Variational Bayesian Parameter-Based Policy Exploration

Tikara Hosino
Technology Research & Innovation
Nihon Unisys, Ltd.
1-1-1 Toyosu, Koto-ku, Tokyo Japan
chikara.hoshino@unisys.co.jp

***Abstract*—Reinforcement learning has shown success in many tasks that cannot provide explicit training samples and can only provide rewards. However, because of a lack of robustness and the need for hard hyperparameter tuning, reinforcement learning is not easily applicable in many new situations. One reason for this problem is that the existing methods do not account for the uncertainties of rewards and policy parameters. In this paper, for parameter-based policy exploration, we use a Bayesian method to define an objective function that explicitly accounts for reward uncertainty. In addition, we provide an algorithm that uses a Bayesian method to optimize this function under the uncertainty of policy parameters in continuous state and action spaces. The results of numerical experiments show that the proposed method is more robust than comparing method against estimation errors on finite samples, because our proposal balances reward acquisition and exploration.**

***Index Terms*—Reinforcement Learning, Parameter-Based method, Bayesian Learning, Variational Approximation, Continuous Control, Exploration and Exploitation Trade-Off**

I. Introduction

Reinforcement learning has been successfully used in many closed simulation domains such as video games [1] and the game Go [2]. However, using reinforcement learning for real-world tasks, such as robot control, remains difficult. This paper focuses on two issues that make it difficult to apply reinforcement learning in continuous control domains.

First, the stochastic policy that determines the action at each time stochastically does not produce smooth trajectories. This results in large variances in the estimated gradient with policy gradient methods such as REINFORCE [3]. Several methods such as the Deep Deterministic Policy Gradient (DDPG) [4] and the Twin Delayed Deep Deterministic policy gradient (TD3) [5] have been proposed to use deterministic policies to avoid this difficulty. Parameter-based policy exploration, such as Policy Gradients with parameter-based exploration [6], is a way to introduce a distribution of policy parameters. This hierarchy maps policy searches to the parameter space and makes deterministic policies available.

Second, the various existence methods do not account for the uncertainties of rewards and policy parameters. Overlooking uncertainty can lead to unstable behavior of policies and overestimation of rewards during exploration. This problem, called the exploration and exploitation trade-off, requires careful adjustment of hyperparameters and makes it difficult to apply reinforcement learning in new situations. Bayesian methods consider estimation uncertainties in a principled way. In this direction, K-Learning [7], [8] addresses the uncertainty of rewards with the Bayesian approach, thus demonstrating its effectiveness in solving exploration and exploitation trade-off. However, the formulation of K-Learning requires problems with discrete state and action spaces; direct application to continuous state and action spaces is not straightforward.

In this paper, we propose a parameter-based deterministic policy exploration algorithm based on the Bayesian method that considers uncertainties of rewards and policy parameters in continuous state and action spaces. The contributions of this paper to reinforcement learning are as follows:

- We explicitly define an objective function that accounts for the uncertainty of the reward with the Bayesian method.
- We provide an algorithm for solving this objective function with uncertainty in policy parameters using the approximate Bayesian method.
- We perform numerical experiments to show that the proposed method is more robust than comparing method against estimation errors on finite samples, because our proposal balances reward acquisition and exploration.

II. Parameter-Based Policy Exploration

A. Problem Definition

Let us assume a Markov decision problem in which the joint distribution of T timestamp of states $x_{1:T+1}$, actions $u_{1:T}$, and rewards $r_{1:T}$ is written by

$$p(x_{1:T+1}, u_{1:T}, r_{1:T}) = p(x_1)\prod_{t=1}^{T} p(x_{t+1}|x_t, u_t)p(r_t|x_t, u_t)p(u_t|x_t),$$

where $p(x_1)$ is the initial state probability, $p(x_{t+1}|x_t, u_t)$ is the transition probability, $p(r_t|x_t, u_t)$ is the reward probability, and $p(u_t|x_t)$ is the policy probability.

In parameter-based policy exploration, the policy π is the deterministic function from the current state x_t and policy parameters θ to the current action u_t.

$$u_t = \pi(x_t, \theta).$$

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

The evaluation of the parameter θ is based on the trajectory of one-episode h and the total reward for trajectory r_h is defined by

$$h \equiv (x_1, u_1, \ldots, x_T, u_T, x_{T+1}), \quad r_h \equiv \sum_{t=1}^{T} r_t.$$

We formalize the problem to find the optimal distribution of policy parameters $p(\theta)$ that maximizes the expectation of the total reward $J(p)$.

$$J(p) = \int r(\theta) p(\theta) d\theta,$$

where

$$r(\theta) = \int r_h p(r_h|\theta) dr_h.$$

Furthermore, the final evaluation of the distribution of policy parameters $p(\theta)$ is made by the action u_t^*, which is the ensemble average of the output of the policy by $p(\theta)$.

$$u_t^* = \int \pi(x_t, \theta) p(\theta) d\theta.$$

B. Policy parameter uncertainty

We consider the problem of maximizing the integral $I(\rho)$ with respect to ρ.

$$I(\rho) = \int w(\theta) p(\theta|\rho) d\theta, \quad w(\theta) \geq 0$$

where $p(\theta|\rho)$ is the probability distribution of the policy parameters θ with the hyperparameter ρ and $w(\theta)$ is the weight function of θ. We show that the maximization of $I(\rho)$ indicates that it can be considered a weighted likelihood problem. This display allows us to apply Bayesian methods to handle uncertainties in policy parameters.

First, we evaluate the ratio of $I(\rho')$ to $I(\rho)$, where ρ is the current parameter and ρ' is the optimization parameter. Using Jensen's inequality, we obtain

$$\begin{aligned} \log \frac{I(\rho')}{I(\rho)} &= \log \int \frac{w(\theta) p(\theta|\rho)}{I(p)} \frac{p(\theta|\rho')}{p(\theta|\rho)} d\theta \\ &\geq \int \frac{w(\theta) p(\theta|\rho)}{I(\rho)} \log \frac{p(\theta|\rho')}{p(\theta|\rho)} d\theta. \end{aligned}$$

If we define $Q(\rho', \rho)$ as

$$Q(\rho', \rho) = \int w(\theta) p(\theta|\rho) \log p(\theta|\rho') d\theta, \tag{1}$$

then we obtain the following inequality:

$$\log I(\rho') \geq \log I(p) + \frac{Q(\rho', \rho) - Q(\rho, \rho)}{I(\rho)}.$$

This inequality shows that maximizing $Q(\rho', \rho)$ with respect to ρ' maximizes the lower bound of $I(\rho')$ [9]. In addition, we approximate (1) with J pairs of θ_j which are samples from $p(\theta|\rho)$ and the corresponding weights $w(\theta_j)$.

$$D_p \equiv \{(\theta_1, w(\theta_1)), \ldots, (\theta_J, w(\theta_J))\}, \quad \theta_j \sim p(\theta|\rho),$$

$$Q(\rho', \rho) \approx \frac{1}{J} \sum_{j=1}^{J} w(\theta_j) \log p(\theta_j|\rho').$$

The first-order condition for maximizing Q with respect to ρ' is

$$\sum_{j=1}^{J} w(\theta_j) \nabla_{\rho'} \log p(\theta_j|\rho') = 0. \tag{2}$$

This equation shows that Q is related to the weighted log likelihood

$$\prod_{j=1}^{J} p(\theta_j|\rho')^{w(\theta_j)}, \tag{3}$$

and [10], [11] show that the maximizer of the weighted likelihood with respect to parameter ρ' converges to the solution of (2) as $J \to \infty$.

Using this relation of the weighted likelihood representation, we obtain a Bayesian estimate of the distribution of policy parameters $p^*(\theta)$ as the predicted distribution given by

$$p^*(\theta) = \int p(\theta|\rho) p(\rho|D_p) d\rho,$$

$$p(\rho|D_p) = \frac{\prod_{j=1}^{J} p(\theta_j|\rho)^{w(\theta_j)} p(\rho)}{\int \prod_{j=1}^{J} p(\theta_j|\rho)^{w(\theta_j)} p(\rho) d\rho}, \ \theta_j \sim p_{old}^*(\theta),$$

where $p(\rho)$ is a prior distribution for the hyperparameter ρ.

C. Mean Reward uncertainty

With parameter-based policy exploration, we cannot know the true distribution of episode rewards given a parameter $p(r_h|\theta)$. Therefore, we need to somehow estimate the mean $r(\theta) = \int r_h p(r_h|\theta) dr_h$. Many studies use a simple sample average of episodes $\tilde{r}(\theta) = \frac{1}{K} \sum_{k=1}^{K} r_{h_k}$, with a fixed θ. However, this estimator does not account for the uncertainty of the estimator. O'Donoghue [7] and O'Donoghue et al. [8] proposed using K-learning for this problem. The authors used the Bayesian method to estimate the cumulant of the mean instead of the simple average. The cumulant gives a more exploratory "optimistic" behavior and has excellent properties such as additivity. Although their algorithm shows an approximation of the cumulant of discrete state and action spaces, extending them to continuous state and action spaces is not straightforward.

In the parameter-based exploration setting, we use Bayesian regression to predict the reward r using the policy parameter θ

$$p(r|\theta, w), \tag{4}$$

where w is the regression parameter and the prior distribution of the regression parameter is $p(w)$. Under given K samples $D \equiv \{(r_1, \theta_1), \ldots, (r_K, \theta_K)\}$, the posterior distribution of the parameter is written by

$$p(w|D) = \frac{\prod_{k=1}^{K} p(r_k|\theta_k, w) p(w)}{\int \prod_{k=1}^{K} p(r_k|\theta_k, w) p(w) dw}.$$

Next, the cumulant generating function of the mean reward parameterized by γ is defined as

$$K(\theta, \gamma) \equiv \log E_{p(w|D)}[\exp(\gamma \int r p(r|\theta, w) dr)],$$

and $K(\theta,1)$ gives the cumulant [12]. The second-order approximation of $K(\theta,1)$ is given by

$$K(\theta,1) \approx \frac{d}{d\gamma}\bigg|_{\gamma=0} K(\theta,\gamma) + \frac{1}{2}\,\frac{d^2}{d\gamma^2}\bigg|_{\gamma=0} K(\theta,\gamma).$$

The first and second derivatives give the mean and variance of the mean, respectively described by

$$\frac{d}{d\gamma}\bigg|_{\gamma=0} K(\theta,\gamma) = \int p(w|D) \int rp(r|\theta,w)drdw \equiv \tilde{r}(\theta),$$

$$\frac{d^2}{d\gamma^2}\bigg|_{\gamma=0} K(\theta,\gamma) = \int p(w|D) \left(\int rp(r|\theta,w)dr \right)^2 dw - \left(\int p(w|D) \int rp(r|\theta,w)drdw \right)^2 \equiv \tilde{v}(\theta).$$

Therefore, the approximation of the cumulant is given by the sum of the mean $\tilde{r}(\theta)$ and the variance of the mean $\tilde{v}(\theta)$ that reflects the uncertainty of the estimator. It is noted that the variance $\tilde{v}(\theta)$ is not the variance of the predicted distribution of r but rather an estimator of the mean strongly associated with Thompson sampling [13].

D. Gaussian Approximation

In this paper, the policy $\pi(x,\theta)$ is represented by a neural network. Furthermore, we restrict both the distribution of policy parameters $p(\theta|\rho)$ and the distribution of the reward $p(r|\theta,w)$ to the Gaussian distribution. Under this restriction, our method can connect to other methods, such as the EM-based policy hyperparameter exploration [9] and the Covariance Matrix Adaptation Evolution Strategy [14]. These algorithms differ in their weight function, covariance structure, and estimation method (with or without Bayes).

The Bayesian estimation of the Gaussian distribution with the conjugate prior follows [15]. The likelihood is assumed by Gaussian distribution.

$$p(x|\mu,\sigma^2) = \mathcal{N}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2 \right).$$

In addition, the prior distribution is conjugate Normal-Gamma

$$\begin{aligned} &\mathcal{NG}(\mu,\lambda,\mu_0,\kappa_0,\alpha_0,\beta_0) = \mathcal{N}(\mu|\mu_0,(\kappa_0\lambda)^{-1})\mathcal{G}(\lambda|\alpha_0,\beta_0) \\ &= \frac{1}{Z_{NG}} \lambda^{\frac{1}{2}} \exp\left(-\frac{\kappa_0\lambda}{2}(\mu-\mu_0)^2 \right) \lambda^{\alpha_0-1} \exp(-\lambda\beta_0), \quad (5) \\ &Z_{NG} = \frac{\Gamma(\alpha_0)}{\beta_0^{\alpha_0}} \left(\frac{2\pi}{\kappa_0} \right)^{\frac{1}{2}}, \end{aligned}$$

and the Student-t Distribution is defined by

$$\begin{aligned} t_\nu(x|\mu,\sigma^2) &= c \left[1 + \frac{1}{\nu} \frac{(x-\mu)}{\sigma}^2 \right]^{-\frac{\nu+1}{2}}, \\ c &= \frac{\Gamma(\nu/2+1/2)}{\Gamma(\nu/2)} \frac{1}{\sqrt{\nu\pi}\sigma}. \end{aligned}$$

In that case, for the distribution of policy parameters $p(\theta|\rho)$ given J pairs of weights and samples $D_p \equiv \{(\theta_1, w(\theta_1)), \ldots, (\theta_J, w(\theta_J))\}$, the predictive distribution is described by

$$p(\theta|D_p) = t_{2\alpha_J}\left(\theta|\mu_J, \frac{\beta_J(\kappa_J+1)}{\alpha_J\kappa_J} \right), \quad (6)$$

where

$$\begin{aligned} \bar{J} &= \sum_{j=1}^{J} w(\theta_j), \quad \bar{\theta} = \frac{\sum_{j=1}^{J} w(\theta_j)\theta_j}{\bar{J}}, \\ \mu_J &= \frac{\kappa_0\mu_0 + \bar{J}\bar{\theta}}{\kappa_0+\bar{J}}, \quad \kappa_J = k_0 + \bar{J}, \\ \alpha_J &= \alpha_0 + \frac{\bar{J}}{2}, \\ \beta_J &= \beta_0 + \frac{1}{2}\sum_{j=1}^{J} w(\theta_j)(\theta_j-\bar{\theta})^2 + \frac{k_0\bar{J}(\bar{\theta}-\mu_0)^2}{2(\kappa_0+\bar{J})}. \end{aligned}$$

For the distribution of mean reward $p_\theta(r|w)$ under fixed θ, the Gaussian assumption is related to a problem that solves the J-arm Gaussian bandit iteratively [16]. Given K samples $D_r \equiv (r_1, \ldots, r_K)$, the mean $\tilde{r}(\theta)$ and the variance of the mean $\tilde{v}(\theta)$ are respectively obtained by

$$\tilde{r}(\theta) = \mu_K, \quad (7)$$

$$\tilde{v}(\theta) = \frac{\beta_K}{(\alpha_K-1)\kappa_K}, \quad (8)$$

where

$$\begin{aligned} \bar{r} &= \frac{\sum_{k=1}^{K} r_k}{K}, \\ \mu_K &= \frac{\kappa_0\mu_0 + K\bar{r}}{\kappa_0+K}, \quad \kappa_K = k_0 + K, \\ \alpha_K &= \alpha_0 + \frac{K}{2}, \\ \beta_K &= \beta_0 + \frac{1}{2}\sum_{k=1}^{K}(r_k-\bar{r})^2 + \frac{k_0K(\bar{r}-\mu_0)^2}{2(\kappa_0+K)}. \end{aligned}$$

Equations (7), (8) are derived from the marginal posterior distribution of the mean.

$$p_\theta(\mu|D_r) = t_{2\alpha_K}\left(\mu|\mu_K, \frac{\beta_K}{\alpha_K\kappa_K} \right).$$

III. Proposed Method

In this section, we describe our proposed algorithm. Let $q(\theta)$ be an any optimization distribution of policy parameters, $p(\theta)$ be the distribution of the current distribution of data generation policy parameters, and $\phi(\theta)$ be the prior distribution of policy parameters. First, we consider minimizing the following KL-divergence under the variational method:

$$KL(q(\theta)||\tilde{p}(\theta)),$$

where

$$\tilde{p}(\theta) = \frac{\exp(E(\theta))}{Z}, \quad Z = \int \exp(E(\theta))d\theta,$$

and

$$E(\theta) \equiv \beta\tilde{r}(\theta) + \frac{\beta^2}{2}\tilde{v}(\theta) + \alpha \log p(\theta) + (1-\alpha)\log\phi(\theta).$$

In this definition, $0 \leq \alpha \leq 1$ determines the strength of the entropy regularization, and $\beta > 0$ is the inverse temperature. In our algorithm, we treat α as a constant specified by the user and β is an optimization variable.

Using the positivity of KL-divergence gives the following key inequality:

$$\begin{aligned}\tilde{J}(q) &= \int \tilde{r}(\theta)q(\theta)d\theta \\ &\leq \int \left(\tilde{r}(\theta) + \frac{\beta}{2}\tilde{v}(\theta)\right) q(\theta)d\theta \\ &\quad + \frac{\alpha}{\beta}\left(c_1 - KL(q(\theta)|p(\theta))\right) \\ &\quad + \frac{1-\alpha}{\beta}(c_2 - KL(q(\theta)|\phi(\theta))) \\ &\leq \frac{1}{\beta}\left(\log Z + \alpha c_1 + (1-\alpha)c_2)\right),\end{aligned} \tag{9}$$

where c_1 and c_2 are constants that satisfy

$$c_1 \geq \mathrm{KL}(q(\theta)||p(\theta)), \quad c_2 \geq \mathrm{KL}(q(\theta)||\phi(\theta)).$$

Therefore, if we define objective function $F(\beta)$ as

$$F(\beta) \equiv \frac{1}{\beta}(\log Z + \alpha c_1 + (1-\alpha)c_2),$$

then minimizing β for $F(\beta)$ gives an upper bound on $\tilde{J}(q)$. The main term of our objective function $\log Z$ with $\alpha = 1$ is the approximation of a cumulant of the mean reward with respect to the posterior distribution of the regression parameters $p(w|D_r)$ and the predictive distribution of the policy parameters $p(\theta|D_p)$:

$$\log Z \approx \log E_{p(\theta|D_p)}[E_{p(w|D_r)}[\exp(\beta \int rp(r|\theta,w)dr)]].$$

It is noted that the second expression of (9) has a clear interpretation and connection to the existing method [17].

The first term relates to K-learning, which uses the cumulant of the mean reward instead of the mean reward to prevent underexploration [7], [8]. The second term, $KL(q(\theta)||p(\theta))$, is KL-divergence from the current distribution of policy parameters to the new distribution of policy parameters, penalizing large movements in the policy. Choosing this term properly prevents policy instability and is used by the Trust Region Policy Optimization [18] and the Relative Entropy Policy Search (REPS) [19]. The third term, $KL(q(\theta)||\phi(\theta))$, is the KL-divergence from the current distribution of policy parameters to the prior distribution of policy parameters. If we set $p(\phi) \equiv 1$ (we use this setting in the numerical experiments), this term is reduced to entropy of $q(\theta)$ and is used by such as Soft Actor Critic (SAC) [20].

Next, we estimate $F(\beta)$ by importance sampling given by

$$F(\beta) \approx \frac{1}{\beta}\log\frac{1}{J}\sum_{j=1}^{J}\exp(H(\theta_j) + \alpha c_1 + (1-\alpha)c_2), \tag{10}$$

where $\theta_j \sim p(\theta)$ and

$$H(\theta) \equiv \beta\tilde{r}(\theta) + \frac{\beta^2}{2}\tilde{v}(\theta) + (1-\alpha)(\log\phi(\theta) - \log p(\theta)). \tag{11}$$

In that case, the weighting coefficient $\tilde{w}(\theta_j)$ for each parameter is determined by

$$\tilde{w}(\theta_j) = \frac{\exp(H(\theta_j))}{\sum_{j=1}^{J}\exp(H(\theta_j))}. \tag{12}$$

The two constants c_1 and c_2 are determined by the following considerations. We set the constant c_2 to satisfy $c_2 \approx KL(q(\theta)||\phi(\theta))$, which gives the tight bound of (9). We approximate the constant c_2 with a data generation policy $p(\theta)$ instead of the target policy $q(\theta)$.

$$c_2 \approx \int p(\theta)\log\frac{p(\theta)}{\phi(\theta)}d\theta \approx \frac{1}{J}\sum_{j=1}^{J}\log\frac{p(\theta_j)}{\phi(\theta_j)},\ \theta_j \sim p(\theta). \tag{13}$$

For the constant c_1, we set the c_1 to satisfy $KL(q(\theta)||p(\theta)) < \delta_1$. This constraint determines the trade-off between learning speed and robustness, similar to the learning coefficient for gradient-based methods. Furthermore, the robustness of the algorithm depends on the accuracy of the importance sampling (10). The accuracy of importance sampling is assessed by the effective sample size (ESS) [21] which is defined by

$$ESS \equiv \frac{1}{\sum_{j=1}^{J}\tilde{w}(\theta_j)^2}.$$

Therefore, we set the c_1 to the maximum value that satisfies the two constraints

$$c_1 \leq \delta_1, \quad ESS \leq \delta_2. \tag{14}$$

We optimize c_1, β using line search of the c_1. Starting from 0, increase c_1 and minimize β at each c_1 while the condition (14) is satisfied.

The description of our proposed algorithm is Alg. 1. The computational cost of the proposed method is almost the same as other parameter-based explorations.

IV. Numerical experiment

We performed numerical experiments to demonstrate the effectiveness of the proposed algorithm. We used two perspectives to evaluate the experiment. One was to confirm the robustness of the proposed algorithm, the other was to confirm that the proposed algorithm solves the trade-off between reward acquisition and exploration which is measured by the entropy of the distribution of policy parameters. For this purpose, we compared the proposed algorithm with the EM-based policy hyperparameter exploration using the REPS weighting scheme (EPHE-RW) [9]. EPHE-RW matches the non-Bayesian version of our algorithm, where the distribution of policy parameters is updated with a maximum likelihood of (3) and no variance of the mean $\tilde{v}(\theta) \equiv 0$. Thus, comparing

Algorithm 1 Algorithm for proposed method

Input: initial distribution of policy parameters $p(\theta_0)$, entropy coefficient α, KL-constraint δ_1, ESS-constraint δ_2,
Output: the distribution of policy parameters $p(\theta)$

for $i = 1$ to iteration I **do**
  for $j = 1$ to population J **do**
    sample $\theta_j \propto p(\theta)$
    $E[j] \leftarrow \log p(\theta_j)$
    for $k = 1$ to episode length K **do**
      execute one episode by policy θ_j and $R[k] \leftarrow r_h$
    end for
    compute $\tilde{r}(\theta_j)$, $\tilde{v}(\theta_j)$ by $R[.]$ (7), (8)
  end for
  calculate c_2 by $E[.]$ (13)
  optimize β, c_1 by (10) under constraints (14)
  calculate $\tilde{w}(\theta_.)$ by (12)
  update $p(\theta)$ by $\theta_.$ and $w(\theta_.) = J \times \tilde{w}(\theta_.)$ (6)
end for
return $p(\theta)$

TABLE I
REWARD OF "PENDULUM-V1" EXPERIMENT

	Reward (Mean)	Reward (Std)
proposed	-159.330	14.269
EPHE-RW	-306.554	193.666

the two algorithms revealed the effectiveness of Bayesian uncertainty handling in parameter-based policy exploration.

We selected the task "Pendulum-v1" in OpenAI Gym [22], which is a non-linear control task with continuous state and action spaces. The policy network was a simple three layered neural network that had two hidden layers with 40 units and we used the "tanh" activation function. We updated $I = 1000$ steps of the distribution of policy parameters with population size $J = 26$ and episode length of $K = 26$. The entropy coefficient α was set to $\alpha = 1 - \frac{1}{|\theta|}$, where $|\theta|$ is the number of policy parameters. The initial distribution of policy parameters $p_0(\theta)$ was the standard normal distribution $\mathcal{N}(\theta|0, 1.0)$, the KL-divergence parameter was $\delta_2 = 0.5$ and the ESS restriction was $\delta_1 = 0.5 \times J$. The parameters of Normal-Gamma prior (5), were set to $\mu_0 = 0$, $\alpha_0 = -0.5$, $\beta_0 = 0.5e - 7$, and $\kappa_0 = 1.0e - 6$. This prior was almost non-informative, and α_0 was set using reference prior [23], [24]. The experiment was performed with 10 different initializations. Over the last 100 steps, we evaluated the reward and entropy of the distribution of policy parameters divided by the number of policy parameters.

The rewards obtained in the experiment are shown in Table. I. This result shows that the proposed method has higher mean reward and lower standard deviation than EPHE-RW.

Fig. 1 shows the profile of the learning steps for a detailed comparison. This figure show that, in the early stage, the speed of getting rewards was faster with EPHE-RW than with the proposed method. However, EPHE-RW often continued to use

TABLE II
ENTROPY OF THE DISTRIBUTION OF POLICY PARAMETERS "PENDULUM-V1" EXPERIMENT

	Entropy (Mean)	Entropy (Std)
proposed	-0.620	0.333
EPHE-RW	-5.344	0.006

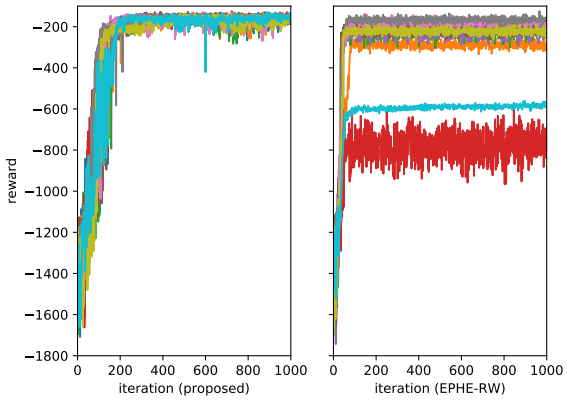

Fig. 1. Reward profiles of 10 trials (left proposed, right EPHE-RW)

the sub-optimal solution and could not escape from it, whereas the proposed method progressively found an almost optimal solution consistently.

Table. II and Fig. 2 show clear differences in the entropy of the distribution of policy parameters between the proposed method and EPHE-RW. Fig. 2 shows that the entropy of the proposed method decreased first and then gradually increased in about 400 steps, at which point the algorithm found a near optimal solution. This behavior indicates that the algorithm balances reward acquisition with the entropy of the distribution of policy parameters and prevents underexploration. However, the entropy of EPHE-RW decreased monotonically to a minimum, indicating no trade-off between reward acquisition and the entropy of the distribution of policy parameters.

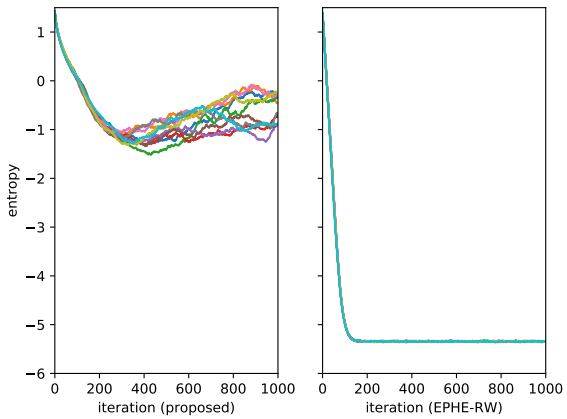

Fig. 2. Entropy profiles of 10 trials (left proposed, right EPHE-RW)

V. Discussion

A. Sample efficiency

There are two ways our algorithm can use samples more efficiently.

First, we explicitly solve the Bayesian reward regression (4) using neural networks. Because this regression problem is stationary over iterations of the algorithm, it can use samples efficiently to estimate the mean $\tilde{r}(\theta)$ and $\tilde{v}(\theta)$ accurately. Furthermore, in many cases, when the sample tends to infinity, $\tilde{v}(\theta)$ converges to 0, guaranteeing the convergence of the algorithm. However, the extra computational cost of Bayesian regression is high. Therefore, we need to design a trade-off between computational cost and sample efficiency for each problem.

The second way of using samples more efficiently is an estimate of the distribution of policy parameters $p(\theta)$. Although, the distribution of policy parameters $p(\theta)$ is non-stationary over iterations of the algorithm, importance sampling can be used [25]. Let $p(\theta)$ be the current policy and $\hat{p}(\theta)$ be the old policy. Using the importance weight $\frac{p(\theta)}{\hat{p}(\theta)}$, we modify $H(\theta)$ (11) as

$$\hat{H}(\theta) \equiv H(\theta) + \log p(\theta) - \log \hat{p}(\theta),$$

where we evaluate $\hat{H}(\theta)$ with samples from the old policy $\theta_j \sim \hat{p}(\theta)$.

VI. Conclusion

For the lack of robustness and underexploration problem in reinforcement learning, we developed a Bayesian framework that explicitly includes uncertainties in rewards and policy parameters. In addition, in continuous state and action spaces, we approximated this formulation and constructed an algorithm whose computational cost was equivalent to the ordinal parameter-based exploration method. The results of numerical experiments showed that the proposed algorithm is more robust than comparing method against estimation errors on finite samples, because our proposal solves the trade-off between reward acquisition and entropy of the distribution of the policy parameters that facilitates exploration. However, to solve a wider problem more accurately, we need to develop an algorithm that does not use the simple Gaussian approximation. For example, using neural networks for reward regression and Gaussian mixture for distribution of policy parameters can be done in future work.

References

[1] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. A. Riedmiller, A. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015. [Online]. Available: https://doi.org/10.1038/nature14236

[2] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. P. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016. [Online]. Available: https://doi.org/10.1038/nature16961

[3] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, pp. 229–256, 1992. [Online]. Available: https://doi.org/10.1007/BF00992696

[4] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," in *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2016. [Online]. Available: http://arxiv.org/abs/1509.02971

[5] S. Fujimoto, H. van Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, ser. Proceedings of Machine Learning Research, J. G. Dy and A. Krause, Eds., vol. 80. PMLR, 2018, pp. 1582–1591. [Online]. Available: http://proceedings.mlr.press/v80/fujimoto18a.html

[6] F. Sehnke, C. Osendorfer, T. Rückstieß, A. Graves, J. Peters, and J. Schmidhuber, "Policy gradients with parameter-based exploration for control," in *Artificial Neural Networks - ICANN 2008 , 18th International Conference, Prague, Czech Republic, September 3-6, 2008, Proceedings, Part I*, ser. Lecture Notes in Computer Science, V. Kurková, R. Neruda, and J. Koutník, Eds., vol. 5163. Springer, 2008, pp. 387–396. [Online]. Available: https://doi.org/10.1007/978-3-540-87536-9_40

[7] B. O'Donoghue, "Variational Bayesian reinforcement learning with regret bounds," *arXiv preprint arXiv:1807.09647*, 2018.

[8] B. O'Donoghue, I. Osband, and C. Ionescu, "Making sense of reinforcement learning and probabilistic inference," in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=S1xitgHtvS

[9] J. Wang, E. Uchibe, and K. Doya, "Adaptive baseline enhances em-based policy search: Validation in a view-based positioning task of a smartphone balancer," *Front. Neurorobot.*, vol. 2017, 2017. [Online]. Available: https://doi.org/10.3389/fnbot.2017.00001

[10] T. Ueno, K. Hayashi, T. Washio, and Y. Kawahara, "Weighted likelihood policy search with model selection," in *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, P. L. Bartlett, F. C. N. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., 2012, pp. 2366–2374. [Online]. Available: http://papers.nips.cc/paper/4815-weighted-likelihood-policy-search-with-model-selection

[11] M. Imaizumi and R. Fujimaki, "Factorized asymptotic Bayesian policy search for pomdps," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, C. Sierra, Ed. ijcai.org, 2017, pp. 4346–4352. [Online]. Available: https://doi.org/10.24963/ijcai.2017/607

[12] S. Watanabe, "Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory," *J. Mach. Learn. Res.*, vol. 11, pp. 3571–3594, 2010. [Online]. Available: http://portal.acm.org/citation.cfm?id=1953045

[13] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3/4, pp. 285–294, 1933.

[14] N. Hansen and A. Ostermeier, "Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation," in *Proceedings of 1996 IEEE International Conference on Evolutionary Computation, Nayoya University, Japan, May 20-22, 1996*, T. Fukuda and T. Furuhashi, Eds. IEEE, 1996, pp. 312–317. [Online]. Available: https://doi.org/10.1109/ICEC.1996.542381

[15] K. P. Murphy, "Conjugate Bayesian analysis of the Gaussian distribution," University of British Columbia, Tech. Rep., 2007.

[16] J. Honda and A. Takemura, "Optimality of Thompson sampling for Gaussian bandits depends on priors," in *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, AISTATS 2014, Reykjavik, Iceland, April 22-25, 2014*, ser. JMLR Workshop and Conference Proceedings, vol. 33. JMLR.org, 2014, pp. 375–383. [Online]. Available: http://proceedings.mlr.press/v33/honda14.html

[17] T. Kozuno, E. Uchibe, and K. Doya, "Theoretical analysis of efficiency and robustness of softmax and gap-increasing operators in reinforcement learning," in *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, ser. Proceedings of Machine Learning Research, K. Chaudhuri

and M. Sugiyama, Eds., vol. 89. PMLR, 2019, pp. 2995–3003. [Online]. Available: http://proceedings.mlr.press/v89/kozuno19a.html

[18] J. Schulman, S. Levine, P. Abbeel, M. I. Jordan, and P. Moritz, "Trust region policy optimization," in *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, ser. JMLR Workshop and Conference Proceedings, F. R. Bach and D. M. Blei, Eds., vol. 37. JMLR.org, 2015, pp. 1889–1897. [Online]. Available: http://proceedings.mlr.press/v37/schulman15.html

[19] J. Peters, K. Mülling, and Y. Altun, "Relative entropy policy search," in *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*, M. Fox and D. Poole, Eds. AAAI Press, 2010. [Online]. Available: http://www.aaai.org/ocs/index.php/AAAI/AAAI10/paper/view/1851

[20] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, ser. Proceedings of Machine Learning Research, J. G. Dy and A. Krause, Eds., vol. 80. PMLR, 2018, pp. 1856–1865. [Online]. Available: http://proceedings.mlr.press/v80/haarnoja18b.html

[21] A. Kong, "A note on importance sampling using standardized weights," University of Chicago, Tech. Rep., 1992.

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

[23] J. M. Bernardo and A. F. Smith, *Bayesian theory*. John Wiley & Sons, 2009, vol. 405.

[24] S. Watanabe, "Bayesian cross validation and waic for predictive prior design in regular asymptotic theory," *arXiv preprint arXiv:1503.07970*, 2015.

[25] T. Zhao, H. Hachiya, V. Tangkaratt, J. Morimoto, and M. Sugiyama, "Efficient sample reuse in policy gradients with parameter-based exploration," *Neural Computation*, vol. 25, no. 6, pp. 1512–1547, 2013. [Online]. Available: https://doi.org/10.1162/NECO_a_00452