ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230288

Enhancing OOD Generalization in Offline Reinforcement Learning with Energy-Based Policy Optimization

Hongye Cao[a], Shangdong Yang[b,a], Jing Huo[a;*], Xingguo Chen[b,a] and Yang Gao[a]

[a]State Key Laboratory for Novel Software Technology, Nanjing University, China
[b]School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, China

Abstract. Offline Reinforcement Learning (RL) is an important research domain for real-world applications because it can avert expensive and dangerous online exploration. Offline RL is prone to extrapolation errors caused by the distribution shift between offline datasets and states visited by behavior policy. Existing offline RL methods constrain the policy to offline behavior to prevent extrapolation errors. But these methods limit the generalization potential of agents in Out-Of-Distribution (OOD) regions and cannot effectively evaluate OOD generalization behavior. To improve the generalization of the policy in OOD regions while avoiding extrapolation errors, we propose an Energy-Based Policy Optimization (EBPO) method for OOD generalization. An energy function based on the distribution of offline data is proposed for the evaluation of OOD generalization behavior, instead of relying on model discrepancies to constrain the policy. The way of quantifying exploration behavior in terms of energy values can balance the return and risk. To improve the stability of generalization and solve the problem of sparse reward in complex environment, episodic memory is applied to store successful experiences that can improve sample efficiency. Extensive experiments on the D4RL datasets demonstrate that EBPO outperforms the state-of-the-art methods and achieves robust performance on challenging tasks that require OOD generalization.

1 Introduction

Reinforcement Learning (RL) has been applied in various real-world applications, including robotics [14], intelligent gaming [29], and sequential recommendation systems [1]. The RL mode of exploring and exploiting enables effective online policy learning for agents [28, 6]. However, in actual domains such as medical care and autonomous driving, the development of RL is difficult due to the expensive and dangerous trial-and-error in online interaction [23]. Hence, offline RL is proposed to learn policies from offline collected datasets without online exploration.

Offline RL is challenging. The learned policy of the agent is susceptible to extrapolation errors caused by the distribution shift between offline datasets and states visited by the behavior policy during training [11, 24]. These errors can be amplified by bootstrapping, resulting in severe estimation errors. The subsequent difficulty in correcting is also challenging due to the lack of online exploration. Existing offline RL methods focus on the policy constraints which prevent extrapolation errors during policy learning. The policy is restricted to offline behavior to prevent extrapolation errors in Out-Of-Distribution (OOD) regions that do not involve offline transitions [2, 10, 19].

Existing model-free offline RL methods [17, 37] introduce the regularization for the value functions to restrict the learned policy within the offline data manifold. These approaches are conservative and can not effectively learn the behavior policy from offline datasets. Model-based methods [24, 36] utilize dynamics models that are built from offline datasets to optimize the policy. These models provide a prior knowledge of how the environment behaves, which makes it easier to learn an optimal policy than model-free approaches. Model-based methods introduce uncertainty factors into policy optimization to prevent divergence. The uncertainty of the current policy reduces the interference of extrapolation errors. However, existing uncertainty factors limit the behavior to offline datasets by estimating the model discrepancies that might overfit the limited and suboptimal offline datasets [32, 35]. The agent is limited to the behavior policy of offline datasets and can not achieve tasks in OOD regions. The learned policy appears to be over-conservative. Hence, it is important to improve the generalization ability of the policy in offline RL. The generalization of offline RL in OOD regions is disturbed by the risk caused by extrapolation errors. How to trade-off the OOD generalization (return) and extrapolation errors (risk) is the most critical study point. The evaluation of exploration behavior can effectively balance the return and risk during OOD generalization. To the best of our knowledge, there are no existing research methods for evaluating exploration behavior during OOD generalization in offline RL.

To solve the abovementioned problem, we propose an Energy-Based Policy Optimization (EBPO) method to enhance OOD generalization in offline RL. An energy function based on the distribution of offline data is proposed for the evaluation of OOD generalization behavior, instead of relying on model discrepancies to constrain the policy. Errors that arise from offline data in modeling result in more extrapolation errors when calculating uncertainties using model discrepancies [39]. The energy model directly built from offline datasets for behavior evaluation can reduce errors and improve the stability of training. The way of quantifying exploration behavior in terms of energy values can balance the return and risk during OOD generalization. This effective approach is to assign a high energy value to the exploration behavior when the policy faces more risk, leading to a reduction in reward to avoid risky actions. Conversely, when the policy achieves higher returns, the energy assigned to exploration behavior is reduced, and the shaped reward is amplified to encourage further exploration. This trade-off can significantly enhance OOD generalization by robustly balancing exploration and exploitation.

* Corresponding Author. Email: huojing@nju.edu.cn

In the practical implementation, we first build offline RL models to learn offline behavior policy. Subsequently, we perform the rollout exploration based on the energy evaluation for OOD generalization. The energy score is obtained from the purely discriminative classification model without explicit reliance on density estimators, which is less susceptible to overfitting problems. To improve the stability of the OOD generalization, especially in the complex problem with sparse rewards, episodic memory (EM) is applied to effectively improve sample efficiency. Episodic memory stores the best rewards in the past and the policy can repeat the best results without gradient-based learning that ensures stable energy-based OOD generalization. Rollout based on successful experiences can robustly conduct OOD generalization to accelerate convergence of the policy. This method improves the OOD generalization with the help of energy and episodic memory. The contributions of this research are:

- The energy function is proposed to evaluate the exploration behavior during OOD generalization. The energy value is introduced into the uncertainty factor to balance the return and risk during rollout exploration.
- Episodic memory is applied to store successful policy experiences to improve sample efficiency and solve the problem of sparse reward in complex environment. We theoretically prove the policy convergence of EBPO.
- Extensive experiments on the D4RL datasets demonstrate that the proposed method outperforms the state-of-the-art (SOTA) methods and achieves robust performance on challenging tasks requiring OOD generalization. Ablation studies further demonstrate the contributions of hyperparameters and components in EBPO.

2 Related work

2.1 Model-based offline RL

Model-based offline RL approaches are based on the supervised learning paradigm [5, 15, 27]. Dynamic models are constructed by learning transitions from offline datasets and interact as a kind of environment simulator for policy optimization of the agent. Existing model-based offline RL methods constrain the policy by incorporating the uncertainty factors during the rollout exploration process [31, 36]. These methods reduce the estimation errors between the learned policy and the real-world environment by uncertainty optimization. Uncertainty through reward shaping is calculated from the inconsistency of ensemble models predictions for each state-action pair. Constrains in policy exploration and optimization to regions with high consistency for better worst-case performances are also present in deployment environments. However, these methods reduce the interference of extrapolation errors, while the generalization of the policy is greatly limited [38, 39]. Moreover, these approaches can not evaluate the OOD generalization behaviors during policy optimization.

2.2 Energy-based models

Energy-based models capture dependencies between variables by associating scalar energy values with variables [18, 20, 21]. An energy function is constructed from the model variables, with observed variable energy values lower than unobserved variable energy values. The energy-based method is to establish a function $E(x) : \mathbb{R}^D \to \mathbb{R}$, which maps each point x of the input space to a non-probabilistic scalar energy. A set of energy values can be transformed into a probability density $p(x)$ by the Gibbs distribution expressed below:

$$p\left(x\right) = \frac{e^{-E(x)/\tau}}{\int e^{-E(x)/\tau}}, \tag{1}$$

where the denominator is called the partition function, and τ is the temperature factor. The energy $E(x)$ for a given data point $x \in \mathbb{R}^D$ can be expressed as the negative of logarithmic partition as follows:

$$E\left(x\right) = -\tau \cdot \log \int e^{-E(x)/\tau}. \tag{2}$$

Model-based offline RL builds models in a supervised way. We propose to build the energy model by the offline datasets. To the best of our knowledge, this is the first time to evaluate OOD behavior by energy model in offline RL.

2.3 Episodic memory-based methods

Following concepts in psychobiology, episodic memory-based methods store promising experiences in nonparametric tables. This human-like memory approach quickly retrieves past successful policies when encountering similar scenarios [7, 22, 25]. Episodic memory proposes a framework to quickly retrieve past successful policies to improve sample efficiency [13]. The agent can repeat the best results without gradient-based learning. Reuse of successful experiences can accelerate convergence of the policy and handle the sparse reward problem of low-quality datasets. Episodic control [4] updates the memory table by taking the largest return $R(s, a)$ among all inferences starting from the same state-action pair (s, a).

$$Q^{EM}\left(s,a\right) = \begin{cases} \max\{R\left(s,a\right), Q^{EM}\left(s,a\right)\}, & \text{if } (s,a) \in EM, \\ R\left(s,a\right), & \text{otherwise}, \end{cases} \tag{3}$$

where EM represents the stored episodic memory. The process of exploring the OOD regions is interrupted by a lot of extrapolation errors. Reuse of good experiences can prevent policy divergence. OOD generalization based on excellent experiences can achieve stable policy generalization. Hence, episodic memory is applied to store successful experiences for generalization in EBPO.

3 Preliminaries

In the RL framework, the process of the agent interacting with the environment is formulated as a Markov Decision Process (MDP). The standard MDP is defined by the tuple of $M = \langle \mathcal{S}, \mathcal{A}, T, \mu_0, r, \gamma \rangle$, where $\mathcal{S}$ denotes the state space, $\mathcal{A}$ represents the action space, $T(s'|s, a)$ is the transition dynamic model, $r(s, a)$ is the reward function and μ_0 is the distribution of the initial state s_0. $\gamma \in [0, 1)$ is the discount factor. The goal of RL is to learn a policy $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$, which can maximize the expected discounted cumulative reward $\eta_M\left(\pi\right) := \mathbb{E}_{s_0 \sim \mu_0, s_t \sim T, a_t \sim \pi}\left[\sum_{t=0}^{\infty} \gamma^t r\left(s_t, a_t\right)\right]$. The value function $V_M^{\pi}\left(s\right) := \mathbb{E}_{s_t \sim T, a_t \sim \pi}\left[\sum_{t=0}^{\infty} \gamma^t r\left(s_t, a_t\right) | s_0 = s\right]$ is the expected discounted return under the policy π that the state starts from s.

In offline RL, the policy is only learned from the collected offline datasets and cannot interact with the real-world environment. We define the offline dataset $D_{env} = \{(s, a, r, s')\}$, which contains all the collected state-action transitions and D_{em} contains the collected state-action transitions and returns of the episodic memory. In the model-based approach, $\hat{T}(s'|s, a)$ is the dynamic model estimated

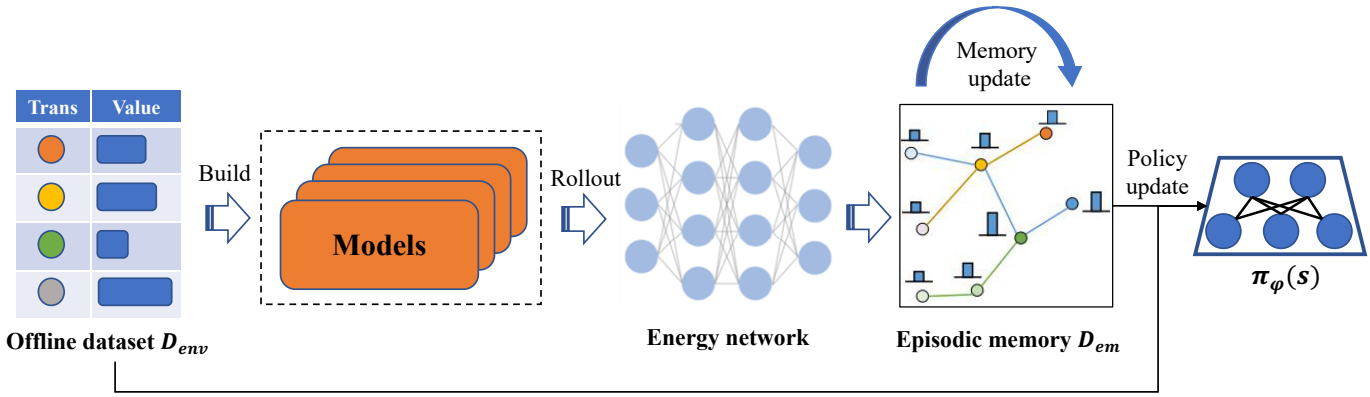

Figure 1. The framework of EBPO. Ensemble models are trained from the offline dataset D_{env}. Energy network is proposed for the rollout exploration evaluation and policy π is updated with the episodic memory D_{em} and offline dataset D_{env}.

from transitions in the D_{env}. This dynamic model defines the estimated MDP $\hat{M} = \langle \mathcal{S}, \mathcal{A}, \hat{T}, \mu_0, r, \gamma \rangle$. $\mathbb{P}^{\pi}_{\hat{T},t}(s)$ is the probability of being in state s at time step t when the actions and transitions are sampled from π and $\hat{T}$. $\rho^{\pi}_{\hat{T}}(s,a) := \pi(a|s)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}^{\pi}_{\hat{T},t}(s)$ denotes the discounted occupancy measure of policy π under $\hat{T}$. We denote the improper expectations of $\eta_{\hat{M}}(\pi) = \mathbb{E}_{\rho^{\pi}_{\hat{T}}}[r(s,a)]$. Our goal is to learn a policy that maximizes $\eta_{\hat{M}}(\pi)$ with the offline datasets D_{env} and episodic memory D_{em}.

4 Method

The framework of EBPO is shown in Figure 1. First, based on the model-based architecture, ensemble models are trained from the offline datasets. Afterwards, we perform the rollout to explore the OOD regions. In the exploration process, an energy network is proposed to evaluate the generalization behavior, and the best transition is stored in the episodic memory. Episodic memory is updated at a fixed frequency. Finally, the policy is updated using the offline dataset and episodic memory. Rollout based on the energy evaluation explores the OOD regions and episodic memory stores successful experiences to ensure steady exploration. This energy-based method can effectively balance the return and risk during exploration process.

The energy network is introduced into rollout exploration for the evaluation of OOD generalization behavior. How to effectively balance the return and risk based on the energy-based rollout exploration is discussed in Section 4.1. Subsequently, episodic memory is applied to capture successful experiences and accelerate convergence, which is given in Section 4.2. Finally, the practical algorithms and implementations are presented in Section 4.3.

4.1 Energy-based OOD generalization evaluation

Due to the lack of interaction with the real environment, offline RL methods are prone to extrapolation errors. The uncertainty of offline RL based on the energy evaluation is introduced into the rollout exploration to avoid extrapolation errors. Energy $E_\pi(s,a)$ is proposed to balance the return and risk of extrapolation errors by reward shaping. Our goal is to learn a policy that can maximize the function $\mathbb{E}_{(s,a)\sim\rho^{\pi}_{\hat{T}}}[r(s,a) - E_\pi(s,a)]$.

High energy values represent high OOD generalization uncertainty, while low energy values represent the rollout exploration behavior closer to in-distribution regions. The reward combined with energy values can effectively balance the return and risk during OOD generalization. A higher reward of the current exploration behavior does not fully represent the stability of the policy. The reward is adaptively adjusted according to energy values of exploration behavior during OOD generalization. The following two parts first introduce the derivation of uncertainty through reward shaping and an energy function is proposed for OOD generalization evaluation.

The following part introduces the derivation of uncertainty through reward shaping in offline RL. Offline RL uses offline datasets to optimize policies. Due to the lack of online interaction, the uncertainty in the rollout exploration process can easily lead to extrapolation errors. We define the uncertainty of the model. The estimator error [39] $U^{\pi}_{\hat{M}}(s,a)$ for the true return between the optimal and actual model is defined as:

$$U^{\pi}_{\hat{M}}(s,a) := \mathbb{E}_{s'\sim\hat{T}(s,a)}\left[V^{\pi}_{\hat{M}}(s')\right] - \mathbb{E}_{s'\sim T(s,a)}\left[V^{\pi}_{M}(s')\right]. \quad (4)$$

We then combine the objective of policy optimization to maximize the discounted cumulative reward. The Equation 4 can be derived as follows:

$$\eta_{\hat{M}}(\pi) - \eta_{M}(\pi) = \gamma\mathbb{E}_{(s,a)\sim\rho^{\pi}_{\hat{T}}}\left[U^{\pi}_{\hat{M}}(s,a)\right]. \quad (5)$$

Based on the estimation errors between the models and the expected discounted return under policy π, the maximized target discounted return can be derived as follows:

$$\begin{aligned}\eta_{M}(\pi) &= \mathbb{E}_{(s,a)\sim\rho^{\pi}_{\hat{T}}}\left[r(s,a) - \gamma U^{\pi}_{\hat{M}}(s,a)\right] \\ &\geq \mathbb{E}_{(s,a)\sim\rho^{\pi}_{\hat{T}}}\left[r(s,a) - |\gamma U^{\pi}_{\hat{M}}(s,a)|\right] \\ &= \mathbb{E}_{(s,a)\sim\rho^{\pi}_{\hat{T}}}\left[r(s,a) - u(s,a)\right] \\ &\geq \eta_{\hat{M}}(\pi),\end{aligned} \quad (6)$$

where $u(s,a)$ is an artificial fixed uncertainty [39] of the policy. The goal is to learn a policy that can maximize $\mathbb{E}_{(s,a)\sim\rho^{\pi}_{\hat{T}}}[r(s,a) - u(s,a)]$. Existing uncertainty computations [32, 39] only calculate the deviation during policy optimization without evaluating OOD generalization. Therefore, we propose an energy function to evaluate the exploration behavior through reward shaping.

The following part introduces the energy evaluation of uncertainty in offline RL. Energy-based methods capture dependencies between variables by associating scalar energy values with variables in the model [18]. The energy function is constructed from the model variables. The observed variable energy values are lower than the unobserved ones. A general theoretical framework is proposed for many

learning models in the form of a non-probabilistic factor. Compared to probabilistic approaches, this framework offers greater flexibility in the design of architecture and training criteria. An energy-based regularization term is applied to the detection of OOD behavior. The energy value can be obtained from a purely discriminative classification model.

For the K-class neural network classifier $f(s,a):\mathbb{R}^D\to\mathbb{R}^K$, the input is mapped to K logarithms. In EBPO, we set two categories: OOD and in-distribution regions. We use the state-action transitions and rewards in OOD and in-distribution regions to train the classifier based on the ensemble models. The probability that the current sample belongs to a certain class is obtained through SoftMax:

$$p\left(y|\left(s,a\right)\right)=\frac{e^{f_y(s,a)/\tau}}{\sum_{i=1}^{K}e^{f_i(s,a)/\tau}},\tag{7}$$

where $f_y(s,a)$ represents the y^{th} index of $f(s,a)$ and denotes the logit corresponding to the y^{th} class label. By connecting Equation 1 and Equation 7, we can define the given input (s,a) of class y as $E_\pi(s,a)=-f_y(s,a)$.

More importantly, we can use the denominator of the SoftMax activation to represent the energy function $E_\pi(s,a)$ over the reward of state-action pair $(s,a)\in\mathbb{R}^D$ without changing the parameterization of the neural network $f(s,a)$ as:

$$E_\pi\left(s,a\right)=-\tau\cdot\log\sum_{i}^{K}e^{f_i(s,a)/\tau}.\tag{8}$$

The energy of the sample (s,a) is converted to a scalar only with respect to $f(s,a)$. We apply the energy value into the reward norm term for OOD behavior exploration without relying on the density estimator while avoiding the difficult optimization in training the model process. Hence, the goal is to learn a policy that can maximize the function $\mathbb{E}_{(s,a)\sim\rho_{\hat{T}}^{\pi}}[r(s,a)-E_\pi(s,a)]$.

The model is trained with the negative log-likelihood loss in in-distribution datasets D_{in}. We can express the negative log-likelihood loss for a model trained on in-distribution data as:

$$L_{nll}=\mathbb{E}_{(s,a)\sim D_{in}}\left(-\log\frac{e^{f_y(s,a)/\tau}}{\sum_{j=1}^{K}e^{f_j(s,a)/\tau}}\right).\tag{9}$$

By converting the logit value to the energy value, the loss function can be rewritten as follows:

$$L_{nll}=\mathbb{E}_{(s,a)\sim D_{in}}\left(\frac{1}{\tau}\cdot E_\pi\left(s,a\right)+\log\sum_{j=1}^{K}e^{-E_\pi(s,a)/\tau}\right),\tag{10}$$

where the first term pushes down the energy of the ground truth answer. The second term can be interpreted as the Free Energy of the ensemble of energies. We apply the loss function into the training of the classifier.

When the reward with energy value can be guaranteed to gradually increase during rollout exploration, it is inferred that the policy can effectively generalize in the OOD region and is not greatly disturbed by extrapolation errors. Therefore, the energy of OOD generalization evaluation is used as the uncertainty factor to balance the return and risk during the rollout exploration. The energy model conducts scalar evaluation of the data instead of solely relying on binary classification. This soft-adaptive approach assigns higher energy values to exploration behavior in riskier situations, reducing rewards to discourage risky actions. This trade-off enhances OOD generalization by effectively balancing exploration and exploitation.

4.2 Episodic memory

The energy value is proposed to balance the return and risk of the policy. However, in complex environment, policy convergence bootstrapping remains challenging due to sparse rewards and unknown exploration behavior during OOD generalization. Effectively using empirical data for policy optimization is crucial. Therefore, value-based episodic memory methods are applied to avoid overly optimistic estimation during policy generalization. For the policy optimization based on episodic memory incorporating energy values, the energy-based evaluation can balance the return and risk for efficient OOD generalization, and episodic memory ensures the safe exploration of the OOD region and prevents divergence.

We use value-based planning to bootstrap more efficiently and implicit memory-based planning schemes plan strictly within offline datasets. This approach avoids overly optimistic estimation during the planning phase. We also optimize the expected state values instead of state-action values. Specifically, we compare the best return along the trajectory so far with the estimated value, and take the maximum between them to obtain the enhanced return. This process proceeds recursively from the last step to the first step and forms an implicit planning scheme in episodic memory which aggregates optimal experiences along and across trajectories. The entire backpropagation process can be expressed as follows:

$$R_t=\begin{cases}r_t, & t=L,\\ r_t+\gamma\max\left(R_{t+1},V\left(s_{t+1}\right)\right), & t<L,\end{cases}\tag{11}$$

where t represents the step size along the trajectory, L is the trajectory length, and $V(s_{t+1})$ generalizes from similar experiences. Furthermore, the backpropagation process in Equation 11 incorporating energy values is extended and rewritten as follows:

$$V_{t,h}=\begin{cases}V\left(s_t\right), & h=0,\\ r_t\left(s,a\right)-E_\pi\left(s,a\right)+\gamma V_{t+1,h-1}, & h>0,\end{cases}\tag{12}$$

$$R_t=V_{t,h^*},h^*=\underset{h>0}{\operatorname{argmax}}V_{t,h},\tag{13}$$

where h represents the horizon of rollout steps. $V_{t,h}=0$ if $t>L$. The best return and state-value transitions will be stored in the episodic memory. In order to prove the convergence of the episodic memory incorporating energy values, we consider the Bellman expectation operator B_μ and Bellman optimality operator B_*.

$$\begin{aligned}B_\mu V\left(s\right):=\mathbb{E}_{a\sim\mu(a|s)}\left[r\left(s,a\right)-E_\pi\left(s,a\right)\right.\\ \left.+\gamma\mathbb{E}_{s'\sim\pi(s'|s,a)}\left[V\left(s'\right)\right]\right],\end{aligned}\tag{14}$$

$$\begin{aligned}B_* V\left(s\right):=\max_a\left(r\left(s,a\right)-E_\pi\left(s,a\right)\right.\\ \left.+\gamma\mathbb{E}_{s'\sim\pi(s'|s,a)}\left[V\left(s'\right)\right]\right),\end{aligned}\tag{15}$$

where μ is the behavior policy.

Lemma 1. The Bellman expectation operator B_μ has the unique fixed point V_μ that $B_\mu V_\mu=V_\mu$. Moreover, the Bellman optimality operator B_* has the unique fixed point V_* that $B_* V_*=V_*$. Based on the Banach fixed point theorem [30], these two Bellman operators are convergent. Hence, the EBPO is convergent.

Proof. For the same behavior policy, the reward and energy functions based on the state-action transitions and reward distributions in the offline dataset are invariant. Based on the Banach fixed point theorem, if the Bellman operators are the compressive map of the metric

space (X, d), the Bellman operators have the unique function sets V_μ and V_*. First, we construct one metric space (X, d). The measure d can be shown by $L-\infty$ [3]:

$$||V||_\infty = \max_{i\in[0,|V|]} |V_i| . \tag{16}$$

We first prove the convergence of the Bellman expectation operator. For any V_1, V_2:

$$\begin{aligned}
&|B_\mu V_1(s) - B_\mu V_2(s)| \\
&= \left|\mathbb{E}_{a\sim\mu(a|s)}\left[r(s,a) - E_\pi(s,a) + \gamma\mathbb{E}_{s'\sim\pi(s'|s,a)}\left[V_1(s')\right]\right.\right. \\
&\quad \left.\left. -r(s,a) + E_\pi(s,a) - \gamma\mathbb{E}_{s'\sim\pi(s'|s,a)}\left[V_2(s')\right]\right]\right| \\
&= \gamma\left|\mathbb{E}_{a\sim\mu(a|s)}\left[\mathbb{E}_{s'\sim\pi(s'|s,a)}\left[V_1(s') - V_2(s')\right]\right]\right| \\
&= \gamma\left|\mathbb{E}_{a\sim\mu(a|s),s'\sim\pi(s'|s,a)}\left[V_1(s') - V_2(s')\right]\right| \\
&\le \gamma\mathbb{E}_{a\sim\mu(a|s),s'\sim\pi(s'|s,a)}\left[\left|V_1(s') - V_2(s')\right|\right] \\
&\le \gamma\mathbb{E}_{a\sim\mu(a|s),s'\sim\pi(s'|s,a)}\left[\max_s |V_1(s) - V_2(s)|\right] \\
&= \gamma\mathbb{E}_{a\sim\mu(a|s),s'\sim\pi(s'|s,a)}\left[||V_1 - V_2||_\infty\right] \\
&= \gamma\,||V_1 - V_2||_\infty ,
\end{aligned} \tag{17}$$

where $\gamma \in [0, 1]$. For any state s, $|B_\mu V_1(s) - B_\mu V_2(s)| \le \gamma||V_1 - V_2||_\infty$. We can derive:

$$\begin{aligned}
||B_\mu V_1 - B_\mu V_2||_\infty &= \max_s |B_\mu V_1(s) - B_\mu V_2(s)| \\
&\le \gamma\,||V_1 - V_2||_\infty .
\end{aligned} \tag{18}$$

Hence, the Bellman expectation operator B_μ is the compression map of $(\mathbb{R}^{|S|}, L_\infty)$. Based on the Banach fixed point theorem [30], B_μ has the unique fixed point V_μ that satisfies $B_\mu V_\mu = V_\mu$. Hence, the Bellman expectation operator is convergent. The following part will prove the convergence of the Bellman optimality operator. For any V_1, V_2:

$$\begin{aligned}
&|B_* V_1(s) - B_* V_2(s)| \\
&= \left|\left[\max_a(r(s,a) - E_\pi(s,a) + \gamma\mathbb{E}_{s'\sim\pi(s'|s,a)}\left[V_1(s')\right])\right]\right. \\
&\quad \left. - \left[\max_a(r(s,a) - E_\pi(s,a) + \gamma\mathbb{E}_{s'\sim\pi(s'|s,a)}\left[V_2(s')\right])\right]\right| \\
&\le \left|\max_a\left(r(s,a) - E_\pi(s,a) + \gamma\mathbb{E}_{s'\sim\pi(s'|s,a)}\left[V_1(s')\right]\right.\right. \\
&\quad \left.\left. -r(s,a) + E_\pi(s,a) - \gamma\mathbb{E}_{s'\sim\pi(s'|s,a)}\left[V_2(s')\right]\right)\right| \\
&= \left|\gamma\max_a\mathbb{E}_{s'\sim\pi(s'|s,a)}\left[V_1(s') - V_2(s')\right]\right| \\
&\le \gamma\max_a\left|\mathbb{E}_{s'\sim\pi(s'|s,a)}\left[V_1(s') - V_2(s')\right]\right| \\
&= \gamma\max_a\left|\sum_{s'}\pi(s'|s,a)\left[V_1(s') - V_2(s')\right]\right| .
\end{aligned} \tag{19}$$

Then, we can derive:

$$\begin{aligned}
&\gamma\max_a\left|\sum_{s'}\pi(s'|s,a)[V_1(s') - V_2(s')]\right| \\
&\le \gamma\max_a\sum_{s'}\pi(s'|s,a)\left|V_1(s') - V_2(s')\right| \\
&= \gamma\sum_{s'}\pi(s'|s,a_*(s))\left|V_1(s') - V_2(s')\right| \\
&\le \gamma\sum_{s'}\pi(s'|s,a_*(s))\max_s|[V_1(s) - V_2(s)]| \\
&= \gamma\sum_{s'}\pi(s'|s,a_*(s))\,||V_1 - V_2||_\infty \\
&= \gamma\,||V_1 - V_2||_\infty ,
\end{aligned} \tag{20}$$

where $\gamma \in [0, 1]$. For any state s, $|B_* V_1(s) - B_* V_2(s)| \le \gamma||V_1 - V_2||_\infty$. We can derive:

$$\begin{aligned}
||B_* V_1 - B_* V_2||_\infty &= \max_s |B_* V_1(s) - B_* V_2(s)| \\
&\le \gamma\,||V_1 - V_2||_\infty .
\end{aligned} \tag{21}$$

Therefore, the Bellman expectation operator B_* is the compression map of $(\mathbb{R}^{|S|}, L_\infty)$. Based on the Banach fixed point theorem [30], B_* has the unique fixed point V_* that satisfies $B_* V_* = V_*$. Hence, the Bellman optimality operator is convergent. These two Bellman operators are convergent, and the EBPO is convergent.

4.3 Practical implementation

We describe the practical implementation of EBPO driven by the abovementioned analysis. Ensemble models are trained in the offline datasets by a supervised learning mode. Supervised learning patterns efficiently mine all state-action transitions in the datasets. After obtaining N ensemble models, we initialize the agent state and episodic memory from the dataset. For each epoch, the model is rolled out to explore the OOD region based on the initialize state. During the rollout exploration process, the energy value is introduced into the reward normalization as an uncertainty term. Afterwards, transitions acquired by rollout are added to episodic memory. The above process is a round of rollout exploration of the OOD region. Finally, SAC [12] algorithm is used to update policy π with D_{env} and D_{em} until convergence. Meanwhile, the episodic memory is updated with a memory update frequency p. Algorithm 1 Energy-Based Policy Optimization is listed below.

Algorithm 1 Energy-Based Policy Optimization (EBPO)

Input: Offline dataset D_{env}, rollout horizon h, episodic memory D_{em}, frequency p.

Train an ensemble of N dynamic models $\{M_i\}_{i=1}^N$ on the dataset D_{env}
Initialize critic network V_ω and actor network π_φ with random parameters ω, φ
Initialize episodic memory D_{em}
for *epoch 1, 2, 3 . . .* **do**
  Sample state s_1, action a_1 from D_{env} for the initialization of the rollout
  for *j = 1, 2 . . . , h* **do**
    Sample an action a_j from $\pi_\varphi(a|s_j)$
    Randomly select dynamics M from $\{M_i\}_{i=1}^N$ and sample $s_{j+1}, r_j \sim M(s_j, a_j)$
    $r_j \leftarrow r_j - E_\pi(s_j, a_j)$
    Add sample (s_j, a_j, r_j, s_{j+1}) to D_{em}
  end
  Use SAC to update critic network V_ω and actor network π_φ with $D_{env} \cup D_{em}$
  if *epoch mod* $p = 0$ **then**
    for *transitions* ε *in episodic memory* D_{em} **do**
      for s_j, a_j, r_j, s_{j+1} *in reversed*(ε) **do**
        Calculate R_j with Equation 13 and add into the episodic memory D_{em}
      end
    end
  end
end

Table 1. Comparative experiment results on the D4RL datasets. Each number is the normalized score proposed in [8], averaged over 5 random seeds. We bold red the highest score among all methods. Suboptimal results are marked in blue. ± standard deviation.

Environment	Dataset	EBPO	TD3-CVAE	F-BRC	COMBO	MOPO	TD3+BC	SAC	CQL	BC
HalfCheetah	random	**39.9±2.3**	28.6±2.0	33.3±1.3	**38.8**	35.4	10.2	30.5	35.4	2.1
HalfCheetah	medium	**55.9±0.2**	43.2±0.4	41.3±0.3	**54.2**	42.3	42.8	-4.3	44.4	36.1
HalfCheetah	mixed	**62.8±1.1**	45.3±0.4	43.2±1.5	**55.1**	53.1	43.3	-2.4	46.2	38.4
HalfCheetah	med-expert	**102.8±0.4**	**96.1±9.7**	93.3±10.2	90.0	63.3	95.9	1.8	62.4	35.8
Hopper	random	**13.3±0.2**	11.7±0.2	11.3±0.2	**17.9**	11.7	11.0	11.3	10.8	1.6
Hopper	medium	**98.9±6.2**	55.9±11.4	**99.4±0.3**	94.9	28.0	98.5	0.8	58.0	29.0
Hopper	mixed	**98.3±0.3**	46.7±17.9	35.6±1.0	**73.1**	67.5	31.4	1.9	48.6	11.8
Hopper	med-expert	106.3±4.9	111.6±2.3	**112.4±0.3**	111.1	23.7	**112.2**	1.6	98.7	111.9
Walker2d	random	**22.5±0.3**	5.5±8.0	1.5±0.7	7.0	**13.6**	1.4	4.1	7.0	9.8
Walker2d	medium	71.4±5.3	68.2±18.7	78.8±1.0	75.5	11.8	**79.7**	0.9	**79.2**	6.6
Walker2d	mixed	**69.4±1.1**	15.4±7.8	41.8±7.9	**56.0**	39.0	25.2	3.5	26.7	11.3
Walker2d	med-expert	99.7±1.6	84.9±20.9	**105.2±3.9**	96.1	44.6	101.1	-0.1	**111.0**	6.4

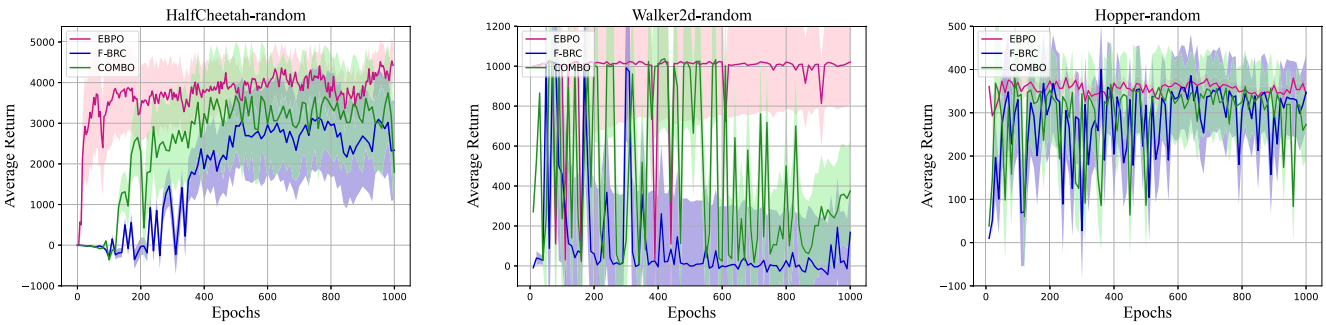

Figure 2. Learning curves on low-quality datasets. Each number is the averaged return during training, averaged over 5 random seeds and the shadow is the standard error.

5 Experiments

In the experiments, we aim to study the following questions: (i) How does the proposed method perform compared with SOTA methods on the standard offline RL benchmark? Moreover, how does EBPO perform on low-quality datasets with sparse reward? (ii) How does EBPO perform on challenging tasks that require generalizing OOD behavior? (iii) How does the hyperparameters affect the performance of EBPO? (iv) What is the effect of each component in EBPO?

5.1 Experimental datasets and settings

The D4RL [8] benchmark based on the MuJoCo [33] simulator is used as the dataset for this experiment. The dataset includes three environments (halfCheetah, hopper and walker2d) and four dataset types (random, medium, mixed and med-expert). We construct a total of 12 sub-datasets for experiments. In all domains, ensemble models are trained in the offline dataset. Each model in ensemble models is parameterized by a 4-layer feedforward neural network with 200 hidden units. The energy network is based on the 4-layer feedforward neural network for evaluation. The discount factor is set to 0.99. The epoch length is set to 1000. The rollout horizon is set according to different offline RL tasks. The frequency p of episodic memory update is set to 10. For the SAC policy optimization update, we sample a batch of 256 transitions, 5% of which are from D_{env} and the rest from D_{em}. The temperature factor τ is set to 5. The selection of temperature factor is determined by counterfactual query method through comparative experiment of hyperparameters.

5.2 Comparative methods

Comparative methods are shown as follows:

- BC [34]. Behavior policy imitated by supervised learning is made as the offline version.
- CQL [17]. A model-free offline RL method optimizes the policy with regularization.
- SAC [12]. SAC is based on maximum entropy and uses a random distributed policy function.
- TD3+BC [9]. An offline RL method with minimal changes.
- MOPO [39]. A model-based offline RL method constructs ensemble models and rollout with the uncertainty return penalty.
- COMBO [38]. An offline model-based conservative policy optimization method by regularization.
- F-BRC [16]. An offline RL approach that parameterizes critics as log behavior policies that generate offline data.
- TD3-CVAE [26]. An offline model-free method with bonus-based exploration.

5.3 Comparative experiments

The results of comparative experiments are shown in Table 1. EBPO achieves 7 best results on 12 tasks. Compared with model-based methods COMBO and MOPO, EBPO is the best in 8 out of the 12 tasks, indicating that EBPO outperforms existing model-based methods. Compared to uncertainty calculation methods based on the model divergence and regularization, energy-based method improves the generalization of the policy. Likewise, against model-free methods, EBPO achieves 8 best results on 12 tasks. CQL which constrains

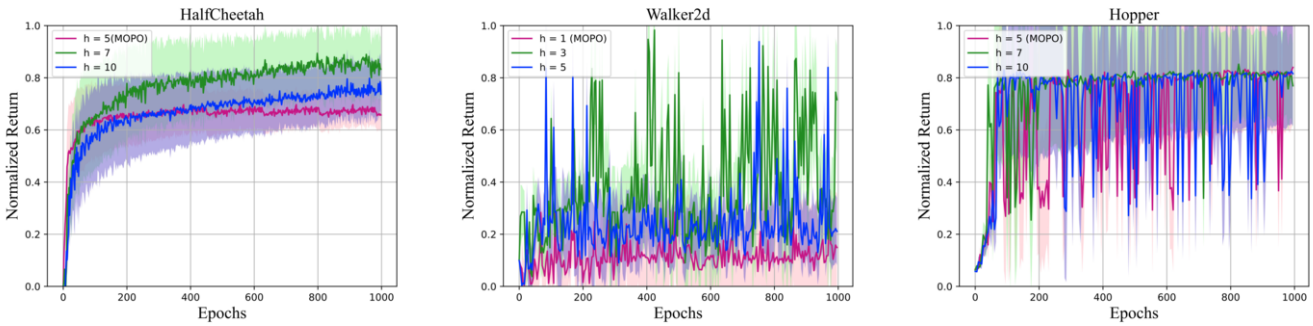

Figure 3. Learning curves on mixed datasets with three different rollout horizons h. Each number is the normalized return during training, averaged over 5 random seeds and the shadow is the standard error.

actions to in-distribution regions outperforms generalization-boosted method EBPO in the walker2d-medium dataset, but EBPO shows strong performance on other datasets. EBPO is not limited to the optimization of a single task. Compared to the bonus-based exploration model-free method TD3-CVAE, EBPO achieves 10 best results on 12 tasks. The proposed method shows better performance in offline RL tasks. Overall, EBPO outperforms the SOTA offline RL methods.

Moreover, we choose F-BRC and COMBO to conduct comparative experiments on three low-quality datasets with sparse reward. The experimental learning curves are shown in Figure 2. EBPO converges faster and has better performance in halfCheetah-random and walker2d-random datasets. Furthermore, EBPO shows stable performance in hopper-random dataset. Rollout based on the energy evaluation can improve OOD generalization during the agent exploration process. Episodic memory can improve sample efficiency and accelerate convergence. The successful experiences can handle the sparse reward problem and prevent the policy divergence.

5.4 *Sensitivity to rollout horizon*

Rollout horizons can reflect the ability of the agent exploration for OOD generalization. We set different rollout horizons of EBPO to conduct experiments on mixed datasets in Figure 3. Under different horizons, the performance of EBPO is greatly affected. In the halfCheetah and hopper environments, the method with the rollout horizon 7 achieves the best results. Meanwhile, in the walker2d environment, the method with rollout horizon 3 reaches the best performance. Compared with the rollout horizons in the SOTA model-based settings, EBPO expands the rollout exploration horizons within a certain range. Moreover, EBPO can improve the rollout horizons in difficult environment of walker2d. The proposed method can improve the exploration range of offline RL agents. Episodic memory stores successful experiences that ensure stable energy-based OOD generalization improvement. EBPO can improve the OOD generalization in offline RL robustly.

5.5 *Evaluation on tasks requiring OOD generalization*

The halfCheetah-jump (half-jump) and ant-angle tasks [8] are proposed to study the generalization performance of OOD in question (ii). Agents in these two environments not only need to execute the behavior policies in the dataset but also need to explore the OOD region to achieve tasks that are different from the transitions in the datasets. We choose model-based methods COMBO, MOPO and model-free method CQL for comparative experiments.

The results of experiments are shown in Table 2. EBPO outperforms the other methods on two OOD generalization tasks. The learning curves of EBPO, COMBO and MOPO for the experiment are shown in Figure 4. EBPO achieves better and stable performance in the halfCheetah-jump task. EBPO also outperforms the other two model-based methods in the ant-angle task. EBPO accelerates the improvement of the policy optimization. Energy can balance the return and risk during policy generalization which enables stable execution of OOD generalization behavior. These results further demonstrate that the proposed method can achieve stable and effective generalization in the OOD region.

Table 2. Average returns of half-jump and ant-angle tasks that require OOD generalization. All results are averaged over 5 random seeds. We bold the highest score across all methods.

Environment	EBPO	COMBO	MOPO	CQL	SAC
Half-jump	**6311.8**	5392.7	4016.6	741.1	-3588.2
Ant-angle	**3207.9**	2764.8	2530.9	2473.4	-966.4

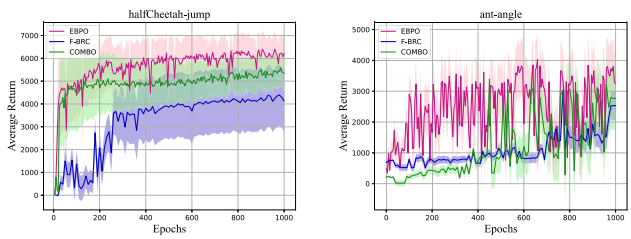

Figure 4. Learning curves on half-jump and ant-angle tasks. Each number is the average return during training, averaged over 5 random seeds and the shadow is the standard error.

6 Conclusion

This research proposes an energy-based policy optimization method for OOD generalization in offline RL. Energy function is proposed to evaluate the exploration behavior during OOD generalization. Episodic memory is then applied to improve sample efficiency and speed up policy convergence. The proposed method effectively balances the return and risk and improves the generalization of the policy. Experiments on the D4RL datasets show that EBPO outperforms the SOTA offline RL methods. Furthermore, EBPO achieves superior performance on two tasks requiring OOD generalization. Future studies are encouraged to combine meta-learning methods with EBPO to improve the adaptability in different RL tasks.

Acknowledgements

This work is supported in part by the Science and Technology Innovation 2030 New Generation Artificial Intelligence Major Project (2021ZD0113303), in part by the National Natural Science Foundation of China (62192783, 62276128, 62276142, 62206133), in part by the Collaborative Innovation Center of Novel Software Technology and Industrialization and in part by State Key Laboratory of Novel Software Technology Project (KFKT2022B12).

References

[1] M Mehdi Afsar, Trafford Crump, and Behrouz Far, 'Reinforcement learning based recommender systems: A survey', *ACM Computing Surveys (CSUR)*, (2021).

[2] Gaon An, Seungyong Moon, Jang-Hyun Kim, and Hyun Oh Song, 'Uncertainty-based offline reinforcement learning with diversified q-ensemble', *Advances in neural information processing systems*, **34**, 7436–7447, (2021).

[3] Marc G Bellemare, Will Dabney, and Rémi Munos, 'A distributional perspective on reinforcement learning', in *International Conference on Machine Learning*, pp. 449–458. PMLR, (2017).

[4] Charles Blundell, Benigno Uria, Alexander Pritzel, Yazhe Li, Avraham Ruderman, Joel Z Leibo, Jack Rae, Daan Wierstra, and Demis Hassabis, 'Model-free episodic control', *arXiv preprint arXiv:1606.04460*, (2016).

[5] Hongye Cao, Qianru Wei, Jiangbin Zheng, and Yanqing Shi, 'Model-based offline adaptive policy optimization with episodic memory', in *International Conference on Artificial Neural Networks*, pp. 50–62. Springer, (2022).

[6] Xingguo Chen, Xingzhou Ma, Yang Li, Guang Yang, Shangdong Yang, and Yang Gao, 'Modified retrace for off-policy temporal difference learning', in *Uncertainty in Artificial Intelligence*, pp. 303–312. PMLR, (2023).

[7] Francisco Cruz, Richard Dazeley, and Peter Vamplew, 'Memory-based explainable reinforcement learning', in *Australasian Joint Conference on Artificial Intelligence*, pp. 66–77. Springer, (2019).

[8] Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine, 'D4rl: Datasets for deep data-driven reinforcement learning', *arXiv preprint arXiv:2004.07219*, (2020).

[9] Scott Fujimoto and Shixiang Shane Gu, 'A minimalist approach to offline reinforcement learning', *Advances in neural information processing systems*, **34**, 20132–20145, (2021).

[10] Scott Fujimoto, David Meger, and Doina Precup, 'Off-policy deep reinforcement learning without exploration', in *International conference on machine learning*, pp. 2052–2062. PMLR, (2019).

[11] Wonjoon Goo and Scott Niekum, 'You only evaluate once: a simple baseline algorithm for offline rl', in *Conference on Robot Learning*, pp. 1543–1553. PMLR, (2022).

[12] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine, 'Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor', in *International conference on machine learning*, pp. 1861–1870. PMLR, (2018).

[13] Hao Hu, Jianing Ye, Guangxiang Zhu, Zhizhou Ren, and Chongjie Zhang, 'Generalizable episodic memory for deep reinforcement learning', *arXiv preprint arXiv:2103.06469*, (2021).

[14] Md Al-Masrur Khan, Md Rashed Jaowad Khan, Abul Tooshil, Niloy Sikder, MA Parvez Mahmud, Abbas Z Kouzani, and Abdullah-Al Nahid, 'A systematic review on reinforcement learning-based robotics within the last decade', *IEEE Access*, **8**, 176598–176623, (2020).

[15] Rahul Kidambi, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims, 'Morel: Model-based offline reinforcement learning', *Advances in neural information processing systems*, **33**, 21810–21823, (2020).

[16] Ilya Kostrikov, Rob Fergus, Jonathan Tompson, and Ofir Nachum, 'Offline reinforcement learning with fisher divergence critic regularization', in *International Conference on Machine Learning*, pp. 5774–5783. PMLR, (2021).

[17] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine, 'Conservative q-learning for offline reinforcement learning', *Advances in Neural Information Processing Systems*, **33**, 1179–1191, (2020).

[18] Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and F Huang, 'A tutorial on energy-based learning', *Predicting structured data*, **1**(0), (2006).

[19] Jinning Li, Chen Tang, Masayoshi Tomizuka, and Wei Zhan, 'Dealing with the unknown: Pessimistic offline reinforcement learning', in *Conference on Robot Learning*, pp. 1455–1464. PMLR, (2022).

[20] Minghuan Liu, Tairan He, Minkai Xu, and Weinan Zhang, 'Energy-based imitation learning', *arXiv preprint arXiv:2004.09395*, (2020).

[21] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li, 'Energy-based out-of-distribution detection', *Advances in Neural Information Processing Systems*, **33**, 21464–21475, (2020).

[22] Xiaoteng Ma, Yiqin Yang, Hao Hu, Qihan Liu, Jun Yang, Chongjie Zhang, Qianchuan Zhao, and Bin Liang, 'Offline reinforcement learning with value-based episodic memory', *arXiv preprint arXiv:2110.09796*, (2021).

[23] Tatsuya Matsushima, Hiroki Furuta, Yutaka Matsuo, Ofir Nachum, and Shixiang Gu, 'Deployment-efficient reinforcement learning via model-based offline optimization', *arXiv preprint arXiv:2006.03647*, (2020).

[24] Thomas M Moerland, Joost Broekens, and Catholijn M Jonker, 'Model-based reinforcement learning: A survey', *arXiv preprint arXiv:2006.16712*, (2020).

[25] Dhruv Ramani, 'A short survey on memory based reinforcement learning', *arXiv preprint arXiv:1904.06736*, (2019).

[26] Shideh Rezaeifar, Robert Dadashi, Nino Vieillard, Léonard Hussenot, Olivier Bachem, Olivier Pietquin, and Matthieu Geist, 'Offline reinforcement learning as anti-exploration', in *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8106–8114, (2022).

[27] Marc Rigter, Bruno Lacerda, and Nick Hawes, 'Rambo-rl: Robust adversarial model-based offline reinforcement learning', *arXiv preprint arXiv:2204.12581*, (2022).

[28] Julian Schrittwieser, Thomas Hubert, Amol Mandhane, Mohammadamin Barekatain, Ioannis Antonoglou, and David Silver, 'Online and offline reinforcement learning by planning with a learned model', *Advances in Neural Information Processing Systems*, **34**, 27580–27591, (2021).

[29] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, and Adrian Bolton, 'Mastering the game of go without human knowledge', *nature*, **550**(7676), 354–359, (2017).

[30] David Roger Smart, *Fixed point theorems*, volume 66, Cup Archive, 1980.

[31] DiJia Su, Jason D Lee, John M Mulvey, and H Vincent Poor, 'Musbo: Model-based uncertainty regularized and sample efficient batch optimization for deployment constrained reinforcement learning', *arXiv preprint arXiv:2102.11448*, (2021).

[32] Phillip Swazinna, Steffen Udluft, Daniel Hein, and Thomas Runkler, 'Comparing model-free and model-based algorithms for offline reinforcement learning', *arXiv preprint arXiv:2201.05433*, (2022).

[33] Emanuel Todorov, Tom Erez, and Yuval Tassa, 'Mujoco: A physics engine for model-based control', in *2012 IEEE/RSJ international conference on intelligent robots and systems*, pp. 5026–5033. IEEE, (2012).

[34] Faraz Torabi, Garrett Warnell, and Peter Stone, 'Behavioral cloning from observation', *arXiv preprint arXiv:1805.01954*, (2018).

[35] Brandon Trabucco, Aviral Kumar, Xinyang Geng, and Sergey Levine, 'Conservative objective models for effective offline model-based optimization', in *International Conference on Machine Learning*, pp. 10358–10368. PMLR, (2021).

[36] Jianhao Wang, Wenzhe Li, Haozhe Jiang, Guangxiang Zhu, Siyuan Li, and Chongjie Zhang, 'Offline reinforcement learning with reverse model-based imagination', *Advances in Neural Information Processing Systems*, **34**, 29420–29432, (2021).

[37] Haoran Xu, Xianyuan Zhan, and Xiangyu Zhu, 'Constraints penalized q-learning for safe offline reinforcement learning', in *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8753–8760, (2022).

[38] Tianhe Yu, Aviral Kumar, Rafael Rafailov, Aravind Rajeswaran, Sergey Levine, and Chelsea Finn, 'Combo: Conservative offline model-based policy optimization', *Advances in neural information processing systems*, **34**, 28954–28967, (2021).

[39] Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma, 'Mopo: Model-based offline policy optimization', *Advances in Neural Information Processing Systems*, **33**, 14129–14142, (2020).