield $\alpha = 0.96$. Thus, the derivation is exact. □

G PROOFS FOR THE ANO KERNEL

This appendix collects the proofs of the structural results stated in Section 4.2. Throughout, $x_0 = 1+\epsilon$, $E(x) = e^{a(x-x_0)}$, and the gain field is $G'(x) = \kappa_+\nu(1-E)/[(1+E)(E+\nu)]$ with $a > 0$, $\nu > 0$, $\kappa_+ > 0$.

G.1 PROOF OF PROPOSITION 4.1 (STRUCTURAL UNIQUENESS)

Proof. Differentiating $G'(x)$ and using $E'(x) = aE(x)$ gives

$$G''(x) = \frac{\kappa_+\nu a\, E\left(E^2 - 2E - (1+2\nu)\right)}{\left[(1+E)(E+\nu)\right]^2}. \tag{35}$$

The denominator is strictly positive and the prefactor $\kappa_+\nu aE$ is strictly positive, so $\operatorname{sign} G'' = \operatorname{sign}\left(E^2 - 2E - (1+2\nu)\right)$. The quadratic $E \mapsto E^2 - 2E - (1+2\nu)$ has roots $1 \pm \sqrt{2+2\nu} = 1 \pm s$. Since $\nu > 0$ implies $s = \sqrt{2+2\nu} > \sqrt{2} > 1$, the root $1 - s$ is negative and therefore inadmissible for $E > 0$; the unique admissible root is $E^* = 1 + s$. As $x \mapsto E(x) = e^{a(x-x_0)}$ is a strictly increasing bijection from $\mathbb{R}$ onto $(0, \infty)$, the equation $E(x) = E^*$ has the unique solution

$$x_p = x_0 + \frac{\ln(1+s)}{a}. \tag{36}$$

The quadratic is negative for $E < E^*$ and positive for $E > E^*$, so G' decreases on $(-\infty, x_p)$ and increases on $(x_p, +\infty)$, whence x_p is the global minimizer of G'. □

G.2 PROOF OF LEMMA 4.2 (DEPTH CONTROL)

Proof. Substituting $E = E^* = 1 + s$ into the gain field and using the identity $1 + \nu + s = \frac{1}{2}s(s+2)$ (valid since $\nu = \frac{1}{2}(s^2 - 2)$):

$$G'(x_p) = \frac{\kappa_+\nu\big(1 - (1+s)\big)}{(2+s)(1+s+\nu)} = \frac{-\kappa_+\nu s}{(2+s)\cdot\frac{1}{2}s(s+2)} = -\,\kappa_+\,\frac{s^2-2}{(s+2)^2} = -\,\kappa_+\, d(\nu), \tag{37}$$

where $d(\nu) = (s^2-2)/(s+2)^2$ depends on ν alone.

For monotonicity, regard d as a function of s: $\frac{dd}{ds} = \frac{4(s+1)}{(s+2)^3} > 0$ for $s > \sqrt{2}$. Since $s(\nu) = \sqrt{2+2\nu}$ is smooth and strictly increasing with $ds/d\nu = 1/s$, the chain rule gives

$$\frac{dd}{d\nu} = \frac{4(s+1)}{s(s+2)^3} > 0. \tag{38}$$

The limits follow from $s \to \sqrt{2}$ as $\nu \to 0^+$ (giving $d \to 0$) and $s \to \infty$ as $\nu \to \infty$ (giving $d \to 1$). Strict monotonicity plus continuity and the two limits yield bijectivity of $d : (0, \infty) \to (0, 1)$ by the intermediate value theorem, and hence $\kappa_- = -\kappa_+ d(\nu)$ sweeps $(-\kappa_+, 0)$ bijectively.

For the inversion, solve $d(s+2)^2 = s^2 - 2$, i.e., $(d-1)s^2 + 4ds + (4d+2) = 0$, whose discriminant simplifies to $8(d+1)$. Retaining the root with $s > \sqrt{2}$:

$$d = \frac{|\kappa_-|}{\kappa_+}, \qquad s = \frac{2d + \sqrt{2(d+1)}}{1-d}, \qquad \nu = \frac{s^2-2}{2}. \tag{39}$$

Finally, the pull-to-push ratio satisfies

$$\frac{|G'(x)|}{\kappa_+} = \frac{\nu(E-1)}{(1+E)(E+\nu)} < 1 \iff 0 < E^2 + E + 2\nu \qquad (E > E(x_0) = 1 \text{ on the pull side}), \tag{40}$$

which holds for all $E, \nu > 0$; equivalently $\sup_\nu d(\nu) = 1$. Hence the restriction $|\kappa_-| < \kappa_+$ is intrinsic to the family. □

G.3 Proof of Proposition 4.3 (Scale Calibration)

Proof. Writing $q := E(1) = e^{-a\epsilon} \in (0, 1)$, the anchor condition $G'(1) = 1$ reads

$$\Psi(a) := \frac{\kappa_+ \nu(1-q)}{(1+q)(q+\nu)} = 1. \tag{41}$$

The map $a \mapsto q = e^{-a\epsilon}$ is a strictly decreasing bijection from $(0, \infty)$ onto $(0, 1)$, and $q \mapsto \kappa_+\nu(1-q)/[(1+q)(q+\nu)]$ is strictly decreasing on $(0, 1)$ (numerator decreasing and positive, denominator increasing and positive). The composition Ψ is therefore strictly increasing, with $\Psi(0^+) = 0$ (as $q \to 1$) and $\Psi(+\infty) = \kappa_+$ (as $q \to 0$). Since $\kappa_+ > 1$, the value 1 lies in the open range $(0, \kappa_+)$, so a unique root exists by the intermediate value theorem and strict monotonicity.

Clearing denominators turns the anchor condition into a quadratic in q:

$$q^2 + Bq - C = 0, \qquad B := \nu(\kappa_+ + 1) + 1 > 0, \quad C := \nu(\kappa_+ - 1) > 0. \tag{42}$$

Since $C > 0$, the product of the two roots is $-C < 0$: exactly one root is positive, and it is automatically less than 1 because substituting $q = 1$ gives $1 + B - C = 2 + 2\nu > 0$. Hence

$$q = \frac{2C}{B + \sqrt{B^2 + 4C}} \in (0, 1), \qquad a = -\frac{\ln q}{\epsilon} > 0. \tag{43}$$

The conjugate form is preferred over the algebraically equivalent $q = \left(-B + \sqrt{B^2 + 4C}\right)/2$: when $C \ll B^2$ the latter subtracts two nearly equal quantities and loses accuracy catastrophically, whereas the conjugate form only ever adds positive terms. □

G.4 Closed-Form Antiderivative

Substituting $E = e^{at}$ with $t = x - x_0$, so that $dx = dE/(aE)$, the gain field integrates via the partial-fraction decomposition

$$\frac{\nu(1-E)}{E(1+E)(E+\nu)} = \frac{1}{E} - \frac{2\nu}{(\nu-1)(1+E)} + \frac{\nu+1}{(\nu-1)(E+\nu)} \qquad (\nu \neq 1), \tag{44}$$

which yields the elementary antiderivative

$$\Phi(t) = \frac{\kappa_+}{a}\left[at - \frac{2\nu}{\nu-1}\ln(1+E) + \frac{\nu+1}{\nu-1}\ln(E+\nu)\right], \qquad E = e^{at}. \tag{45}$$

The point $\nu = 1$ is a removable singularity: taking the limit $\nu \to 1$ (e.g., by L'Hôpital's rule applied to the numerator) gives

$$\Phi(t) = \frac{\kappa_+}{a}\left[at - \ln(1+E) + \frac{2}{1+E}\right], \qquad \nu = 1. \tag{46}$$

Both branches satisfy $\Phi' = G'$ exactly (differentiate and compare with Eq. 9). Anchoring at $x = 1$ to enforce $G(1) = 1$ yields $G(x) = 1 + \Phi(x - x_0) - \Phi(1 - x_0)$.

G.5 Asymptotic Behaviour

Since $G' \to \kappa_+$ as $x \to -\infty$ and $G' \to 0$ as $x \to +\infty$, the graph of G possesses an oblique asymptote on the left and a horizontal one on the right:

$$x \to -\infty: \quad G(x) = \kappa_+ x + C_- + o(1), \qquad C_- = 1 - \kappa_+ x_0 + \frac{\kappa_+}{a}\big(C_\nu - \Phi(1 - x_0)\big), \tag{47}$$

$$x \to +\infty: \quad G(x) \longrightarrow 1 - \Phi(1 - x_0), \tag{48}$$

where $C_\nu = \frac{\nu+1}{\nu-1}\ln\nu$ for $\nu \neq 1$ and $C_1 = 2$. The right-hand limit is a derived quantity determined by $(\epsilon, \kappa_+, \kappa_-)$, not independently prescribed.

Consistency checks. Both hold identically by construction:

$$G'(x_p) = -\kappa_+ \frac{s^2 - 2}{(s+2)^2} = \kappa_-, \qquad G'(1) = \frac{\kappa_+ \nu(1-q)}{(1+q)(q+\nu)} = 1. \tag{49}$$

The special value $\nu = 1$ corresponds to $s = 2$ and hence to the single depth $d = \frac{1}{8}$, i.e., $\kappa_- = -\kappa_+/8$.

H Proof of Lemma 4.4

Proof. Let $h = G - \mathrm{id}$. By Proposition 4.1, G' decreases on $(-\infty, x_p)$ and increases on (x_p, ∞), with limits $\kappa_+ > 1$ at $-\infty$ and 0 at $+\infty$ and minimum $\kappa_- < 0$ at x_p; since $G'(1) = 1$, we get $G'(x) > 1$ for $x < 1$ and $G'(x) < 1$ for $x > 1$. Hence h increases up to its global maximum $h(1) = 0$ and decreases thereafter, i.e., $G(x) \leq x$ everywhere. Then $g(r) = 2 - G(2 - r) \geq 2 - (2 - r) = r$. □