Proximal Policy Optimization Suffices for On-Policy Reinforcement Learning

Rayane Bouftini[1,2] **Mohammed Benzaouia**[1]

[1]School of Science and Engineering, Al Akhawayn University, Ifrane, Morocco
[2]UM6P College of Computing, Benguerir, Morocco

Abstract

Proximal Policy Optimization (PPO) has been applied to a wide range of applications, from robotics to large language models (LLMs). Yet, despite its success, it still suffers from unstable updates, performance collapse, and extreme sensitivity to hyperparameters. Many alternatives have been proposed to alleviate these issues, often demonstrating empirically stronger performance on standard reinforcement learning (RL) benchmarks. In this work, we observe that PPO's training instability fundamentally stems from the brittle inter-coupling between its surrogate optimization hyperparameters which jointly dictate the effective size of policy updates. This complex coupling makes tuning quite difficult and, in practice, often leaves PPO under-performing. To mitigate this, we introduce a criterion on trajectories generated by behavior and target policies which can be used as an early stopping rule. This formulation effectively decouples the optimization dynamics, allowing the algorithm to adaptively scale its updates using a single threshold parameter. We additionally find a scaling law between this threshold parameter and the number of trajectory batches collected, further reducing the tuning burden. Using this simple criterion, we evaluate PPO on continuous control tasks from the DM Control Suite, outperforming prior policy gradient methods and revealing that many reported improvements were largely artifacts of under-performing PPO baselines. Our code is available at: https://github.com/rbouftini/adappo

1 INTRODUCTION

Large-scale pretraining complemented by online reinforcement learning (RL) has emerged as a successful paradigm for building capable and reliable agents. In robotics, behavior cloning [Schaal, 1996, Pomerleau, 1988] on expert demonstrations provides a strong initialization for robot policies [Dasari et al., 2019, Mandlekar et al., 2021, Lee et al., 2024, Sundaresan et al., 2025], but the resulting policy is typically suboptimal [Ross and Bagnell, 2010, Block et al., 2024]; online RL is then used to improve performance beyond the limits of the demonstrations [Rajeswaran et al., 2017, Torne et al., 2024]. Similarly, language models are pretrained on massive corpora to acquire broad knowledge [Radford et al., 2019, Brown et al., 2020], yet they are neither aligned with human preferences nor reliably capable of performing advanced cognitive tasks [Wei et al., 2022]. Reinforcement Learning from Human Feedback (RLHF) addresses alignment [Ouyang et al., 2022], and recent work on Reinforcement Learning with Verifiable Rewards (RLVR) has further improved the reasoning capabilities in latest models [Abdin et al., 2025, Guo et al., 2025, OpenAI, 2025].

Given the importance of online reinforcement learning for fine-tuning policies and achieving high performance, policy gradient (PG) methods [Williams, 1992, Sutton et al., 1999] emerged as the de facto standard within this setting. PG methods parameterize the policy and directly optimize the RL objective through local search techniques such as stochastic gradient ascent to find an optimal policy. However, vanilla PG is sample inefficient and unstable for large gradient steps. Trust Region Policy Optimization (TRPO) [Schulman et al., 2015a] remedies these issues by constructing a local approximation of the objective as a surrogate and maximizing it within a constrained trust region, i.e., a region in which the surrogate remains a reliable approximation, with the constraint based on policy divergence and motivated by monotonic improvement theory. TRPO can also be viewed as a variant of the natural policy gradient [Amari, 1998, Kakade, 2001], preconditioning the gradient with the inverse of the Fisher information matrix to account for the geometry of the parameter space. Because of TRPO's second-order optimization, Schulman et al. [2017] introduced Proximal Policy Optimization (PPO) as a sim-

Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence (UAI 2026), PMLR 337:640–657.

pler alternative. It preserves the constrained optimization idea through a heuristic clipping mechanism, and its first-order nature, simplicity, and empirical performance made it widely used.

Nevertheless, since PPO does not strictly enforce the divergence constraint, its performance is highly sensitive to hyperparameters that are required to ensure stable updates [Engstrom et al., 2020, Andrychowicz et al., 2021]. Since then, many algorithms have been proposed to remedy the training instability of PPO, mainly addressing the fact that the clipping mechanism fails to establish any rigorous constraint on policy divergences, which can lead to excessive policy updates and performance collapse [Wang et al., 2020, Xie et al., 2025]. Yet, across the many applications of policy optimization, training instability is still observed and ultimately hinders the final policy capabilities [Shao et al., 2024, Zhao et al., 2025, Liu et al., 2025].

In this work, we revisit the fact that the hyperparameter tuning of PPO is the primary cause of training instability. The proposed changes in the literature that show improved performance over PPO on standard RL benchmarks are then often merely artifacts of suboptimal hyperparameters leading to underperforming PPO baselines. Because the main PPO hyperparameters for surrogate optimization–the learning rate, number of policy updates per batch, and clipping range–are tightly coupled and jointly determine the effective policy update, this coupling makes hyperparameter tuning quite challenging in practice. To that end, we propose to decouple these hyperparameters and let them be implicitly governed by a single criterion, significantly simplifying the search of optimal training parameters.

The main observation is that the core idea of surrogate construction is entirely based on the accuracy of the local approximation. The clipping mechanism alone does not guarantee this accuracy, making it necessary to explicitly incorporate a criterion to control the estimation error. When viewing the surrogate construction as an importance sampling procedure over trajectory distributions, it is sufficient to control the divergence between these distributions to trust the surrogate optimization. The maximum tolerated divergence, which can be seen as a threshold condition, is then directly related to the number of samples (trajectory batches) used to construct the surrogate. Since the forward KL divergence is not tractable, the clipping mechanism merely ensures a bounded relationship between the forward and reverse KL divergences, allowing the computable reverse KL to serve as a reliable substitute. This formulation allows us to derive a scaling law for the threshold criterion based on the number of samples collected, with the clipping parameter contributing only as a scaling factor. The iterative optimization of the surrogate then relies entirely on this criterion, effectively absorbing the impact of the learning rate and number of policy updates, rendering the optimization more stable.

In summary, our main contributions are as follows:

- We theoretically show that the trajectory divergence between the behavior and target policies controls the accuracy of the surrogate objective. We further demonstrate how constraining the policy probability ratio through a clipping mechanism is necessary to bound the geometry of the space and arrive at a practical criterion.
- From this framework, we derive a scaling law showing that the maximum tolerated trajectory divergence scales as $\frac{1-\epsilon}{1+\epsilon} \ln \frac{N}{c_0}$, where N is the number of trajectory batches and ϵ is the clipping parameter. We empirically confirm this relationship across multiple settings.
- This theory yields a simple trajectory-based criterion that decouples PPO's three key hyperparameters: the learning rate and number of policy updates adapt implicitly to the criterion, while the clipping range enters only as a known scaling factor, reducing hyperparameter tuning to a single parameter.
- We evaluate PPO equipped with this criterion against 7 policy gradient methods on 13 continuous control tasks from the DM Control Suite, achieving greater stability and overall best performance.

2 BACKGROUND AND NOTATION

We formulate the reinforcement learning problem with an episodic Markov Decision Process (MDP) [Puterman, 2014] that is described by the tuple $(\mathcal{S}, \mathcal{A}, r, P, \mu, T)$, where $\mathcal{S}$ and $\mathcal{A}$ are the state and action spaces respectively; $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function; $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the transition probability function; and μ is the initial state distribution. The agent's goal is to learn a policy $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$ as a map from states to probability distributions over actions, with $\pi(a|s)$ denoting the probability of taking action a in state s. We define the performance measure $J(\pi)$ as the expected return over a finite horizon T, $J(\pi) := \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T-1} r(s_t, a_t)\right]$, where $\tau = (s_0, a_0, s_1, a_1, \ldots, s_T)$ denotes a trajectory consisting of a sequence of states and actions under π.

We define the following useful quantities in the standard way: the action-value and value functions

$$Q^\pi(s_t, a_t) := \mathbb{E}_{s_{t+1}, a_{t+1}, \cdots \sim \pi}\left[\sum_{l=t}^{T-1} r(s_l, a_l)\right], \qquad (1)$$
$$V^\pi(s_t) := \mathbb{E}_{a_t\sim\pi(\cdot|s_t)}[Q^\pi(s_t, a_t)],$$

and the advantage function $A^\pi(s_t, a_t) := Q^\pi(s_t, a_t) - V^\pi(s_t)$, which measures how good the chosen action is compared to the average action at state s_t, and finally the unnormalized episodic state-visitation frequency $\rho_\pi(s) := \sum_{t=0}^{T-1} \Pr(s_t = s \mid \pi)$. Policy gradient methods parametrize

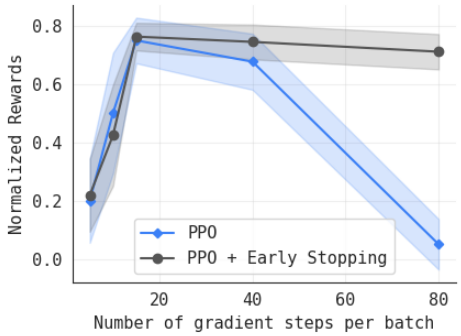


Figure 1: **PPO with Early Stopping.** We report the performance of PPO and PPO with early stopping for different number of gradient steps K (equivalently, the number of epochs). For PPO with early stopping, we use a threshold $\delta = 0.05$ found best after tuning. The performance is aggregated over six environments from DM Control Suite, each for 10 random seeds, and we report the average normalized returns for the last 15% of training steps. We observe that early stopping based on *policy* KL does not yield higher performance than simple PPO with tuned K, but reduces the dependency on it.

the policy (e.g., with a neural network) by a vector of parameters $\theta \in \mathbb{R}^n$ and optimize $J(\theta)$ in search of optimal parameters θ^* via gradient ascent. We slightly abuse notation and write $J(\theta)$ instead of $J(\pi_\theta)$. The gradient of the objective has the tractable form

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t \mid s_t)\, R(\tau)\right], \quad (2)$$

where $R(\tau)$ is the sum of rewards in a trajectory. The gradient in (2) is taken under the current policy, so each update requires new trajectories. This, in addition to the difficulty of choosing a suitable step size, makes policy gradient methods sample inefficient.

Kakade and Langford [2002] derived a useful expression that relates the performance of a policy π' to that of another policy π:

$$\begin{aligned} J(\pi') &= J(\pi) + \mathbb{E}_{\tau\sim\pi'}\left[\sum_{t=0}^{T-1} A^\pi(s_t, a_t)\right] \\ &= J(\pi) + \sum_s \rho_{\pi'}(s) \sum_a \pi'(a \mid s)\, A^\pi(s,a). \end{aligned} \quad (3)$$

Maximizing the last term guarantees an improved (or at least no worse) policy, but still requires access to both the state-visitation frequency and action distribution of π'. This prompted the construction of the following local approximation, where the state-visitation frequency is taken under the policy π Schulman et al. [2015a]:

$$L_\pi(\pi') = J(\pi) + \sum_s \rho_\pi(s) \sum_a \pi'(a \mid s)\, A^\pi(s,a). \quad (4)$$

In order to quantify the gap between $L_\pi(\pi')$ and the true objective $J(\pi')$, it is useful to introduce the following statistical divergences. These will also be helpful throughout the paper.

Definition 1. *Let $f : \mathbb{R}_+ \to \mathbb{R}$ be a convex function such that $f(1) = 0$. For two probability mass functions P and Q on discrete sample space $\mathcal{X}$, the* f-divergence *between P and Q is defined as*

$$D_f(P\|Q) := \sum_{x\in\mathcal{X}} Q(x)\, f\left(\frac{P(x)}{Q(x)}\right). \quad (5)$$

Taking $f(x) = \frac{1}{2}|x-1|$ gives the total variation divergence (D_{TV}), and $f(x) = x\log x$ the Kullback-Leibler divergence (D_{KL}). It was then shown that [Schulman et al., 2015a]:

Theorem 1. *Given $\alpha := \max_s D_{\mathrm{TV}}\big(\pi(\cdot \mid s) \,\|\, \pi'(\cdot \mid s)\big)$, the following bound holds:*

$$J(\pi') \;\geq\; L_\pi(\pi') \;-\; 2\,\varepsilon\, T(T-1)\,\alpha^2. \quad (6)$$

where $\varepsilon := \max_{s,a}\big|A^\pi(s,a)\big|$.

This is the analogue of the TRPO's Theorem 1 bound for an undiscounted, finite-horizon MDP, where $\sum_{t=0}^{T-1} t = \frac{T(T-1)}{2}$ shows up instead of $\sum_{t=0}^{\infty}\gamma^t t = \frac{\gamma}{(1-\gamma)^2}$.

In practice, within the set of parameterized policies Π_θ, TRPO does not maximize this lower bound directly. Instead, it maximizes $L_\pi(\pi')$ while enforcing a trust region constraint with hyperparameter δ. This can be interpreted as maximizing a local approximation of $J(\pi')$; a '*surrogate objective*' that disregards the discrepancy between the state distributions $d_\pi(s) := \frac{1}{T}\rho_\pi(s)$, which is a reasonable approximation when π and π' are close. The actions sampled from π can be corrected to reflect π' through importance sampling weights $\frac{\pi'(a|s)}{\pi(a|s)}$. This leads to the TRPO objective:

$$\begin{aligned} \max_{\pi'} \quad & \mathbb{E}_{s\sim d_\pi,\, a\sim\pi}\left[\frac{\pi'(a \mid s)}{\pi(a \mid s)}\, A^\pi(s,a)\right] \\ \text{subject to} \quad & \mathbb{E}_{s\sim d_\pi}\Big[D_{\mathrm{KL}}\big(\pi(\cdot \mid s) \,\|\, \pi'(\cdot \mid s)\big)\Big] \;\leq\; \delta. \end{aligned} \quad (7)$$

The complexity of TRPO's constrained optimization motivated PPO, with a simpler first-order objective that clips the importance ratio to $[1-\epsilon,\, 1+\epsilon]$, heuristically discouraging large policy updates:

$$L^{\mathrm{PPO}} = \mathbb{E}_{\substack{s\sim d_\pi \\ a\sim\pi}}\Big[\min\big(r(s,a)\,A^\pi,\; \tilde{r}(s,a)\,A^\pi\big)\Big], \quad (8)$$

where $r(s,a) = \frac{\pi'(a|s)}{\pi(a|s)}$, $\tilde{r}(s,a) = \mathrm{clip}(r(s,a),\, 1-\epsilon,\, 1+\epsilon)$, and A^π is shorthand for $A^\pi(s,a)$.

3 METHODOLOGY

3.1 CONSTRUCTION OF THE SURROGATE OBJECTIVE

The instability of policy updates in PPO can be attributed to the absence of an explicit constraint on the divergence between policies, allowing the new policy to move further from the current one. One natural modification is to monitor the KL divergence from exceeding a predefined threshold, thereby ensuring that the surrogate objective is optimized only while it remains reliable and within the trust region. As shown in Figure 1, this early stopping mechanism makes PPO less sensitive to the number of gradient steps K. However, existing methods that incorporate this criterion [Achiam, 2018, Wang et al., 2020, Dossa et al., 2021, Sun et al., 2022] introduce the maximum tolerated divergence as an additional hyperparameter that must be jointly tuned with the learning rate and clipping range, making hyperparameters tuning even more brittle.

Our first observation is that the number of gradient steps K should not be held fixed and tuned a priori. Instead, the iterative optimization of the surrogate should be governed entirely by the KL criterion. By taking computationally cheap steps in the Euclidean parameter space while strictly monitoring the divergence in the distribution space, we actively prevent the policy from overshooting. This dynamically couples the step size and the number of updates per batch to the KL boundary. Geometrically, this adaptive approach ensures that the sequence of Euclidean updates respects the curvature of the distribution space, achieving a similar goal to the natural gradient without requiring second-order computation. In contrast, a fixed number of Euclidean steps disregards the underlying geometry entirely, risking overshooting or undershooting the trust region depending on the local curvature.

However, to make this geometric constraint truly reliable, we must reconsider how we measure this divergence. While bounding the action distribution at individual states is standard practice, we observe that it is far more faithful to directly constrain the space of *trajectories* induced by the policies. Writing P^π for the distribution over trajectories τ induced by policy π with initial state distribution μ, we provide the following results:

Theorem 2. *Given* $\alpha := D_{\mathrm{TV}}\big(P^\pi \,\|\, P^{\pi'}\big)$*, then the following bound holds:*

$$J(\pi') \;\geq\; L_\pi(\pi') \;-\; 2\,\varepsilon\,T\,\alpha^2. \tag{9}$$

where $\varepsilon := \max_{s,a}\big|A^\pi(s,a)\big|$.

Theorem 2 guarantees that when the trajectory distributions P^π and $P^{\pi'}$ are close, within α, the local approximation $L_\pi(\pi')$ is more accurate than when only the action distributions are constrained to be close. This yields a better surrogate for optimization, since constraining trajectories provides tighter control over the state distribution mismatch. To make this precise, we introduce the following lemmas where we denote the state marginals at timestep t for a policy π as $p^t_\pi(s) := \Pr(S_t = s \mid \pi)$.

Lemma 1. *For any two policies* π *and* π' *with initial state distribution* μ*. Given* $\alpha := \max_{s\in\mathcal{S}} D_{\mathrm{TV}}\big(\pi'(\cdot \mid s) \,\|\, \pi(\cdot \mid s)\big)$*, then for all* $t \in \{0,\dots,T-1\}$ *we have* $D_{\mathrm{TV}}(p^t_{\pi'} \,||\, p^t_\pi) \;\leq\; \alpha\, t$.

By bounding the difference between π and π' to be at most α in total variation at every state, the mismatch between the state distributions after t steps grows at most linearly in t. This lemma is key to controlling state distribution shift in Theorem 1.

Lemma 2. *For any two policies* π *and* π' *with initial state distribution* μ*. Given* $\alpha := D_{\mathrm{TV}}\big(P^\pi \,\|\, P^{\pi'}\big)$*, then for all* $t \in \{0,\dots,T-1\}$ *we have* $D_{\mathrm{TV}}(p^t_{\pi'} \,||\, p^t_\pi) \leq \alpha$.

Whereas Lemma 1 shows that errors accumulate linearly with the number of timesteps, a direct constraint on the trajectory divergence uniformly bounds the state distribution mismatch at every time step by α, yielding tighter control of the surrogate gap.

We first note the relationship between TV divergence and KL divergence $D_{\mathrm{TV}}(p||q) \leq \sqrt{\frac{1}{2}D_{KL}(p||q)}$ to get:

$$\begin{aligned} J(\pi') \;&\geq\; L_\pi(\pi') \;-\; 2\,\varepsilon\,T\,D_{\mathrm{TV}}\big(P^\pi \,\|\, P^{\pi'}\big)^2 \\ &\geq\; L_\pi(\pi') - \varepsilon\,T D_{\mathrm{KL}}\big(P^\pi \,\|\, P^{\pi'}\big). \end{aligned} \tag{10}$$

Consequently, the local approximation $L_\pi(\pi')$, constructed to disregard the state distribution mismatch, remains faithful to the true objective provided that the trajectory distributions induced by the two policies are close. This motivates optimizing the surrogate subject to a trust-region constraint on the trajectory divergence:

$$\begin{aligned} \max_{\pi'} \quad & \mathbb{E}_{s\sim d_\pi,\, a\sim\pi}\left[\frac{\pi'(a \mid s)}{\pi(a \mid s)}\, A^\pi(s,a)\right] \\ \text{subject to} \quad & D_{\mathrm{KL}}\big(P^\pi \,\|\, P^{\pi'}\big) \leq \delta. \end{aligned} \tag{11}$$

The construction of the constraint does not require any heuristic approximation to monotonic improvement theory Schulman et al. [2015a]. In contrast, the formulation in TRPO based on the average KL divergence between policies $\mathbb{E}_s\big[D_{\mathrm{KL}}(\pi \,\|\, \pi')\big]$, deviates from the theory in Theorem 1 which requires a uniform pointwise bound over all states $\max_s D_{\mathrm{KL}}(\pi \,\|\, \pi')$, and this is harder to maintain in practice; thus, the former is typically applied instead.

In the next section, we view the introduction of this criterion from an importance sampling perspective, which reveals that constraining the probability ratio is a necessary condition for the criterion to be reliable, while also shedding light on how the trust region size should be set.

3.2 OPTIMIZING POLICY IMPROVEMENT VIA IMPORTANCE SAMPLING

Another view is to optimize policy improvement directly via importance sampling at the trajectory level:

$$\begin{aligned} J(\pi') - J(\pi) &= \mathbb{E}_{\tau\sim\pi'}\left[\sum_{t=0}^{T-1} A^{\pi}(s_t, a_t)\right] \\ &= \mathbb{E}_{\tau\sim\pi}\left[\frac{P^{\pi'}(\tau)}{P^{\pi}(\tau)}\sum_{t=0}^{T-1} A^{\pi}(s_t, a_t)\right], \end{aligned} \tag{12}$$

with $\frac{P^{\pi'}(\tau)}{P^{\pi}(\tau)} = \prod_{t=0}^{T-1}\frac{\pi'(a_t|s_t)}{\pi(a_t|s_t)}$. However, the variance of the estimator grows exponentially with T due to the product of action probabilities [Meuleau et al., 2000, Jie and Abbeel, 2010]. Although numerical tricks can stabilize computation, they do not reduce variance. For this reason, the surrogate $L_{\pi}(\pi')$ provides a lower variance estimate by ignoring state-distribution mismatch and avoiding the full product of ratios; meanwhile, these two objectives remain close within the trust region.

Nevertheless, since $L_{\pi}(\pi')$ is a good *local* approximation of $J(\pi')$, as shown in Section 3.1, the trajectory-level importance sampling perspective is valuable for guiding the choice of δ, as it relates to the sample size N. In fact, Sanz-Alonso [2018] and Chatterjee and Diaconis [2018] establish that with $N = \exp(D_{\mathrm{KL}}(P^{\pi'}\|P^{\pi}) - t)$, for some $t > 0$, the trajectory-level IS estimate may fail with substantial probability. Therefore, a *necessary* condition for the IS estimate to be reliable, i.e., close in the L^1 sense to the policy improvement $J(\pi') - J(\pi)$, is to have $N \geq \exp(D_{\mathrm{KL}}(P^{\pi'}\|P^{\pi}) - t)$. We can then use this as a condition to find when the surrogate is no longer trustworthy. Therefore, at each optimization step, taking $P^{\pi'}$ as the target and P^{π} as the sampling distribution, we continue optimizing on the current batch as long as

$$c_0 \cdot \exp\big(\mathrm{D}_{\mathrm{KL}}(P^{\pi'}\|P^{\pi})\big) \leq N, \text{ for some } c_0 \in (0, 1]. \tag{13}$$

During optimization, the target distribution $P^{\pi'}$ changes at every gradient descent step. Once the updated target violates the divergence condition in Equation 13, the estimator (and hence the surrogate) can no longer be considered accurate on that batch, and new samples should be collected to construct a new objective.

Although theoretically appealing that the KL criterion scales with the natural logarithm of the number of batches collected, it is practically not very useful since it depends on the unknown target distribution $P^{\pi'}$, in fact the very issue importance sampling aims to avoid. Furthermore, bounding the tractable reverse KL, $\mathrm{D}_{\mathrm{KL}}(P^{\pi} \,\|\, P^{\pi'})$, is not geometrically equivalent. However, by explicitly trimming the per-step probability ratio and effectively constraining the geometry of the space, the following theorem allows us to safely use the reverse KL as a stopping criterion while providing its exact scaling relationship:

Theorem 3 (Informal)**.** *Let the probability ratio* $r_t := \frac{\pi'(a_t|s_t)}{\pi(a_t|s_t)} \in [1-\epsilon,\ 1+\epsilon]$*, with* $\epsilon \in (0, 1)$*. Under mild assumptions, with a number of trajectories collected* $N > 0$*, and constant* $c_0 \in (0, 1]$*, we can reformulate the criterion in Equation 13 as:*

$$\mathrm{D}_{\mathrm{KL}}(P^{\pi} \,\|\, P^{\pi'}) \leq \frac{1-\epsilon}{1+\epsilon}\ln\left(\frac{N}{c_0}\right)\left(1 + \mathcal{O}\left(\frac{1}{\sqrt{T}}\right)\right). \tag{14}$$

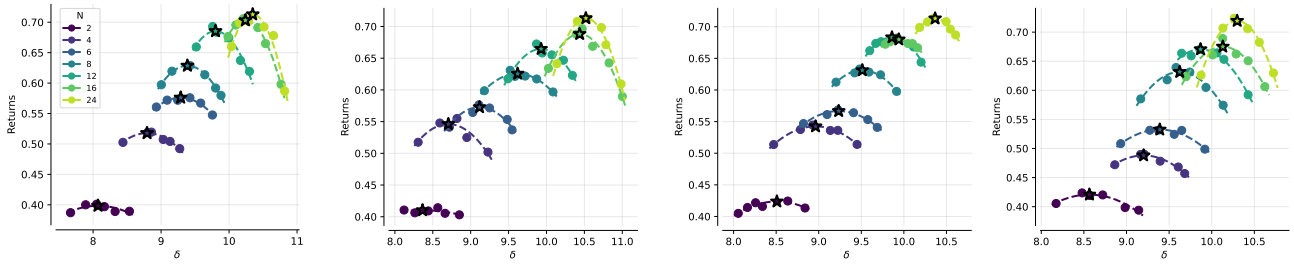
Intuitively, the trajectory distributions P^{π} and $P^{\pi'}$ can be viewed as two distinct points on a statistical manifold. The shortest path, or geodesic, connecting these two distributions is defined by the exponential twist density [Dabak and Johnson, 2003]. This geodesic induces a Fisher information metric on the manifold, which corresponds exactly to the variance of the log-probability ratio with respect to the intermediate twisted densities. Because both the forward and reverse Kullback-Leibler divergences are obtained by integrating this shared metric along the path with different time-weightings, bounding the reverse KL does not universally bound the forward KL unless the underlying geometry of the space is explicitly constrained. Essentially, in the long-horizon regime, restricting the per-step probability ratio to be within $[1-\epsilon,\ 1+\epsilon]$ strictly limits the curvature of the statistical manifold. This guarantees a condition number of $\frac{1+\epsilon}{1-\epsilon}$ for the Fisher metric, which provides the necessary geometric bound to safely control the forward KL using the reverse KL. Proof is deferred to Appendix A.

Therefore, within PPO, it is sufficient to control the reverse KL weighted inversely by the condition number to maintain the surrogate accuracy guarantee. Consequently, the larger the allowed deviation ϵ of the per-step probability ratio, the stricter the reverse KL bound must become.

Finally, introducing the KL criterion $\mathrm{D}_{\mathrm{KL}}(P^{\pi} \,\|\, P^{\pi'}) \leq \delta(N, \epsilon)$ into the PPO algorithm as an "early stopping" mechanism yields Algorithm 1. For the long-horizon regime $T \to \infty$, this threshold is explicitly defined as $\delta(N, \epsilon) = \frac{1-\epsilon}{1+\epsilon}\ln\left(\frac{N}{c_0}\right)$, where c_0 acts as a precision parameter.

3.3 PRACTICAL ADVANTAGES

The KL criterion decouples PPO's three key surrogate optimization hyperparameters. The learning rate no longer dictates stability; it merely controls wall-clock efficiency. Any step size within a stable optimization regime is valid, as the trajectory divergence criterion absorbs variations in step size by dynamically scaling the number of epochs, as shown in Table 1. Similarly, from Theorem 3, the clip epsilon enters the criterion only through the scaling factor $\frac{1-\epsilon}{1+\epsilon}$, so any choice within $\epsilon \in (0, 1)$ is automatically reflected in the trust region size without separate tuning.



(a) $\epsilon = 0.05$ (b) $\epsilon = 0.1$ (c) $\epsilon = 0.15$ (d) $\epsilon = 0.2$

Figure 2: **Threshold sweeps across different clipping ranges.** Performance (returns) is plotted as a function of the trajectory divergence threshold δ for varying numbers of collected batches N. For each N, a quadratic fit is applied (dashed lines), and the empirical optimal threshold is marked. Across all clipping values ϵ, we consistently observe that the optimal δ increases monotonically as more batches are collected.

Learning Rate	Returns	Average Epochs (K)
1×10^{-5}	757 ± 26	28
6×10^{-5}	760 ± 20	21
1×10^{-4}	762 ± 18	14
3×10^{-4}	760 ± 23	12

Table 1: **Impact of varying the learning rate on final performance and the average number of gradient steps (K) per batch.** The trajectory divergence criterion automatically scales K inversely to the step size, maintaining policy stability and consistent returns on DM Control Suite across a wide range of learning rates without the need for co-tuning.

4 RESULTS

4.1 SCALING LAW

In practice, Equation 14 provides a heuristic scaling relationship between the threshold δ, the sample size N, and the clipping parameter ϵ. To empirically verify this, we conduct large-scale experiments (necessary due to the inherent randomness in RL) to identify the optimal δ for various values of N under a fixed ϵ. To evaluate the performance of each (δ, N) pair, we use the normalized average reward over the last 15% of training steps across six environments from the DeepMind Control Suite [Tassa et al., 2018, Tunyasuvunakool et al., 2020] and 10 random seeds, allocating a fixed training budget of 400 episodes per run.

The results, shown in Figure 2, demonstrate that the optimal threshold δ increases monotonically with N. Concretely, for each ϵ setting, after extracting the optimal δ, we are able to fit a scaling law that matches the logarithmic scaling predicted by Theorem 3.

Furthermore, as illustrated in Figure 3 for a specific choice of ϵ, the empirical fit closely recovers the predicted $\frac{1-\epsilon}{1+\epsilon}$ scaling factor. This relationship holds consistently across other values of ϵ, with the corresponding fitted equations provided in Appendix D.

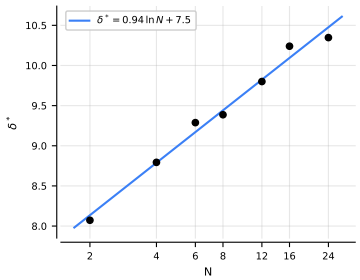


Figure 3: **Optimal threshold δ as a function of the number of samples for $\epsilon = 0.05$.** The N-axis is on *logarithmic* scale.

4.2 HYPERPARAMETER TUNING

Although our proposed trajectory-based criterion significantly reduces PPO's sensitivity to hyperparameters, conducting a fair performance evaluation requires strictly tuned baselines. Therefore, we perform extensive hyperparameter tuning for 7 other policy optimization algorithms: AlphaPPO [Xu et al., 2023], ESPO [Sun et al., 2022], RPO [Gan et al., 2024], SPO [Xie et al., 2025], SPU [Vuong et al., 2018], TRPO [Schulman et al., 2015a], and TR-PPO-RB [Wang et al., 2020]. We reuse the official implementation whenever it was made available by the authors. For the tuning method, we employ Bayesian Optimization with Gaussian Processes and an Upper Confidence Bound (UCB) acquisition function via OSS Vizier [Song et al., 2022, 2024]. The optimization objective is the average normalized return computed across 5 Gymnasium environments [Towers et al., 2024] (Walker2d, Hopper, Swimmer, HalfCheetah, and InvertedPendulum), evaluated over 8 random seeds each. Importantly, we deliberately tune on a disjoint set of environments from those used for our final testing to prevent benchmark overfitting. In total, we conduct 40 Bayesian optimization trials to search for the optimal hyperparameters for each algorithm.

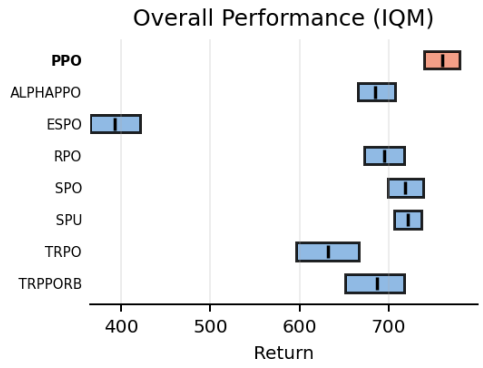


Figure 4: **Aggregated performance.** Overall results across 13 DM Control Suite tasks and 10 random seeds. PPO achieves the strongest overall performance among all evaluated baselines.

To isolate the effects of the algorithms' core mechanics, we only tune algorithm-specific hyperparameters. For general hyperparameters, such as network architecture, discount factor, and advantage estimation, we adopt optimal settings firmly established in the literature which can be applied to any policy gradient method [Engstrom et al., 2020, Andrychowicz et al., 2021, Huang et al., 2022]. Specifically, the policy is parameterized by a neural network with two hidden layers of 64 units each and Tanh activations. The network is orthogonally initialized with a standard deviation of $\sqrt{2}$ for the hidden layers and 0.01 for the final layer to encourage early exploration. A wider architecture is used for the separate value function network, comprising two hidden layers of 256 units, which is fitted using a mean squared error loss against the empirical sum of observed rewards. For continuous action spaces, the policy outputs the mean of a Gaussian distribution with state-dependent means and independent standard deviations [Duan et al., 2016]. Optimization is performed using the Adam optimizer [Kingma and Ba, 2014] with a linear learning rate warmup and a stable schedule [Wen et al., 2024]. We also apply gradient clipping [Zhang et al., 2019] except for and TRPO. Advantages are normalized across the batch and estimated via Generalized Advantage Estimation (GAE) [Schulman et al., 2015b] with $\lambda = 0.97$ and discount factor $\gamma = 0.99$.

4.3 PERFORMANCE EVALUATION

Once the hyperparameters are tuned for each algorithm, we evaluate them on downstream tasks from the DM Control Suite, specifically considering 13 new state-based continuous control tasks. In our experiments, we use 10 random seeds per environment with horizon $T = 1000$, $N = 16$ and 400 training episodes, reporting 95% bootstrap confidence intervals in all plots and standard deviations in the performance tables. The optimal hyperparameters used following tuning can be found in Appendix C.

Figure 4 shows the overall comparison results between algorithms. To assess the aggregated performance across all tasks robustly, we use the interquartile mean (IQM) metric, following the recommendations of Agarwal et al. [2021]. Additionally, we report the final performance for each individual environment in Table 2. These results suggest that with this simple addition to PPO, we are able to obtain stable results and achieve the overall best performance.

Notably, Early Stopping Policy Optimization (ESPO), which removes the clipping mechanism entirely and relies solely on a policy KL divergence criterion for early stopping, performs poorly across these tasks. The theoretical framework established in Section 3.2 directly explains this failure since bounding the reverse KL divergence without simultaneously constraining the geometry of the space fails to bound the forward KL divergence. As we established, strictly bounding the forward KL is necessary to maintain an accurate surrogate objective.

4.4 ABLATION STUDY

Our proposed approach allows for an *adaptive* number of gradient steps on the policy objective, governed by the trajectory trust region criterion. In practice, however, training involves *joint* optimization of both the policy and the value function (for advantage estimation). Consequently, the number of gradient steps taken on the value network could also independently influence overall performance.

We investigated whether the number of value function updates should be held fixed or allowed to adapt alongside the policy updates. As shown in Table 3, coupling the value function epochs to the adaptive policy criterion yields stable and comparable performance. Nevertheless, we recommend keeping the number of value function updates fixed. Treating it as an independent hyperparameter, much like the GAE parameter λ or the discount factor γ, properly isolates the core policy optimization mechanism from the specific implementation details of the critic.

5 RELATED WORK

Implementation and hyperparameter sensitivity. A series of large-scale studies have shown that the performance of policy optimization algorithms, or specifically of PPO, depends critically on implementation details and hyperparameter choices [Tucker et al., 2018, Engstrom et al., 2020, Andrychowicz et al., 2021, Huang et al., 2022]. In particular, they argue that code-level optimizations often matter more than the policy algorithm itself. Our results align with these findings in the sense that we also demonstrate that a properly stabilized PPO outperforms many newer policy gradient methods that previously claimed superiority

Table 2: **Environment performance comparison on the DeepMind Control Suite.** PPO achieves competitive and stable performance across a wide variety of tasks compared to the tuned baselines.

Environment	PPO	ALPHAPPO	ESPO	RPO	SPO	SPU	TRPO	TRPPORB
ball-in-cup-catch-v0	941 ± 7	972 ± 5	865 ± 139	974 ± 4	968 ± 4	971 ± 3	956 ± 9	969 ± 5
cartpole-balance-sparse-v0	721 ± 155	759 ± 85	960 ± 44	648 ± 212	587 ± 102	714 ± 84	839 ± 172	880 ± 100
cartpole-balance-v0	748 ± 58	597 ± 70	625 ± 119	646 ± 99	697 ± 138	700 ± 75	655 ± 141	732 ± 161
cheetah-run-v0	596 ± 35	567 ± 13	339 ± 61	605 ± 15	685 ± 26	586 ± 8	513 ± 26	526 ± 10
dog-walk-v0	546 ± 16	603 ± 34	181 ± 48	455 ± 61	436 ± 76	482 ± 50	313 ± 29	631 ± 81
finger-spin-v0	799 ± 21	788 ± 21	387 ± 65	750 ± 71	802 ± 21	783 ± 39	673 ± 112	684 ± 58
fish-upright-v0	941 ± 7	934 ± 13	869 ± 55	930 ± 8	945 ± 4	941 ± 7	924 ± 16	926 ± 11
quadruped-fetch-v0	396 ± 34	381 ± 34	338 ± 36	365 ± 6	355 ± 16	379 ± 19	369 ± 6	390 ± 24
quadruped-walk-v0	872 ± 101	743 ± 121	441 ± 205	713 ± 115	742 ± 158	827 ± 130	831 ± 101	809 ± 142
reacher-easy-v0	953 ± 6	961 ± 5	693 ± 263	957 ± 9	949 ± 6	966 ± 5	923 ± 53	946 ± 7
reacher-hard-v0	915 ± 10	925 ± 14	571 ± 281	907 ± 57	934 ± 11	943 ± 12	748 ± 74	872 ± 26
walker-stand-v0	964 ± 3	977 ± 2	805 ± 131	968 ± 2	963 ± 3	968 ± 5	974 ± 1	970 ± 2
walker-walk-v0	902 ± 61	821 ± 85	506 ± 142	915 ± 17	938 ± 8	923 ± 14	835 ± 93	788 ± 102

upon their introduction. Ultimately, our method, rather than simply documenting this well-established hyperparameter sensitivity within the RL community, attempts to resolve it.

Value Function Epochs	Returns
Fixed	760 ± 23
Adaptive	758 ± 34

Table 3: **Ablation comparing fixed versus adaptive epochs for value function fitting.** Returns are evaluated following the setup detailed in Section 4.3 on the DM Control Suite, using a learning rate of $\alpha = 3 \times 10^{-4}$.

Early stopping in PPO. Several works incorporate early stopping by monitoring the per-state policy KL divergence during surrogate optimization [Achiam, 2018, Dossa et al., 2021, Sun et al., 2022, Wang et al., 2020]. While this prevents excessive updates, these methods simply introduce the divergence threshold as an additional free hyperparameter that must be jointly tuned with existing ones. Our approach is similar in that we rely on iterative surrogate optimization governed by a stopping criterion. However, we adopt a trajectory-level perspective, which cannot be captured by relying on per-step policies alone. This fundamentally shifts the threshold from a free variable to a predictable boundary, ultimately allowing us to derive a principled scaling law and eliminate the reliance on ad-hoc tuning. Notably, ESPO [Sun et al., 2022] removes clipping entirely and relies solely on policy KL early stopping. As our theory predicts and experiments confirm, this performs poorly since, without clipping to constrain the geometry of the space, the computable reverse KL cannot reliably bound the intractable forward KL.

6 DISCUSSION AND LIMITATIONS

In this work, we introduced a simple criterion for assessing the accuracy of the surrogate objective in PPO. The key observation is that each optimization step constructs the surrogate using a fixed behavior policy while the target policy changes at every gradient step. By reinterpreting this as an importance sampling problem at the trajectory level, we formulated a condition under which the surrogate can no longer be trusted to accurately reflect the true objective, and further optimization steps cannot be expected to yield an improved policy. While this statistical perspective naturally involves the intractable forward KL divergence, we showed that explicitly constraining the geometry of the space–as PPO already does through its clipping mechanism–allows us to derive a practical criterion based on the practical reverse KL. Incorporating this criterion into PPO effectively decouples the hyperparameters that jointly controlled the effective size of policy updates, removing the brittleness to hyperparameter tuning. Furthermore, in the long-horizon regime, we derived a scaling law for this threshold, showing how it can be set as a function of the number of trajectory batches and the clipping parameter without relying on ad-hoc tuning.

Our work has three main limitations. First, while we focused on decoupling PPO's core optimization hyperparameters, our experimental setup adopted the many code-level optimizations established in the RL literature–such as network initialization, architecture choices, and advantage estimation–that are shared across policy gradient methods. A deeper investigation into how these implementation choices interact with exploration and ultimately affect the final policy remains an important direction for future work. Second, the derived scaling law relies on the long-horizon regime $T \to \infty$. While our experiments show that the scaling holds well for typical RL tasks on DM Control Suite, for shorter-horizon tasks, the correction term becomes non-negligible and the exact scaling curve may deviate from the theoretical prediction. Nevertheless, even without a closed-form scaling

law, the criterion can still be tuned as a single parameter while retaining the hyperparameter decoupling benefits described above. Finally, our empirical evaluation is limited to continuous control on the DM Control Suite. While the criterion and its scaling law are not tied to this setting, validating them in other domains and most notably language-model fine-tuning with RLHF and RLVR, where PPO's training instability is itself a central concern, remains an important direction for future work.

References

Marah Abdin, Sahaj Agarwal, Ahmed Awadallah, Vidhisha Balachandran, Harkirat Behl, Lingjiao Chen, Gustavo de Rosa, Suriya Gunasekar, Mojan Javaheripi, Neel Joshi, et al. Phi-4-reasoning technical report. *arXiv preprint arXiv:2504.21318*, 2025.

Joshua Achiam. Spinning Up in Deep Reinforcement Learning. 2018.

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.

Shun-Ichi Amari. Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276, 1998.

Marcin Andrychowicz, Anton Raichuk, Piotr Stańczyk, Manu Orsini, Sertan Girgin, Raphaël Marinier, Leonard Hussenot, Matthieu Geist, Olivier Pietquin, Marcin Michalski, Sylvain Gelly, and Olivier Bachem. What matters for on-policy deep actor-critic methods? a large-scale study. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=nIAxjsniDzg.

Adam Block, Dylan J Foster, Akshay Krishnamurthy, Max Simchowitz, and Cyril Zhang. Butterfly effects of SGD noise: Error amplification in behavior cloning and autoregression. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=CgPs04l9TO.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Sourav Chatterjee and Persi Diaconis. The sample size required in importance sampling. *The Annals of Applied Probability*, 28(2):1099–1135, 2018.

Anand Dabak and Don Johnson. Relations between kullback-leibler distance and fisher information. 02 2003.

Sudeep Dasari, Frederik Ebert, Stephen Tian, Suraj Nair, Bernadette Bucher, Karl Schmeckpeper, Siddharth Singh, Sergey Levine, and Chelsea Finn. Robonet: Large-scale multi-robot learning. *arXiv preprint arXiv:1910.11215*, 2019.

Rousslan Fernand Julien Dossa, Shengyi Huang, Santiago Ontañón, and Takashi Matsubara. An empirical investigation of early stopping optimizations in proximal policy optimization. *IEEE access*, 9:117981–117992, 2021.

Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. In *International conference on machine learning*, pages 1329–1338. PMLR, 2016.

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Firdaus Janoos, Larry Rudolph, and Aleksander Madry. Implementation matters in deep rl: A case study on ppo and trpo. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=r1etN1rtPB.

Yaozhong Gan, Renye Yan, Zhe Wu, and Junliang Xing. Reflective policy optimization. *arXiv preprint arXiv:2406.03678*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Shengyi Huang, Rousslan Fernand Julien Dossa, Antonin Raffin, Anssi Kanervisto, and Weixun Wang. The 37 implementation details of proximal policy optimization. *The ICLR Blog Track 2023*, 2022.

Tang Jie and Pieter Abbeel. On a connection between importance sampling and the likelihood ratio policy gradient. *Advances in Neural Information Processing Systems*, 23, 2010.

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

Sham M Kakade. A natural policy gradient. *Advances in neural information processing systems*, 14, 2001.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Seungjae Lee, Yibin Wang, Haritheja Etukuru, H Jin Kim, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. Behavior generation with latent actions. *arXiv preprint arXiv:2403.03181*, 2024.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025.

Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *5th Annual Conference on Robot Learning*, 2021. URL https://openreview.net/forum?id=JrsfBJtDFdI.

Nicolas Meuleau, Leonid Peshkin, Leslie P Kaelbling, and Kee-Eung Kim. Off-policy policy search. *MIT Articical Intelligence Laboratory*, 2000.

OpenAI. Introducing openai o3 and o4-mini. https://openai.com/index/introducing-o3-and-o4-mini/, April 2025.

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.

Dean A Pomerleau. Alvinn: An autonomous land vehicle in a neural network. *Advances in neural information processing systems*, 1, 1988.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *arXiv preprint arXiv:1709.10087*, 2017.

Stéphane Ross and Drew Bagnell. Efficient reductions for imitation learning. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 661–668. JMLR Workshop and Conference Proceedings, 2010.

Daniel Sanz-Alonso. Importance sampling and necessary sample size: An information theory approach. *SIAM/ASA Journal on Uncertainty Quantification*, 6(2):867–879, 2018.

Stefan Schaal. Learning from demonstration. *Advances in neural information processing systems*, 9, 1996.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 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.

Xingyou Song, Sagi Perel, Chansoo Lee, Greg Kochanski, and Daniel Golovin. Open source vizier: Distributed infrastructure and api for reliable and flexible blackbox optimization. In *International Conference on Automated Machine Learning*, pages 8–1. PMLR, 2022.

Xingyou Song, Qiuyi Zhang, Chansoo Lee, Emily Fertig, Tzu-Kuo Huang, Lior Belenki, Greg Kochanski, Setareh Ariafar, Srinivas Vasudevan, Sagi Perel, et al. The vizier gaussian process bandit algorithm. *arXiv preprint arXiv:2408.11527*, 2024.

Mingfei Sun, Vitaly Kurin, Guoqing Liu, Sam Devlin, Tao Qin, Katja Hofmann, and Shimon Whiteson. You may not need ratio clipping in ppo. *arXiv preprint arXiv:2202.00079*, 2022.

Priya Sundaresan, Hengyuan Hu, Quan Vuong, Jeannette Bohg, and Dorsa Sadigh. What's the move? hybrid imitation learning via salient points. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=r0pLGGcuY6.

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12, 1999.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

Marcel Torne, Anthony Simeonov, Zechu Li, April Chan, Tao Chen, Abhishek Gupta, and Pulkit Agrawal. Reconciling reality through simulation: A real-to-sim-to-real approach for robust manipulation. *arXiv preprint arXiv:2403.03949*, 2024.

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.

George Tucker, Surya Bhupatiraju, Shixiang Gu, Richard Turner, Zoubin Ghahramani, and Sergey Levine. The mirage of action-dependent baselines in reinforcement learning. In *International conference on machine learning*, pages 5015–5024. PMLR, 2018.

Saran Tunyasuvunakool, Alistair Muldal, Yotam Doron, Siqi Liu, Steven Bohez, Josh Merel, Tom Erez, Timothy Lillicrap, Nicolas Heess, and Yuval Tassa. dm_control: Software and tasks for continuous control. *Software Impacts*, 6:100022, 2020.

Quan Vuong, Yiming Zhang, and Keith W Ross. Supervised policy update for deep reinforcement learning. *arXiv preprint arXiv:1805.11706*, 2018.

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

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Kaiyue Wen, Zhiyuan Li, Jason Wang, David Hall, Percy Liang, and Tengyu Ma. Understanding warmup-stable-decay learning rates: A river valley loss landscape perspective. *arXiv preprint arXiv:2410.05192*, 2024.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

Elizabeth L Wilmer, David A Levin, and Yuval Peres. Markov chains and mixing times. *American Mathematical Soc., Providence*, 107, 2009.

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.

Haotian Xu, Zheng Yan, Junyu Xuan, Guangquan Zhang, and Jie Lu. Improving proximal policy optimization with alpha divergence. *Neurocomputing*, 534:94–105, 2023.

Jingzhao Zhang, Tianxing He, Suvrit Sra, and Ali Jadbabaie. Why gradient clipping accelerates training: A theoretical justification for adaptivity. *arXiv preprint arXiv:1905.11881*, 2019.

Yuzhong Zhao, Yue Liu, Junpeng Liu, Jingye Chen, Xun Wu, Yaru Hao, Tengchao Lv, Shaohan Huang, Lei Cui, Qixiang Ye, et al. Geometric-mean policy optimization. *arXiv preprint arXiv:2507.20673*, 2025.

A PROOF OF THEOREMS AND LEMMAS

Lemma 1. *For any two policies π and π' with initial state distribution μ. Given $\alpha := \max_{s\in\mathcal{S}} D_{\mathrm{TV}}\big(\pi'(\cdot \mid s) \,\|\, \pi(\cdot \mid s)\big)$, then for all $t \in \{0, \dots, T-1\}$ we have $D_{\mathrm{TV}}(p^t_{\pi'} \,||\, p^t_{\pi}) \;\le\; \alpha\, t$.*

Proof. By coupling lemma at each matched state Wilmer et al. [2009], when $S_t = S'_t = s$ and $D_{\mathrm{TV}}\big(\pi'(\cdot \mid s) \,\|\, \pi(\cdot \mid s)\big) \le \alpha$, one can couple $A_t \sim \pi(\cdot \mid s)$ and $A'_t \sim \pi'(\cdot \mid s)$ so that $\Pr(A_t = A'_t \mid s) = 1 - \alpha$. Let E_t be the event that the two trajectories have made *no* action mismatch up to time t. We have $\Pr(E_t) = (1-\alpha)^t$.

Thus there exists a distribution m_t on $\mathcal{S}$ such that

$$p^t_{\pi'} \;=\; \Pr(E_t)\, p^t_{\pi} \;+\; \big(1 - \Pr(E_t)\big)\, m_t \;=\; (1-\alpha)^t\, p^t_{\pi} \;+\; \big(1 - (1-\alpha)^t\big)\, m_t \tag{15}$$

Then

$$\big|p^t_{\pi'} - p^t_{\pi}\big| = \Big|\, (1-\alpha)^t\, p^t_{\pi} + \big(1 - (1-\alpha)^t\big)\, m_t - p^t_{\pi}\Big| = \big(1 - (1-\alpha)^t\big)\, \big|m_t - p^t_{\pi}\big| \tag{16}$$

After summing over all s and using the trivial bound $D_{\mathrm{TV}}(p \,||\, q) \le 1$, we get: $D_{\mathrm{TV}}(p^t_{\pi'} \,||\, p^t_{\pi}) \le 1 - (1-\alpha)^t$

Finally, using the inequality, $(1-\alpha)^t \ge 1 - \alpha t$ for $\alpha \in [0, 1]$, gives $1 - (1-\alpha)^t \le \alpha t$, proving the lemma. □

Lemma 3. *Let $\varepsilon := \max_{s,a} |A^\pi(s,a)|$. Given $\alpha := \max_{s\in\mathcal{S}} D_{\mathrm{TV}}\big(\pi'(\cdot \mid s) \,\|\, \pi(\cdot \mid s)\big)$, we have*

$$|\mathbb{E}_{a\sim\pi'}[A^\pi(s,a)]| \le 2\alpha\,\varepsilon \tag{17}$$

Proof. By construction we know

$$\mathbb{E}_{a\sim\pi}[A^\pi(s,a)] = \mathbb{E}_{a\sim\pi}[Q^\pi(s,a)] - V^\pi(s) = 0 \tag{18}$$

Thus,

$$\begin{aligned} |\mathbb{E}_{a\sim\pi'}[A^\pi(s,a)]| &= |\sum_a (\pi'(a|s) - \pi(a|s)) A^\pi(a,s)| \\ &\le \sum_a |\pi'(a \mid s) - \pi(a \mid s)|\, \varepsilon = 2\, D_{\mathrm{TV}}(\pi'(\cdot \mid s) \,||\, \pi(\cdot \mid s))\, \varepsilon \;\le\; 2\alpha\,\varepsilon \end{aligned} \tag{19}$$

□

Theorem 1. *[Schulman et al., 2015a] Given $\alpha := \max_s D_{\mathrm{TV}}\big(\pi(\cdot \mid s) \,\|\, \pi'(\cdot \mid s)\big)$, then the following bound holds:*

$$J(\pi') \;\ge\; L_\pi(\pi') \;-\; 2\,\varepsilon\, T(T-1)\,\alpha^2 \tag{20}$$

where $\varepsilon := \max_{s,a} \big|A^\pi(s,a)\big|$.

Proof.

$$\begin{aligned} |J(\pi') - L_\pi(\pi')| &= |\sum_{t=0}^{T-1} \Big(\sum_s p^t_{\pi'}(s) - p^t_{\pi}(s)\Big)\, \mathbb{E}_{a\sim\pi'}[A^\pi(s,a)]| \\ &\le \sum_{t=0}^{T-1} \Big(\sum_s |p^t_{\pi'}(s) - p^t_{\pi}(s)|\Big)\, |\mathbb{E}_{a\sim\pi'}[A^\pi(s,a)]| \\ &\le \sum_{t=0}^{T-1} \big(2 D_{\mathrm{TV}}(p^t_{\pi'} \,||\, p^t_{\pi})\big)\, (2\alpha\varepsilon) = 4\alpha\varepsilon \sum_{t=0}^{T-1} \alpha t = 4\alpha^2\varepsilon\, \frac{T(T-1)}{2} \end{aligned} \tag{21}$$

□

Lemma 2. *For any two policies π and π' with initial state distribution μ. Given $\alpha := D_{\mathrm{TV}}\big(P^{\pi} \,\|\, P^{\pi'}\big)$, then for all $t \in \{0, \ldots, T-1\}$ we have: $D_{\mathrm{TV}}(p^t_{\pi'} \,||\, p^t_{\pi}) \leq \alpha$.*

Proof. For state marginals at timestep t for policies π and π' we have: $p^t_{\pi}(s) = \sum_{\tau : s_t = s} P^{\pi}(\tau)$ and $p^t_{\pi'}(s) = \sum_{\tau : s_t = s} P^{\pi'}(\tau)$. Thus

$$D_{\mathrm{TV}}(p^t_{\pi} || p^t_{\pi'}) = \frac{1}{2} \sum_s |p^t_{\pi}(s) - p^t_{\pi'}(s)| = \frac{1}{2} \sum_s |\sum_{\tau : s_t = s} P^{\pi}(\tau) - P^{\pi'}(\tau)|$$

$$\leq \frac{1}{2} \sum_s \sum_{\tau : s_t = s} |P^{\pi}(\tau) - P^{\pi'}(\tau)| = \frac{1}{2} \sum_{\tau} |P^{\pi}(\tau) - P^{\pi'}(\tau)| \; = \alpha \tag{22}$$

□

Theorem 2. *Given $\alpha := D_{\mathrm{TV}}\big(P^{\pi} \,\|\, P^{\pi'}\big)$, then the following bound holds:*

$$J(\pi') \;\geq\; L_{\pi}(\pi') \;-\; 2\,\varepsilon\,T\,\alpha^2. \tag{23}$$

where $\varepsilon := \max_{s,a} \big|A^{\pi}(s,a)\big|$.

Proof. Similarly as in Theorem 1, here we use Lemma 2 and follow the same procedure to obtain the claimed bound. □

Theorem 3. *Let the probability ratio $r_t := \frac{\pi'(a_t|s_t)}{\pi(a_t|s_t)} \in [1-\epsilon,\; 1+\epsilon]$, with $\epsilon \in (0,1)$. Suppose Assumptions 1–4 hold. Then, for any number of trajectories collected $N > 0$, and constant $c_0 \in (0,1]$, we can reformulate the criterion in Equation 13 as:*

$$\mathrm{D_{KL}}(P^{\pi} \,\|\, P^{\pi'}) \leq \frac{1-\epsilon}{1+\epsilon} \ln\!\left(\frac{N}{c_0}\right) \left(1 + \mathcal{O}\left(\frac{1}{\sqrt{T}}\right)\right). \tag{24}$$

Proof. Proof relies on the following key assumptions:

Assumption 1 (Bounded Importance Ratio)**.**

$$r_t := \frac{\pi'(a_t \mid s_t)}{\pi(a_t \mid s_t)} \in [1-\epsilon,\; 1+\epsilon], \qquad \epsilon \in (0,1). \tag{25}$$

Assumption 2 (Geometric α-mixing)**.** *For each policy π_c with $c \in [0,1]$, let $\mathcal{F}_i^j$ be the σ-algebra generated by the state sequence $(s_i, \ldots, s_j)$. There exist constants $C_\alpha > 0$ and $\rho \in (0,1)$, such that for all integers $k \geq 1$ and all $t \geq 0$:*

$$\alpha_{\pi_c}(k) := \sup_{A \in \mathcal{F}_0^t,\, B \in \mathcal{F}_{t+k}^{\infty}} \big|\Pr(A \cap B) - \Pr(A)\Pr(B)\big| \leq C_\alpha\, \rho^k. \tag{26}$$

Equivalently, writing $\tau := \frac{1}{|\ln \rho|}$ for the effective mixing time, we have:

$$\alpha_{\pi_c}(k) \leq C_\alpha e^{-k/\tau}. \tag{27}$$

Intuitively, this assumption guarantees rapid mixing of the environment dynamics. The α-mixing coefficient bounds the maximum deviation between the joint distribution of events separated by k steps and the product of their marginals, ensuring that long-horizon state dependencies vanish exponentially fast.

Assumption 3 (Uniform policy proximity)**.** *There exist constants $C_{\max} \geq 1$ and $L > 0$ such that:*

(a) *The max-to-average divergence ratio is bounded:*

$$D_{\mathrm{KL}}^{\max}(\pi, \pi') \;:=\; \sup_s D_{\mathrm{KL}}(\pi \| \pi')(s) \;\leq\; C_{\max} \cdot \bar{d}\,, \qquad \bar{d} := \frac{\delta}{T}\,, \tag{28}$$

where $\delta := D_{\mathrm{KL}}(P^{\pi} \| P^{\pi'})$ is the trajectory-level reverse KL.

(b) *The environments dynamics increase the per-step policy perturbations by at most a constant:*

$$\sup_{t\geq 0} \mathrm{D_{TV}}\big(p^t_{\pi_c},\, p^t_{\pi}\big) \;\leq\; L \cdot \sup_{s} \mathrm{D_{TV}}\big(\pi_c(\cdot|s),\, \pi(\cdot|s)\big) \qquad \forall\, c \in [0,1]. \tag{29}$$

Part (a) formalizes how controlling trajectory-level KL controls worst-case per-state divergence. Part (b) follows from standard Markov-chain perturbation bounds and holds with L depending only on the environment's mixing constants.

Assumption 4 (Long-horizon regime)**.** *$T \to \infty$ with $\delta = \mathcal{O}(1)$ and $T/\tau \to \infty$.*

We start the proof by first defining $\mathcal{P}$ to be the manifold of probability distributions, with P^{π} and $P^{\pi'}$ representing the trajectory distributions of the old and new policies respectively. We define the trajectory log-likelihood ratio S_T as:

$$S_T := \ln \frac{P^{\pi'}(\tau)}{P^{\pi}(\tau)} = \sum_{t=0}^{T-1} \ln \frac{\pi'(a_t|s_t)}{\pi(a_t|s_t)} = \sum_{t=0}^{T-1} \xi_t, \tag{30}$$

where the per-step log-ratio is bounded: $\xi_t := \ln r_t \in \big[\ln(1-\epsilon),\ \ln(1+\epsilon)\big]$.

Because ξ_t is strictly bounded, all moments of S_T are finite. Consequently, the exponential twist density (the geodesic connecting P^{π} and $P^{\pi'}$) is well-defined for $c \in [0,1]$:

$$P_c(\tau) := \frac{\big[P^{\pi}(\tau)\big]^{1-c}\big[P^{\pi'}(\tau)\big]^{c}}{\mathcal{Z}_c}, \tag{31}$$

$$\pi_c(a \mid s) := \frac{\pi(a \mid s)^{1-c}\,\pi'(a \mid s)^{c}}{Z_c(s)}, \tag{32}$$

where $\mathcal{Z}_c$ and $Z_c(s)$ are the trajectory and per-state normalization constants, respectively.

The statistical distance between the tranjectories along this geodesic is governed by the Fisher Information Metric:

$$F(c) := \sum_{\tau} \left(\frac{\partial \ln P_c(\tau)}{\partial c}\right)^2 P_c(\tau) = \mathrm{Var}_{P_c}[S_T]. \tag{33}$$

This induces the following exact integral representations for the Kullback-Leibler divergences:

$$\mathrm{D_{KL}}(P^{\pi'} \| P^{\pi}) = \int_0^1 c\, F(c)\, dc\,, \tag{34}$$

$$\mathrm{D_{KL}}(P^{\pi} \| P^{\pi'}) = \int_0^1 (1-c)\, F(c)\, dc\,. \tag{35}$$

To analyze $F(c)$, we define the Cumulant Generating Functions (CGFs) for the per-state log-ratio and the trajectory log-ratio. For a fixed state s, the per-state log-ratio CGF is:

$$f(c,s) := \ln Z_c(s) = \ln\, \mathbb{E}_{a\sim\pi}[e^{c\,\xi(a,s)}]. \tag{36}$$

By the standard properties of exponential families, the derivatives of this CGF correspond exactly to the cumulants of ξ under the tilted density π_c:

$$f'(c,s) = \mathbb{E}_{\pi_c}[\xi], \quad f''(c,s) = \mathrm{Var}_{\pi_c}[\xi], \quad f'''(c,s) = \mathbb{E}_{\pi_c}\big[(\xi - \mathbb{E}_{\pi_c}[\xi])^3\big]. \tag{37}$$

For a fixed state sequence $\mathbf{s} = (s_0, \ldots, s_{T-1})$, the conditional trajectory CGF decomposes additively. Because the actions are conditionally independent given the state sequence, the CGF of the sum S_T is the sum of the per-state CGFs:

$$\begin{aligned}
\Phi_{\mathbf{s}}(c) &:= \ln\, \mathbb{E}_{P^{\pi}}\big[e^{c\,S_T} \mid \mathbf{s}\big] \\
&= \ln\, \mathbb{E}_{P^{\pi}}\left[\prod_{t=0}^{T-1} e^{c\,\xi_t} \,\middle|\, \mathbf{s}\right] \\
&= \ln \left(\prod_{t=0}^{T-1} \mathbb{E}_{\pi(\cdot|s_t)}\big[e^{c\,\xi_t} \mid s_t\big]\right) \\
&= \sum_{t=0}^{T-1} \ln\, \mathbb{E}_{\pi}\big[e^{c\,\xi_t} \mid s_t\big] = \sum_{t=0}^{T-1} f(c, s_t).
\end{aligned} \tag{38}$$

Lemma 4 (Per-state Grönwall bound)**.** *For any state sequence* $\mathbf{s} = (s_0, \ldots, s_{T-1})$ *such that* $\Phi''_{\mathbf{s}}(0) > 0$*, and state* s *with* $\pi(\cdot|s) \neq \pi'(\cdot|s)$*. Under Assumption 1 with range* $d_\epsilon := \ln \frac{1+\epsilon}{1-\epsilon}$ *and all* $c \in [0, 1]$*:*

$$e^{-d_\epsilon c}\, \Phi''_{\mathbf{s}}(0) \leq \Phi''_{\mathbf{s}}(c) \leq e^{d_\epsilon c}\, \Phi''_{\mathbf{s}}(0). \tag{39}$$

Proof. We have:

$$\begin{aligned} \left|\mathbb{E}\big[(\xi - \mathbb{E}[\xi])^3\big]\right| &\leq \mathbb{E}\big[|\xi - \mathbb{E}[\xi]|^3\big] && (40)\\ &\leq d_\epsilon\, \mathbb{E}\big[|\xi - \mathbb{E}[\xi]|^2\big] && (41)\\ &= d_\epsilon f''(c, s) && (42) \end{aligned}$$

Step 41 uses the fact that $|\xi - \mathbb{E}[\xi]| \leq \ln \frac{1+\epsilon}{1-\epsilon}$, and since $f'''(c, s) = \mathbb{E}\big[(\xi - \mathbb{E}[\xi])^3\big]$, we get:

$$|f'''(c, s)| \leq d_\epsilon\, f''(c, s) \qquad \forall\, c \in [0, 1],\, \forall\, s. \tag{43}$$

Utilizing (43):

$$\begin{aligned} -d_\epsilon\, f''(c, s) \leq f'''(c, s) &\leq d_\epsilon\, f''(c, s) && (44)\\ e^{-d_\epsilon c}\, f''(0, s) \leq f''(c, s) &\leq e^{d_\epsilon c}\, f''(0, s) \quad \text{(by Gronwall inequality)} && (45)\\ e^{-d_\epsilon c}\, \Phi''_{\mathbf{s}}(0) \leq \Phi''_{\mathbf{s}}(c) &\leq e^{d_\epsilon c}\, \Phi''_{\mathbf{s}}(0). \quad \text{(after summing over t)} && (46) \end{aligned}$$

□

We have the per-step average KL is $\bar{d} = \delta/T \to 0$. By Assumption 3(a), each per-state KL satisfies $D_{\mathrm{KL}}(\pi\|\pi')(s) = \mathcal{O}(\bar{d})$. By definition of the CGF:

$$\begin{aligned} |f'(0, s)| = D_{\mathrm{KL}}(\pi\|\pi')(s) &= \mathcal{O}(\bar{d}), && (47)\\ f''(0, s) = \mathrm{Var}_\pi[\xi|s] &= \mathcal{O}(\bar{d}), && (48) \end{aligned}$$

where (48) uses the fact that $\mathrm{Var}_\pi[\xi|s] = 2\, D_{\mathrm{KL}}(\pi\|\pi')(s)$ when $\bar{d} \to 0$. Then by the Fundamental Theorem of Calculus, and triangle inequality, we have:

$$\begin{aligned} |f'(c, s)| &\leq |f'(0, s)| + \left| \int_0^c f''(x, s)\, dx \right| && (49)\\ &\leq |f'(0, s)| + \int_0^c e^{d_\epsilon}\, f''(0, s)\, dx && (50)\\ &\leq |f'(0, s)| + \ e^{d_\epsilon}\, f''(0, s) = \mathcal{O}(\bar{d}), && (51) \end{aligned}$$

where Step 50 uses Lemma 4 and $f''(x, s) \geq 0$, and Step 51 uses the definition of $c \in [0, 1]$.

By the law of total variance, conditioning on the state sequence $\mathbf{s}$, the Fisher Information metric decomposes into action variance and state coupling:

$$F(c) = \mathrm{Var}_{P_c}[S_T] = \underbrace{\mathbb{E}_{P_c}\big[\mathrm{Var}_{P_c}[S_T \mid \mathbf{s}]\big]}_{=:\, F_{\mathrm{act}}(c)} + \underbrace{\mathrm{Var}_{P_c}\big[\mathbb{E}_{P_c}[S_T \mid \mathbf{s}]\big]}_{=:\, F_{\mathrm{state}}(c)}. \tag{52}$$

Writing $g_t := f'(c, s_t)$ with $|g_t| = \mathcal{O}(\bar{d})$ we decompose the variance $F_{\mathrm{state}}(c)$:

$$F_{\mathrm{state}}(c) = \mathrm{Var}_{P_c}\Big[\sum_t f'(c, s_t)\Big] = \sum_t \mathrm{Var}[g_t] + 2 \sum_{t<t'} \mathrm{Cov}[g_t, g_{t'}]. \tag{53}$$

Each diagonal term satisfies $\mathrm{Var}[g_t] \leq \mathbb{E}[g_t^2] = \mathcal{O}(\bar{d}^2)$. By Assumption 2, the off-diagonal covariances decay as $|\,\mathrm{Cov}[g_t, g_{t+k}]| = \mathcal{O}(\bar{d}^2) \cdot \rho^k$, then the double sum evaluates to:

$$F_{\mathrm{state}}(c) = T \cdot \mathcal{O}(\bar{d}^2) \cdot \mathcal{O}(\tau) = \mathcal{O}\left(\frac{\delta^2 \tau}{T}\right). \tag{54}$$

Meanwhile $F(0) \geq F_{\text{act}}(0) = \sum_t \mathbb{E}_{p^t_\pi}[f''(0, s_t)] = \mathcal{O}(\delta)$. Therefore:

$$\frac{F_{\text{state}}(c)}{F(0)} = \mathcal{O}\left(\frac{\delta\tau}{T}\right). \tag{55}$$

By Lemma 4, $\Phi''_{\mathbf{s}}(c) \leq e^{d_\epsilon c}\, \Phi''_{\mathbf{s}}(0)$ for every state sequence. Taking expectations under P_c:

$$F_{\text{act}}(c) = \mathbb{E}_{P_c}[\Phi''_{\mathbf{s}}(c)] \leq e^{d_\epsilon c}\, \mathbb{E}_{P_c}[\Phi''_{\mathbf{s}}(0)]. \tag{56}$$

The quantity $\mathbb{E}_{P_c}[\Phi''_{\mathbf{s}}(0)] = \sum_t \mathbb{E}_{p^t_{\pi_c}}[f''(0, s_t)]$ differs from $\mathbb{E}_{P_0}[\Phi''_{\mathbf{s}}(0)] = \sum_t \mathbb{E}_{p^t_\pi}[f''(0, s_t)]$ only because the state marginals $p^t_{\pi_c}$ differ from p^t_π. Bounding this shift at each timestep using the L_1 definition of Total Variation distance:

$$\begin{aligned}
\left|\mathbb{E}_{p^t_{\pi_c}}[f''(0,s)] - \mathbb{E}_{p^t_\pi}[f''(0,s)]\right| &= \left|\sum_s f''(0,s)\left(p^t_{\pi_c}(s) - p^t_\pi(s)\right)\right| \\
&\leq \|f''(0,\cdot)\|_\infty \sum_s \left|p^t_{\pi_c}(s) - p^t_\pi(s)\right| \\
&= 2\,\|f''(0,\cdot)\|_\infty \cdot \mathrm{D}_{\mathrm{TV}}(p^t_{\pi_c},\, p^t_\pi) \\
&= \mathcal{O}\left(\frac{\delta\, L}{T}\sqrt{\frac{\delta}{T}}\right).
\end{aligned} \tag{57}$$

The final equality follows because the maximum per-state variance is bounded as $\|f''(0,\cdot)\|_\infty = \mathcal{O}(\bar{d})$ by Assumption 3, and the state marginal shift is bounded by $\mathrm{D}_{\mathrm{TV}}(p^t_{\pi_c},\, p^t_\pi) \leq L \cdot \mathcal{O}(\sqrt{\bar{d}})$ by combining Assumption 3 with Pinsker's inequality on the per-state policy divergence.

Summing over T timesteps and dividing by $\mathbb{E}_{P_0}[\Phi''_{\mathbf{s}}(0)] = F_{\text{act}}(0) = \mathcal{O}(\delta)$ we get:

$$\mathbb{E}_{P_c}[\Phi''_{\mathbf{s}}(0)] = \left(1 + \mathcal{O}(L/\sqrt{T})\right) F_{\text{act}}(0). \tag{58}$$

Substituting (58) into (56):

$$F_{\text{act}}(c) \leq \left(1 + \mathcal{O}(L/\sqrt{T})\right) e^{d_\epsilon c}\, F_{\text{act}}(0). \tag{59}$$

The analogous lower bound $F_{\text{act}}(c) \geq (1 + \mathcal{O}(L/\sqrt{T}))\, e^{-d_\epsilon c}\, F_{\text{act}}(0)$ follows identically from the lower half of Lemma 4.

Then, the previous variance decomposition now gives:

$$\begin{aligned}
F(c) &= F_{\text{act}}(c) + F_{\text{state}}(c) \\
&\leq \left(1 + \mathcal{O}(L/\sqrt{T})\right) e^{d_\epsilon c}\, F_{\text{act}}(0) + \mathcal{O}(\delta^2\tau/T) \\
&= \left(1 + \mathcal{O}(1/\sqrt{T})\right) e^{d_\epsilon c}\, F(0),
\end{aligned} \tag{60}$$

where the last step uses $F_{\text{act}}(0) = F(0)\left(1 + \mathcal{O}(\delta\tau/T)\right)$ from (55) and factors out $F(0)$, absorbing the relative errors since $\mathcal{O}(L/\sqrt{T}) + \mathcal{O}(\delta\tau/T) = \mathcal{O}(1/\sqrt{T})$ as $T \to \infty$.

The analogous lower bound gives:

$$F(c) \geq \left(1 + \mathcal{O}(1/\sqrt{T})\right) e^{-d_\epsilon c}\, F(0). \tag{61}$$

Applying (60) and (61) to the integral representations (34) and (35):

$$\mathrm{D}_{\mathrm{KL}}(P^{\pi'} \| P^\pi) = \int_0^1 c\, F(c)\, dc \leq \left(1 + \mathcal{O}(1/\sqrt{T})\right) F(0) \int_0^1 c\, e^{d_\epsilon c}\, dc\,, \tag{62}$$

$$\mathrm{D}_{\mathrm{KL}}(P^{\pi} \| P^{\pi'}) = \int_0^1 (1-c)\, F(c)\, dc \geq \left(1 + \mathcal{O}(1/\sqrt{T})\right) F(0) \int_0^1 (1-c)\, e^{-d_\epsilon c}\, dc\,. \tag{63}$$

Both integrals evaluate to the same quantity up to a factor of e^{d_ϵ}. The numerator integral:

$$\int_0^1 c\, e^{d_\epsilon c}\, dc = \frac{d_\epsilon e^{d_\epsilon} - e^{d_\epsilon} + 1}{d_\epsilon^2}\,.$$

The denominator integral, via the substitution $u = 1 - c$:

$$\int_0^1 (1-c)\, e^{-d_\epsilon c}\, dc = e^{-d_\epsilon} \int_0^1 u\, e^{d_\epsilon u}\, du = e^{-d_\epsilon} \cdot \frac{d_\epsilon e^{d_\epsilon} - e^{d_\epsilon} + 1}{d_\epsilon^2}\,.$$

Dividing (62) by (63), the $F(0)$ and the common polynomial cancel:

$$\frac{\mathrm{D_{KL}}(P^{\pi'} \| P^{\pi})}{\mathrm{D_{KL}}(P^{\pi} \| P^{\pi'})} \leq \left(1 + \mathcal{O}(1/\sqrt{T})\right) e^{d_\epsilon}\,. \tag{64}$$

Since $e^{-d_\epsilon} = e^{-\ln \frac{1+\epsilon}{1-\epsilon}} = \frac{1-\epsilon}{1+\epsilon}$, concludes the proof. □

B ALGORITHM

Algorithm 1 PPO with Trajectory Divergence Criterion

Input: initial policy θ_0 and value parameters ϕ_0, clipping parameter ϵ, divergence threshold δ, batch size N, learning rates α_π and α_v
for $i \leftarrow 0, 1, 2, \ldots$ **do**
  Collect N trajectories $\{\tau_j\}_{j=1}^N$ using π_{θ_i}
  Compute advantage estimates $\{\hat{A}_t^{(j)}\}$ and returns $\{R_t^{(j)}\}$
  Initialize $\theta \leftarrow \theta_i$, $\phi \leftarrow \phi_i$
  while true **do**
    Policy loss:

$$L^{\mathrm{PPO}}(\theta) \leftarrow -\frac{1}{NT} \sum_{j,t} \min\left(\frac{\pi_\theta(a_t^{(j)}|s_t^{(j)})}{\pi_{\theta_i}(a_t^{(j)}|s_t^{(j)})} \hat{A}_t^{(j)}, \mathrm{clip}\left(\frac{\pi_\theta(a_t^{(j)}|s_t^{(j)})}{\pi_{\theta_i}(a_t^{(j)}|s_t^{(j)})}, 1-\epsilon, 1+\epsilon \right) \hat{A}_t^{(j)} \right)$$

    Compute KL divergence estimate over trajectories:

$$\widehat{D}_{\mathrm{KL}} \leftarrow \frac{1}{N} \sum_{j=1}^{N} \sum_{t=0}^{T-1} \left[\log \pi_{\theta_i}(a_t^{(j)}|s_t^{(j)}) - \log \pi_\theta(a_t^{(j)}|s_t^{(j)}) \right]$$

    if $\widehat{D}_{\mathrm{KL}} > \delta$ **then break**
    end if
    Value loss: $L^{\mathrm{V}}(\phi) \leftarrow \frac{1}{NT} \sum_{j,t} \left(V_\phi(s_t^{(j)}) - R_t^{(j)} \right)^2$
    Update parameters: $\theta \leftarrow \theta - \alpha_\pi \nabla_\theta L^{\mathrm{PPO}}(\theta),\ \phi \leftarrow \phi - \alpha_v \nabla_\phi L^{\mathrm{V}}(\phi)$
  end while
  $\theta_{i+1} \leftarrow \theta$, $\phi_{i+1} \leftarrow \phi$
end for

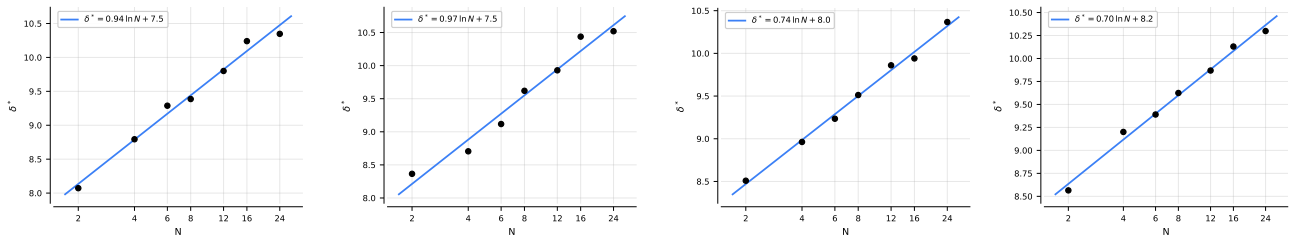
C HYPERPARAMETER SETTINGS

To ensure a fair and rigorous evaluation, we systematically tuned the key hyperparameters for all baseline algorithms. Table 4 defines the search space and scale types used for Bayesian optimization. After 40 optimization trials per algorithm, we extracted the optimal configurations, which are summarized in Table 5. For completeness, we also include the final parameters used for our method in PPO, as motivated by the theoretical framework in the main text. For instance, the specific ϵ and δ choice is motivated by the collection of $N = 16$ batches per iteration.

D FITTED SCALING LAW

Table 4: **Hyperparameter Search Space per Algorithm.**

Algorithm	Hyperparameter	Search Range	Scale Type
All except TRPO	Learning Rate (α_π, α_v)	$[10^{-5}, 10^{-2}]$	Log
	Epochs (K)	$\{4, \ldots, 20\}$	Linear
	Max Gradient Norm	$[0.1, 2.0]$	Linear
RPO	Beta (β)	$[0.0, 1.0]$	Linear
	Epsilon (ϵ)	$[0.01, 0.6]$	Linear
	Epsilon1 (ϵ_1)	$[0.01, 0.6]$	Linear
SPO	Epsilon (ϵ)	$[0.01, 0.6]$	Linear
ESPO	Delta (δ)	$[0.01, 0.5]$	Linear
TRPO	CG Iterations	$\{1, \ldots, 20\}$	Linear
	CG Damping	$[10^{-5}, 10^{-1}]$	Log
	Max KL (D_{KL})	$[0.001, 0.1]$	Log
	Backtrack Coefficient	$[0.1, 0.9]$	Linear
	Backtrack Iterations	$\{1, \ldots, 20\}$	Linear
TR-PPO-RB	Delta (δ)	$[0.001, 0.1]$	Log
	Alpha (α)	$[0.0, 1.0]$	Linear
AlphaPPO	Alpha (α)	$[-2.0, 2.5]$	Linear
	Beta (β)	$[0.0, 1.0]$	Linear
SPU	Delta (δ)	$[0.001, 0.1]$	Linear
	Epsilon (ϵ)	$[0.001, 0.1]$	Linear
	Lambda (λ)	$[0.1, 5.0]$	Linear



(a) $\epsilon = 0.05$ (b) $\epsilon = 0.1$ (c) $\epsilon = 0.15$ (d) $\epsilon = 0.2$

Figure 5: Corresponding fitted scaling law for each ϵ with the optimal δ for each N extracted after sweeping. The x-axis is in logarithmic scale.

Table 5: **Optimal Hyperparameters.** Best hyperparameter configurations discovered via Bayesian optimization across 40 trials for each baseline, alongside the parameters utilized for PPO.

Algorithm	Learning Rate	Epochs (K)	Max Grad Norm	Algorithm-Specific Parameters
PPO	3×10^{-4}	Adaptive	–	$\epsilon = 0.15$, $\delta = 10.0$
AlphaPPO	1×10^{-4}	13	0.10	$\alpha = 2.5$, $\beta = 0.45$
ESPO	2×10^{-4}	8	1.90	$\delta = 0.25$
RPO	5×10^{-4}	15	1.85	$\beta = 0.15$, $\epsilon = 0.2$, $\epsilon_1 = 0.1$
SPO	8×10^{-4}	11	1.50	$\epsilon = 0.45$
SPU	3×10^{-4}	15	0.90	$\delta = 0.1$, $\epsilon = 0.04$, $\lambda = 1.9$
TRPO	1×10^{-2}	15	–	CG Iters = 8, CG Damp = 0.1, Max KL = 0.03 BT Coeff = 0.4, BT Iters = 14
TR-PPO-RB	1×10^{-4}	16	1.90	$\delta = 0.1$, $\alpha = 0.2$