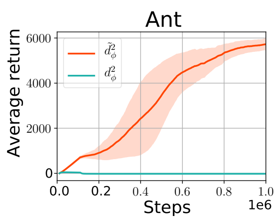

Figure A

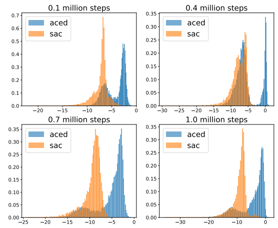

Figure B

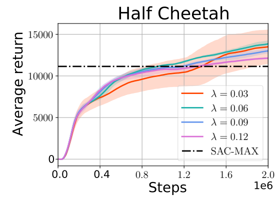

Figure C

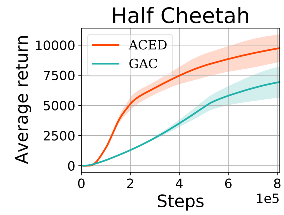

Figure D

We thank the reviewers for insightful comments. We are improving the paper by incorporating the reviewers' suggestions.

Response to reviewer #1:

Why we present two SBR instances (Table 2) The first d_ϕ^2 requires sampling from a uniform distribution to approximate f, which leads to a large variance of gradient and then hurts the performance (see Figure A). Therefore, we propose $\tilde{d}_\phi^2$ to reduce the variance, which provides a closed-form expression of f and does not require sampling. We will add more discussions about these two instances in the final version, if accepted.

Discussion about Theorem 5 Theorem 5 proposes a bound on the performance gap between the regularized optimal policy π_α^* and the original optimal policy π^*. This theorem shows that the gap depends on $\Delta = \phi(U^2) - \phi(0)$. Though larger Δ can encourage actions away from each other better, it may lead to worse performance of π_α^*. We will add the discussion about Theorem 5 in the final version, if accepted.

Performance difference between SAC and our method (ACED) When we use Gaussian policies, the hyperparameters of ACED (Table 6 in Appendix) are the same as that of SAC except for the target value T. Therefore, the performance difference mainly comes from different regularization.

More examples that demonstrate the benefits of ACED To show the improved efficiency of ACED, we evaluate ACED ($N = 2$) against SAC with an ensemble of policies. The ensemble size is 5, and each policy outputs a Gaussian distribution. Experiments show that 1000 updates in ACED cost 21.4s while those in SAC cost 33.7s. The core reason is that ACED does not need to compute probability density, which requires the forward prediction of all networks. We will provide results in details in the final version, if accepted.

Response to reviewer #2:

Compare the two SBR instances In practice, $\tilde{d}_\phi^2$ is better than d_ϕ^2. Please see the response to reviewer #1 for details.

Histograms of entropies of ACED and SAC In Figure B, we provide the histograms of entropies of SAC and ACED during training. It shows that the difference of stochasticity exists through almost the whole training procedure.

Response to reviewer #3:

The meaning of "we did not tune the hyperparameters" For SAC and TD3, we use the hyperparameters provided by their authors. Therefore, we did not tune them again. For our algorithm, we use the same hyperparameters (see Table 6 in Appendix) as SAC, if possible. We did not tune N and T for the results in Section 5.1.

Examples of complex policies We provided evaluation with normalizing flow policies as an example in Section 5.2. Moreover, expressing the policies by a noisy network [10] requires an additional classification network to estimate entropy [44]. We will provide more examples in the final version, if accepted.

Response to reviewer #4:

Discussion about Theorem 5 Please refer to the response to reviewer #1.

Should the community switch from SAC to ACED? If computing the probability density of policies is time-consuming or infeasible, the answer is "yes". For example, when parameterizing policies by noisy networks, ACED is a better choice than SAC. Otherwise, the answer depends on the performance of SAC and ACED.

How sensitive ACED is to hyperparameters? ACED is insensitive to the sample number N and the target value T. The sensitivity analysis for N can be found in Section 5.3 and Appendix C.5. Figure C shows the sensitivity analysis for T. Here, we set T as $\mathcal{F}(\mathcal{N}(\mathbf{0}, \lambda\mathbf{I}))$, where $\mathcal{N}(\mathbf{0}, \lambda\mathbf{I})$ is a Gaussian distribution with the covariance matrix $\lambda\mathbf{I}$.

Connections with related work and our novelty contributions Most existing regularization [18,23,51] takes the form of $\mathbb{E}_{a\sim\pi(\cdot|s)}\left[f(\pi(a|s))\right]$. We propose a novel regularization form (Equation 4) to encourage stochasticity. Due to the different forms, previous regularization often requires computing probability density to estimate entropy but our regularization does not. Moreover, unlike previous regularization, our regularization considers the distances between actions and thus incorporates geometric information.

What is Gini mean difference? Gini mean difference is a measure of statistical dispersion. Given a distribution D, it is defined by $\mathbb{E}|X_1 - X_2|$, where X_1 and X_2 are sampled from D independently.

What is the difference between Figures 1(a) and 1(d)? The reward function in Figure 1(a) is unimodal, while the other is trimodal. Figure 1(d) shows that our regularization can lead to a multi-modal distribution.

What does the notation of $\phi^{(n+1)}$ mean on line 66? $\phi^{(n+1)}$ denote the $(n+1)$th order derivative of ϕ.

Comparison with methods that do not require a specific form of policy: We compare ACED with GAC [47] in HalfCheetah-v2 . Figure D shows that our method outperforms GAC. We note that GAC is expensive in computation. GAC takes more than 55h to train the policy with about 0.8 million steps, while ACED takes less about 6h. As the experiments are still running, we will provide more results in the final version, if accepted.