Truncated Gaussian Policy for Debiased Continuous Control

Ganghun Lee[1,3], **Minji Kim**[2], **Minsu Lee**[*4], **Byoung-Tak Zhang**[*1,2,3]

[1]Interdisciplinary Program in Artificial Intelligence, Seoul National University
[2]Department of Computer Science, Seoul National University
[3]AIIS, Seoul National University
[4]School of AI Convergence, Sungshin Women's University
zxc8594@snu.ac.kr, cyqkcy01@snu.ac.kr, mslee@sungshin.ac.kr, btzhang@snu.ac.kr

Abstract

In continuous domains, reinforcement learning policies are often based on Gaussian distributions for their generality. However, the unbounded support of Gaussian policy can cause a bias toward sampling boundary actions in many continuous control tasks that impose action limits due to physical constraints. This "boundary action bias" can negatively impact training in algorithms like Proximal Policy Optimization. Despite this, it has been overlooked in many existing research and applications. In this paper, we revisit this issue by presenting illustrative explanations and analysis from the sampling point of view. Then, we introduce a truncated Gaussian policy with inherent bounds as a minimal alternative to mitigate the bias. However, we find that the plain truncated Gaussian policy may lay the counter-bias, preferring interior actions: to balance the bias, we ultimately propose a scale-adjusted truncated Gaussian policy, where the distribution scale shrinks if the location is near the boundaries. This property makes boundary actions deterministic more than in plain truncated Gaussian, but still less than in original Gaussian. Extensive empirical studies and comparisons on various continuous control tasks demonstrate that the truncated Gaussian policies significantly reduce the rate of boundary action usage, while scale-adjusted ones successfully balance the bias and counter-bias. It generally outperforms the Gaussian policy and shows competitive results compared to other approaches designed to counteract the bias.

Introduction

Reinforcement learning (RL) has emerged as a promising decision-making model for solving wide range of challenging problems such as games (Mnih et al. 2013; Silver et al. 2016; Fan et al. 2022), robotic manipulation (Kalashnikov et al. 2018; Han et al. 2023), locomotion (Margolis et al. 2024), visual tasks (Le et al. 2022; Franceschelli and Musolesi 2024), recommendation system (Afsar, Crump, and Far 2022), automation (Kabbani and Duman 2022; Hu et al. 2023; Ordouei et al. 2024), and training recent language models (Ouyang et al. 2022; Achiam et al. 2023). Across these diverse applications, actions generally fall into two categories: discrete and continuous. While discrete actions are sampled from a categorical distribution (Sutton et al. 1999; Schulman et al. 2017), continuous actions are usually sampled from noise (Lillicrap et al. 2015) or density distribution such as Gaussian (Mnih et al. 2016; Schulman et al. 2017; Haarnoja et al. 2018a).

In continuous domains, Gaussian policies are dominant for their strong generality and mathematical advantages, such as differentiability, appropriate stochasticity for continuous actions, and simple, intuitive parameters like location and scale, which are advantageous for modulating exploration and exploitation (Ribeiro 2004; Engel, Mannor, and Meir 2005; Kuss and Rasmussen 2003). However, in many continuous control tasks where the action range is practically limited, the unbounded support of Gaussian distributions requires careful regulation to ensure actions stay within these limits. A common solution is to clip the out-of-bound actions into the nearest boundary, as in proximal policy optimization (PPO) (Schulman et al. 2017), one of the most popular policy optimization methods. However, this approach can introduce a bias toward boundary actions (Chou, Maturana, and Scherer 2017; Fujita and Maeda 2018), inducing performance degradation.

In this paper, we revisit this bias as "boundary action bias" and analyze its nature in PPO with illustrative explanations and formalization. A fundamental solution to mitigate the bias is using bounded distributions to avoid out-of-bound samples, such as Beta (Chou, Maturana, and Scherer 2017; Petrazzini and Antonelo 2021; Xiao et al. 2023) or logit-normal distribution (squashed Gaussian) (Haarnoja et al. 2018b; Ciosek and Whiteson 2020; Jang 2021) employed in prior studies. However, the basic shapes and properties of such distributions differ from Gaussian to some extent. Instead, we propose introducing a truncated Gaussian for a policy, which applies minimal modifications to Gaussian to accommodate bounded support. However, our observation implies that a plain truncated Gaussian policy can induce counter-bias, preferring interior actions. This perspective suggests that a balanced distributional trend should be taken into account when designing continuous control policies. To balance the boundary action bias and its counter-bias in a truncated Gaussian policy, we finally propose the scale-adjusted truncated Gaussian policy, where a discounted scale near the boundaries facilitates the sampling of necessary boundary actions. Our extensive empirical studies on various continuous control tasks demon-

*Corresponding authors.

strate how distribution choice affects boundary action usage and performance, showing a great reduction of boundary actions with truncated Gaussian policies. Moreover, the scale-adjusted truncated Gaussian policy generally outperformed the original Gaussian policies in MuJoCo (Todorov, Erez, and Tassa 2012) locomotion and manipulation tasks and high-dimensional locomotion tasks in HumanoidBench (Sferrazza et al. 2024) and Deepmind Control Suite (Tunyasuvunakool et al. 2020). It also demonstrated competitive performance compared to existing methods designed to counteract boundary action bias, achieving the highest normalization scores in MuJoCo tasks. Our findings could help advance the understanding of how distributional sampling trends influence the continuous RL domain. Our contributions are summarized as follows:

- We revisit the phenomenon in which actions are concentrated near the boundary under the Gaussian policy of PPO and suggest truncated Gaussian as an alternative to regulate boundary actions.
- We explore the truncated Gaussian policy in PPO and present illustrative and empirical insights that plain truncated Gaussian clearly reduces the boundary actions but may excessively regulate them, leading to impaired performance.
- We ultimately propose the scale-adjusted truncated Gaussian policy to balance the trend in boundary actions, which generally outperforms other approaches for countering boundary action bias and improves performance over the plain Gaussian policy across various continuous control tasks.

Related Work

We introduce three categories of existing works that can directly or indirectly counteract boundary action bias.

Discretizing Continuous Actions. Continuous action space is generally believed to be more complicated than categorical actions (Lillicrap et al. 2015). Some previous works bypassed continuous action space and built other policy structures. Discretizing continuous action (Tang and Agrawal 2020; Zhu et al. 2024b) is an obvious approach, but it is being exposed to the curse of dimensionality and oversimplification of the task. Using the Bernoulli policy ((Seyde et al. 2021)) to formalize the problem as selecting only two boundary actions was shown to improve performance in specific tasks; however, many other control tasks will still require continuous actions.

Gradient Correction. In PPO, actions are sampled from Gaussian distribution and clipped into boundaries if they are out-of-bounds. From the agent's perspective, it does not realize that the actions are actually clipped and perceives all out-of-bound actions as distinct. Clipped Action Policy Gradient (CAPG) (Fujita and Maeda 2018; Xiao et al. 2022; Markowitz et al. 2023; Mohamadi et al. 2024) pointed out the above issue and provided an explanation that the Gaussian policy in PPO can suffer from boundary action bias due to the high variance of policy gradient estimation for out-of-bound actions. Then, CAPG corrected the policy gradient for out-of-bound actions to reduce the estimation variance, thereby informing the agent about action clipping. This way, the agent can more deliberately choose boundary actions than in the standard PPO, while maintaining its unbounded Gaussian policy. CAPG improved policy performance by reducing the burden of estimating the policy gradient. However, the usage of boundary actions remains at a similar ratio to that of standard PPO, raising questions about whether the boundary action bias has truly been mitigated.

Alternative Distributions. Studies using a Beta distribution (Chou, Maturana, and Scherer 2017; Petrazzini and Antonelo 2021; Jerome, Palmer, and Savani 2022; Xiao et al. 2023; Chen et al. 2023; Xu et al. 2024) indicated that oversampled out-of-bound actions, clipped into the same boundary actions, can confuse agents by exaggerating the value of boundary actions. They used Beta policy, with distinct shape parameters representing a wider variety of shapes than the Gaussian. However, Beta is mathematically more complex and less straightforward, making optimization harder. Instead of instituting other distribution, applying a Jacobian transformation is another approach. Soft Actor-Critic (SAC) (Haarnoja et al. 2018b) used hyperbolic tangent as a squashing function to enforce unbounded actions to be within (-1, 1), which belongs to logit-normal distribution (Atchison and Shen 1980; Ciosek and Whiteson 2020; Jang 2021) with different boundary scale. Although the base distribution is Gaussian and shares the same parameters, the logit-normal differs noticeably in aspects such as overall shape, mode-location coincidence, and unimodality. Compared to these distributions, the truncated Gaussian (Burkardt 2014) largely retains the shape assumptions of the original Gaussian while having bounded support, making it widely used in practical fields such as medicine and biology, as well as in neural network initialization (Glorot and Bengio 2010; He et al. 2015). While it remains unclear how these features affect various tasks, to the best of our knowledge, the use of the truncated Gaussian for RL policy has seldom been explored and has received much less attention compared to other distributions.

Advanced Distributions. Some interesting approaches using advanced distributions also exist, although they fall outside the primary scope of this paper. For instance, quasi-optimal learning (Li, Zhou, and Zhu 2023) incorporates an estimation process for support regions that contain only near-optimal actions. This approach alters the policy distribution to be more "twisty", aiming to prevent unsafe or unethical actions, which is particularly relevant for medical applications. Addtionally, studies on Q-exponential families (Zhu et al. 2024a; Kobayashi 2019) propose heavy-tailed Gaussian policies, demonstrating that such policies are generally more effective and improve on Gaussian.

Boundary Action Bias in Gaussian Policy

In this section, we explain the nature of boundary action bias in standard PPO and potential impacts on policy learning.

Boundary Action Bias. We define boundary action bias as an underlying tendency to sample boundary or near-

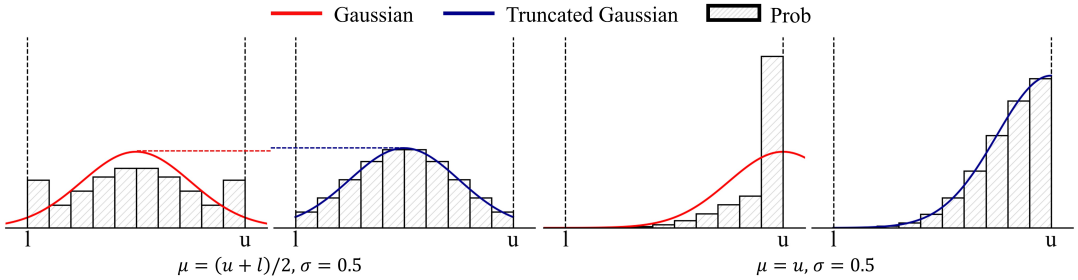

Figure 1: Comparison of Gaussian and truncated Gaussian policies. (Left) The Gaussian PDF (red line) and the truncated Gaussian PDF (blue line) are centered within the two action bounds l and u, sharing the same parameters $\mu = (u + l)/2$ and $\sigma = 0.5$. Gaussian policy samples actions from the Gaussian PDF but clips any out-of-bound actions to the nearest boundary, resulting in distorted action probabilities (hatched bars). In contrast, since the truncated Gaussian policy does not need to clip actions, it preserves the consistency between PDF and action probabilities. Given the same parameters, truncated Gaussian PDF is always taller than Gaussian PDF because the finite support of truncated Gaussian PDF requires a higher height to ensure the area under PDF sum up to 1. (Right) The two PDFs moved to the right action bound, where $\mu = u$. The distortion has deepened with the Gaussian policy, but the truncated Gaussian policy maintains PDF-action consistency.

boundary actions during policy learning, irrespective of task properties. Various causes may contribute to the bias, but we focus on the structure of the policy distribution and the mechanism of action sampling. We attribute the occurrence of the bias to the combination of Gaussian policy and action clipping. As a visual example, refer to the leftmost graph in Figure 1, where the red line represents the Gaussian PDF when the location μ is at the center of the action boundaries l and u (dashed lines). The hatch bars represent the actual action probabilities. The difference between the Gaussian PDF and action probabilities is noticeable because the unclipped samples $\overline{a} \sim \mathcal{N}((u + l)/2, 0.5)$ outside the boundaries are clipped to $a = \text{clip}(\overline{a}, l, u)$. As a result, the Gaussian overly imposes the boundary action probabilities even though they should have been the lowest, also violating the Gaussian unimodality. This also counteracts the strategy of relatively large initial scale parameters to promote exploration. The third graph in Figure 1 is where the Gaussian location parameter μ is moved to the boundary u. Since the chances of out-of-bound samples being clipped become much higher, the exaggeration of boundary actions is further intensified, strengthening the bias.

Relationship Between Bias and Gaussian Parameters. Standard PPO policy learns to adjust two Gaussian parameters μ and σ. As described in the previous paragraph, the distributional bias toward boundary actions grows as μ moves farther away from the center of two bounds. In fact, μ can even go outside the bounds, for example, $\mu = 3$ when $u = 1$ (see the left graph in Figure 2). This indeed happens even in the basic scenario like learning `HalfCheetah-v4` in MuJoCo with a standard PPO setting, where the maximum $|\mu|$ of matured policy reaches 3-5. In this case, most of the samples of $\overline{a}$ would be out-of-bound and clipped to u or l. This not only makes the policy extremely biased but also diminishes the role of scale parameter σ, since the variation in out-of-bound samples does not actually impact the final actions. If so, the policy will lose scale control and rely solely on the binary directions of μ. Combining these relationships, a feedback loop that reinforces bias can be formed. Since the policy itself tends to sample boundary actions more easily, the proportion of boundary actions that occupy the experience is also likely to be relatively large. This biased property of "reinforced exploration of boundary actions" gives more chances to behave boundary actions and again increases the probability of boundary actions if they are not significantly unsuitable, forming a loop. This loop can hinder the policy to identify the value of delicate actions. Moreover, the fact that an agent perceives clipped actions as unclipped also contributes to the loop, thinking it is exploring new actions and continuing to shift μ toward and beyond the boundary as in the above `HalfCheetah-v4` example. Once μ moves beyond the boundary for certain states, chances for interior actions become very scarce, losing elasticity and getting entrenched, leaving the possibility of premature or polarized policy.

Formalization. The amount of boundary action bias B_f in standard PPO can be quantified simply by the area under Gaussian PDF $f(x; \mu, \sigma)$ for outside of the bounds l, u:

$$B_f = \int_{-\infty}^{l} f(x)dx + \int_{u}^{\infty} f(x)dx \tag{1}$$

Since Gaussian has infinite support, $0 < B_f < 1$. This implies that standard PPO is always under the influence of the bias. The outside area increases as μ deviates further from the center $x = (u + l)/2$, thus $B_f \propto |\mu - (u + l)/2|$. This confirms that the bias grows according to the location deviated. For σ, if $l \leq \mu \leq u$, $B_f \propto \sigma$, and the lower bound of the bias $\inf_{\sigma>0} B_f(\sigma) = 0$. However, when $\mu > u$ or $\mu < l$, $B_f(\sigma)$ becomes convex, and the lower bound grows as $\inf_{\sigma>0} B_f(\sigma) \propto \mu$. This confirms that if the location is inside the boundaries, the large scale increases the bias, and when the location starts to move beyond the bounds, the amount that the scale can lower the bias decreases: this aligns with the diminution of the role of scale explained in the previous paragraph.

Method

In this section, we introduce truncated Gaussian distribution as preliminary, then propose a truncated Gaussian policy based on the distribution, presenting an illustrative comparison with the Gaussian policy. Finally, we propose a scale-adjusted truncated Gaussian policy to compensate for the potential counter-bias of plain truncated Gaussian policy.

Preliminaries

Markov Decision Process. In RL, the standard problem is defined as Markov Decision Process (MDP) represented as the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$ where $\mathcal{S}$ is state space, $\mathcal{A}$ is action space, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ is state transition function, and $\gamma \in [0, 1)$ is a discount factor. From the time $t = 0$, the agent starts with an initial state $s_0 \in \mathcal{S}$ and takes action $a_t \in \mathcal{A}$ when the state is s_t at every time t. The state at time t transfers to $s_{t+1} \sim \mathcal{P}(s_t, a_t)$, while environment emits the reward $r_t = \mathcal{R}(s_t, a_t)$. A policy $\pi \in \Pi$ maps state to action distributions, $\pi : \mathcal{S} \to p(\mathcal{A})$, and the action can be sampled from the policy by $a_t \sim \pi(s_t)$ where Π is a family of policies. The expected return under π is defined as $J_\pi = \mathbb{E}\left[\Sigma_{t=0}^{\infty}\right] \gamma^t r_t$, and the objective of RL is to find optimal policy $\pi^* = \operatorname{argmax}_\pi J_\pi$.

Truncated Gaussian Distribution. Truncated Gaussian distribution (Burkardt 2014) is a variant of Gaussian that has a bounded support. The probability density function (PDF) of truncated Gaussian for $l \leq x \leq u$ is given by

$$f(x; \mu, \sigma, l, u) = \frac{1}{\sigma} \cdot \frac{\varphi(\frac{x-\mu}{\sigma})}{\Phi(\frac{u-\mu}{\sigma}) - \Phi(\frac{l-\mu}{\sigma})}, \tag{2}$$

where μ, σ are location and scale parameters, l, u are lower and upper sample bounds. The φ and Φ are PDF and cumulative distribution function (CDF) of standard Gaussian distribution, respectively:

$$\varphi(x) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}x^2), \tag{3}$$

$$\Phi(x) = \frac{1}{2}(1 + \operatorname{erf}(x/\sqrt{2})), \tag{4}$$

where erf is the error function. The inverse CDF of truncated Gaussian, which is calculated as

$$F^{-1}(x; \mu, \sigma, l, u) = \Phi^{-1}\left(\Phi\left(\frac{l-\mu}{\sigma}\right) + x\left(\Phi\left(\frac{u-\mu}{\sigma}\right) - \Phi\left(\frac{l-\mu}{\sigma}\right)\right)\right)\sigma + \mu, \tag{5}$$

can be simulated using a uniform random variable x based on the standard Gaussian.

Truncated Gaussian Policy

We propose a new policy based on the truncated Gaussian to mitigate the boundary action bias. While it maintains the key assumptions of Gaussian, it directly samples actions from bounded support, hence not requiring any clippings, unlike Gaussian. To implement the truncated Gaussian policy, it is noteworthy that the gradient can be computationally unstable if the location parameter μ goes beyond the boundaries. Thus, we bound the location parameter with hyperbolic tangent function. Let g be a policy network without activation, and then the action of truncated Gaussian policy is described as follows:

$$a \sim f(x;\, \mu = \frac{u-l}{2} \cdot (\tanh(g(s)) + 1) + l,\, \sigma,\, l,\, u). \tag{6}$$

Now a bounded action can directly be simulated with the function f.

The brief comparison between the Gaussian and truncated Gaussian policies is illustrated in Figure 1. Two graphs on the left compare them with the same shape parameters where μ is located in the center of action bounds. Unlike Gaussian policies (red line) that distort actual action probabilities from original distributions, the truncated Gaussian policies (blue line) show consistent action probabilities with its PDF. Since the area under PDF should be 1, the truncated Gaussian PDF, which has finite support, is slightly taller than the Gaussian PDF, making all interior actions rise. Two graphs on the right are where μ is moved to the right bound, displaying more differences between them.

Since truncated Gaussian allows the policy to focus on actions more inside the boundaries, it is expected to enhance the exploration of delicate actions. However, this property may fall into the counter-bias, which can overly impose the policy of seeking small actions while avoiding boundary actions too much. The two graphs in Figure 2 clearly show this tendency. While Gaussian policy (red line) beyond the boundary samples boundary actions almost deterministic, truncated Gaussian policy (transparent blue line) shows limitation to raise the probability of boundary action due to the restriction of μ.

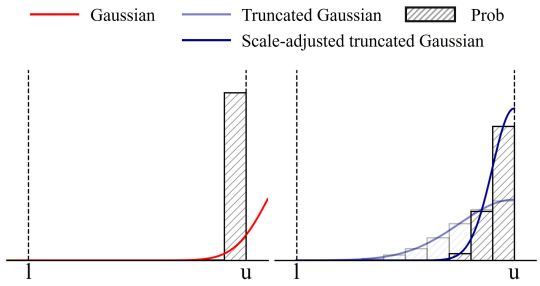

Figure 2: Comparison between Gaussian and truncated Gaussian policies when the location is extreme. While the Gaussian policy (red line) can decisively sample boundary actions, the truncated Gaussian policy (transparent blue line) inevitably allows more within-boundary actions due to the location limit. If the scale is adjusted, the truncated Gaussian policy can become more deterministic on boundary actions (dark blue line).

Scale-Adjusted Truncated Gaussian Policy

To avoid potential overcompensation of truncated Gaussian policy, we propose a scale-adjusted truncated Gaussian policy. The concept is illustrated in the right graph in Figure 2.

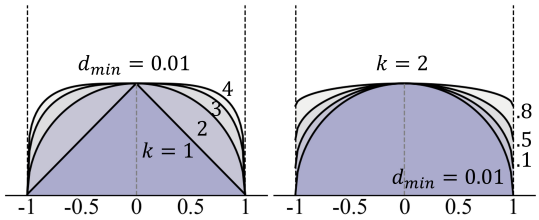

Figure 3: Scale-adjustment functions with different hyperparameters. k adjusts the kurtosis, and d_{min} regulates the maximum discount rate. The x-axis represents the location μ, where $l = -1$ and $u = 1$.

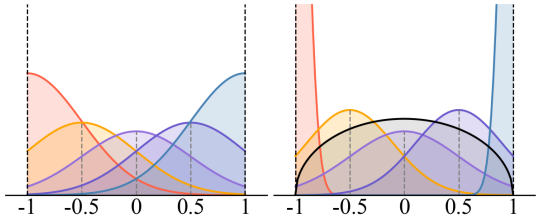

Figure 4: Shape comparison of a plain truncated Gaussian policy (left) and a scale-adjusted policy (right), where $l = -1$ and $u = 1$. The same colors indicate the same location and scale parameter. The scale-adjusted one is more decisive on boundary actions than the plain truncated Gaussian policy. The black line in the right graph is an example of scale-adjustment function.

Suppose the shape of plain truncated Gaussian (transparent blue line) was pressed close to the boundary u for the same location and scale parameters as the dark blue line. In that case, the policy becomes more deterministic for boundary actions. Because the scale is changed while the location is unchanged, we call this the scale-adjusted truncated Gaussian policy.

We suggest discounting scale parameters as the location approaches the boundaries. The adjusted scale σ' of truncated Gaussian is described with scale-adjustment function d as follows:

$$\sigma' = \sigma \cdot d. \tag{7}$$

The scale-adjustment function d can be freely set; however, since our objective is to adjust the scale to be discounted depending on the relation between location value μ and boundaries l and u, d becomes the function about them, as $d(\mu; l, u)$. We desire no discounting if the μ is at the center of actions, where $\mu = (u + l)/2$, so $d((u + l)/2; l, u) = 1$. Meanwhile, assuming the full discount (100% discount) is applied for the locations at boundaries, where $\mu = l$ or $\mu = u$, the discount factor should be $d(l; l, u) = 0$ and $d(u; l, u) = 0$. One of the continuous functions satisfying this is a semi-ellipse centered at $(u + l)/2$ and having a major axis $u - l$. In practice, the full discount of scale is not available since $\sigma' > 0$, we set a minimum amount of discounting ratio d_{min} and applied to the semi-ellipse. Taking the above into account, the full description of discount function d is as follows:

$$d(\mu; l, u) = \frac{\sqrt[k]{\left|(\frac{u-l}{2})^k - (\mu - \frac{u+l}{2})^k\right|}}{(u - l)/2} \cdot (1 - d_{min}) + d_{min}, \tag{8}$$

where k determines the kurtosis of the semi-ellipse; in $0 < k < 2$, scales are more globally discounted, while in $k > 2$, scale discount is more focused on actions around the boundary rather than neutral locations. The scale-adjustment functions according to different k and d_{min} are shown in Figure 3.

Figure 4 compares standard truncated Gaussian with scale-discounted version, using five different locations and the same scale parameter, where $l = -1$ and $u = 1$. In the center, where $\mu = 0$, there are no changes in scale between the two distributions. However, the adjusted scales become narrower as the locations approach the boundaries. Please note that while the scale is adjusted, the locations remain unchanged: the mode values always equal the given locations. With the settings of d_{min} and k, we can balance the distributional trend of the truncated Gaussian to how much to be deterministic on boundary actions.

Experiments

In this section, we conduct experiments to answer the following key questions: (1) How does each distributional trend, including our methods and existing approaches, influence boundary action usage? (2) How do boundary actions affect tasks differently? (3) Are the proposed methods competitive against other approaches? (4) Do the methods also perform well in high-dimensional action spaces? (5) How do scale-adjustment hyperparameters impact performance?

Experimental Setup

Tasks. We conduct experiments on eight continuous control tasks from MuJoCo (Todorov, Erez, and Tassa 2012), including six locomotion tasks (`HalfCheetah`, `Walker2d`, `Ant`, `Hopper`, `Humanoid`, and `Swimmer`) and two manipulation tasks (`Pusher` and `Reacher`), with version `v4`. For high-dimensional action space experiments, we use two locomotion tasks from Humanoid-Bench (`h1hand-walk`, `h1hand-reach`) (61 dimensions) (Sferrazza et al. 2024) and two from the DeepMind Control Suite (DMC) (`dog-walk`, `dog-run`) (Tunyasuvunakool et al. 2020) (38 dimensions). Action ranges are rescaled to (-1, 1), setting the boundaries to $l = -1$ and $u = 1$ for all tasks. Each MuJoCo result is averaged over 10 seeds, and HumanoidBench and DeepMind results are averaged over 5 seeds.

Methods. We selected two alternative distributions, Beta and logit-normal, and a gradient correction approach, CAPG, as baselines. Discretization methods were excluded since they do not involve continuous actions and could fundamentally change the nature of the problem. To summarize, we compare our truncated Gaussian **(TGaussian)** and scale-adjusted truncated Gaussian **(SA-TGaussian)** policy with (1) Gaussian policy with action clipping **(Gaussian)**, (2) Gaussian policy with gradient correction **(CAPG)**, (3) Beta

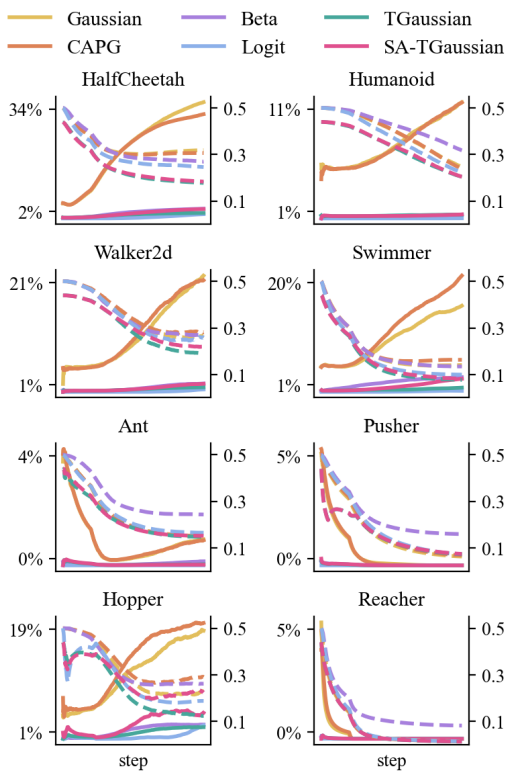

Figure 5: Average Boundary Action Rates (aBAR) (solid lines, left axis) and average standard deviations (dashed lines, right axis) for each method across tasks.

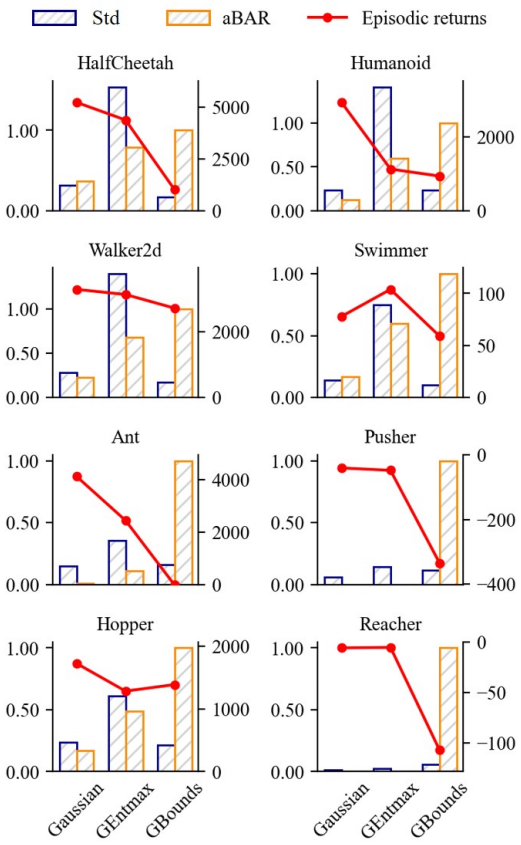

Figure 6: Relationship between aBAR, the average standard deviation of policy distribution, and episodic return. aBAR (decimal) and standard deviation share the left axis, and episodic return uses the right axis.

policy **(Beta)**, and (4) Hyperbolic tangent-based sample-squashing policy **(Logit)**. All policies are trained using PPO. We refer to ClearRL (Huang et al. 2022) for the standard PPO setup. All statistics are aggregated with 10 runs for each experimental setting. We use $k = 2$, $d_{min} = 0.01$ for our SA-TGaussian. The initial scale for all methods is set to $\sigma_{init} = 0.5$, which is a learnable parameter and independent of states, as in standard PPO setup.

Empirical Study on Boundary Action Bias

Measurement of Boundary Action Usage. To investigate the policy trend in the selection of boundary actions, we define average Boundary Action Rates (aBAR) as the average usage rates of actions falling within the top and bottom 1% of the action range for a single episode. Formally, the near-boundary action set as $\mathcal{A}' = \{a \in \mathcal{A} \,|\, |a| > 0.99\}$ assuming $u = 1$ and $l = -1$. The threshold value 0.99 was chosen based on intuition derived from a two-tailed test with a 2% significance level.

Figure 5 shows the aBAR during the training for baselines and our methods. It is obvious that the Gaussian family, Gaussian and CAPG, that uses action clipping, exhibited relatively higher aBAR (solid lines) than other methods. Considering the fact that the interval of aBAR corresponds to only 2% of the entire range of actions, approximately 10-30% high aBAR of Gaussian family in `HalfCheetah`, `Walker2d`, `Hopper`, `Humanoid`, and `Swimmer` is significantly high, indicating the occurrence of bias. The aBAR also steadily increased in those tasks, which implies the actions are getting more polarized as the training continues. However, in the two manipulation tasks `Pusher` and `Reacher`, aBAR drops to almost zero, even for the Gaussian family. Since those tasks require delicate actions due to the target positions, a smaller-scale of actions would have been more advantageous for task achievement than in locomotion tasks, even though the Gaussian family generally sampled more boundary actions initially. In contrast, the unbounded distribution family recorded a much lower aBAR of about 0-2%, demonstrating the debiasing effects of those distributions.

Impact of Boundary Action Bias. Another notable observation in Figure 5 is that the aBAR had a positive correlation with the standard deviations of policy distributions, which is directly related to the scale parameter. Consider a scenario where the Gaussian location is far beyond the boundary, deterministically resulting in boundary actions due to clipping. As the location continues to go further over the boundary, the influence of the standard deviation on action sampling

$\sigma_{init} = 0.5$	**Gaussian**	**CAPG**	**Beta**	**Logit**	**TGaussian**	**SA-TGaussian**
HalfCheetah	5165 ± 173.2	5399 ± 101.3	4749 ± 247.5	5175 ± 364.3	5216 ± 187.9	**5554 ± 115.3**
Walker2d	3153 ± 437.6	2871 ± 468.3	2622 ± 297.1	3559 ± 430.9	2746 ± 388.9	**3619 ± 376.2**
Ant	4001 ± 447.8	4124 ± 419.2	3393 ± 318.5	4099 ± 314.4	**4164 ± 299.1**	3972 ± 452.3
Hopper	1705 ± 267.3	1723 ± 276.4	1448 ± 181.0	1365 ± 158.2	**1825 ± 234.3**	1488 ± 239.6
Humanoid	3059 ± 553.5	3045 ± 622.6	**4102 ± 482.2**	2434 ± 405.2	2074 ± 277.1	3599 ± 577.0
Swimmer	77.23 ± 1.49	**81.32 ± 0.59**	70.99 ± 0.64	62.85 ± 0.86	59.74 ± 0.57	72.54 ± 0.55
Pusher	**-38.20 ± 2.50**	-42.34 ± 1.68	-48.16 ± 1.81	-42.31 ± 2.40	-40.91 ± 2.93	-41.06 ± 2.92
Reacher	-6.04 ± 0.92	-5.28 ± 0.82	-5.03 ± 0.54	-5.63 ± 0.61	**-4.75 ± 0.62**	-4.88 ± 0.61
Norm	0.609	0.679	0.311	0.451	0.554	**0.747**
$\sigma_{init} = 1.0$	**Gaussian**	**CAPG**	**Beta**	**Logit**	**TGaussian**	**SA-TGaussian**
HalfCheetah	1631 ± 26.55	2454 ± 34.81	-	3665 ± 143.4	**4508 ± 73.93**	4473 ± 89.45
Walker2d	3214 ± 450.9	3111 ± 348.4	-	3227 ± 477.4	2392 ± 379.8	**3279 ± 338.6**
Ant	3423 ± 435.9	3760 ± 356.1	-	3878 ± 356.6	3793 ± 353.6	**4169 ± 323.6**
Hopper	1502 ± 283.3	**1630 ± 201.4**	-	1432 ± 285.6	1465 ± 166.5	1462 ± 186.1
Humanoid	2729 ± 390.8	3184 ± 416.0	-	**3348 ± 578.6**	1856 ± 264.6	2742 ± 926.3
Swimmer	85.24 ± 0.60	82.91 ± 0.87	-	90.33 ± 0.61	51.56 ± 0.64	**95.62 ± 1.38**
Pusher	-49.92 ± 2.19	-49.18 ± 2.51	-	-52.26 ± 3.21	-46.91 ± 2.63	**-45.73 ± 2.76**
Reacher	-6.89 ± 0.80	-6.45 ± 1.25	-	-6.36 ± 1.23	**-4.81 ± 0.55**	-5.30 ± 0.92
Norm	0.373	0.604	-	0.549	0.435	**0.812**

Table 1: Score table of all methods in the setting of initial scale 0.5 and 1.0.

diminishes since samples outside the boundary are clipped to the boundary value. Consequently, if boundary actions had been frequent, the standard deviation would have decreased less. Indeed, in Figure 5, the Gaussian family with high aBAR exhibited a phenomenon where the standard deviation decrease stagnates relative to other methods. Since it is generally expected that a policy would fine-tune actions by reducing the standard deviation throughout training (Wang, Zariphopoulou, and Zhou 2018), this observation suggests the policy with high aBAR possibly causes premature convergence.

To investigate how aBAR affects the policy more deeply, we experimented with two additional setups using Gaussian policy.: (1) amplifying aBAR by inducing larger standard deviation with entropy maximization (Ahmed et al. 2019) **(GEntmax)** (using coefficient 0.01. Please note that standard continuous PPO setup uses coefficient 0), and (2) forcing actions to select only boundary actions to produce 100% aBAR **(GBounds)**. Figure 6 shows the final standard deviation, aBAR (decimal), and performance for each setup (The standard deviation for GBounds is exceptional due to the forced action space). Overall, as aBAR increased, the performance tended to degrade, implying that using boundary actions too often can be detrimental to policy performance. Additionally, the result explains why most existing PPO implementations for continuous action space set the entropy coefficient to zero despite the original intention of the entropy maximization term being to enhance exploration. However, particularly in the cases of `Swimmer` and `Hopper`, the observed trend was either attenuated or reversed, indicating that boundary actions may not be significantly detrimental to certain tasks. Since `Ant`, `Pusher`, and `Reacher` did not show a high aBAR, enforcing boundary actions severely impaired the performance. These results suggest that the boundary action bias in Gaussian may act differently depending on the given context.

Comparison

Overall Effectiveness. Table 1 summarizes the performance of all methods across tasks (see the upper section of the table where $\sigma_{init} = 0.5$). The corresponding learning curve is shown in Figure 7. We provide the normalization scores for each method on the eight continuous control tasks to facilitate the comparison. The overall performance demonstrated that our SA-TGaussian achieved the highest results, followed by CAPG. Interestingly, Gaussian ranked third, followed by our TGaussian, while Logit and Beta were positioned at the lower end. For CAPG, correcting the policy gradient enabled agents to select boundary actions actively rather than passively, likely contributing to the performance gain compared to Gaussian. Meanwhile, alternative distributions with bounded support, such as TGaussian, Logit, and Beta, underperformed Gaussian despite significantly reducing aBAR. This suggests that the overcompensation inherent in their distributional design my have limited their ability to leverage the potential benefits of boundary actions. However, by adjusting the scale to make boundary actions more deterministic, SA-TGaussian demonstrated significant performance improvements over Gaussian, TGaussian, and even surpassed CAPG. This result highlights the importance of balancing the trade-off between boundary actions and interior actions for effective policy learning. Our scale-adjusted truncated Gaussian policy generally appears to provide an effective solution for achieving this balance. Additionally, among the three alternative distributions, TGaussian achieved the highest performance compared to previous methods, Logit and Beta. This suggests that preserving the generality of Gaussian distributions can positively con-

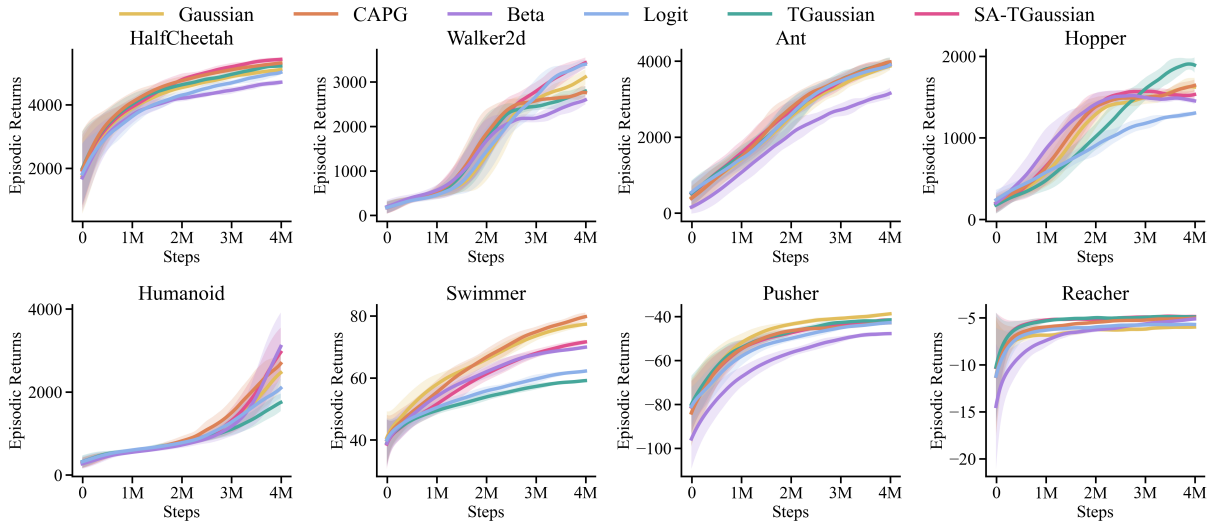

Figure 7: Learning curves during training for each method across MuJoCo tasks.

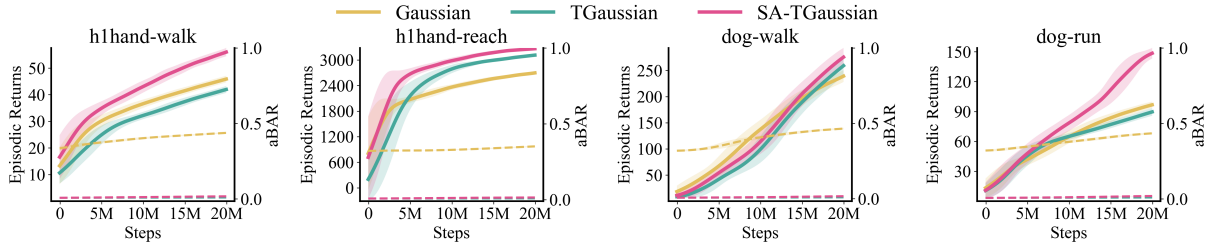

Figure 8: Learning curves depicting episodic returns (solid lines, left axis) and aBAR (dashed lines, right axis) for each method across tasks from HumanoidBench (`h1hand-walk`, `h1hand-reach`) and DMC (`dog-walk`, `dog-run`).

tribute to achieving overall good performance. Moreover, the shape properties of Logit and Beta, which feature longer tails than TGaussian and tend to avoid boundary actions when the location is near the boundary, may have exacerbated the negative impact of missing boundary actions.

Task-Wise Anaylsis. Although SA-TGaussian achieved the highest overall normalization score, each method tended to exhibit strengths in different tasks. For `HalfCheetah` and `Walker2d`, where GEntmax and GBounds showed intermediate levels of performance loss compared to Gaussian, SA-TGaussian performed the best. These tasks originally had high aBAR with Gaussian, suggesting that boundary action bias might have negatively affected performance. SA-TGaussian appears to effectively mitigate this bias by balancing boundary and interior actions. For `Ant`, which experienced significant performance degradation in GEntmax and GBounds while Gaussian showed moderate aBAR, our TGaussian performed slightly better than the others but remained similar to the Gaussian family. For `Hopper`, which exhibited minor degradation in GEntmax but recovered in GBounds, possibly reflecting a more neutral relationship to boundary actions, TGaussian achieved the highest score. `Humanoid` followed a pattern similar to `HalfCheetah` and `Walker2d`, with SA-TGaussian demonstrating strong performance. However, in this case, the Beta policy slightly outperformed other methods. For `Swimmer`, where GEntmax exhibited an inverse trend with relatively less performance loss in GBounds, the Gaussian family excelled. This result may reflect the need for boundary actions to enable the dramatic body movements required for forward motion, akin to a snake's slithering (Franceschetti et al. 2022). In manipulation tasks like `Pusher` and `Reacher`, Gaussian initially showed almost zero aBAR, indicating that boundary action effects might be minimal in these tasks. While Gaussian emerged as the best performer in `Pusher`, both TGaussian and SA-TGaussian also showed strong results. Furthermore, SA-TGaussian achieved the best performance in `Reacher`.

Ablation Study

Initial Scale. We report additional comparison with starting scale as $\sigma_{init} = 1.0$ in the lower section in Table 1, which tends to degrade the policy performance than the previous setting $\sigma_{init} = 0.5$ (Andrychowicz et al. 2020). Also, the large initial scale increases the likelihood of boundary action bias. As the standard deviation of Beta cannot exceed 0.5 for $\alpha > 1$ and $\beta > 1$, we excluded the Beta from this

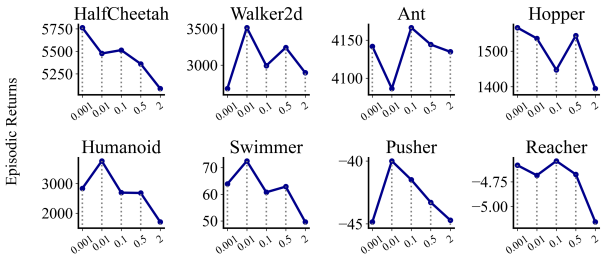

Figure 9: Results of the ablation study on d_{min}.

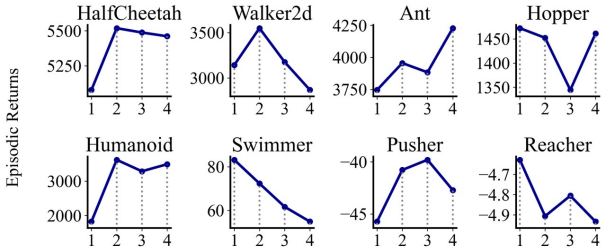

Figure 10: Results of the ablation study on k.

experiment. Overall, the performance of the Gaussian family diminished from $\sigma_{init} = 0.5$, while our SA-TGaussian policy less diminished, still attaining the best normalization score. General performance increase in `Swimmer` is exceptional, which was most beneficial from boundary actions.

High-Dimensional Action Space. To evaluate the effectiveness of our methods in tasks with much higher action dimensions, we conducted additional experiments in two distinct domains: HumanoidBench and the DMC. Figure 8 presents the learning curves from these experiments, where solid lines represent episode returns and dashed lines indicate aBAR. Consistent with the results observed in MuJoCo, the Gaussian policy recorded high aBAR near 50% for each task, while TGaussian and SA-TGaussian maintained almost 0% aBAR. Although high-dimensional tasks generally require precise and delicate actions, the Gaussian policy severely struggled due to polarized actions. Interestingly, for `h1hand-walk` and `dog-run`, TGaussian underperformed Gaussian, possibly due to overcompensation for boundary actions. In contrast, SA-TGaussian demonstrated superior performance across all tasks, emphasizing the importance of its balanced approach.

Scale-Adjustment Hyperparameters. d_{min} is the minimum of d, where $\sigma' = \sigma \cdot d$. Thus, with smaller d_{min}, the distribution is more decisive for boundary actions. Figure 9 represents the final episodic returns for each case $d_{min} = 0.001, 0.01, 0.1, 0.5, 2.0$. We found the decreasing trend in episodic returns when d_{min} gets larger. $d_{min} = 0.01$ generally performed the best, but the detailed trend of specific task was slightly differ from each other.

k is the kurtosis of the scale-discounting ellipse. A larger k means that the scale discount becomes less significant at locations farther from the boundary. Figure 10 shows the final episodic returns for each case $k = 1, 2, 3, 4$. While the generally best-performing value was $k = 2$, trends varied across tasks, indicating that the optimal k depended more on the task than on d_{min}. For example, `Swimmer` exhibited a clear decreasing trend in episodic returns as k increased, consistent with the main experiments. A larger k means being less decisive toward boundary actions during training, which could have negatively affected `Swimmer`, where boundary actions are mostly advantageous. Conversely, `Ant` showed a clear increasing trend in episodic returns as k increased. A larger k, being less decisive toward boundary actions, may have positively affected `Ant`, which is also consistent with the main experiments.

In summary, two ablation studies showed that the combination of $d_{min} = 0.01$ and $k = 2$ generally performs well, but the optimal setup may vary depending on the task, particularly in the context of distributional trends related to boundary actions.

Discussion

In this study, we revisited the issue of boundary action bias in standard PPO and proposed a truncated Gaussian policy as a mitigation strategy. Recognizing the counter-bias introduced by the truncated Gaussian, we further developed a scale-adjusted truncated Gaussian policy to address this limitation. Our results demonstrate that the scale-adjusted truncated Gaussian effectively balances the two sides of the bias, improving upon the plain Gaussian policy. The competitive performance of our method suggests that it could serve as a viable option not only for other tasks but also for other algorithms. Beyond proposing a method, we emphasize the importance of understanding the overall impact and roles of boundary and interior actions in RL tasks. This perspective, which has been largely overlooked due to the conventional reliance on Gaussian policies, offers valuable insights for advancing RL applications.

However, this study did not fully explore the internal mechanisms through which suppressing boundary actions can lead to better performance, leaving this as a key area for future research. Moreover, the experimental results showed irregular performance patterns across tasks for different methods, suggesting that the effect of distributional differences on final performance is influenced by more complex factors, including the nature of the tasks. Understanding these factors will be important for further improvement.

It also seems that even truncated Gaussian or other non-Gaussian distributions significantly reduced the usage of boundary actions, the performance gap was not that large. If the patterns of action in policies differ but the performances are similar, this may suggest an inherent ability of RL to adapt to the given distribution rather than being solely constrained by it. This could be attributed to the fact that continuous control tasks provide highly diverse pathways to solutions, even though these pathways are extremely sparse within the scope of the entire population. Designing toy problems to better observe the impact of distributions or conducting experiments with tighter control over variables influencing learning would be a valuable direction for future research.

Acknowledgments

This work was partly supported by the IITP (RS-2021-II212068-AIHub/10%, RS-2021-II211343-GSAI/15%, RS-2022-II220951-LBA/15%, RS-2022-II220953-PICA/15%), NRF (RS-2024-00358416/15%, RS-2024-00353991-SPARC/10%, RS-2023-00274280-HEI/10%), and KEIT (RS-2024-00423940/10%) grant funded by the Korean government.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Afsar, M. M.; Crump, T.; and Far, B. 2022. Reinforcement learning based recommender systems: A survey. *ACM Computing Surveys*, 55(7): 1–38.

Ahmed, Z.; Le Roux, N.; Norouzi, M.; and Schuurmans, D. 2019. Understanding the impact of entropy on policy optimization. In *International conference on machine learning*, 151–160. PMLR.

Andrychowicz, M.; Raichuk, A.; Stańczyk, P.; Orsini, M.; Girgin, S.; Marinier, R.; Hussenot, L.; Geist, M.; Pietquin, O.; Michalski, M.; et al. 2020. What matters in on-policy reinforcement learning? a large-scale empirical study. *arXiv preprint arXiv:2006.05990*.

Atchison, J.; and Shen, S. M. 1980. Logistic-normal distributions: Some properties and uses. *Biometrika*, 67(2): 261–272.

Burkardt, J. 2014. The truncated normal distribution. *Department of Scientific Computing Website, Florida State University*, 1(35): 58.

Chen, W.; Peng, J.; Chen, J.; Zhou, J.; Wei, Z.; and Ma, C. 2023. Health-considered energy management strategy for fuel cell hybrid electric vehicle based on improved soft actor critic algorithm adopted with Beta policy. *Energy Conversion and Management*, 292: 117362.

Chou, P.-W.; Maturana, D.; and Scherer, S. 2017. Improving stochastic policy gradients in continuous control with deep reinforcement learning using the beta distribution. In *International conference on machine learning*, 834–843. PMLR.

Ciosek, K.; and Whiteson, S. 2020. Expected policy gradients for reinforcement learning. *Journal of Machine Learning Research*, 21(52): 1–51.

Engel, Y.; Mannor, S.; and Meir, R. 2005. Reinforcement learning with Gaussian processes. In *Proceedings of the 22nd international conference on Machine learning*, 201–208.

Fan, L.; Wang, G.; Jiang, Y.; Mandlekar, A.; Yang, Y.; Zhu, H.; Tang, A.; Huang, D.-A.; Zhu, Y.; and Anandkumar, A. 2022. Minedojo: Building open-ended embodied agents with internet-scale knowledge. *Advances in Neural Information Processing Systems*, 35: 18343–18362.

Franceschelli, G.; and Musolesi, M. 2024. Reinforcement learning for generative ai: State of the art, opportunities and open research challenges. *Journal of Artificial Intelligence Research*, 79: 417–446.

Franceschetti, M.; Lacoux, C.; Ohouens, R.; Raffin, A.; and Sigaud, O. 2022. Making reinforcement learning work on swimmer. *arXiv preprint arXiv:2208.07587*.

Fujita, Y.; and Maeda, S.-i. 2018. Clipped action policy gradient. In *International Conference on Machine Learning*, 1597–1606. PMLR.

Glorot, X.; and Bengio, Y. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 249–256. JMLR Workshop and Conference Proceedings.

Haarnoja, T.; Zhou, A.; Abbeel, P.; and Levine, S. 2018a. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, 1861–1870. PMLR.

Haarnoja, T.; Zhou, A.; Hartikainen, K.; Tucker, G.; Ha, S.; Tan, J.; Kumar, V.; Zhu, H.; Gupta, A.; Abbeel, P.; et al. 2018b. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*.

Han, D.; Mulyana, B.; Stankovic, V.; and Cheng, S. 2023. A survey on deep reinforcement learning algorithms for robotic manipulation. *Sensors*, 23(7): 3762.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, 1026–1034.

Hu, M.; Zhang, J.; Matkovic, L.; Liu, T.; and Yang, X. 2023. Reinforcement learning in medical image analysis: Concepts, applications, challenges, and future directions. *Journal of Applied Clinical Medical Physics*, 24(2): e13898.

Huang, S.; Dossa, R. F. J.; Ye, C.; Braga, J.; Chakraborty, D.; Mehta, K.; and AraÃšjo, J. G. 2022. Cleanrl: High-quality single-file implementations of deep reinforcement learning algorithms. *Journal of Machine Learning Research*, 23(274): 1–18.

Jang. 2021. *Policy Gradient in Bounded Continuous Action Space using Logitnormal distribution*. Ph.D. thesis, Graduate School of Seoul National University.

Jerome, J.; Palmer, G.; and Savani, R. 2022. Market making with scaled beta policies. In *Proceedings of the Third ACM International Conference on AI in Finance*, 214–222.

Kabbani, T.; and Duman, E. 2022. Deep reinforcement learning approach for trading automation in the stock market. *IEEE Access*, 10: 93564–93574.

Kalashnikov, D.; Irpan, A.; Pastor, P.; Ibarz, J.; Herzog, A.; Jang, E.; Quillen, D.; Holly, E.; Kalakrishnan, M.; Vanhoucke, V.; et al. 2018. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on robot learning*, 651–673. PMLR.

Kobayashi, T. 2019. Student-t policy in reinforcement learning to acquire global optimum of robot control. *Applied Intelligence*, 49(12): 4335–4347.

Kuss, M.; and Rasmussen, C. 2003. Gaussian processes in reinforcement learning. *Advances in neural information processing systems*, 16.

Le, N.; Rathour, V. S.; Yamazaki, K.; Luu, K.; and Savvides, M. 2022. Deep reinforcement learning in computer vision: a comprehensive survey. *Artificial Intelligence Review*, 1–87.

Li, Y.; Zhou, W.; and Zhu, R. 2023. Quasi-optimal reinforcement learning with continuous actions. *arXiv preprint arXiv:2301.08940*.

Lillicrap, T. P.; Hunt, J. J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; and Wierstra, D. 2015. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.

Margolis, G. B.; Yang, G.; Paigwar, K.; Chen, T.; and Agrawal, P. 2024. Rapid locomotion via reinforcement learning. *The International Journal of Robotics Research*, 43(4): 572–587.

Markowitz, J.; Gardner, R. W.; Llorens, A.; Arora, R.; and Wang, I.-J. 2023. A risk-sensitive approach to policy optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 15019–15027.

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

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; and Riedmiller, M. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

Mohamadi, N.; Niaki, S. T. A.; Taher, M.; and Shavandi, A. 2024. An application of deep reinforcement learning and vendor-managed inventory in perishable supply chain management. *Engineering Applications of Artificial Intelligence*, 127: 107403.

Ordouei, M.; Broumandnia, A.; Banirostam, T.; and Gilani, A. 2024. Optimization of energy consumption in smart city using reinforcement learning algorithm. *International Journal of Nonlinear Analysis and Applications*, 15(1): 277–290.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744.

Petrazzini, I. G.; and Antonelo, E. A. 2021. Proximal policy optimization with continuous bounded action space via the beta distribution. In *2021 IEEE symposium series on computational intelligence (SSCI)*, 1–8. IEEE.

Ribeiro, M. I. 2004. Gaussian probability density functions: Properties and error characterization. *Institute for Systems and Robotics, Lisboa, Portugal*.

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

Seyde, T.; Gilitschenski, I.; Schwarting, W.; Stellato, B.; Riedmiller, M.; Wulfmeier, M.; and Rus, D. 2021. Is bang-bang control all you need? solving continuous control with bernoulli policies. *Advances in Neural Information Processing Systems*, 34: 27209–27221.

Sferrazza, C.; Huang, D.-M.; Lin, X.; Lee, Y.; and Abbeel, P. 2024. Humanoidbench: Simulated humanoid benchmark for whole-body locomotion and manipulation. *arXiv preprint arXiv:2403.10506*.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. 2016. Mastering the game of Go with deep neural networks and tree search. *nature*, 529(7587): 484–489.

Sutton, R. S.; McAllester, D.; Singh, S.; and Mansour, Y. 1999. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12.

Tang, Y.; and Agrawal, S. 2020. Discretizing continuous action space for on-policy optimization. In *Proceedings of the aaai conference on artificial intelligence*, volume 34, 5981–5988.

Todorov, E.; Erez, T.; and Tassa, Y. 2012. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, 5026–5033. IEEE.

Tunyasuvunakool, S.; Muldal, A.; Doron, Y.; Liu, S.; Bohez, S.; Merel, J.; Erez, T.; Lillicrap, T.; Heess, N.; and Tassa, Y. 2020. dm_control: Software and tasks for continuous control. *Software Impacts*, 6: 100022.

Wang, H.; Zariphopoulou, T.; and Zhou, X. 2018. Exploration versus exploitation in reinforcement learning: A stochastic control approach. *arXiv preprint arXiv:1812.01552*.

Xiao, L.; Hong, S.; Xu, S.; Yang, H.; and Ji, X. 2022. IRS-aided energy-efficient secure WBAN transmission based on deep reinforcement learning. *IEEE Transactions on Communications*, 70(6): 4162–4174.

Xiao, Q.; Jiang, L.; Wang, M.; and Zhang, X. 2023. An Improved Distributed Sampling PPO Algorithm Based on Beta Policy for Continuous Global Path Planning Scheme. *Sensors*, 23(13): 6101.

Xu, G.; Lin, Z.; Wu, Q.; Tan, J.; and Chan, W. K. V. 2024. Bi-level hierarchical model with deep reinforcement learning-based extended horizon scheduling for integrated electricity-heat systems. *Electric Power Systems Research*, 229: 110195.

Zhu, L.; Shah, H.; Wang, H.; Nagai, Y.; and White, M. 2024a. q-exponential family for policy optimization. *arXiv preprint arXiv:2408.07245*.

Zhu, Y.; Wang, Z.; Zhu, Y.; Chen, C.; and Zhao, D. 2024b. Discretizing Continuous Action Space with Unimodal Probability Distributions for On-Policy Reinforcement Learning. *arXiv preprint arXiv:2408.00309*.