Sparse Actor-Critic: Sparse Tsallis Entropy Regularized Reinforcement Learning in a Continuous Action Space

Jaegoo Choy, Kyungjae Lee, and Songhwai Oh

Abstract— **In case of deep reinforcement learning (RL) algorithms, to achieve high performance in complex continuous control tasks, it is necessary to exploit the goal and at the same time explore the environment. In this paper, we introduce a novel off-policy actor-critic reinforcement learning algorithm with a sparse Tsallis entropy regularizer. The sparse Tsallis entropy regularizer has the effect of maximizing the expected returns while maximizing the sparse Tsallis entropy for its policy function. Maximizing the sparse Tsallis entropy makes the actor to explore the large action and state space efficiently, thus it helps us to find the optimal action at each state. We derive the iteration update rules and modify a policy iteration rule for an off-policy method. In experiments, we demonstrate the effectiveness of the proposed method in continuous reinforcement learning problems in terms of the convergence speed. The proposed method outperforms former on-policy and off-policy RL algorithms in terms of the convergence speed and performance.**

I. Introduction

Recent advances of reinforcement learning (RL) have enabled to learn a complex behavior in many robotic tasks, such as rich contact manipulation [1], collision avoidance [2] and visual navigation [3]. One challenge of such robotic problems is difficulty to model a stochastic environment. Since the environment is veiled, a robot should explore to find what actions lead to a high reward while improving its policy based on the gathered clues. Scheduling such a trade-off between exploration and exploitation is the most crucial factor to be considered when designing an RL algorithm.

Although several applications of RL in robotics [4] successfully handled the exploration-exploitation trade-off, searching an optimal policy is even more challenging problem in continuous state and action spaces where many robotic problems are defined. In order to handle a continuous action space, actor-critic methods have been widely investigated where both policy and value functions are estimated by a function approximation method, such as a neural network. The policy function is directly optimized using a stochastic gradient of average return with respect to the parameter of policy function. The value function provides an estimation of a future return to reduce the variance of the stochastic gradient.

While the actor-critic method is powerful due to its scalability, it still suffers from the exploration and sample inefficiency. Recently, some of actor critic methods have focused on improving exploration efficiency using an entropy regularization [5], [6], [7]. In [8], [6], it has been revealed that adding an additional objective of maximizing the entropy of a policy to the original RL problem encourages the exploration and helps to find a better policy than the one without the entropy. In [9], Lee et al. further extended the Shannon entropy to a sparse Tsallis entropy which has less regularization effect and prefers a sparse distribution. [9] also proved that the Shannon entropy is not appropriate when the action space is large, since it assigns a non-zero probability mass to the entire actions. In contrast, since the Tsallis entropy induces a sparse distribution, it has an effect of pruning harmful actions. Such phenomenon also has been investigated in [10]. However, as the analyses and practices shown in [9], [10] only focus on discrete action spaces, they are not directly applicable to a continuous action space. Thus, the discretization of an action space is essential, yet it often leads to the curse of dimensionality.

In this paper, we propose a novel actor-critic method based on Tsallis entropy regulariztion to encourage the exploration in a continuous action space. To apply sparse Tsallis entropy regularization to continuous control problems, we derive a novel policy gradient method with the sparse Tsallis entropy which can exploit the benefit of the sparse Tsallis entropy to optimize the policy function. Furthermore, thanks to the fact that the sparse Tsallis entropy regularizer is less strict than the Shannon-Gibbs entropy regularizer, our approach can find the maximum reward faster while still exploring unknown environments. To validate the effectiveness of the proposed actor-critic method with the sparse Tsallis entropy, we conduct simulations on two multi-goal problems and continuous control problems. The proposed method outperforms prior actor-critic methods in terms of the number of steps required to achieve the maximum performance.

II. Related Work

Deep reinforcement learning has been widely investigated in robotics and machine learning. The actor-critic method is one of the crucial techniques in robotics since it can handle continuous action spaces. To handle continuous action spaces, most existing actor-critic methods are based on the policy gradient method [11], [12]. In [11], the authors have proposed a basic policy gradient method, which is called REINFORCE where the gradient of the expected return with respect to the parameters of a policy function. It is proposed and proved in [12] in the first time. In [13], trust region policy optimization (TRPO) has been introduced in which the trust

J. Choy, K. Lee and S. Oh are with the Department of Electrical and Computer Engineering and ASRI, Seoul National University, Seoul, Korea (e-mail: kyungjae.lee@rllab.snu.ac.kr, songhwai@snu.ac.kr).

This work was supported by the ICT R&D program of MSIT/IITP (No. 2019-0-01309, Development of AI Technology for Guidance of a Mobile Robot to its Goal with Uncertain Maps in Indoor/Outdoor Environments).

region method is applied to the policy gradient for numerical stability and avoiding the local optima by restricting the region where parameters can be updated. In [14], proximal policy optimization (PPO) has been introduced in which it updates policy parameters with clipped surrogate TRPO objective or adaptive Kullback-Leibler divergence penalized objective, instead of using the trust region method. Furthermore, authors of [15] proposed generalized advantages estimation (GAE) which is to reduce the variance of the estimation of the policy gradient using the state value network. While TRPO and PPO with GAE obtained numerical stability and variance reduction, these methods still have weakness in that they are on-policy learning methods which require much more samples than off-policy learning methods.

In [16], deep deterministic policy gradient (DDPG) has been proposed, which is an off-policy actor-critic method based on deterministic policy gradient. Since it saves sampled episodes into the replay memory, DDPG has a benefit of sample efficiency. However, the deterministic policy often suffers from falling into the local optima since it is unimodal.

To store the informative states or actions into the replay memory, an agent has to explore an environment with multiple directions. In [6], [7], the authors proposed a soft Q-learning (SQL) which utilizes entropy regularization to encourage exploration. In particular, in order to search multiple directions, the policy function should be modeled as a sampling network which can represent the multi-modal distribution. Furthermore, soft actor-critic (SAC) method has been proposed by extending SQL to an actor-critic method. In [17], Nachum et al. have proposed the path consistency learning (PCL) which is an off-policy actor-critic method based on the patch consistency equation. The authors proved that if a state value, state-action value, and policy functions satisfy the path consistency equation, then, they are optimal in the entropy regularized RL problem. The PCL also employs the benefit of entropy regularization in the exploration phase.

Existing actor-critic methods have addressed the problem of numerical stability and variance reduction of policy gradient and have improved the exploration efficiency using an entropy regularization and a multi-modal policy distribution. In this paper, we focus on utilizing a different type of an entropy regularization, which is called the sparse Tsallis entropy.

III. Backgrounds

In this section, we introduce Markov decision processes (MDPs) and its extensions using Shannon-Gibbs entropy maximization and sparse Tsallis entropy maximization, which are also called soft MDP and sparse MDP, respectively. And we review soft actor-critic (SAC) [8] which utilizes the Shannon-Gibbs entropy maximization.

A. Markov Decision Processes

An MDP is a well-known mathematical framework for a sequential decision making problem. A general MDP is defined as a tuple $fS; A; \ ; d; P; \ ; \mathbf{r}g$, where S is a state space, A is an action space, is a set of stochastic policies, which are a conditional distribution over the action space given a state, $d(s_0)$ is an initial state distribution, $P(s^0 js; a)$ is a transition probability from $s \ 2 \ S$ to $s^0 \ 2 \ S$ by taking $a \ 2 \ A$, $\ 2 \ (0; 1)$ is a discount factor, and $\mathbf{r}$ is a reward function from a state-action pair to a real value.

The main goal of an MDP is to find an optimal policy $\ 2$ which maximizes the expected sum of discounted rewards, i.e., $\mathsf{E}[\sum_{t=0}^{1} \ ^t \mathbf{r}(s_t; a_t) j \ ; d]$, which will be denoted as $\mathsf{E} \ [\mathbf{r}(s; a)]$ for the sake of simplicity of notation. Similarly, note that, for any function $f(s; a)$, $\mathsf{E}[\sum_{t=0}^{1} \ ^t f(s_t; a_t) j \ ; d]$ will be denoted as $\mathsf{E} \ [f(s; a)]$.

B. Entropy Regularized Markov Decision Processes

Since entropy regularization has an effect of penalizing deterministic behaviors, it induces stochastic policies and encourages to explore a wide area of state action spaces during the training phase. An entropy regularized MDP is formulated as follows:

$$\begin{aligned} &\text{maximize} \quad \mathsf{E} \ [\mathbf{r}(s_t; a_t)] + \ C(\ ); \\ &\text{subject to} \quad 8s \sum_{a^0} (a^0 js) = 1; \ 8s; a \ \ (a^0 js) \quad 0; \end{aligned} \tag{1}$$

where $C(\ )$, $\mathsf{E} \ [C(\ (a_t js_t))]$ indicates a -discounted entropy regularization of a policy, $C(x)$ indicates an entropy function, and is a regularization coefficient.

There are two widely used entropy regularization methods: soft MDP and sparse MDP. By setting $C(x)$ to $\log(x)$, we can obtain soft MDP, which utilizes Shannon-Gibbs entropy as a regularization, and it is highly investigated in [7], [18]. In [18], Bloem et al. derived the Bellman optimality equation of the soft MDP and showed that the optimal policy distribution is a softmax distribution of a state action value $Q^{soft}(s; a)$ as follows:

$$(ajs) = \frac{\exp \ \frac{Q^{soft}(s;a)}{}}{\sum_{a^0} \exp \ \frac{Q^{soft}(s;a^0)}{}}: \tag{2}$$

For the sparse Tsallis entropy, we can obtain an MDP with sparse Tsallis entropy maximization by setting $C(x)$ to $(1 \ x)=2$, which is also known as sparse MDP. In case of the sparse Tsallis entropy, the Bellman optimality equation can be derived from the Karush—Kuhn—Tucker conditions. In [9], Lee et al. showed that the sparse Tsallis entropy makes the optimal policy distribution to be multi-modal but a sparse distribution as follows:

$$(ajs) = \max \ \frac{Q^{sp}(s; a)}{} \quad \frac{Q^{sp}(s; \ )}{} \ ; 0 \ ; \tag{3}$$

where $\frac{Q^{sp}(s; \ )}{} = \frac{\sum_{a \ 2 \ S(s)} \frac{Q^{sp}(s;a)}{} \ 1}{K_s}$, $S(s)$ is a set of actions satisfying $1 + i \frac{Q^{sp}(s;a_{(i)})}{} > \sum_{j=0}^{i} \frac{Q^{sp}(s;a_{(j)})}{}$ with $a_{(i)}$ indicating the action with the ith largest action value $Q^{sp}(s; a_{(i)})$, and K_s is the cardinality of $S(s)$. V^{sp} and Q^{sp}

indicate the state and state-action value of the sparse MDP defined as,

$$V^{sp}(s) = \mathrm{E}\left[r(s_t; a_t) + \frac{}{2}(1 \quad (a_t|s_t))|s_0 = s \right];$$
$$Q^{sp}(s; a) = \mathrm{E}\left[r(s_t; a_t) + \frac{}{2}(1 \quad (a_t|s_t))|s_0 = s; a_0 = a \right]:$$

Although an entropy regularizer encourages exploration, there is a side effect that the optimal performance must be smaller than the optimal performance of the original MDP. The sparse MDP can make the probability of some actions zero unlike the soft MDP, so the policy function of the sparse MDP can be more goal-oriented than the soft MDP. However, the Tsallis entropy regularization method is only applicable to a discrete action space so far because of difficulty in finding the threshold in (3) in a continuous action space.

C. Soft Actor-Critic

The soft actor-critic (SAC) [8] is an off-policy actor-critic algorithm using the Shannon-Gibbs (SG) entropy. SAC is theoretically based on soft policy iteration method, which consists of two steps: soft policy evaluation and soft policy improvement. Harrnoja et al. [8] showed that soft policy iteration converges to the soft Bellman optimality equations (2). SAC is an extension of the soft policy iteration to a high-dimensional control problem by using three parameterized functions: a state value function $V(s_t)$, a state-action value function $Q(s_t; a_t)$, and a policy function $(a_t|s_t)$.

IV. Proposed Method

In this section, we propose a novel actor-critic algorithm with sparse Tsallis entropy regularization which can be derived from sparse MDP. First, we develop a policy gradient theorem with sparse Tsallis entropy regularization and then extend it to an off-policy actor-critic method.

A. Policy Gradient Theorem

In continuous action spaces, the objective of sparse MDP can be specified as follows:

$$J( ) = \mathrm{E}\left[r(s; a) + \frac{}{2}(1 \quad (a|s)) \right] = \int_s d(s) V(s) \mathrm{d}s;$$

where $d(s)$ is the initial state distribution. The optimal policy of sparse MDP described in (3) is also satisfied in continuous action spaces. In continuous action spaces, the threshold can be inferred through the following constraint:

$$\int_a (a|s) \mathrm{d}a = \int_a \max\left(\frac{Q(s; a)}{} \quad (s); 0 \right) \mathrm{d}a = 1:$$

However, since it is generally intractable to obtain (s) by solving the constraint, we directly compute the policy gradient by differentiating the objective of sparse MDP J with respect to the policy parameters as follows:

$$\frac{@J( )}{@} = \int_s d(s) \frac{@}{@} V(s) \mathrm{d}s$$
$$= \int_s d(s) \int_a \frac{@}{@} Q(s; a) + \frac{}{2}(1 \quad (a|s)) \quad (a|s) \mathrm{d}a \mathrm{d}s$$
$$= \int_s d(s) \int_a \frac{@Q(s; a)}{@} \quad \frac{}{2} \frac{@ \quad (a|s)}{@} \quad (a|s) +$$
$$Q(s; a) + \frac{}{2}(1 \quad (a|s)) \quad \frac{@ \quad (a|s)}{@} \mathrm{d}a \mathrm{d}s:$$

Here, the two terms are derived by the product rule in calculus and it can be rearranged as

$$= \int_s d(s) \int_a Q(s; a) + \frac{}{2}(1 \quad 2 \quad (a|s)) \quad \frac{@ \quad (a|s)}{@} \mathrm{d}a \mathrm{d}s +$$
$$\int_s d(s) \int_a \frac{@}{@} \left[r(s; a) + \int_{s'} P(s'|s; a) V(s') \mathrm{d}s' \right] \quad (a|s) \mathrm{d}a \mathrm{d}s$$
$$= \int_s d(s) \int_a Q(s; a) + \frac{}{2}(1 \quad 2 \quad (a|s)) \quad \frac{@ \quad (a|s)}{@} \mathrm{d}a \mathrm{d}s +$$
$$\int_s d(s) \int_a \int_{s'} P(s'|s; a) \frac{@V(s')}{@} \mathrm{d}s' \quad (a|s) \mathrm{d}a \mathrm{d}s$$
$$= \int_s (s) \int_a Q(s; a) + \frac{}{2}(1 \quad 2 \quad (a|s)) \quad \frac{@ \quad (a|s)}{@} \mathrm{d}a \mathrm{d}s$$
$$= \mathrm{E}_{s \sim \quad ; a \sim} \left[\frac{@ \log \quad (a|s)}{@} \quad Q(s; a) + \frac{}{2}(1 \quad 2 \quad (a|s)) \right];$$

where (s) is a stationary distribution of an initial distribution $d(s)$ and a policy . Although the gradient can be computed by an on-policy algorithm, we change it to an off-policy algorithm by replacing the stationary distribution (s) with the replay memory. In [19], [20], they showed that the gradient holds the same value approximately in spite of the replacement even if the stationary distribution (s) is replaced with the replay memory M. Therefore, we can estimate the gradient as

$$\mathrm{E}_{s \sim M; a \sim}\left[\nabla \log \quad (a|s) \quad Q(s; a) + \frac{}{2}(1 \quad 2 \quad (a|s)) \right]:$$

B. Sparse Actor Critic

In this section, we will discuss an off-policy actor-critic algorithm based on the policy gradient theorem and sparse Bellman equation. Our actor-critic algorithm suggests loss functions for three network parameters, a state value function $V(s)$, a state-action value function $Q(s; a)$ and a policy distribution $(a|s)$.

The loss function of the state value function $V(s)$ is designed to minimize the squared residual error induced from the sparse Bellman equation as follows:

$$L_V( ) = \mathrm{E}_{s_t \quad M}\left[\frac{1}{2}(V(s_t) \quad \mathrm{E}_{a_t} \quad [Q(s_t; a_t) + \frac{}{2}(1 \quad (a_t|s_t))])^2 \right];$$

Then, the gradient of the error can be estimated as

$$\nabla L_V( ) = \nabla V(s_t) \quad V(s_t) \quad Q(s_t; a_t) \quad \frac{}{2}(1 \quad (a_t|s_t)) \quad ; \tag{4}$$

where the expectation with respect to the action is approximated as an one-sample Monte-Carlo estimation.

The parameters of the state-action value function $Q(s;a)$ can be updated by minimizing a loss

$$L_Q(\ ) = \mathsf{E}_{(s_t;a_t)\ M}\ \frac{1}{2}(Q\ (s_t;a_t)\quad (r(s_t;a_t)+ \mathsf{E}_{s_{t+1}\ p_s}[V\ (s_{t+1})]))^2\ ;$$

which is derived from the Bellman equation $Q_{sp}(s_t;a_t) = r(s_t;a_t) + \mathsf{E}_{s_{t+1}\ p_s}[V_{sp}(s_{t+1})]$. Then, its gradient is

$$\mathsf{O}\ L_Q(\ ) = \mathsf{O}\ Q\ (s_t;a_t)(Q\ (s_t;a_t)\quad r(s_t;a_t) \quad V\ (s_{t+1}))\text{:}$$

When we update the state-action value function, we use a target value function V , moving average of the source value function V , for stability of the training as utilized in [8], [21].

The gradients for the policy parameters are

$$\mathsf{O}\ L\ (\ ) = \mathsf{O}\ \log(\quad (a_t|s_t))(Q\ (s_t;a_t)+ \frac{}{2}(1\quad 2\quad (a_t|s_t)))$$

as stated earlier. Note that the direction of gradients of is opposite to and , because we aim to maximize the objective $L\ (\ )$ and minimize $L_V(\ )$ and $L_Q(\ )$. The overall algorithm of our proposed method, Sparse Actor Critic (SPAC), is presented in Algorithm 1.

C. Mixture Density Network for a Policy Function

To define a multi-modal policy in continuous action spaces, we use a mixture of Gaussian distributions as a policy function. It is the weighted sum of Gaussian distributions, so output of the policy network consists of weight $w\ (s)$, mean (s), and standard deviation (s) of each mixture. Then the probability of action a at state s is equal to $(a|s) = \sum_{i=1}^{K} w^i(s)\mathcal{N}(a|\ ^i(s);\ ^2{}_i(s))$ where K is the number of Gaussians in the mixture.

Algorithm 1 SPAC: Sparse Actor Critic

```
, , ,        initialize parameters.
M            empty memory.
for i = 0 to N do
  Initialize state s_0.
  for t = 0 to T do
    Sample an action a_t        (a|s_t).
    Observe next state s_{t+1} from the environment.
    Add the experience to memory.
  end for
                _V O L_V( )
              _Q O L_Q( )
          +     O L ( )
            + (1      )
end for
```

V. Experiments

In experiment, we verify the effect of the effect of the sparse Tsallis entropy on exploration and compare the proposed methods to the prior actor-critic methods. First, we examine exploration and exploitation performance of our algorithm in multi-modal environments by comparing to SAC. In addition, we conduct simulations on MuJoCo [22] and Torcs, the open racing car simulator which is used as a reinforcement learning testbed in [16], [23], to investigate whether our algorithm can be applied to more complex continuous tasks. The proposed method is compared with three prior methods: soft actor-critic (SAC), Deep deterministic policy gradient (DDPG), and Proximal Policy Optimization (PPO).

A. Multiple Goals

We consider two multi-goal problems with a two-dimensional continuous action space and a two-dimensional coordinate continuous state space. The first multi-goal environment is same to the multi-goal environment of [6] which has four goals equally spaced and far from its origin. The second multi-goal environment has an additional repulsive reward which is negatively proportional to the distance from the agent to the closest repulsive point.

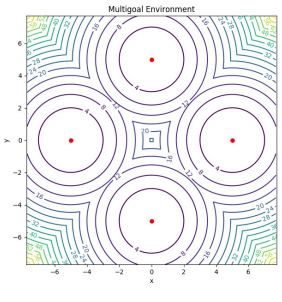
The structure of networks is a simple multi-layer perceptron which has two hidden layers with 100 hidden units. We use mixtures of four Gaussian distributions described in IV as a policy function and compare total rewards of our method to total rewards of SAC. The reward map of multi-goal problems and experiment results are shown in Figure 1.

As shown in Figure 1, the proposed method shows faster convergence speed and higher performance than SAC.

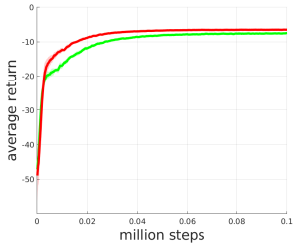
As stated before, entropy regularization methods have the effect of preventing a policy function from converging to a deterministic policy function. Our algorithm has the effect of restricting the exploration because the degree of regularization to the policy function in our algorithm is less than that in SAC. Hence, the policy learned with SPAC is less deviated from the optimal policy than the policy learned with SAC, resulting in higher performance and convergence speed.

B. MuJoCo

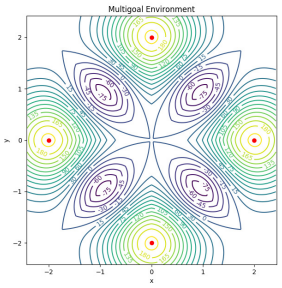
To validate efficiency, we compare our method with prior methods on MuJoCo continuous tasks: InvertedPendulum-v2, InvertedDoublePendulum-v2, and Swimmer-v2. Detail problem settings such as state and action representations or a definition of rewards are explained in [22]. Both our method and soft actor-critic utilize a Gaussian distribution to model a policy function. The network structures used in all comparison methods are identical multi-layer perceptrons. The MuJoCo experiment results are shown in Figure 2. Our method shows higher performance and convergence speed than SAC, since our method encourages exploration performance while it does not harm exploitation performance much.



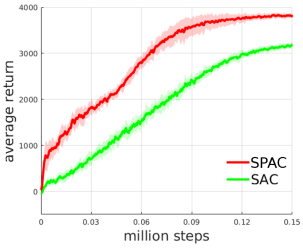
(a) The first multi-goal problem



(b) Evaluation



(c) The second multi-goal problem



(d) Evaluation

Fig. 1: Two multi-goal problems. (a) The reward map of the first multi-goal problem. It has goals on (5, 0) and (0, 5). (b) The total rewards of SPAC (red) and SAC (green) of the first multi-goal problem. The average of three different random seed results is shown in bold lines and the variance is shown in shaded areas. Our method outperforms SAC with respect to convergence speed and performance. (c) The reward map of the second multi-goal problem. It has goals on (2, 0) and (0, 2) and repulsive points on (1, 0) and (0, 1). (d) The total rewards of SPAC and SAC of the second multi-goal problem. Also, our method outperforms SAC in terms of convergence speed and performance.

C. Torcs

To demonstrate the applicability of our method in high dimensional continuous tasks and autonomous driving problem, we test our method on Torcs, the open racing car simulator, which is used in [16], [23]. The goal of Torcs is to complete a lap as fast as it can while minimizing the damage caused by collisions. And the experiment is divided into two cases when the input is a low-dimensional features and high-dimensional images. When the input is 28-dimensional features, steering, acceleration, and brake are used to control a car. On the other hand, when the input is a 64 * 64 pixel RGB image, only steering is used to control a car to reduce the training time.

We use two hidden layers with 256 nodes as a network architecture for . When the input is an image, we use a convolutional neural network to reduce it to a 28-dimensional vector and the rest of the network is identical to the feature based problem. As in previous experiments, the same network structure used by our algorithm is used in comparison methods. The results are shown in Figure 3. Because the image based problem has much more complex state space than the feature based problem, it shows a lower convergence speed than when using features although the action space is less complicate. Entropy regularized methods show higher performance than other actor-critic algorithms and our method shows the fastest convergence speed and the highest performance in both feature based and image based problems.

In addition, we measure the performance of SPAC and SAC with varying the entropy regularization coefficient in feature based Torcs problem. The results are shown in Figure 4. We observe that SPAC is more stable than SAC with a change of regularization coefficient within a certain range of regularization coefficieint. Furthermore, SPAC shows the higher performance than SAC at all value. Given that the is the most important parameter to train, robustness for value suggests that our method is more convenient than SAC.

VI. Conclusion

In this paper, we propose a new off-policy actor-critic algorithm using the sparse Tsallis entropy regularization method. Since the proposed method takes advantage of the sparse Tsallis entropy, it enables us to find the maximum reward while increasing the entropy of the policy. Thus, it leads to achieve both the exploitation performance (finding the maximum reward) and the exploration performance (increasing the entropy of the policy). Moreover, the sparse Tsallis entropy regularization restricts the degree of exploration in contrast with the Shannon-Gibbs entropy, so our method does not excessively impair the exploitation performance of the actor-critic algorithm. We derive iteration update rules of value functions based on the Bellman equation of the sparse MDP and a policy function by differentiating the objective function of the sparse MDP and extend it to the off-policy algorithm. In experiments, we have shown that the proposed method shows the state-of-the-art quality in terms of convergence speed and performance at various continuous tasks.

References

[1] S. Gu, E. Holly, T. Lillicrap, and S. Levine, "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 3389–3396.

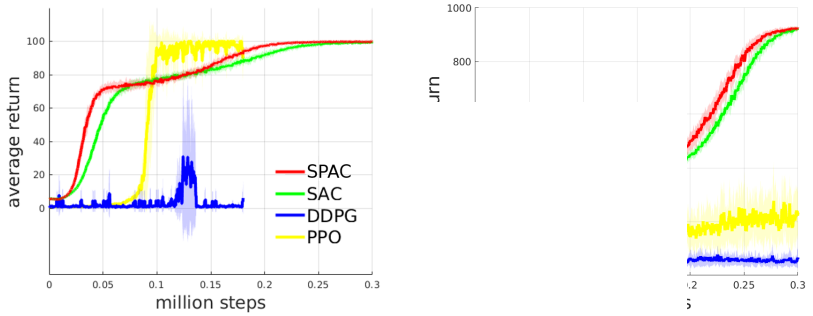
[2] G. Kahn, A. Villaflor, V. Pong, P. Abbeel, and S. Levine, "Uncertainty-aware reinforcement learning for collision avoidance," *arXiv preprint arXiv:1702.01182*, 2017.

[3] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi, "Target-driven visual navigation in indoor scenes using deep reinforcement learning," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 3357–3364.

[4] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1238–1274, 2013.

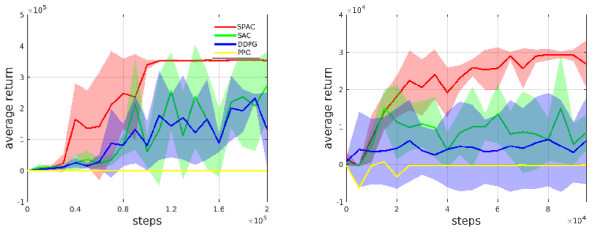
[5] N. Heess, D. Silver, and Y. W. Teh, "Actor-critic reinforcement learning with energy-based policies." in *EWRL*, 2012, pp. 43–58.

[6] T. Haarnoja, H. Tang, P. Abbeel, and S. Levine, "Reinforcement learning with deep energy-based policies," *arXiv preprint arXiv:1702.08165*, 2017.



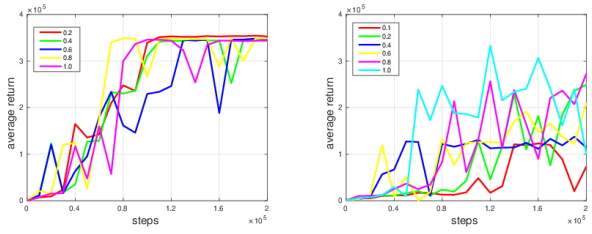
(a) InvertedPendulum-v2 (b) InvertedDoublePendulum-v2 (c) Swimmer-v2

Fig. 2: Results of MuJoCo problems. It compares the total rewards of SPAC (red) to SAC (green), DDPG (blue), and PPO (yellow) of (a) InvertedPendulum-v2, (b) InvertedDoublePendulum-v2, and (c) Swimmer-v2. The average of three different random seed results is shown in bold lines and the variance is shown in shaded areas.



(a) Feature based input result (b) Image based input result

Fig. 3: (a) Results of the total rewards of SPAC (red) to SAC (green), DDPG (blue), and PPO (yellow) when input is based on (a) feature and (b) image in Torcs experiment. SPAC outperforms prior actor critic algorithms in both problems.



(a) Sparse actor critic (b) Soft actor critic

Fig. 4: Results of feature based Torcs problem with different in (a) sparse actor critic and (b) soft actor critic. Soft actor critic is more sensitive to the entropy regularization coefficient than sparse actor critic. Sparse actor critic shows the consistent and high performance at the end of the training phase.

[7] J. Schulman, X. Chen, and P. Abbeel, "Equivalence between policy gradients and soft q-learning," *arXiv preprint arXiv:1704.06440*, 2017.

[8] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," *arXiv preprint arXiv:1801.01290*, 2018.

[9] K. Lee, S. Choi, and S. Oh, "Sparse markov decision processes with causal sparse tsallis entropy regularization for reinforcement learning," *arXiv preprint arXiv:1709.06293*, 2017.

[10] Y. Chow, O. Nachum, and M. Ghavamzadeh, "Path consistency learning in tsallis entropy regularized mdps," in *International Conference on Machine Learning*, 2018, pp. 978–987.

[11] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, vol. 8, no. 3-4, pp. 229–256, 1992.

[12] R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Advances in neural information processing systems*, 2000, pp. 1057–1063.

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

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

[15] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," *arXiv preprint arXiv:1506.02438*, 2015.

[16] 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," *arXiv preprint arXiv:1509.02971*, 2015.

[17] O. Nachum, M. Norouzi, K. Xu, and D. Schuurmans, "Bridging the gap between value and policy based reinforcement learning," in *Advances in Neural Information Processing Systems*, 2017, pp. 2775–2785.

[18] M. Bloem and N. Bambos, "Infinite time horizon maximum causal entropy inverse reinforcement learning," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*. IEEE, 2014, pp. 4911–4916.

[19] T. Degris, M. White, and R. S. Sutton, "Off-policy actor-critic," *arXiv preprint arXiv:1205.4839*, 2012.

[20] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *ICML*, 2014.

[21] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[22] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 5026–5033.

[23] J. Koutník, J. Schmidhuber, and F. Gomez, "Evolving deep unsupervised convolutional networks for vision-based reinforcement learning," in *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*. ACM, 2014, pp. 541–548.