Truly Proximal Policy Optimization

Yuhui Wang*, Hao He*, Xiaoyang Tan

College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China
MIIT Key Laboratory of Pattern Analysis and Machine Intelligence, China
Collaborative Innovation Center of Novel Software Technology and Industrialization, China
{*y.wang, hugo, x.tan*}*@nuaa.edu.cn*

Abstract

Proximal policy optimization (PPO) is one of the most successful deep reinforcement learning methods, achieving state-of-the-art performance across a wide range of challenging tasks. However, its optimization behavior is still far from being fully understood. In this paper, we show that PPO could neither strictly restrict the probability ratio as it attempts to do nor enforce a well-defined trust region constraint, which means that it may still suffer from the risk of performance instability. To address this issue, we present an enhanced PPO method, named Trust Region-based PPO with Rollback (TR-PPO-RB). Two critical improvements are made in our method: 1) it adopts a new clipping function to support a rollback behavior to restrict the ratio between the new policy and the old one; 2) the triggering condition for clipping is replaced with a trust region-based one, which is theoretically justified according to the trust region theorem. It seems, by adhering more truly to the "proximal" property – restricting the policy within the trust region, the new algorithm improves the original PPO on both stability and sample efficiency.

1 INTRODUCTION

Deep model-free reinforcement learning has achieved great successes in recent years, notably in video games (Mnih et al., 2015), board games (Silver et al., 2017), robotics (Levine et al., 2016), and challenging control tasks (Schulman et al., 2016; Duan et al., 2016). Policy gradient (PG) methods are useful model-free policy search algorithms, updating the policy with an estimator of the gradient of the expected return (Peters & Schaal, 2008). One major challenge of PG-based methods is to estimate the right step size for the policy updating, and an improper step size may result in severe policy degradation due to the fact that the input data strongly depends on the current policy (Kakade & Langford, 2002; Schulman et al., 2015). For this reason, the trade-off between learning stability and learning speed is an essential issue to be considered for a PG method.

The well-known trust region policy optimization (TRPO) method addressed this problem by imposing onto the objective function a trust region constraint so as to control the KL divergence between the old policy and the new one (Schulman et al., 2015). This can be theoretically justified by showing that optimizing the policy within the trust region leads to guaranteed monotonic performance improvement. However, the complicated second-order optimization involved in TRPO makes it computationally inefficient and difficult to scale up for large scale problems when extending to complex network architectures. Proximal Policy Optimization (PPO) significantly reduces the complexity by adopting a clipping mechanism so as to avoid imposing the hard constraint completely, allowing it to use a first-order optimizer like the Gradient Descent method to optimize the objective (Schulman et al., 2017). As for the mechanism for dealing with the learning stability issue, in contrast with the trust region method of TRPO, PPO tries to remove the incentive for pushing the policy away from the old one when the probability ratio between them is out of a clipping range. PPO is proven to be very effective in dealing with a wide range of challenging tasks, while being simple to implement and tune.

However, despite its success, the actual optimization behavior of PPO is less studied, highlighting the need to study the proximal property of PPO. Some researchers have raised concerns about whether PPO could restrict the probability ratio as it attempts to do (Wang et al.,

*Authors contributed equally.

2019; Ilyas et al., 2018), and since there exists an obvious gap between the heuristic probability ratio constraint and the theoretically-justified trust region constraint, it is natural to ask whether PPO enforces a trust region-like constraint as well to ensure its stability in learning?

In this paper, we formally address both the above questions and give negative answers to both of them. In particular, we found that PPO could neither strictly restrict the probability ratio nor enforce a trust region constraint. The former issue is mainly caused by the fact that PPO could not entirely remove the incentive for pushing the policy away, while the latter is mainly due to the inherent difference between the two types of constraints adopted by PPO and TRPO respectively.

Inspired by the insights above, we propose an enhanced PPO method, named Trust Region-based PPO with Rollback (TR-PPO-RB). In particular, we apply a negative incentive to prevent the policy from being pushed away during training, which we called a *rollback* operation. Furthermore, we replace the triggering condition for clipping with a trust region-based one, which is theoretically justified according to the trust region theorem that optimizing the policy within the trust region lead to guaranteed monotonic improvement (Schulman et al., 2015). TR-PPO-RB actually combines the strengths of TRPO and PPO – it is theoretically justified and is simple to implement with first-order optimization. Extensive results on several benchmark tasks show that the proposed methods significantly improve both the policy performance and the sample efficiency. Source code is available at `https://github.com/wangyuhuix/TrulyPPO`.

2 RELATED WORK

Many researchers have extensively studied different approach to constrain policy updating in recent years. The natural policy gradient (NPG) (Kakade, 2001) improves REINFORCE by computing an ascent direction that approximately ensures a small change in the policy distribution. Relative entropy policy search (REPS) (Peters et al., 2010) constrains the state-action marginals, limits the loss of information per iteration and aims to ensure a smooth learning progress. While this algorithm requires a costly nonlinear optimization in the inner loop, which is computationally expansive. TRPO is derived from the conservative policy iteration (Kakade & Langford, 2002), in which the performance improvement lower bound has been first introduced.

There has been a focus on the problem of constraining policy update, and attention is being paid to TRPO and PPO in recent years. Wu et al. (2017) proposed an actor critic method which uses Kronecker-factor trust regions (ACKTR). Hmlinen et al. (2018) proposed a method to improve exploration behavior with evolution strategies. Chen et al. (2018) presented a method adaptively adjusts the scale of policy gradient according to the significance of state-action.

Several studies focus on investigating the clipping mechanism of PPO. Wang et al. (2019) found that the ratio-based clipping of PPO could lead to limited sample efficiency when the policy is initialized from a bad solution. To address this problem, the clipping ranges are adaptively adjusted guided by a trust region criterion. This paper also works on a trust region criterion, but it is used as a triggering condition for clipping, which is much simpler to implement. Ilyas et al. (2018) performed a fine-grained examination and found that the PPO's performance depends heavily on optimization tricks but not the core clipping mechanism. However, as we found, although the clipping mechanism could not strictly restrict the policy, it does exert an important effect in restricting the policy and maintain stability. We provide detail discussion in our experiments.

3 PRELIMINARIES

A Markov Decision Processes (MDP) is described by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, c, \rho_1, \gamma)$. $\mathcal{S}$ and $\mathcal{A}$ are the state space and action space; $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the transition probability distribution; $c : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function; ρ_1 is the distribution of the initial state s_1, and $\gamma \in (0, 1)$ is the discount factor. The performance of a policy π is defined as $\eta(\pi) = \mathbb{E}_{s\sim\rho^\pi, a\sim\pi}[c(s,a)]$ where $\rho^\pi(s) = (1-\gamma)\sum_{t=1}^{\infty}\gamma^{t-1}\rho_t^\pi(s)$, ρ_t^π is the density function of state at time t.

Policy gradients methods (Sutton et al., 1999) update the policy by the following surrogate performance objective,

$$L_{\pi_{\text{old}}}(\pi) = \mathbb{E}_{s,a}\left[r_\pi(s,a)A^{\pi_{\text{old}}}(s,a)\right] + \eta(\pi_{\text{old}}) \quad (1)$$

where $\pi(a|s)/\pi_{\text{old}}(a|s)$ is the *probability ratio* between the new policy π and the old policy π_{old}, $A^{\pi_{\text{old}}}(s,a) = \mathbb{E}[R_t^\gamma|s_t = s, a_t = a; \pi_{\text{old}}] - \mathbb{E}[R_t^\gamma|s_t = s; \pi_{\text{old}}]$ is the advantage value function of the old policy π_{old}. Schulman et al. (2015) derived the following performance bound:

Theorem 1. *Let* $C = \max_{s,a}|A^{\pi_{\text{old}}}(s,a)|4\gamma/(1-\gamma)^2$, $D_{\text{KL}}^s(\pi_{\text{old}},\pi) \triangleq D_{\text{KL}}(\pi_{\text{old}}(\cdot|s)||\pi(\cdot|s))$, $M_{\pi_{\text{old}}}(\pi) = L_{\pi_{\text{old}}}(\pi) - C\max_{s\in\mathcal{S}} D_{\text{KL}}^{\text{s}}(\pi_{\text{old}},\pi)$ *. We have*

$$\eta(\pi) \geq M_{\pi_{\text{old}}}(\pi), \eta(\pi_{\text{old}}) = M_{\pi_{\text{old}}}(\pi_{\text{old}}). \quad (2)$$

This theorem implies that maximizing $M_{\pi_{\text{old}}}(\pi)$ guarantee non-decreasing of the performance of the new policy π. TRPO imposed a constraint on the KL divergence:

$$\max_{\pi} L_{\pi_{\text{old}}}(\pi) \tag{3a}$$

$$\text{s.t.} \max_{s \in \mathcal{S}} D_{\text{KL}}^{\text{s}}\left(\pi_{\text{old}}, \pi\right) \leq \delta \tag{3b}$$

Constraint (3b) is called the *trust region-based constraint*, which is a constraint on the KL divergence between the old policy and the new one.

To faithfully investigate how the algorithms work in practice, we consider a parametrized policy. In practical Deep RL algorithms, the policy are usually parametrized by Deep Neural Networks (DNNs). For discrete action space tasks where $|\mathcal{A}|= D$, the policy is parametrized by $\pi_\theta(s_t) = f_\theta^p(s_t)$. where f_θ^p is the DNN outputting a vector which represents a D-dimensional discrete distribution. For continuous action space tasks, it is standard to represent the policy by a Gaussian policy, i.e., $\pi_\theta(a|s_t) = \mathcal{N}(a|f_\theta^\mu(s_t), f_\theta^\Sigma(s_t))$ (Williams, 1992; Mnih et al., 2016), where f_θ^μ and f_θ^Σ are the DNNs which output the mean and covariance matrix of the Gaussian distribution. For simplicity, we will use the notation of θ rather than π in the our paper, e.g., $D_{\text{KL}}^s(\theta_{\text{old}}, \theta) \triangleq D_{\text{KL}}^s(\pi_{\theta_{\text{old}}}, \pi_\theta)$.

4 ANALYSIS OF THE "PROXIMAL" PROPERTY OF PPO

In this section, we will first give a brief review of PPO and then investigate the "proximal" property of PPO. We refer to "proximal" property as whether the algorithm could restrict the policy difference, regarding the probability ratio or the KL divergence between the new policy and the old one.

PPO employs a clipped surrogate objective to prevent the new policy from straying away from the old one. The clipped objective function of state-action (s_t, a_t) is

$$\begin{aligned} & L_t^{\text{CLIP}}(\theta) \\ & = \min\left(r_t(\theta, \theta_{\text{old}})A_t, \mathcal{F}^{\text{CLIP}}\left(r_t(\theta, \theta_{\text{old}}), \epsilon\right) A_t\right) \end{aligned} \tag{4}$$

where θ and θ_{old} are the parameters of the new policy and the old one respectively; $r_t(\theta, \theta_{\text{old}}) \triangleq \pi_\theta(a_t|s_t)/\pi_{\theta_{\text{old}}}(a_t|s_t)$ is the probability ratio, we will omit writing the parameter of the old policy θ_{old} explicitly; $s_t \sim \rho^{\pi_{\theta_{\text{old}}}}, a_t \sim \pi_{\theta_{\text{old}}}(\cdot|s_t)$ are the sampled states and actions; A_t is the estimated advantage value of $A^{\pi_{\theta_{\text{old}}}}(s_t, a_t)$; The clipping function $\mathcal{F}^{\text{CLIP}}$ is defined as

$$\mathcal{F}^{\text{CLIP}}(r_t(\theta), \epsilon) = \begin{cases} 1-\epsilon & r_t(\theta) \leq 1-\epsilon \\ 1+\epsilon & r_t(\theta) \geq 1+\epsilon \\ r_t(\theta) & else \end{cases} \tag{5}$$

where $(1-\epsilon, 1+\epsilon)$ is called the *clipping range*, $0 < \epsilon < 1$ is the parameter. The overall objective function is

$$L^{\text{CLIP}}(\theta) = \frac{1}{T}\sum_{t=1}^{T} L_t^{\text{CLIP}}(\theta) \tag{6}$$

To faithfully analyse how PPO works in practice, we assume that $s_i \neq s_j$ for all $i \neq j$ $(1 \leq i, j \leq T)$, since we could hardly meet exactly the same states in finite trials in large or continuous state space. This assumption means that only one action is sampled on each sampled states.

PPO restricts the policy by clipping the probability ratio between the new policy and the old one. Recently, researchers have raised concerns about whether this clipping mechanism can really restrict the policy (Wang et al., 2019; Ilyas et al., 2018). We investigate the following questions of PPO. The first one is that whether PPO could bound the probability ratio as it attempts to do. The second one is that whether PPO could enforce a well-defined trust region constraint, which is primarily concerned since that it is a theoretical indicator on the performance guarantee (see eq. (2)) (Schulman et al., 2015). We give an elaborate analysis of PPO to answer these two questions.

Question 1. *Could PPO bound the probability ratio within the clipping range as it attempts to do?*

In general, PPO could generate an effect of preventing the probability ratio from exceeding the clipping range too much, but it could not strictly bound the probability ratio. To see this, $L_t^{\text{CLIP}}(\theta)$ in eq. (4) can be rewritten as:

$$L_t^{\text{CLIP}}(\theta) = \begin{cases} (1-\epsilon)A_t & \begin{array}{l} r_t(\theta) \leq 1-\epsilon \\ \text{and } A_t < 0 \end{array} \quad (7a) \\ (1+\epsilon)A_t & \begin{array}{l} r_t(\theta) \geq 1+\epsilon \\ \text{and } A_t > 0 \end{array} \quad (7b) \\ r_t(\theta)A_t & \text{otherwise} \end{cases}$$

The case (7a) and (7b) are called the *clipping condition*. As the equation implies, once $r_t(\theta)$ is out of the clipping range (with a certain condition of A_t), the gradient of $L_t^{\text{CLIP}}(\theta)$ w.r.t. θ will be zero. As a result, the incentive, deriving from $L_t^{\text{CLIP}}(\theta)$, for driving $r_t(\theta)$ to go farther beyond the clipping range is removed.

However, in practice the probability ratios are known to be not bounded within the clipping range (Ilyas et al., 2018). The probability ratios on some tasks could even reach a value of 40, which is much larger than the upper clipping range 1.2 ($\epsilon = 0.2$, see our empirical results in Section 6). One main factor for this problem is that the clipping mechanism could not entirely remove incentive deriving from the overall objective $L^{\text{CLIP}}(\theta)$, which possibly push these out-of-the-range $r_t(\theta)$ to go farther beyond the clipping range. We formally describe this claim in following.

Theorem 2. *Given θ_0 that $r_t(\theta_0)$ satisfies the clipping condition (either 7a or 7b). Let $\nabla L^{\text{CLIP}}(\theta_0)$ denote the gradient of L^{CLIP} at θ_0, and similarly $\nabla r_t(\theta_0)$. Let $\theta_1 = \theta_0 + \beta \nabla L^{\text{CLIP}}(\theta_0)$, where β is the step size. If*

$$\langle \nabla L^{\text{CLIP}}(\theta_0), \nabla r_t(\theta_0) \rangle A_t > 0 \tag{8}$$

then there exists some $\bar{\beta} > 0$ such that for any $\beta \in (0, \bar{\beta})$, we have

$$|r_t(\theta_1) - 1| > |r_t(\theta_0) - 1| > \epsilon. \tag{9}$$

We provide the proof in Appendix A. As this theorem implies, even the probability ratio $r_t(\theta_0)$ is already out of the clipping range, it could be driven to go farther beyond the range (see eq. (9)). The condition (8) requires the gradient of the overall objective $L^{\text{CLIP}}(\theta_0)$ to be similar in direction to that of $r_t(\theta_0)A_t$. This condition possibly happens due to the similar gradients of different samples or optimization tricks. For example, the Momentum optimization methods preserve the gradients attained before, which could possibly make this situation happen. Such condition occurs quite often in practice. We made statistics over 1 million samples on benchmark tasks in Section 6, and the condition occurs at a percentage from 25% to 45% across different tasks.

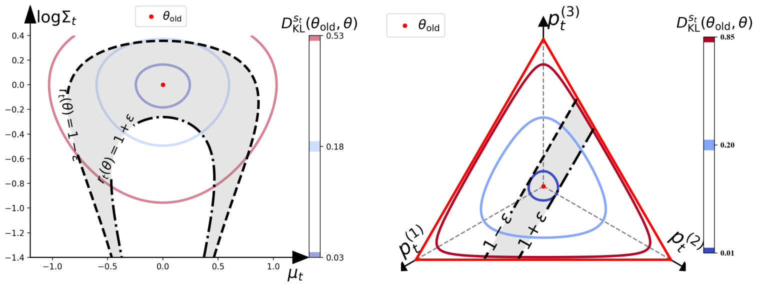
Question 2. *Could PPO enforce a trust region constraint?*

PPO does not explicitly attempt to impose a trust region constraint, i.e., the KL divergence between the old policy and the new one. Nevertheless, Wang et al. (2019) revealed that a different scale of the clipping range can affect the scale of the KL divergence. As they stated, under state-action (s_t, a_t), if the probability ratio $r_t(\theta)$ is not bounded, then *neither* could the corresponding KL divergence $D_{\text{KL}}^{s_t}(\theta_{\text{old}}, \theta)$ be bounded. Thus, together with the previous conclusion in Question 1, we can know that PPO could not bound KL divergence. In fact, even the probability ratio $r_t(\theta)$ is bounded, the corresponding KL divergence $D_{\text{KL}}^{s_t}(\theta_{\text{old}}, \theta)$ is not necessarily bounded. Formally, we have the following theorem.

Theorem 3. *Assume that for discrete action space tasks where $|\mathcal{A}| \geq 3$ and the policy is $\pi_\theta(s) = f_\theta^p(s)$, we have $\{f_\theta^p(s_t) | \theta \in \mathbb{R}\} = \{p | p \in \mathbb{R}^{+D}, \sum_d^D p^{(d)} = 1\}$; for continuous action space tasks where the policy is $\pi_\theta(a|s) = \mathcal{N}(a | f_\theta^\mu(s), f_\theta^\Sigma(s))$, we have $\{(f_\theta^\mu(s_t), f_\theta^\Sigma(s_t)) | \theta \in \mathbb{R}\} = \{(\mu, \Sigma) | \mu \in \mathbb{R}^D, \Sigma$ is a symmetric semidefinite $D \times D$ matrix$\}$. Let $\Theta = \{\theta | 1 - \epsilon \leq r_t(\theta) \leq 1 + \epsilon\}$. We have $\sup_{\theta \in \Theta} D_{\text{KL}}^{s_t}(\theta_{\text{old}}, \theta) = +\infty$ for both discrete and continuous action space tasks.*

To attain an intuition on how this theorem holds, we plot the sublevel sets of $r_t(\theta)$ and the level sets of $D_{\text{KL}}^{s_t}(\theta_{\text{old}}, \theta)$ for the continuous and discrete action space tasks respectively. As Fig. 1 illustrates, the KL divergences (solid lines) within the sublevel sets of probability ratio (grey area) could go to infinity.

It can be concluded that there is an obvious gap between



(a) Case of a Continuous Action Space Task
(b) Case of a Discrete Action Space Task

Figure 1: The grey area shows the sublevel sets of $r_t(\theta)$, i.e., $\Theta = \{\theta | 1 - \epsilon \leq r_t(\theta) \leq 1 + \epsilon\}$. The solid lines are the level sets of the KL divergence, i.e., $\{\theta | D_{\text{KL}}^{s_t}(\theta_{\text{old}}, \theta) = \delta\}$. (a) The case of a continuous action space case, where $dim(\mathcal{A}) = 1$. The action distribution under state s_t is $\pi_\theta(s_t) = \mathcal{N}(\mu_t, \Sigma_t)$, where $\mu_t = f_\theta^\mu(s_t), \Sigma_t = f_\theta^\Sigma(s_t)$. (b) The case of a discrete action space task, where $|\mathcal{A}| = 3$. The policy under state s_t is parametrized by $\pi_\theta(s_t) = (p_t^{(1)}, p_t^{(2)}, p_t^{(3)})$. Note that the level sets are plotted on the hyperplane $\sum_{d=1}^3 p_t^{(d)} = 1$ and the figure is showed from the view of elevation= 45° and azimuth= 45°.

bounding the probability ratio and bounding the KL divergence. Approaches which manage to bound the probability ratio could not necessarily bound KL divergence theoretically.

5 METHOD

In the previous section, we have shown that PPO could neither strictly restrict the probability ratio nor enforce a trust region constraint. We address these problems in the scheme of PPO with a general form

$$L_t(\theta) = \min\left(r_t(\theta)A_t, \mathcal{F}\left(r_t(\theta), \cdot\right) A_t\right) \tag{10}$$

where $\mathcal{F}$ is a clipping function which attempts to restrict the policy, "$\cdot$" in $\mathcal{F}$ means any hyperparameters of it. For example, in PPO, $\mathcal{F}$ is a ratio-based clipping function $\mathcal{F}^{\mathrm{CLIP}}(r_t(\theta), \epsilon)$ (see eq. (5)). We modify this function to promote the ability in bounding the probability ratio and the KL divergence. We now detail how to achieve this goal in the following sections.

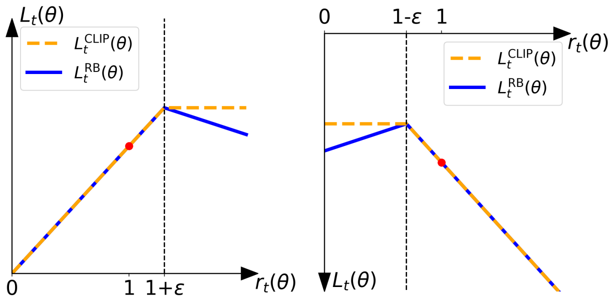
5.1 PPO WITH ROLLBACK (PPO-RB)

As discussed in Question 1, PPO could not strictly restrict the probability ratio within the clipping range: the clipping mechanism could not entirely remove the incentive for driving $r_t(\theta)$ to go beyond the clipping range, even $r_t(\theta)$ has already exceeded the clipping range. We address this issue by substituting the clipping function with a *rollback function*, which is defined as

$$\mathcal{F}^{\mathrm{RB}}(r_t(\theta), \epsilon, \alpha) = \begin{cases} -\alpha r_t(\theta) + (1+\alpha)(1-\epsilon) & r_t(\theta) \le 1-\epsilon \\ -\alpha r_t(\theta) + (1+\alpha)(1+\epsilon) & r_t(\theta) \ge 1+\epsilon \\ r_t(\theta) & \text{otherwise} \end{cases} \tag{11}$$

where $\alpha > 0$ is a hyperparameter to decide the force of the rollback. The corresponding objective function at timestep t is denoted as $L_t^{\mathrm{RB}}(\theta)$ and the overall objective function is $L^{\mathrm{RB}}(\theta)$. The rollback function $\mathcal{F}^{\mathrm{RB}}\left(r_t(\theta), \epsilon, \alpha\right)$ generates a negative incentive when $r_t(\theta)$ is outside of the clipping range. Thus it could somewhat neutralize the incentive deriving from the overall objective $L^{\mathrm{FB}}(\theta)$. Fig. 2 plots L_t^{RB} and L_t^{CLIP} as functions of the probability ratio $r_t(\theta)$. As the figure depicted, when $r_t(\theta)$ is over the clipping range, the slope of L_t^{RB} is reversed, while that of L_t^{CLIP} is zero.

The rollback operation could more forcefully prevent the probability ratio from being pushed away compared to the original clipping function. Formally, we have the following theorem.



(a) $A_t > 0$ (b) $A_t < 0$

Figure 2: Plots showing L_t^{RB} and L_t^{CLIP} as functions of the probability ratio $r_t(\theta)$, for positive advantages (left) and negative advantages (right). The red circle on each plot shows the starting point for the optimization, i.e., $r_t(\theta) = 1$. When $r_t(\theta)$ crosses the clipping range, the slope of L_t^{RB} is reversed, while that of L_t^{CLIP} is zero.

Theorem 4. *Let $\theta_1^{\mathrm{CLIP}} = \theta_0 + \beta\nabla L^{\mathrm{CLIP}}(\theta_0)$, $\theta_1^{\mathrm{RB}} = \theta_0 + \beta\nabla L^{\mathrm{RB}}(\theta_0)$. The indexes of the samples which satisfy the clipping condition is denoted as $\Omega = \{t | 1 \le t \le T, (A_t > 0 \text{ and } r_t(\theta_0) \ge 1+\epsilon) \text{ or } (A_t < 0 \text{ and } r_t(\theta_0) \le 1-\epsilon)\}$. If $t \in \Omega$ and $r_t(\theta_0)$ satisfies $\sum_{t' \in \Omega} \langle \nabla r_t(\theta_0), \nabla r_{t'}(\theta_0)\rangle A_t A_{t'} > 0$, then there exists some $\bar{\beta} > 0$ such that for any $\beta \in (0, \bar{\beta})$, we have*

$$\left|r_t(\theta_1^{\mathrm{RB}}) - 1\right| < \left|r_t(\theta_1^{\mathrm{CLIP}}) - 1\right|. \tag{12}$$

This theorem implies that the rollback function can improve its ability in preventing the out-of-the-range ratios from going farther beyond the range.

5.2 TRUST REGION-BASED PPO (TR-PPO)

As discussed in Question 2, there is a gap between the ratio-based constraint and the trust region-based one: bounding the probability ratio is not sufficient to bound the KL divergence. However, bounding the KL divergence is what we primarily concern about, since it is a theoretical indicator on the performance guarantee (see eq. (2)). Therefore, new mechanism incorporating the KL divergence should be taken into account.

The original clipping function uses the probability ratio as the element of the trigger condition for clipping (see eq. (5)). Inspired by the thinking above, we substitute the ratio-based clipping with a trust region-based one. Formally, the probability ratio is clipped when the policy

π_θ is out of the trust region,

$$\mathcal{F}^{\mathrm{TR}}(r_t(\theta),\delta)=\begin{cases} r_t(\theta_{\mathrm{old}}) & D_{\mathrm{KL}}^{s_t}(\theta_{\mathrm{old}},\theta)\geq\delta \\ r_t(\theta) & \text{otherwise}\end{cases} \tag{13}$$

where δ is the parameter, $r_t(\theta_{\mathrm{old}})=1$ is a constant. The incentive for updating policy is removed when the policy π_θ is out of the trust region, i.e., $D_{\mathrm{KL}}^{s_t}(\theta_{\mathrm{old}},\theta)\geq\delta$. Although the clipped value $r_t(\theta_{\mathrm{old}})$ may make the surrogate objective discontinuous, this discontinuity does not affect the optimization of the parameter θ at all, since the value of the constant does not affect the gradient.

In general, TR-PPO could combine both the strengths of TRPO and PPO: it is somewhat theoretically-justified (by the trust region constraint) while is simple to implement and only requires first-order optimization. Compared to TRPO, TR-PPO doesn't need to optimize θ through the KL divergence term $D_{\mathrm{KL}}^{s_t}(\theta_{\mathrm{old}},\theta)$. The KL divergence is just calculated to decide whether to clip $r_t(\theta)$ or not. Compared to PPO, TR-PPO uses a different metric of policy difference to restrict the policy. PPO applies a ratio-based metric, i.e., $\pi(a_t|s_t)/\pi_{\mathrm{old}}(a_t|s_t)$, which imposes an element-wise constraint on the sampled action point. While TR-PPO uses a trust region-based one, i.e., the KL divergence $\sum_a \pi_{\mathrm{old}}(a|s_t)\log(\pi_{\mathrm{old}}(a|s_t)/\pi(a|s_t))$, which imposes a summation constraint over the action space. The ratio-based constraint could impose a relatively strict constraint on actions which are not preferred by the old policy (i.e., $\pi_{\mathrm{old}}(a_t|s_t)$ is small), which may lead to limited sample efficiency when the policy is initialized from a bad solution (Wang et al., 2019). While the trust region-based one has no such bias and tends to perform more sample efficient in practice.

Finally, we should note the importance of the $\min(\cdot,\cdot)$ operation for all variants of PPO. Take TR-PPO as an example, the objective function incorporating the extra $\min(\cdot,\cdot)$ operation is

$$L_t^{\mathrm{TR}}(\theta)=\min\left(r_t(\theta)A_t,\mathcal{F}^{\mathrm{TR}}\left(r_t(\theta),\delta\right)A_t\right) \tag{14}$$

Schulman et al. (2017) stated that this extra $\min(\cdot,\cdot)$ operation makes $L_t^{\mathrm{TR}}(\theta)$ be a lower bound on the unclipped objective $r_t(\theta)A_t$. It should also be noted that such operation is important for optimization. As eq. (13) implies, the objective without $min(\cdot,\cdot)$ operation, i.e., $\mathcal{F}^{\mathrm{TR}}(r_t(\theta),\delta)A_t$, would stop updating once the policy violates the trust region, even the objective value is worse than the initial one, i.e., $r_t(\theta)A_t < r_t(\theta_{\mathrm{old}})A_t$. The $min(\cdot,\cdot)$ operation actually provides a remedy for this issue. To see this, eq. (14) is rewritten as

$$L_t^{\mathrm{TR}}(\theta)=\begin{cases} r_t(\theta_{\mathrm{old}})A_t & \begin{array}{l} D_{\mathrm{KL}}^{s_t}(\theta_{\mathrm{old}},\theta)\geq\delta \text{ and} \\ r_t(\theta)A_t\geq r_t(\theta_{\mathrm{old}})A_t \end{array} \\ r_t(\theta)A_t & \text{otherwise}\end{cases} \tag{15}$$

As can be seen, the ratio is clipped only if the objective value is improved (and the policy violates the constraint). We also experimented with the direct-clipping method, i.e., $\mathcal{F}^{\mathrm{TR}}(r_t(\theta),\delta)A_t$, and found it performs extremely bad in practice.

5.3 COMBINATION OF TR-PPO AND PPO-RB (TR-PPO-RB)

The trust region-based clipping still possibly suffers from the unbounded probability ratio problem, since we do not provide any negative incentive when the policy is out of the trust region. Thus we integrate the trust region-based clipping with the rollback mechanism.

$$\begin{aligned}&\mathcal{F}^{\mathrm{TR-RB}}(r_t(\theta),\delta,\alpha)\\ &=\begin{cases} -\alpha r_t(\theta) & D_{\mathrm{KL}}^{s_t}(\theta_{\mathrm{old}},\theta)\geq\delta \\ r_t(\theta) & \text{otherwise}\end{cases}\end{aligned} \tag{16}$$

As the equation implies, $\mathcal{F}^{\mathrm{TR-RB}}(r_t(\theta),\delta,\alpha)$ generates a negative incentive when π_θ is out of the trust region.

6 EXPERIMENT

We conducted experiments to investigate whether the proposed methods could improve ability in restricting the policy and accordingly benefit the learning.

To measure the behavior and the performance of the algorithm, we evaluate the probability ratio, the KL divergence, and the episode reward during the training process. The probability ratio and the KL divergence are measured between the new policy and the old one at each epoch. We refer one epoch as: 1) sample state-actions from a behavior policy $\pi_{\theta_{\mathrm{old}}}$; 2) optimize the policy π_θ with the surrogate function and obtain a new policy.

We evaluate the following algorithms. (a) *PPO*: the original PPO algorithm. We used $\epsilon=0.2$, which is recommended by (Schulman et al., 2017). We also tested PPO with $\epsilon=0.6$, denoted as *PPO-0.6*. (b) *PPO-RB*: PPO with the extra rollback trick. The rollback coefficient is set to be $\alpha=0.3$ for all tasks (except for the Humanoid task we use $\alpha=0.1$). (c) *TR-PPO*: trust region-based PPO. The trust region coefficient is set to be $\delta=0.025$ for all tasks (except for the Humanoid and HalfCheetah task we use $\delta=0.03$). (d) *TR-PPO-RB*: trust region-based PPO with rollback. The coefficients are set to be $\delta=0.03$ and $\alpha=0.05$ (except for the Humanoid and HalfCheetah task we use $\alpha=0.1$). The δ of TR-PPO-RB is set to be slightly larger than that of TR-PPO due to the existence of the rollback mechanism. (e) *TR-PPO-simple*: A vanilla version of TR-PPO, which does not include the $\min(\cdot,\cdot)$ operation. The δ is same as TR-PPO. (f) *A2C*: a classic policy gradient method. A2C has the exactly same implementations and hyperparameters as PPO except the clipping mechanism is removed. (g) *SAC*: Soft Actor-Critic, a state-of-the-art off-policy RL algorithm (Haarnoja et al., 2018). We adopt the implementations provided in (Haarnoja et al., 2018). (h) *PPO-SAC* and *TR-PPO-SAC*: two variants of SAC which use ratio-based clipping with $\epsilon = 0.2$ and trust region-based clipping with $\delta = 0.02$ respectively. All our proposed methods and PPO adopt exactly the same implementations and hyperparameters given in (Dhariwal et al., 2017) except the clipping function. This ensures that the differences are due to the algorithm changes instead of the implementations.

The algorithms are evaluated on continuous and discrete control benchmark tasks implemented in OpenAI Gym (Brockman et al., 2016), simulated by MuJoCo (Todorov et al., 2012) and Arcade Learning Environment (Bellemare et al., 2013). For continuous control tasks, we evaluate algorithms on 6 benchmark tasks (including a challenging high-dimensional Humanoid locomotion task). All tasks were run with 1 million timesteps except for the Humanoid task was 20 million timesteps. Each algorithm was run with 4 random seeds. The experiments on discrete control tasks are detailed in Appendix B.

Question 1. *Does PPO suffer from the issue in bounding the probability ratio and KL divergence as we have analysed?*

In general, PPO could not strictly bound the probability ratio within the predefined clipping range. As shown in Fig. 3, a reasonable proportion of the probability ratios of PPO are out of the clipping range on all tasks. Especially on Humanoid-v2, HalfCheetah-v2, and Walker2d-v2, even half of the probability ratios exceed. Moreover, as can be seen in Fig. 4, the maximum probability ratio of PPO can achieve more than 3 on all tasks (the upper clipping range is 1.2). In addition, the maximum KL divergence also grows as timestep increases (see Fig. 5).

Nevertheless, PPO still exerts an important effect on restricting the policy. To show this, we tested two variants of PPO: one uses $\epsilon = 0.6$, denoted as *PPO-0.6*; another one entirely removes the clipping mechanism, which collapses to the vanilla *A2C* algorithm. As expected, the probability ratios and the KL divergences of these two variants are much larger than that of PPO (we put the results in Appendix B, since the values are too large). Moreover, the performance of these two methods fluctuate dramatically during the training process (see Fig. 6).

In summary, it could be concluded that although the core clipping mechanism of PPO could not strictly restrict the probability ratio within the predefined clipping range, it could somewhat generate the effect on restricting the policy and benefit the learning. This conclusion is partly different from that of Ilyas et al. (2018). They drew a conclusion that “PPO’s performance depends heavily

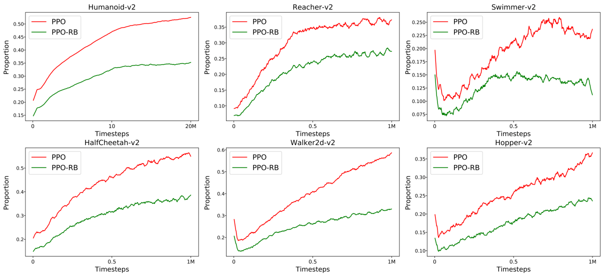


Figure 3: The proportions of the probability ratios which are out of the clipping range. The proportions are calculated over all sampled state-actions at that epoch. We only show the results of PPO and PPO-RB, since only these two methods have the clipping range parameter to judge whether the probability ratio is out of the clipping range.

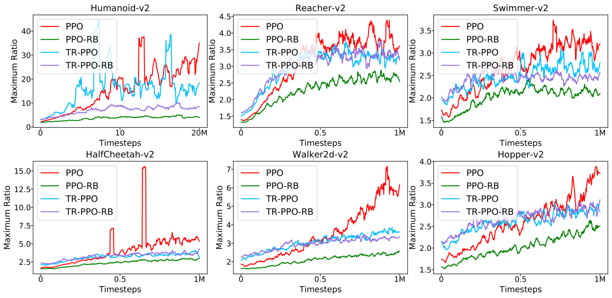


Figure 4: The maximum ratio over all sampled sates of each update during the training process. The results of TR-PPO-simple and PPO-0.6 are provided in Appendix. since their value are too large.

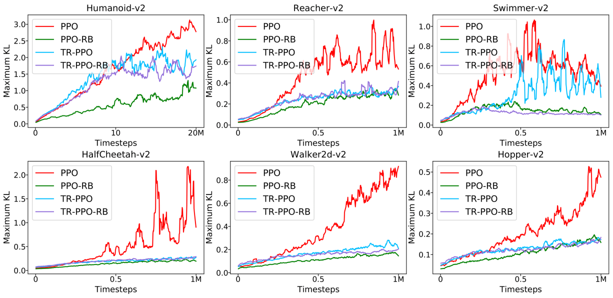


Figure 5: The maximum KL divergence over all sampled states of each update during the training process. The results of TR-PPO-simple and PPO-0.6 are plotted in Appendix, since their value are too large.

Table 1: a) Timesteps to hit thresholds within 1 million timesteps (except Humanoid with 20 million). b) Averaged top 10 episode rewards during training process. These results are averaged over 4 random seeds.

	(a) Timesteps to hit threshold ($\times 10^3$)								(b) Averaged top 10 episode tewards						
	Threshold	PPO	PPO-RB	TR-PPO	TR-PPO-RB	SAC	PPO-SAC	TR-PPO-SAC	PPO	PPO-RB	TR-PPO	TR-PPO-RB	SAC	PPO-SAC	TR-PPO-SAC
Hopper	3000	273	179	153	**130**	187	144	136	3612	3604	**3788**	3653	3453	3376	3439
Walker2d	3000	528	**305**	345	320	666	519	378	4036	4992	4874	**5011**	3526	3833	4125
Humanoid	5000	8410	8344	7580	6422	**314**	/	/	7510	7366	6842	6501	**7636**	/	/
Reacher	-4.5	230	206	211	**161**	352	367	299	-3.55	-1.61	-1.55	**-1.5**	-3.81	-3.44	-4.21
Swimmer	70	721	359	**221**	318	/	/	/	101	**126**	110	112	53	54	56
HalfCheetah	2100	/	374	227	266	39	45	**36**	1623	3536	4672	4048	10674	10826	**10969**

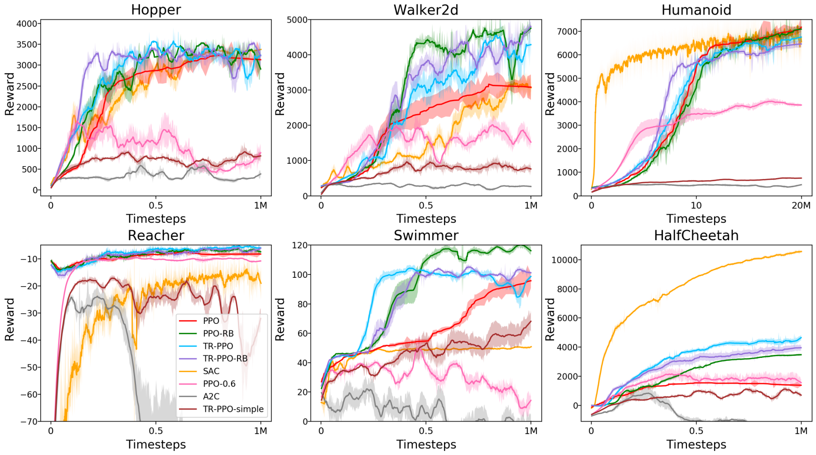


Figure 6: Episode rewards of the policy during the training process averaged over 4 random seeds. The shaded area depicts the mean ± the standard deviation.

on optimization tricks but not the core clipping mechanism". They got this conclusion by examining a variant of PPO which implements only the core clipping mechanism and removes additional optimization tricks (e.g., clipped value loss, reward scaling). This variant also fails in restricting policy and learning. However, as can be seen in our results, arbitrarily enlarging the clipping range or removing the core clipping mechanism can also result in failure. These results means that the core clipping mechanism also plays a critical and indispensable role in learning.

Question 2. *Could the rollback mechanism and the trust region-based clipping improve its ability in bounding the probability ratio or the KL divergence? Could it benefit policy learning?*

In general, our new methods could take a significant effect in restricting the policy compared to PPO. As can be seen in Fig. 3, the proportions of out-of-range probability ratios of PPO-RB are much less than those of the original PPO during the training process. The probability ratios and the KL divergences of PPO-RB are also much smaller than those of PPO (see Fig. 4 and 5). Although PPO-RB focuses on restricting the probability ratio, it seems that the improved ability of restriction on the probability ratio also leads to better restriction on the KL divergence. For the trust region-based clipping methods (TR-PPO and TR-PPO-RB), the KL divergences are also smaller than those of PPO (see Fig. 5). Especially, TR-PPO possesses the enhanced restriction ability on the KL divergence even it does not incorporate the rollback mechanism.

Our new methods could benefit policy learning in both sample efficiency and policy performance. As listed in Table 1 (a), all the three new methods require less samples to hit the threshold on all tasks. Especially, these new methods requires about 3/5 samples of PPO on Hopper, Walker2d and Swimmer. As Table 1 (b) lists, all the three proposed methods achieve much higher episode rewards than PPO does on Walker2d, Reacher, Swimmer, HalfCheetah (while performs fairly good as PPO on the remaining tasks).

The improvement on policy learning of the newly proposed methods may be considered as a success of the "trust region" theorem, which makes the algorithm perform less greedy to the advantage value of the old policy. To show this, we plot the entropy of the policy during the

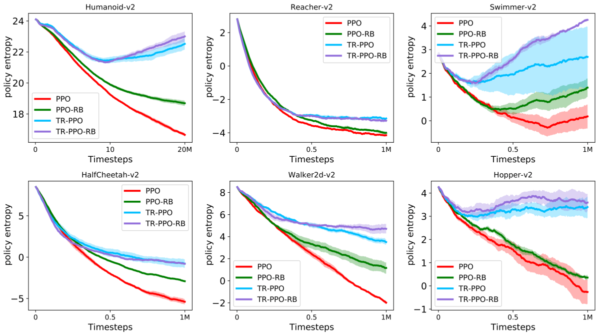


Figure 7: The policy entropy during the training process, averaged over 4 random seeds. The shaded area depicts the mean ± the standard deviation.

training process. As can be seen in Fig. 7, the entropy of the three proposed methods are much larger than that of PPO on almost all tasks, which means the policy performs less greedy and explores more sufficiently.

Question 3. *How well do the ratio-based methods perform compared to trust region-based ones?*

The ratio-based and trust region-based methods restrict the probability ratio and KL divergence respectively. The ratio-based methods include PPO and PPO-RB, while the trust region-based methods include TR-PPO and TR-PPO-RB. We consider two groups of comparisons, that is, PPO vs. TR-PPO and PPO-RB vs. TR-PPO-RB, since the only difference within each group is the the measurement of the policies.

In general, the trust region-based methods are more sample efficient than the ratio-based ones, and they could obtain a better policy on most of the tasks. As listed in Table 1 (a), both TR-PPO and TR-PPO-RB require much fewer episodes to achieve the threshold than PPO and PPO-RB do on all the tasks. Notably, on Hopper and Swimmer, TR-PPO requires almost half of the episodes of PPO. Besides, as listed in Table 1 (b), the episode rewards of TR-PPO and TR-PPO-RB are better than those of PPO and PPO-RB on 4 of the 6 tasks except Humanoid and Swimmer. As Fig. 7 plots, the entropies of trust region-based tends to be larger than those of ratio-based on all tasks, and the entropies even increase at the latter stages of training process. On the one hand, larger entropy may make trust region-based methods explore more sufficiently. On the other hand, it may make the policy hardly converge to the optimal policy.

Comparison with the state-of-art method: We compare our methods with soft actor critic (SAC) (Haarnoja et al., 2018). As Table 1 lists, our methods are fairly better than SAC on 4 of 6 tasks. In addition, the variants of PPO are much more computationally efficient than SAC. Within one million timesteps, the training wall-clock time for all variants of PPO is about 32 min; for SAC, 182 min (see Appendix B.4 for more detail). Furthermore, the variants of PPO require relatively less effort on hyperparameter tuning as we use the same hyperparameter across most of the tasks.

Our methods perform worse than SAC on the remaining 2 tasks. This may due to that we adopt an on-policy approach to learn the actor and critic while SAC adopts an off-policy one. We have also evaluated two variants of SAC which incorporate the clipping technique, termed as PPO-SAC and TR-PPO-SAC, which use ratio-based and trust region-based clipping respectively. As Table 1 lists, the introduced clipping mechanism could improve SAC on both sample efficiency and policy performance on 5 of 6 tasks (see Appendix B.2 for more detail).

7 CONCLUSION

Despite the effectiveness of the well-known PPO, it somehow lacks theoretical justification, and its actual optimization behaviour is less studied. To our knowledge, this is the first work to reveal the reason why PPO could neither strictly bound the probability ratio nor enforce a well-defined trust region constraint. Based on this observation, we proposed a trust region-based clipping objective function with a rollback operation. The trust region-based clipping is more theoretically justified while the rollback operation could enhance its ability in restricting policy. Both these two techniques significantly improve ability in restricting policy and maintaining training stability. Extensive results show the effectiveness of the proposed methods.

Deep RL algorithms have been notorious in its tricky implementations and require much effort to tune the hyperparameters (Islam et al., 2017; Henderson et al., 2018). Our three variants of the proposed methods are equally simple to implement and tune as PPO. They may be considered as useful alternatives to PPO.

Acknowledgements

This work is partially supported by National Science Foundation of China (61672280, 61373060, 61732006), AI+ Project of NUAA(56XZA18009), Postgraduate Research & Practice Innovation Program of Jiangsu Province (KYCX19_0195).

References

Bellemare, M. G., Naddaf, Y., Veness, J., and Bowling, M. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. Openai gym, 2016.

Chen, G., Peng, Y., and Zhang, M. An adaptive clipping approach for proximal policy optimization. *CoRR*, abs/1804.06461, 2018.

Dhariwal, P., Hesse, C., Klimov, O., Nichol, A., Plappert, M., Radford, A., Schulman, J., Sidor, S., and Wu, Y. Openai baselines. `https://github.com/openai/baselines`, 2017.

Duan, Y., Chen, X., Houthooft, R., Schulman, J., and Abbeel, P. Benchmarking deep reinforcement learning for continuous control. In *International Conference on Machine Learning*, pp. 1329–1338, 2016.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning*, pp. 1856–1865, 2018.

Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., and Meger, D. Deep reinforcement learning that matters. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Hmlinen, P., Babadi, A., Ma, X., and Lehtinen, J. PPO-CMA: Proximal policy optimization with covariance matrix adaptation. *arXiv preprint arXiv:1810.02541*, 2018.

Ilyas, A., Engstrom, L., Santurkar, S., Tsipras, D., Janoos, F., Rudolph, L., and Madry, A. Are deep policy gradient algorithms truly policy gradient algorithms? *arXiv preprint arXiv:1811.02553*, 2018.

Islam, R., Henderson, P., Gomrokchi, M., and Precup, D. Reproducibility of benchmarked deep reinforcement learning tasks for continuous control. *arXiv preprint arXiv:1708.04133*, 2017.

Kakade, S. and Langford, J. Approximately optimal approximate reinforcement learning. In *International Conference on Machine Learning (ICML)*, volume 2, pp. 267–274, 2002.

Kakade, S. M. A natural policy gradient. In *Advances in Neural Information Processing Systems 14*, volume 14, pp. 1531–1538, 2001.

Levine, S., Finn, C., Darrell, T., and Abbeel, P. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D., and Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning. In *International Conference on Machine Learning*, pp. 1928–1937, 2016.

Peters, J. and Schaal, S. Reinforcement learning of motor skills with policy gradients. *Neural networks*, 21(4): 682–697, 2008.

Peters, J., Mlling, K., and Altn, Y. Relative entropy policy search. In *AAAI'10 Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, pp. 1607–1612, 2010.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. Trust region policy optimization. In *International Conference on Machine Learning*, pp. 1889–1897, 2015.

Schulman, J., Moritz, P., Levine, S., Jordan, M. I., and Abbeel, P. High-dimensional continuous control using generalized advantage estimation. *International Conference on Learning Representations*, 2016.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., et al. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*, 2017.

Sutton, R. S., McAllester, D. A., Singh, S. P., and Mansour, Y. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems 12*, pp. 1057–1063, 1999.

Todorov, E., Erez, T., and Tassa, Y. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033. IEEE, 2012.

Wang, Y., He, H., Tan, X., and Gan, Y. Trust region-guided proximal policy optimization. *arXiv preprint arXiv:1901.10314*, 2019.

Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

Wu, Y., Mansimov, E., Grosse, R. B., Liao, S., and Ba, J. Scalable trust-region method for deep reinforcement learning using kronecker-factored approximation. In *Advances in Neural Information Processing Systems*, pp. 5279–5288, 2017.