Conservative Dual Policy Optimization for Efficient Model-Based Reinforcement Learning

Shenao Zhang
Georgia Institute of Technology
Atlanta, GA 30332
shenao@gatech.edu

Abstract

Provably efficient Model-Based Reinforcement Learning (MBRL) based on optimism or posterior sampling (PSRL) is ensured to attain the global optimality asymptotically by introducing the complexity measure of the model. However, the complexity might grow exponentially for the simplest nonlinear models, where global convergence is impossible within finite iterations. When the model suffers a large generalization error, which is quantitatively measured by the model complexity, the uncertainty can be large. The sampled model that current policy is greedily optimized upon will thus be unsettled, resulting in aggressive policy updates and over-exploration. In this work, we propose *Conservative Dual Policy Optimization* (CDPO) that involves a *Referential Update* and a *Conservative Update*. The policy is first optimized under a reference model, which imitates the mechanism of PSRL while offering more stability. A conservative range of randomness is guaranteed by maximizing the expectation of model value. Without harmful sampling procedures, CDPO can still achieve the same regret as PSRL. More importantly, CDPO enjoys monotonic policy improvement and global optimality simultaneously. Empirical results also validate the exploration efficiency of CDPO.

1 Introduction

Model-Based Reinforcement Learning (MBRL) involves acquiring a model by interacting with the environment and learning to make the optimal decision using the model [55, 32]. MBRL is appealing due to its significantly reduced sample complexity compared to its model-free counterparts. However, greedy model exploitation that assumes the model is sufficiently accurate lacks guarantees for global optimality. The policies can be suboptimal and get stuck at local maxima even in simple tasks [10].

As such, several provably-efficient MBRL algorithms have been proposed. Based on the principle of *optimism in the face of uncertainty* (OFU) [56, 49, 10], OFU-RL achieves the global optimality by ensuring that the optimistically biased value is close to the real value in the long run. Based on Thompson Sampling [62], Posterior Sampling RL (PSRL) [57, 42, 43] explores by greedily optimizing the policy in an MDP which is sampled from the posterior distribution over MDPs. Beyond finite MDPs, to obtain a general bound that permits sample efficiency in various cases, we need to introduce additional complexity measure. For example, [49, 43] provide an $\widetilde{O}(\sqrt{d_E T})$ regret for both OFU and PSRL with eluder dimension d_E capturing how effectively the model generalizes. However, it is recently shown [13, 33] that the eluder dimension for even the simplest nonlinear models cannot be polynomially bounded. The effectiveness of the algorithms will thus be crippled.

The underlying reasons for such ineffectiveness are the aggressive policy updates and the over-exploration issue. Specifically, when a nonlinear model is used to fit complex transition functions, its generalizability will be poor compared to simple linear problems. If a random model is selected from the large hypothesis, e.g., optimistically chosen or sampled from the posterior, it is “unsettled".

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

In other words, the selected model can change dramatically between successive iterations. Policy updates under this model will also be aggressive and thus cause value degradation. What's worse, large epistemic uncertainty results in an unrealistic model, which drives agents for uninformative exploration. An exploration step can only eliminate an exponentially small portion of the hypothesis.

In this work, we present *Conservative Dual Policy Optimization* (CDPO), a simple yet provable MBRL algorithm. As the sampling process in PSRL harms policy updates due to the unsettled model during training, we propose the *Referential Update* that greedily optimizes an intermediate policy under a *reference model*. It mimics the sampling-then-optimization procedure in PSRL but offers more stability since we are free to set a steady reference model. We show that even without a sampling procedure, CDPO can match the expected regret of PSRL up to constant factors for any proper reference model, e.g., the least squares estimate where the confidence set is centered at. The *Conservative Update* step then follows to encourage exploration within a reasonable range. Specifically, the objective of a reactive policy is to maximize the *expectation* of model value, instead of a single model's value. These two steps are performed in an iterative manner in CDPO.

Theoretically, we show the statistical equivalence between CDPO and PSRL with the same order of expected regret. Additionally, we give the iterative policy improvement bound of CDPO, which guarantees monotonic improvement under mild conditions. We also establish the sublinear regret of CDPO, which permits its global optimality equipped with any model function class that has a bounded complexity measure. To our knowledge, the proposed framework is the first that *simultaneously* enjoys global optimality and iterative policy improvement. Experimental results verify the existence of the over-exploration issue and demonstrate the practical benefit of CDPO.

2 Background

2.1 Model-Based Reinforcement Learning

We consider the problem of learning to optimize an infinite-horizon γ-discounted Markov Decision Process (MDP) over repeated episodes of interaction. Denote the state space and action space as $\mathcal{S}$ and $\mathcal{A}$, respectively. When taking action $a \in \mathcal{A}$ at state $s \in \mathcal{S}$, the agent receives reward $r(s, a)$ and the environment transits into a new state according to probability $s' \sim f^*(\cdot|s, a)$. Here, f^* is a dirac measure for deterministic dynamics and is a probability distribution for probabilistic dynamics.

In model-based RL, the true dynamical model f^* is unknown and needs to be learned using the collected data through episodic (or iterative) interaction. The history data up to iteration t then forms $\mathcal{H}_t = \{\{s_{h,i}, a_{h,i}, s_{h+1,i}\}_{h=0}^{H-1}\}_{i=1}^{t-1}$, where H is the actual timesteps agents run in an episode. The posterior distribution of the dynamics model is estimated as $\phi(\cdot|\mathcal{H}_t)$. Alternatively, the frequentist model of the mean and uncertainty can also be estimated. Specifically, consider the model function class $\mathcal{F} = \{f : \mathcal{S} \times \mathcal{A} \to \mathcal{S}\}$ with size $|\mathcal{F}|$, which contains the real model $f^* \in \mathcal{F}$. The confidence set (or model hypothesis set) $\mathcal{F}_t \subset \mathcal{F}$ is introduced to represent the range of dynamics that is statistically plausible [49, 43, 10]. To ensure that $f^* \in \mathcal{F}_t$ with high probability, one way is to construct the confidence set as $\mathcal{F}_t := \{f \in \mathcal{F} \,|\, \|f - \widehat{f}_t^{LS}\|_{2,E_t} \leq \sqrt{\beta_t}\}$. Here, β_t is an appropriately chosen confidence parameter (via concentration inequality), the cumulative empirical 2-norm is defined by $\|g\|_{2,E_t}^2 := \sum_{i=1}^{t-1} \|g(x_i)\|_2^2$. The least squares estimate is

$$\widehat{f}_t^{LS} := \operatorname*{argmin}_{f \in \mathcal{F}} \sum_{(s,a,s') \in \mathcal{H}_t} \|f(s, a) - s'\|_2^2. \tag{2.1}$$

Denote the state and state-action value function associated with π on model f by $V_\pi^f : \mathcal{S} \to \mathbb{R}$ and $Q_\pi^f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, respectively, which are defined as

$$V_\pi^f(s) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \,\middle|\, s_0 = s, \pi, f\right], \; Q_\pi^f(s, a) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \,\middle|\, s_0 = s, a_0 = a, \pi, f\right].$$

The objective of RL is to learn a policy $\pi^* = \operatorname{argmax}_\pi J(\pi)$ that maximizes the expected return $J(\pi)$. Denote the initial state distribution as ζ. Under policy π, the state visitation measure $\nu_\pi(s)$ over $\mathcal{S}$ and the state-action visitation measure $\rho_\pi(s, a)$ over $\mathcal{S} \times \mathcal{A}$ in the true MDP are defined as

$$\nu_\pi(s) = (1 - \gamma) \cdot \sum_{h=0}^{\infty} \gamma^h \cdot \mathbb{P}(s_h = s), \quad \rho_\pi(s, a) = (1 - \gamma) \cdot \sum_{h=0}^{\infty} \gamma^h \cdot \mathbb{P}(s_h = s, a_h = a), \tag{2.2}$$

where $s_0 \sim \zeta$, $a_h \sim \pi(\cdot|s_h)$ and $s_{h+1} \sim f^*(\cdot|s_h, a_h)$. The objective $J(\pi)$ is then

$$J(\pi) = \mathbb{E}_{s_0\sim\zeta}[V_\pi^{f^*}(s_0)] = \mathbb{E}_{(s,a)\sim\rho_\pi}[r(s,a)] \tag{2.3}$$

2.2 Cumulative Regret and Asymptotic Optimality

A common criterion to evaluate RL algorithms is the cumulative regret, defined as the cumulative performance discrepancy between policy π_t at each iteration t and the optimal policy π^* over the run of the algorithm. The (cumulative) regret up to iteration T is defined as:

$$\mathrm{Regret}(T, \pi, f^*) := \sum_{t=1}^{T} \int_{s\in\mathcal{S}} \zeta(s)(V_{\pi^*}^{f^*}(s) - V_{\pi_t}^{f^*}(s)), \tag{2.4}$$

In the Bayesian view, the model f^*, the learning policy π, and the regret are random variables that must be learned from the gathered data. The Bayesian expected regret is defined as:

$$\mathrm{BayesRegret}(T, \pi, \phi) := \mathbb{E}\left[\mathrm{Regret}(T, \pi, f^*) \mid f^* \sim \phi\right]. \tag{2.5}$$

One way to prove the asymptotic optimality is to show that the (expected) regret is sublinear in T, so that π_t converges to π^* within sufficient iterations. To obtain the regret bound, the *width of confidence set* $\omega_t(s,a)$ is introduced to represent the maximum deviation between any two members in $\mathcal{F}_t$:

$$\omega_t(s,a) = \sup_{\underline{f},\overline{f}\sim\mathcal{F}_t} \|\overline{f}(s,a) - \underline{f}(s,a)\|_2. \tag{2.6}$$

3 Provable Model-Based Reinforcement Learning

In this section, we analyze the central ideas and limitations of greedy algorithms as well as two popular theoretically justified frameworks: optimistic algorithms and posterior sampling algorithms.

Greedy Model Exploitation. Before introducing provable algorithms, we first analyze greedy model-based algorithms. In this framework, the agent takes actions assuming that the fitted model sufficiently accurately resembles the real MDP. Algorithms that lie in this category can be roughly divided into two groups: model-based planning and model-augmented policy optimization. For instance, Dyna agents [61, 20, 17] optimize policies using model-free learners with model-generated data. The model can also be exploited in first-order gradient estimators [18, 12, 9] or value expansion [15, 6]. On the other hand, model-based planning, or model-predictive control (MPC) [40, 41], directly generates optimal action sequences under the model in a receding horizon fashion.

However, greedily exploiting the model without *deep exploration* [45] will lead to suboptimal performance. The resulting policy can suffer from premature convergence, leaving the potentially high-reward region unexplored. Since the transition data is generated by the agent taking actions in the real MDP, the dual effect [4, 27] that current action influences *both* the next state and the model uncertainty is not considered by greedy model-based algorithms.

Optimism in the Face of Uncertainty. A common provable exploration mechanism is to adopt the principle of *optimism in the face of uncertainty* (OFU) [56, 49, 10]. With OFU, the agent assigns to its policy an optimistically biased estimate of virtual value by *jointly* optimizing over the policies and models inside the confidence set $\mathcal{F}_t$. At iteration t, the OFU-RL policy π_t is defined as:

$$\pi_t = \operatorname*{argmax}_{\pi} \max_{f_t\in\mathcal{F}_t} V_\pi^{f_t}. \tag{3.1}$$

Most asymptotic analyses of optimistic RL algorithms can be abstracted as showing two properties: the virtual value V_π^f is sufficiently high, and it is close to the real value $V_\pi^{f^*}$ in the long run. However, in complex environments where the generalizability of nonlinear models is limited, large epistemic uncertainty will result in an unrealistically large optimistic return that drives agents for uninformative exploration. What's worse, such suboptimal exploration steps eliminate only a small portion of the model hypothesis [13], leading to a slow converging process and suboptimal practical performance.

Posterior Sampling Reinforcement Learning. An alternative exploration mechanism is based on *Thompson Sampling* (TS) [62, 52], which involves selecting the maximizing action from a statistically

plausibly set of action values. These values can be associated with the MDP sampled from its posterior distribution, thus giving its name *posterior sampling for reinforcement learning* (PSRL) [57, 42, 43]. The algorithm begins with a prior distribution of f^*. At each iteration t, a model f_t is sampled from the posterior $\phi(\cdot|\mathcal{H}_t)$, and π_t is updated to be optimal under f_t:

$$f_t \sim \phi(\cdot|\mathcal{H}_t),\ \pi_t = \operatorname*{argmax}_{\pi} V_{\pi}^{f_t}. \tag{3.2}$$

The insight is to keep away from actions that are unlikely to be optimal in the real MDP. Exploration is guaranteed by the randomness in the sampling procedure. Unfortunately, executing actions that are optimally associated with a single sampled model can cause similar over-exploration issues [52, 51]. Specifically, an imperfect model sampled from the large hypothesis can cause aggressive policy updates and value degradation between successive iterations. The suboptimality degree of the resulting policies depends on the epistemic model uncertainty. Besides, executing π_t is not intended to offer performance improvement for follow-up policy learning, but only to narrow down the model uncertainty. However, this elimination procedure will be slow when the model suffers a large generalization error, which is quantitatively formulated in the model complexity measure below.

Complexity Measure and Generalization Bounds. In RL, we seek to have the sample complexity for finding a near-optimal policy or estimating an accurate value function. When given access to a generative model (i.e., an abstract sampling model) in finite MDPs, it is known that the (minimax) number of transitions the agent needs to observe can be sublinear in the model size, i.e. smaller than $O(|\mathcal{S}|^2|\mathcal{A}|)$. Beyond finite MDPs where the number of states is large (or countably or uncountably infinite), we are interested in the learnability or generalization of RL. Unfortunately, it is impossible for agnostic reinforcement learning that finds the best hypothesis in some given policy, value, or model hypothesis class: the number of needed samples depends exponentially on the problem horizon [24]. Despite of the structural assumptions, e.g. linear MDPs [66, 22, 65] or low-rank MDPs [21, 38], we focus on the generalization bounds that can cover various cases. This can be done with additional complexity measure, e.g. eluder dimension [49], witness rank [60], or bilinear rank [14].

By introducing the eluder dimension d_E [49], previous work [43, 44] established regret $\widetilde{O}(\sqrt{d_E T})$ for both OFU-RL and PSRL. Intuitively, the eluder dimension captures how effectively the model learned from observed data can extrapolate to future data, and permits sample efficiency in various (linear) cases. Nevertheless, it is shown in [13, 33] that even the simplest nonlinear models do not have a polynomially-bounded eluder dimension. The following result is from Thm. 5.2 in Dong et al. [13] and similar results are also established in [33].

Theorem 3.1 (Eluder Dimension of Nonlinear Models [13])**.** The eluder dimension $dim_E(\mathcal{F}, \varepsilon)$ (c.f. Definition 5.6) of one-layer ReLU neural networks is at least $\Omega(\varepsilon^{-(d-1)})$, where d is the state-action dimension, i.e. $(s, a) \in \mathbb{R}^d$. With more layers, the requirement of ReLU activation can be relaxed.

As a result, additional complexity is hidden in the eluder dimension, e.g. when we choose $\varepsilon = T^{-1}$, regret $\widetilde{O}(\sqrt{d_E T})$ contains $d_E = \Omega(T^{d-1})$ and is no longer sublinear in T. In this case, previous provable exploration mechanisms will lose the desired property of global optimality and sample efficiency, which is the underlying reason for the over-exploration issue.

4 Conservative Dual Policy Optimization

When using nonlinear models, e.g. neural networks, the over-exploration issue causes unfavorable performance in practice, in terms of slow convergence and suboptimal asymptotic values. To tackle this challenge, the key is to abandon the sampling process and have guarantees *during* training.

In this regard, we propose *Conservative Dual Policy Optimization* (CDPO) that is simple yet provably efficient. By optimizing the policy following two successive update procedures iteratively, CDPO simultaneously enjoys monotonic policy value improvement and global optimality properties.

4.1 CDPO Framework

To begin with, consider the problem of maximizing the expected value, $\pi_t = \operatorname{argmax}_{\pi} \mathbb{E}[V_{\pi}^{f^*} \mid \mathcal{H}_t]$, where $\mathbb{E}[V^{f^*} \mid \mathcal{H}_t]$ denotes the expected values over the posterior. Obviously, we have the expected value improvement guarantee $\mathbb{E}[V_{\pi_t}^{f^*} \mid \mathcal{H}_t] \geq \mathbb{E}[V_{\pi_{t-1}}^{f^*} \mid \mathcal{H}_t]$. We can also perform expected value

maximization in a trust-region to guarantee iterative improvement under any f^*. However, such updates will lose the desired global convergence guarantee and may get stuck at local maxima even with linear models. For this reason, we propose a dual procedure of policy optimization.

Referential Update. The first update step returns an intermediate policy, denoted as q_t. This step is a greedy one in the sense that q_t is optimal with respect to the value of a *single* model $\widetilde{f}_t$, which we call a *reference model*. Selecting a reference model and optimizing a policy w.r.t. it imitates the sampling-optimization procedure of PSRL. We will show in Section 5.1 that if we pose the constraint $\widetilde{f}_t \in \mathcal{F}_t$, then CDPO achieves the same expected regret as PSRL, which implies global optimality.

More importantly, policy optimization under $\widetilde{f}_t$ is more stable and can avoid the over-exploration issue in PSRL since we are free to set it as a steady reference between successive iterations. For example, we fix the reference model $\widetilde{f}_t$ as the least squares estimate $\widehat{f}_t^{LS}$ defined in (2.1), instead of a random model *sampled* from the large hypothesis that causes aggressive policy update. This gives us:

$$\text{Referential Update (with LS Reference):} \qquad q_t = \operatorname*{argmax}_q V_q^{\widehat{f}_t^{LS}}. \tag{4.1}$$

Constrained Conservative Update. The conservative update then follows as the second stage of CDPO, which takes input q_t and returns the reactive policy π_{t+1}:

$$\text{Conservative Update: } \pi_t = \operatorname*{argmax}_\pi \mathbb{E}\big[V_\pi^{f_t} \,\big|\, \mathcal{H}_t\big], \text{ s.t. } \mathbb{E}_{s\sim\nu_{q_t}}\Big[D_{\mathrm{TV}}\big(\pi_t(\cdot|s), q_t(\cdot|s)\big)\Big] \le \eta, \tag{4.2}$$

where $D_{\mathrm{TV}}(\cdot,\cdot)$ stands for the total variation distance and η is the hyperparameter that characterizes the trust-region constraint and controls the degree of exploration.

Compared with OFU-RL and PSRL, the above exploration and policy updates are conservative since the policy maximizes the *expectation* of the model value, instead of a *single* model's value (i.e. the optimistic model in OFU-RL and the sampled model in PSRL). The conservative update (4.2) avoids the pitfalls when the optimistic model or the posterior sampled model suffers large bias, which leads to aggressive policy updates and over-exploration during training. Notably, the term *conservative* in our work differs from previous use, e.g. Conservative Policy Iteration [23, 53]. While the latter refers to policy updates with constraints, ours is to emphasize the conservative range of randomness and the reduction of unnecessary over-exploration by shelving the sampling process.

In our analysis, we follow previous work [43, 59, 10, 35] and assume access to a policy optimization oracle. In practice, the problem of finding an optimal policy under a given model can be approximately solved by model-based solvers listed below. More fine-grained analysis can be obtained by applying off-the-shelf results established for policy gradient or MPC for specific policy or model function classes. This, however, is beyond the scope of this paper.

4.2 Practical Algorithm

The pseudocode of CDPO is in Alg. 1. The model-based solver $\texttt{MBPO}(\pi, f, \mathcal{J})$ outputs the policy (q_t or π_t) that optimizes the objective $\mathcal{J}$ with access to model f. Several different types of solvers can be leveraged, e.g., model-augmented model-free policy optimization such as Dyna [61], model-based reparameterization gradient [18, 9], or model-predictive control [63]. Details of different optimization choices can be found in Appendix E. In experiments, we use Dyna and MPC solvers.

Algorithm 1 Practical CDPO Algorithm

Input: Prior ϕ, model-based policy optimization solver $\texttt{MBPO}(\pi, f, \mathcal{J})$.

for iteration $t = 1, ..., T$ **do**
  $q_t \leftarrow \texttt{MBPO}(\cdot, \widehat{f}_t^{LS}, (4.1))$
  Sample N models $\{f_{t,n}\}_{n=1}^N$
  $\pi_t \leftarrow \texttt{MBPO}(q_t, \{f_{t,n}\}_{n=1}^N, (4.2))$
  Execute π_t in the real MDP
  Update $\mathcal{H}_{t+1} = \mathcal{H}_t \cup \{s_{h,t}, a_{h,t}, s_{h+1,t}\}_h$
  Update $\widehat{f}_{t+1}^{LS}$ and ϕ
end for
return policy π_T

With Pinsker's inequality, the total variation constraint in (4.2) is replaced by the KL divergence [53, 2] in experiments. We follow previous work [34] to use neural network ensembles [10, 25] for model estimation and use calibrations [29, 10] for accurate uncertainty measure.

5 Analysis

In this section, we first show the statistical equivalence between CDPO and PSRL in terms of the same BayesRegret bound. Then we give the iterative policy value bound with monotonic improvement. Finally, we prove the global convergence of CDPO. The missing proofs can be found in the Appendix.

5.1 Statistical Equivalence between CDPO and PSRL

We begin our analysis by highlighting the connection between CDPO and PSRL with the following theorem, from which we also show the role of the dual update procedure and the reference model.

Theorem 5.1 (CDPO Matches PSRL in BayesRegret)**.** Let π^{PSRL} be the policy of any posterior sampling algorithm for reinforcement learning optimized by (3.2). If the BayesRegret bound of π^{PSRL} satisfies that for any $T > 0$, $\text{BayesRegret}(T, \pi^{\text{PSRL}}, \phi) \leq \mathcal{D}$, then for all $T > 0$, we have for the CDPO policy π^{CDPO} that $\text{BayesRegret}(T, \pi^{\text{CDPO}}, \phi) \leq 3\mathcal{D}$.

Sketch proof. We first sketch the general strategy in the PSRL analysis. Recall the definition of the Bayesian expected regret $\text{BayesRegret}(T, \pi, \phi) := \mathbb{E}[\sum_{t=1}^{T} \mathfrak{R}_t]$, where $\mathfrak{R}_t = V_{\pi^*}^{f^*} - V_{\pi_t}^{f^*}$. PSRL breaks down $\mathfrak{R}_t$ by adding and subtracting $V_{\pi_{f_t}}^{f_t}$, the value of the *imagined* optimal policy π_{f_t} under a sampled model f_t, i.e. $\pi_{f_t} = \text{argmax}_\pi V_\pi^{f_t}$.

$$\text{PSRL:} \qquad \mathfrak{R}_t = V_{\pi^*}^{f^*} - V_{\pi_t}^{f^*} = V_{\pi^*}^{f^*} - V_{\pi_{f_t}}^{f^*} = V_{\pi^*}^{f^*} - V_{\pi_{f_t}}^{f_t} + V_{\pi_{f_t}}^{f_t} - V_{\pi_{f_t}}^{f^*}, \tag{5.1}$$

where the second equality follows from the definition of the PSRL policy. Following the law of total expectation and the Posterior Sampling Lemma (e.g. Lemma 1 in [42]), we have $\mathbb{E}[V_{\pi^*}^{f^*} - V_{\pi_{f_t}}^{f_t}] = 0$ by noting that f^* and f_t are identically distributed conditioned upon $\mathcal{H}_t$. Then we obtain

$$\begin{aligned}\text{BayesRegret}(T, \pi^{\text{PSRL}}, \phi) &= \sum_{t=1}^{T} \mathbb{E}[V_{\pi_{f_t}}^{f_t} - V_{\pi_{f_t}}^{f^*}] \leq \gamma \sum_{t=1}^{T} \mathbb{E}\Big[\mathbb{E}_\rho\big[L\|f_t(s_h, a_h) - f^*(s_h, a_h)\|_2\big]\Big] \\ &\leq \gamma \frac{L}{1-4\delta} \sum_{t=1}^{T} \mathbb{E}[\omega_t] + 4\gamma\delta T \leq \mathcal{D},\end{aligned} \tag{5.2}$$

where the first inequality follows from the simulation lemma under the L-Lipschitz value assumption [43]. The second inequality follows from the definition of ω_t in (2.6) and the construction of confidence set such that $\mathbb{P}(f^* \in \bigcap \mathcal{F}_t) \geq 1 - 2\delta$ and $\mathbb{P}(f_t \in \bigcap \mathcal{F}_t, f^* \in \bigcap \mathcal{F}_t) \geq 1 - 4\delta$ via a union bound. As more data is collected, the model uncertainty is reduced and the sum of confidence set width ω_t will be sublinear in T (c.f. Lemma B.5 and B.6), indicating sublinear regret.

When it comes to CDPO, we decompose the regret as

$$\text{CDPO:} \qquad \mathfrak{R}_t = V_{\pi^*}^{f^*} - V_{\pi_t}^{f^*} = V_{\pi^*}^{f^*} - V_{\pi_{f_t}}^{f_t} + V_{\pi_{f_t}}^{f_t} - V_{\pi_t}^{f_t} + V_{\pi_t}^{f_t} - V_{\pi_t}^{f^*}, \tag{5.3}$$

where the CDPO policy π_t is defined in (4.2). Since $\mathbb{E}[V_{\pi^*}^{f^*} - V_{\pi_{f_t}}^{f_t}] = 0$, we have

$$\begin{aligned}\text{BayesRegret}(T, \pi^{\text{CDPO}}, \phi) &= \sum_{t=1}^{T} \mathbb{E}[V_{\pi_{f_t}}^{f_t} - V_{\pi_t}^{f_t} + V_{\pi_t}^{f_t} - V_{\pi_t}^{f^*}] \\ &\leq \sum_{t=1}^{T} \mathbb{E}[V_{\pi_{f_t}}^{f_t} - V_{\pi_{f_t}}^{\widetilde{f}_t} + V_{q_t}^{\widetilde{f}_t} - V_{q_t}^{f_t} + V_{\pi_t}^{f_t} - V_{\pi_t}^{f^*}] \leq \frac{L}{1-4\delta} \sum_{t=1}^{T} 3\mathbb{E}[\omega_t] + 8\gamma\delta T \leq 3\mathcal{D},\end{aligned} \tag{5.4}$$

where the first inequality follows from the greediness of q_t and π_t in the dual update steps, i.e., $V_{\pi_{f_t}}^{\widetilde{f}_t} \leq V_{q_t}^{\widetilde{f}_t}$ for any π_{f_t} as well as $\mathbb{E}[V_{\pi_t}^{f_t}] \geq \mathbb{E}[V_{q_t}^{f_t}]$. The $8\delta T$ term is introduced since $\widetilde{f}_t \in \mathcal{F}_t$ and $\mathbb{P}(f_t \in \bigcap \mathcal{F}_t, \widetilde{f}_t \in \bigcap \mathcal{F}_t) \geq 1 - 2\delta$. □

Theorem 5.1 indicates that although CDPO performs conservative updates and abandons the sampling process, it matches the statistical efficiency of PSRL up to constant factors.

The importance of the reference model and the dual procedure is also reflected in the proof. The referential update builds the bridge between $V_{\pi_{f_t}}^{f_t}$ and $V_{\pi_t}^{f_t}$. Policy optimization under the reference model mimics the sampling-then-optimization procedure of PSRL while offering more stability when the reference is steady, e.g., the least squares estimate we use. We formalize this idea below.

5.2 CDPO Policy Iterative Improvement

One motivation for the conservative update is that it maximizes (thus improves) the expected value over the posterior. In this section, we are interested in the policy value improvement under any unknown f^*. Namely, we seek to have the iterative improvement bound $J(\pi_t) - J(\pi_{t-1})$, where the true objective J is defined in (2.3).

We impose the following regularity conditions on the underlying MDP transition and the state-action visitation.

Assumption 5.2 (Regularity Condition on MDP Transition)**.** Assume that the MDP transition function $f^* : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ is with additive σ-sub-Gaussian noise and bounded norm, i.e., $\|s\|_2 \leq C$.

Assumption 5.3 (Regularity Condition on State-Action Visitation)**.** We assume that there exists $\kappa > 0$ such that for any policy π_t, $t \in [1, T]$,

$$\left\{ \mathbb{E}_{\rho_{\pi_t}} \left[\left(\frac{d\rho_{q_{t+1}}}{d\rho_{\pi_t}}(s,a) \right)^2 \right] \right\}^{1/2} \leq \kappa, \tag{5.5}$$

where $d\rho_{q_{t+1}}/d\rho_{\pi_t}$ is the Radon-Nikodym derivative of $\rho_{q_{t+1}}$ with respect to ρ_{π_t}.

Theorem 5.4 (Policy Iterative Improvement)**.** Suppose we have $\|\widetilde{f}(\cdot,\cdot)\| \leq C$ for $\widetilde{f} \in \mathcal{F}$ where the model class $\mathcal{F}$ is finite. Define $\iota := \max_{s,a} |A_\pi^{f^*}(s,a)|$, where $A_\pi^{f^*}$ is the advantage function defined as $A_\pi^{f^*}(s,a) := Q_\pi^{f^*}(s,a) - V_\pi^{f^*}(s)$. With probability at least $1-\delta$, the policy improvement between successive iterations is bounded by

$$J(\pi_t) - J(\pi_{t-1}) \geq \Delta(t) - (1+\kappa) \cdot \frac{22\gamma C^2 \ln(|\mathcal{F}|/\delta)}{(1-\gamma)H} - \frac{2\eta\iota}{1-\gamma}, \tag{5.6}$$

where $\Delta(t) := \mathbb{E}_{s\sim\zeta}\left[V_{q_t}^{\widetilde{f}_t}(s) - V_{q_{t-1}}^{\widetilde{f}_t}(s)\right] \geq 0$ due to the greediness of q_t.

The above theorem provides the iterative improvement bound following the CDPO algorithm. When H is large enough, the policy value improvement is at least $\Delta(t)$ by choosing a properly small η.

In particular, the first term $\Delta(t)$ characterizes the policy improvement brought by the greedy exploitation in (4.1), and $\Delta(t) \geq 0$ since q_t is optimal under the reference model $\widetilde{f}_t$. The second term in (5.6) accounts for the generalization error of least square methods. Specifically, model $\widetilde{f}_t = \widehat{f}_t^{LS} \in \mathcal{F}_t$ is trained to fit the history samples. However, we seek to have the model error bound over the state-action visitation measure, which requires the deviation from the empirical mean to its expectation using Bernstein's inequality and union bound. Finally, the trust-region constraint in (4.2) brings the $4\eta\alpha/(1-\gamma)$ term, which reduces to zero if η is small. This makes intuitive sense as η controls the degree of conservative exploration.

5.3 Global Optimality of CDPO

We now analyze the global optimality of CDPO by studying its expected regret. As discussed in Section 3, agnostic reinforcement learning is impossible. Without structural assumptions, additional complexity measure is required for a generalization bound beyond finite settings. For this reason, we adopt the notation of *eluder dimension* [49, 43], defined as follows:

Definition 5.5 ($(\mathcal{F},\varepsilon)$-Dependence)**.** If we say $(s,a) \in \mathcal{S}\times\mathcal{A}$ is $(\mathcal{F},\varepsilon)$-dependent on $\{(s_i,a_i)\}_{i=1}^n \subseteq \mathcal{S}\times\mathcal{A}$, then

$$\forall f_1, f_2 \in \mathcal{F}, \sum_{i=1}^n \left\| f_1(s_i,a_i) - f_2(s_i,a_i) \right\|_2^2 \leq \varepsilon^2 \Rightarrow \left\| f_1(s,a) - f_2(s,a) \right\|_2 \leq \varepsilon.$$

Conversely, $(s,a) \in \mathcal{S}\times\mathcal{A}$ is $(\mathcal{F},\varepsilon)$-independent of $\{(s_i,a_i)\}_{i=1}^n$ if and only if it does not satisfy the definition for dependence.

Definition 5.6 (Eluder Dimension)**.** The eluder dimension $dim_E(\mathcal{F},\varepsilon)$ is the length of the longest possible sequence of elements in $\mathcal{S}\times\mathcal{A}$ such that for some $\varepsilon' \geq \varepsilon$, every element is $(\mathcal{F},\varepsilon')$-independent of its predecessors.

We make the following assumption on the Lipschitz continuity of the value function.

Assumption 5.7 (Lipschitz Continuous Value)**.** At iteration t, assume the value function $V_\pi^{f_t}$ for any policy π is Lipschitz continuous in the sense that $|V_\pi^{f_t}(s_1) - V_\pi^{f_t}(s_2)| \leq L_t \|s_1 - s_2\|_2$.

Notably, Assumption 5.7 holds under certain regularity conditions of the MDP, e.g. when the transition and rewards are Lipschitz continuous [5, 47]. Under this assumption, many RL settings can be satisfied [13], e.g., nonlinear models with stochastic Lipschitz policies and Lipschitz reward models, and is thus adopted by various model-based RL work [35, 7, 13].

We now study the global optimality of CDPO by the following expected regret theorem, which can be seen as a direct consequence of Theorem 5.1 that states the statistical equivalence between CDPO and PSRL.

Theorem 5.8 (Expected Regret of CDPO)**.** Let $N(\mathcal{F}, \alpha, \|\cdot\|_2)$ be the α-covering number of $\mathcal{F}$. Denote $d_E := \dim_E(\mathcal{F}, T^{-1})$ for the eluder dimension of $\mathcal{F}$ at precision $1/T$. Under Assumption 5.2 and 5.7, the cumulative expected regret of CDPO in T iterations is bounded by

$$\text{BayesRegret}(T, \pi, \phi) \leq \frac{\gamma T(3T-5)L}{(T-1)(T-2)} \cdot \left(1 + \frac{1}{1-\gamma} C d_E + 4\sqrt{T d_E \beta}\right) + 4\gamma C, \tag{5.7}$$

where $\beta := 8\sigma^2 \log\Big(2N\big(\mathcal{F}, 1/(T^2), \|\cdot\|_2\big)T\Big) + 2\big(8C + \sqrt{8\sigma^2 \log(8T^3)}\,\big)/T$ and $L := \mathbb{E}[L_t]$.

Here, the covering number is introduced since we are considering $\mathcal{F}$ that may contain infinitely many functions, for which we cannot simply apply a union bound. Besides, β is the confidence parameter that contains f^* with high probability (via concentration inequality).

To clarify the asymptotics of the expected regret bound, we introduce another measure of dimensionality that captures the sensitivity of $\mathcal{F}$ to statistical overfitting.

Corollary 5.9 (Asymptotic Bound)**.** Define the Kolmogorov dimension w.r.t. function class $\mathcal{F}$ as

$$d_K = \dim_K(\mathcal{F}) := \limsup_{\alpha \downarrow 0} \frac{\log(N(\mathcal{F}, \alpha, \|\cdot\|_2))}{\log(1/\alpha)}.$$

Under the assumptions of Theorem 5.8 and by omitting terms logarithmic in T, the regret of CDPO is

$$\text{BayesRegret}(T, \pi, \phi) = \widetilde{O}(L\sigma\sqrt{d_K d_E T}). \tag{5.8}$$

The sublinear regret result permits the global optimality and sample efficiency for any model class with a reasonable complexity measure. Meanwhile, the iterative improvement theorem guarantees efficient exploration and good performance even when the model class is highly nonlinear.

6 Empirical Evaluation

6.1 Understanding Different Exploration Mechanisms

We first provide insights and evidence of why CDPO exploration can be more efficient in the tabular N-Chain MDPs, which have optimal *right* actions and suboptimal *left* actions at each of the N states. Settings and full results are provided in Appendix F.2. In Figure 1, we compare the posterior of CDPO and PSRL at the state that is the furthest away from the initial state, i.e. the state that is the hardest for the agents to reach and explore.

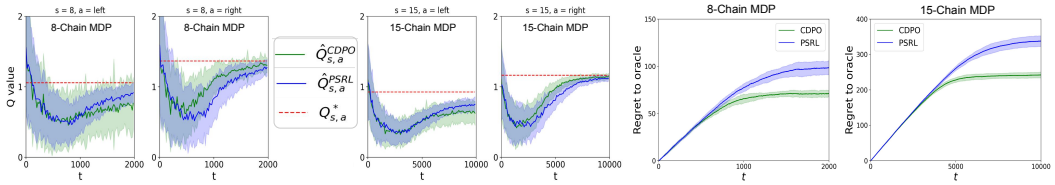


Figure 1: CDPO and PSRL posterior on an 8-Chain MDP and a 15-Chain MDP, where the *right* actions are optimal.

Figure 2: Regret curve of CDPO and PSRL when $N = 8$ and $N = 15$.

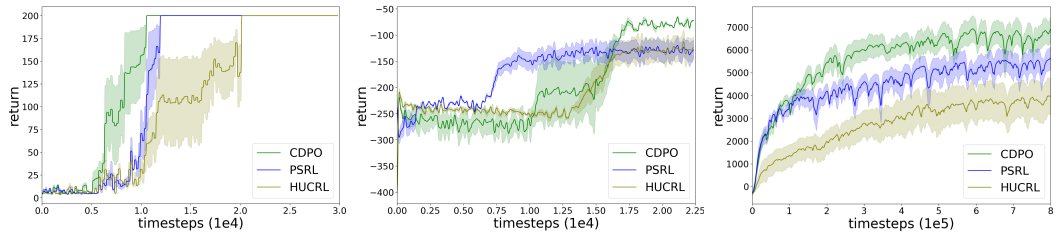
When training starts, both algorithms have a large variance of value estimation. However, as training progresses, CDPO gives more accurate and certain estimates, but *only* for the optimal *right* actions not

for the suboptimal *left* actions, while PSRL agents explore *both* directions. This verifies the potential over-exploration issue in PSRL: as long as the uncertainty contains unrealistically large values, PSRL agents can perform uninformative exploration by acting suboptimally according to an inaccurate *sampled* model. In contrast, CDPO replaces the sampled model with a stable mean estimate and cares about the *expected* value, thus avoiding such pitfalls. We see in Figure 2 that although CDPO has much larger uncertainty for the suboptimal *left* actions, its regret is lower.

6.2 Exploration Efficiency with Nonlinear Model Class

In finite MDPs, PSRL-style agents can specify and try every possible action to finally obtain an accurate high-confidence prediction. However, our discussion in Section 3 indicates that a similar over-exploration issue in more complex environments can lead to less informative exploration steps, which only eliminate an exponentially small portion of the uncertainty.

To see its impact on the training performance, we report the results of provable algorithms with nonlinear models on several MuJoCo tasks in Figure 3. For OFU-RL, we mainly evaluate HUCRL [10], a deep algorithm proposed to deal with the intractability of the joint optimization. We observe that all algorithms achieve asymptotic optimality in the inverted pendulum. Since the dimension of the pendulum task is low, learning an accurate (and thus generalizable) model poses no actual challenge. However, in higher dimensional tasks such as half-cheetah, CDPO achieves a higher asymptotic value with faster convergence. Implementation details and hyperparameters are provided in Appendix F.1.

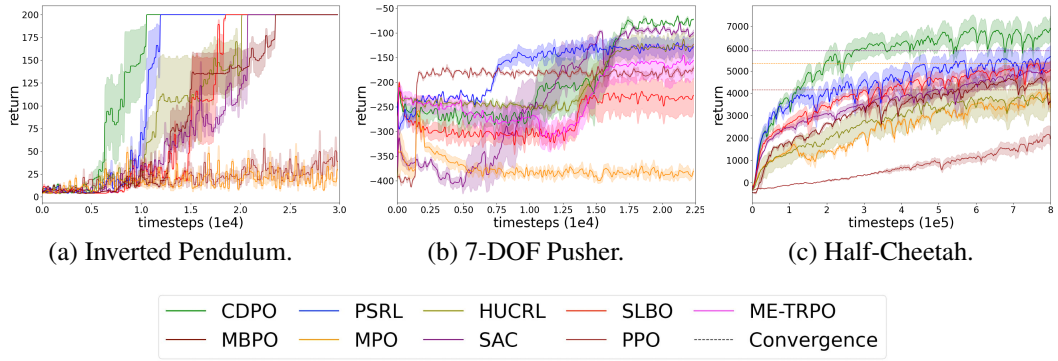


(a) Inverted Pendulum. (b) 7-DOF Pusher. (c) Half-Cheetah.

Figure 3: Performance of CDPO, PSRL, and HUCRL equipped with nonlinear models in several MuJoCo tasks: inverted pendulum swing-up, pusher goal-reaching, and half-cheetah locomotion.

6.3 Comparison with Prior RL Algorithms

We also examine a broader range of MBRL algorithms, including MBPO [20], SLBO [35], and ME-TRPO [30]. The model-free baselines include SAC [16], PPO [54], and MPO [2]. The results are shown in Figure 4. We observe that CDPO achieves competitive or higher asymptotic performance while requiring fewer samples compared to both the model-based and the model-free baselines.

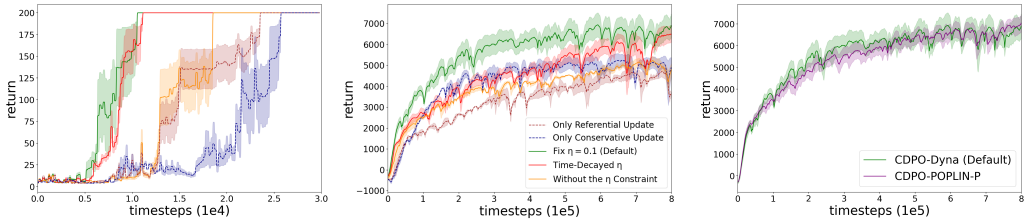


(a) Inverted Pendulum. (b) 7-DOF Pusher. (c) Half-Cheetah.

Figure 4: Comparison between CDPO and model-free, model-based RL baseline algorithms.

6.4 Ablation Study

We conduct ablation studies to provide a better understanding of the components in CDPO. One can observe from Figure 5 that the policies updated with only Referential Update or Conservative Update lag behind the dual framework. We also test the necessity and sensitivity of the constraint hyperparameter η. We see that a constant η and a time-decayed η achieve similar asymptotic values with a similar convergence rate, showing the robustness of CDPO. However, removing the constraint will lose the policy improvement guarantee, thus causing degradation. Ablation on different choices of `MBPO` solver (Dyna and POPLIN-P [63]) shows the generalizability of CDPO.



(a) Inverted Pendulum. (b) Half-Cheetah. (c) Half-Cheetah.

Figure 5: Ablation studies on the effect of the dual update steps and the trust-region constraint. The robustness and generalizability of the CDPO framework are demonstrated by the results of different choices of the constraint threshold and different solvers.

7 Conclusions & Future Work

In this work, we present *Conservative Dual Policy Optimization* (CDPO), a simple yet provable model-based algorithm. By iterative execution of the *Referential Update* and *Conservative Update*, CDPO explores within a reasonable range while avoiding aggressive policy update. Moreover, CDPO gets rid of the harmful sampling procedure in previous provable approaches. Instead, an intermediate policy is optimized under a stable *reference* model, and the agent conservatively explore the environment by maximizing the *expected* policy value. With the same order of regret as PSRL, the proposed algorithm can achieve global optimality while monotonically improving the policy. Considering our naive choice of the reference model, other more sophisticated designs should be a fruitful future direction. It will also be interesting to explore different choices of the `MBPO` solvers, which we would like to leave as future work.

References

[1] Yasin Abbasi-Yadkori and Csaba Szepesvári. Regret bounds for the adaptive control of linear quadratic systems. In *Proceedings of the 24th Annual Conference on Learning Theory*, pages 1–26. JMLR Workshop and Conference Proceedings, 2011.

[2] Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. Maximum a posteriori policy optimisation. *arXiv preprint arXiv:1806.06920*, 2018.

[3] Alekh Agarwal, Nan Jiang, Sham M Kakade, and Wen Sun. Reinforcement learning: Theory and algorithms. *CS Dept., UW Seattle, Seattle, WA, USA, Tech. Rep*, 2019.

[4] Yaakov Bar-Shalom and Edison Tse. Dual effect, certainty equivalence, and separation in stochastic control. *IEEE Transactions on Automatic Control*, 19(5):494–500, 1974.

[5] Osbert Bastani. Sample complexity of estimating the policy gradient for nearly deterministic dynamical systems. In *International Conference on Artificial Intelligence and Statistics*, pages 3858–3869. PMLR, 2020.

[6] Jacob Buckman, Danijar Hafner, George Tucker, Eugene Brevdo, and Honglak Lee. Sample-efficient reinforcement learning with stochastic ensemble value expansion. *arXiv preprint arXiv:1807.01675*, 2018.

[7] Sayak Ray Chowdhury and Aditya Gopalan. Online learning in kernelized markov decision processes. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3197–3205. PMLR, 2019.

[8] Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. *arXiv preprint arXiv:1805.12114*, 2018.

[9] Ignasi Clavera, Violet Fu, and Pieter Abbeel. Model-augmented actor-critic: Backpropagating through paths. *arXiv preprint arXiv:2005.08068*, 2020.

[10] Sebastian Curi, Felix Berkenkamp, and Andreas Krause. Efficient model-based reinforcement learning through optimistic policy search and planning. *arXiv preprint arXiv:2006.08684*, 2020.

[11] Richard Dearden, Nir Friedman, and Stuart Russell. Bayesian q-learning. In *Aaai/iaai*, pages 761–768, 1998.

[12] Marc Deisenroth and Carl E Rasmussen. Pilco: A model-based and data-efficient approach to policy search. In *Proceedings of the 28th International Conference on machine learning (ICML-11)*, pages 465–472. Citeseer, 2011.

[13] Kefan Dong, Jiaqi Yang, and Tengyu Ma. Provable model-based nonlinear bandit and reinforcement learning: Shelve optimism, embrace virtual curvature. *arXiv preprint arXiv:2102.04168*, 2021.

[14] Simon Du, Sham Kakade, Jason Lee, Shachar Lovett, Gaurav Mahajan, Wen Sun, and Ruosong Wang. Bilinear classes: A structural framework for provable generalization in rl. In *International Conference on Machine Learning*, pages 2826–2836. PMLR, 2021.

[15] Vladimir Feinberg, Alvin Wan, Ion Stoica, Michael I Jordan, Joseph E Gonzalez, and Sergey Levine. Model-based value estimation for efficient model-free reinforcement learning. *arXiv preprint arXiv:1803.00101*, 2018.

[16] 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*, pages 1861–1870. PMLR, 2018.

[17] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

[18] Nicolas Heess, Gregory Wayne, David Silver, Timothy Lillicrap, Tom Erez, and Yuval Tassa. Learning continuous control policies by stochastic value gradients. *Advances in neural information processing systems*, 28, 2015.

[19] Thomas Jaksch, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(4), 2010.

[20] Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. *arXiv preprint arXiv:1906.08253*, 2019.

[21] Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. Contextual decision processes with low bellman rank are pac-learnable. In *International Conference on Machine Learning*, pages 1704–1713. PMLR, 2017.

[22] Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, pages 2137–2143. PMLR, 2020.

[23] Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *In Proc. 19th International Conference on Machine Learning*. Citeseer, 2002.

[24] Michael Kearns, Yishay Mansour, and Andrew Ng. Approximate planning in large pomdps via reusable trajectories. *Advances in Neural Information Processing Systems*, 12, 1999.

[25] Rahul Kidambi, Jonathan Chang, and Wen Sun. Optimism is all you need: Model-based imitation learning from observation alone. *arXiv preprint arXiv:2102.10769*, 2021.

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[27] Edgar D Klenske and Philipp Hennig. Dual control for approximate bayesian reinforcement learning. *The Journal of Machine Learning Research*, 17(1):4354–4383, 2016.

[28] Vijay R Konda and John N Tsitsiklis. Actor-critic algorithms. In *Advances in neural information processing systems*, pages 1008–1014. Citeseer, 2000.

[29] Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *International Conference on Machine Learning*, pages 2796–2804. PMLR, 2018.

[30] Thanard Kurutach, Ignasi Clavera, Yan Duan, Aviv Tamar, and Pieter Abbeel. Model-ensemble trust-region policy optimization. *arXiv preprint arXiv:1802.10592*, 2018.

[31] Sergey Levine and Pieter Abbeel. Learning neural network policies with guided policy search under unknown dynamics. In *NIPS*, volume 27, pages 1071–1079. Citeseer, 2014.

[32] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

[33] Gene Li, Pritish Kamath, Dylan J Foster, and Nathan Srebro. Eluder dimension and generalized rank. *arXiv preprint arXiv:2104.06970*, 2021.

[34] Xiuyuan Lu and Benjamin Van Roy. Ensemble sampling. *Advances in neural information processing systems*, 30, 2017.

[35] Yuping Luo, Huazhe Xu, Yuanzhi Li, Yuandong Tian, Trevor Darrell, and Tengyu Ma. Algorithmic framework for model-based deep reinforcement learning with theoretical guarantees. *arXiv preprint arXiv:1807.03858*, 2018.

[36] Horia Mania, Stephen Tu, and Benjamin Recht. Certainty equivalence is efficient for linear quadratic control. *arXiv preprint arXiv:1902.07826*, 2019.

[37] Efstratios Markou and Carl E. Rasmussen. Bayesian methods for efficient reinforcement learning in tabular problems, 2019.

[38] Aditya Modi, Jinglin Chen, Akshay Krishnamurthy, Nan Jiang, and Alekh Agarwal. Model-free representation learning and exploration in low-rank mdps. *arXiv preprint arXiv:2102.07035*, 2021.

[39] William Montgomery and Sergey Levine. Guided policy search as approximate mirror descent. *arXiv preprint arXiv:1607.04614*, 2016.

[40] Manfred Morari and Jay H Lee. Model predictive control: past, present and future. *Computers & Chemical Engineering*, 23(4-5):667–682, 1999.

[41] Anusha Nagabandi, Gregory Kahn, Ronald S Fearing, and Sergey Levine. Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7559–7566. IEEE, 2018.

[42] Ian Osband, Daniel Russo, and Benjamin Van Roy. (more) efficient reinforcement learning via posterior sampling. *arXiv preprint arXiv:1306.0940*, 2013.

[43] Ian Osband and Benjamin Van Roy. Model-based reinforcement learning and the eluder dimension. *arXiv preprint arXiv:1406.1853*, 2014.

[44] Ian Osband and Benjamin Van Roy. Why is posterior sampling better than optimism for reinforcement learning? In *International Conference on Machine Learning*, pages 2701–2710. PMLR, 2017.

[45] Ian Osband, Benjamin Van Roy, Daniel J Russo, Zheng Wen, et al. Deep exploration via randomized value functions. *J. Mach. Learn. Res.*, 20(124):1–62, 2019.

[46] Brendan O'Donoghue, Ian Osband, Remi Munos, and Volodymyr Mnih. The uncertainty bellman equation and exploration. In *International Conference on Machine Learning*, pages 3836–3845, 2018.

[47] Matteo Pirotta, Marcello Restelli, and Luca Bascetta. Policy gradient in lipschitz markov decision processes. *Machine Learning*, 100(2):255–283, 2015.

[48] Stephane Ross and J Andrew Bagnell. Agnostic system identification for model-based reinforcement learning. *arXiv preprint arXiv:1203.1007*, 2012.

[49] Daniel Russo and Benjamin Van Roy. Eluder dimension and the sample complexity of optimistic exploration. In *NIPS*, pages 2256–2264. Citeseer, 2013.

[50] Daniel Russo and Benjamin Van Roy. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39(4):1221–1243, 2014.

[51] Daniel Russo and Benjamin Van Roy. Satisficing in time-sensitive bandit learning. *arXiv preprint arXiv:1803.02855*, 2018.

[52] Daniel Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, and Zheng Wen. A tutorial on thompson sampling. *arXiv preprint arXiv:1707.02038*, 2017.

[53] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR, 2015.

[54] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

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

[56] Alexander L Strehl and Michael L Littman. A theoretical analysis of model-based interval estimation. In *Proceedings of the 22nd international conference on Machine learning*, pages 856–863, 2005.

[57] Malcolm Strens. A bayesian framework for reinforcement learning. In *ICML*, volume 2000, pages 943–950, 2000.

[58] HJ Suh, Max Simchowitz, Kaiqing Zhang, and Russ Tedrake. Do differentiable simulators give better policy gradients? *arXiv preprint arXiv:2202.00817*, 2022.

[59] Wen Sun, Geoffrey J Gordon, Byron Boots, and J Andrew Bagnell. Dual policy iteration. *arXiv preprint arXiv:1805.10755*, 2018.

[60] Wen Sun, Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, and John Langford. Model-based rl in contextual decision processes: Pac bounds and exponential improvements over model-free approaches. In *Conference on Learning Theory*, pages 2898–2933. PMLR, 2019.

[61] Richard S Sutton. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Machine learning proceedings 1990*, pages 216–224. Elsevier, 1990.

[62] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[63] Tingwu Wang and Jimmy Ba. Exploring model-based planning with policy networks. *arXiv preprint arXiv:1906.08649*, 2019.

[64] Jie Xu, Viktor Makoviychuk, Yashraj Narang, Fabio Ramos, Wojciech Matusik, Animesh Garg, and Miles Macklin. Accelerated policy learning with parallel differentiable simulation. *arXiv preprint arXiv:2204.07137*, 2022.

[65] Lin Yang and Mengdi Wang. Sample-optimal parametric q-learning using linearly additive features. In *International Conference on Machine Learning*, pages 6995–7004. PMLR, 2019.

[66] Lin Yang and Mengdi Wang. Reinforcement learning in feature space: Matrix bandit, kernels, and regret bound. In *International Conference on Machine Learning*, pages 10746–10756. PMLR, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] Some components in our algorithm are naively designed.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Appendix H.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Assumption 5.2 and 5.7.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix A.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A] Our code can be found in the supplemental material.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Proofs

A.1 Proof of Theorem 5.4

Proof. We lay out the proof in two major steps. Firstly, we characterize the performance difference between $J(q_t)$ and $J(\pi_{t-1})$, which can be done by applying Lemma B.3. Specifically, we set π_1, π_2 in Lemma B.3 to q_t, π_{t-1} and set f as the reference model $\widetilde{f}_t$. Then we obtain

$$\begin{aligned} &J(q_t) - J(\pi_{t-1}) \\ &\quad = \Delta(t) - \frac{\gamma}{2(1-\gamma)}\left(\mathbb{E}_{\rho_{q_t}}\left[\left\|\widetilde{f}_t(s,a) - f^*(\cdot|s,a)\right\|_1\right] + \mathbb{E}_{\rho_{\pi_{t-1}}}\left[\left\|\widetilde{f}_t(s,a) - f^*(\cdot|s,a)\right\|_1\right]\right), \end{aligned} \tag{A.1}$$

where $\Delta(t) := \mathbb{E}_{s\sim\zeta}\left[V_{q_t}^{\widetilde{f}_t}(s) - V_{\pi_{t-1}}^{\widetilde{f}_t}(s)\right] \geq 0$ due to the optimality of q_t under $\widetilde{f}_t$, i.e., $q_t = \operatorname{argmax}_q V_q^{\widetilde{f}_t}$.

Recall that the reference model is the least squares estimate, i.e.,

$$\widetilde{f}_t = \widehat{f}_t^{LS} = \operatorname*{argmin}_{f\in\mathcal{F}} \sum_{(s,a,s')\in\mathcal{H}_{t-1}} \left\|f(s,a) - s'\right\|_2^2,$$

where $\mathcal{H}_{t-1}$ is the trajectory in the real environment when following policy π_{t-1}.

From the simulation property of continuous distribution, we have the following equivalence between the direct and indirect ways of drawing samples:

$$s' \sim f^*(\cdot|s,a) \;\equiv\; s' = f^*(s,a) + \epsilon, \epsilon \sim p(\epsilon),$$

where $p(\epsilon)$ is some noise distribution.

Therefore, according to the Gaussian noise assumption, we obtain from the least squares generalization bound in Lemma B.4 that

$$\mathbb{E}_{\rho_{\pi_{t-1}}}\left[\left\|\widetilde{f}_t(s,a) - f^*(\cdot|s,a)\right\|_1\right] \leq \frac{22C^2\ln(|\mathcal{F}|/\delta)}{H}, \tag{A.2}$$

where $\epsilon_{\text{approx}} = 0$ in the generalization bound as the realizability is guaranteed since $\widehat{f}_t^{LS}$ and f^* are from the same function class $\mathcal{F}$.

Similarly, we have for the intermediate policy q_t that

$$\begin{aligned} \mathbb{E}_{\rho_{q_t}}\left[\left\|\widetilde{f}_t(s,a) - f^*(\cdot|s,a)\right\|_1\right] &\leq \mathbb{E}_{\rho_{\pi_{t-1}}}\left[\left\|\widetilde{f}_t(s,a) - f^*(\cdot|s,a)\right\|_1\right] \cdot \left\{\mathbb{E}_{\rho_{\pi_{t-1}}}\left[\left(\frac{d\rho_{q_t}}{d\rho_{\pi_{t-1}}}(s)\right)^2\right]\right\}^{1/2} \\ &\leq \kappa \cdot \frac{22C^2\ln(|\mathcal{F}|/\delta)}{H}. \end{aligned} \tag{A.3}$$

Now we can bound (A.1) by

$$J(q_t) - J(\pi_{t-1}) \geq \Delta(t) - (1+\kappa)\cdot\frac{22\gamma C^2\ln(|\mathcal{F}|/\delta)}{(1-\gamma)H}. \tag{A.4}$$

The second step of the proof is to characterize the performance difference between $J(\pi_t)$ and $J(q_t)$.

From the Performance Difference Lemma B.2, we obtain

$$\begin{aligned} J(q_t) - J(\pi_t) &= \frac{1}{1-\gamma}\cdot\mathbb{E}_{(s,a)\sim\rho_{q_t}}\left[A_{\pi_t}^{f^*}(s,a)\right] \\ &= \frac{1}{1-\gamma}\cdot\mathbb{E}_{s\sim\nu_{q_t}}\left[\mathbb{E}_{a\sim q_t}\left[A_{\pi_t}^{f^*}(s,a)\right]\right] \\ &= \frac{1}{1-\gamma}\cdot\mathbb{E}_{s\sim\nu_{q_t}}\left[\mathbb{E}_{a\sim q_t}\left[A_{\pi_t}^{f^*}(s,a)\right] - \mathbb{E}_{a\sim\pi_t}\left[A_{\pi_t}^{f^*}(s,a)\right]\right], \end{aligned} \tag{A.5}$$

where recall that $\iota := \max_{s,a} |A_\pi^{f^*}(s,a)|$ and the third equality holds due to $\mathbb{E}_{a\sim\pi_t}\left[A_{\pi_t}^{f^*}(s,a)\right] = 0$ for any s.

By the definition of the total variation distance, we can further bound the absolute difference as

$$|J(q_t) - J(\pi_t)| \leq \frac{2\eta\iota}{1-\gamma}, \tag{A.6}$$

Thus, we have $J(\pi_t) - J(q_t) \geq -2\eta\iota/(1-\gamma)$ and similarly $J(q_{t-1}) - J(\pi_{t-1}) \geq -2\eta\iota/(1-\gamma)$. Combining with (A.4) gives us the iterative improvement bound as follows:

$$\begin{aligned} J(\pi_t) - J(\pi_{t-1}) &= J(\pi_t) - J(q_t) + J(q_t) - J(\pi_{t-1}) \\ &\geq \Delta(t) - (1+\kappa)\cdot\frac{22\gamma C^2 \ln(|\mathcal{F}|/\delta)}{(1-\gamma)H} - \frac{2\eta\iota}{1-\gamma}. \end{aligned} \tag{A.7}$$

□

A.2 Proof of Theorem 5.8

Proof. We are interested in the expected regret defined as $\text{BayesRegret}(T,\pi,\phi) := \mathbb{E}[\sum_{t=1}^T \mathfrak{R}_t]$, where $\mathfrak{R}_t = V_{\pi^*}^{f^*} - V_{\pi_t}^{f^*}$.

Recall the definition of the reactive policy π_t in CDPO (i.e. (4.2)) and the imagined best-performing policy π_{f_t} under a sampled model f_t, i.e., $\pi_{f_t} = \max_\pi V_\pi^{f_t}$.

From the Posterior Sampling Lemma, we know that if ψ is the distribution of f^*, then for any sigma-algebra $\sigma(\mathcal{H}_t)$-measurable function g,

$$\mathbb{E}[g(f^*)\,|\,\mathcal{H}_t] = \mathbb{E}[g(f_t)\,|\,\mathcal{H}_t]. \tag{A.8}$$

The PS Lemma together with the law of total expectation gives us

$$\mathbb{E}[V_{\pi^*}^{f^*} - V_{\pi_{f_t}}^{f_t}] = 0, \tag{A.9}$$

where the equality holds since the true f^* and the sampled f_t are identically distributed when conditioned on $\mathcal{H}_t$. Therefore, we obtain the expected regret for CDPO as

$$\begin{aligned} \text{BayesRegret}(T,\pi,\phi) &= \sum_{t=1}^T \mathbb{E}[V_{\pi_{f_t}}^{f_t} - V_{\pi_t}^{f_t} + V_{\pi_t}^{f_t} - V_{\pi_t}^{f^*}] \\ &= \sum_{t=1}^T \mathbb{E}[V_{\pi_{f_t}}^{f_t} - V_{\pi_{f_t}}^{\widetilde{f}_t} + V_{\pi_{f_t}}^{\widetilde{f}_t} - V_{\pi_t}^{f_t} + V_{\pi_t}^{f_t} - V_{\pi_t}^{f^*}] \\ &\leq \sum_{t=1}^T \mathbb{E}[V_{\pi_{f_t}}^{f_t} - V_{\pi_{f_t}}^{\widetilde{f}_t} + V_{q_t}^{\widetilde{f}_t} - V_{q_t}^{f_t} + V_{\pi_t}^{f_t} - V_{\pi_t}^{f^*}], \end{aligned} \tag{A.10}$$

where the inequality follows from the greediness of q_t and the optimality of π_t within a trust-region centered around q_t,i.e., $V_{\pi_{f_t}}^{\widetilde{f}_t} \leq V_{q_t}^{\widetilde{f}_t}$ for any π_{f_t} and $V_{\pi_t}^{f_t} \geq V_{q_t}^{f_t}$.

From the Simulation Lemma B.1, we have the bound of $\mathbb{E}\left[\left|V_\pi^{f_t} - V_\pi^{\widetilde{f}_t}\right|\right]$ for any policy π as follows:

$$\begin{aligned} \mathbb{E}\left[\left|V_\pi^{f_t} - V_\pi^{\widetilde{f}_t}\right|\right] &= \gamma\mathbb{E}\left[\left|\mathbb{E}_{(s,a)\sim\widetilde{\rho}_\pi}[(f_t(\cdot|s,a) - \widetilde{f}_t(\cdot|s,a))\cdot V_\pi^{f_t}(s,a)]\right|\right] \\ &\leq \gamma\mathbb{E}\left[\left|\mathbb{E}_{(s,a)\sim\widetilde{\rho}_\pi}\left[L_t\cdot\|f_t(s,a) - \widetilde{f}_t(s,a)\|_2\right]\right|\right], \end{aligned} \tag{A.11}$$

where the first equation follows from Lemma B.1 and $\widetilde{\rho}_\pi$ is the state-action visitation measure under model $\widetilde{f}_t$, the second inequality follows the simulation property of continuous distribution and the Lipschitz value function assumption.

We define the event $A = \left\{ \widetilde{f}_t \in \bigcap_t \mathcal{F}_t, f_t \in \bigcap_t \mathcal{F}_t \right\}$. Recall that the model is bounded by $\|f\|_2 \leq C$. Then we can reduce the expected regret to a sum of set widths:

$$\mathbb{E}\big[V_\pi^{f_t} - V_\pi^{\widetilde{f}_t}\big] \leq \gamma \mathbb{E}\Big[\Big|\mathbb{E}_{(s,a)\sim\widetilde{\rho}_\pi}\big[\mathbb{E}[L_t|A]\omega_t(s,a) + \big(1-\mathbb{P}(A)\big)C\big]\Big|\Big]. \tag{A.12}$$

We can further know from the construction of the confidence set (c.f. Lemma B.5) that $\mathbb{P}\Big(f^* \in \bigcap_t \mathcal{F}_t\Big) \geq 1-2\delta$ and $\mathbb{P}(A) \geq 1-2\delta$ since f_t, f^* are identically distributed and $\mathbb{P}\big(\widetilde{f}_t \in \mathcal{F}_t\big) = 1$ as $\mathcal{F}_t$ is centered at the least squares model for all t.

Besides, we have for

$$\mathbb{E}[L_t|A] \leq \frac{L_t}{P(A)} \leq \frac{L_t}{1-2\delta}. \tag{A.13}$$

Plugging into (A.21), we have

$$\begin{aligned}\mathbb{E}\big[V_\pi^{f_t} - V_\pi^{\widetilde{f}_t}\big] &\leq \gamma \mathbb{E}\Big[\Big|\mathbb{E}_{(s,a)\sim\widetilde{\rho}_\pi}\big[L_t/(1-2\delta)\omega_t(s,a) + 2\delta C\big]\Big|\Big] \\ &\leq \gamma \mathbb{E}\Big[\frac{L_t}{1-2\delta}\cdot\Big|\mathbb{E}_{(s,a)\sim\widetilde{\rho}_\pi}\big[\omega_t(s,a)\big]\Big|\Big] + 2\gamma\delta C. \end{aligned} \tag{A.14}$$

Summing over T iterations gives us

$$\sum_{t=1}^T \mathbb{E}\big[V_\pi^{f_t} - V_\pi^{\widetilde{f}_t}\big] \leq \gamma \sum_{t=1}^T \mathbb{E}\Big[\frac{L_t}{1-2\delta}\cdot\Big|\mathbb{E}_{(s,a)\sim\widetilde{\rho}_\pi}\big[\omega_t(s,a)\big]\Big|\Big] + 2\gamma\delta CT. \tag{A.15}$$

By setting $\delta = 1/(2T)$, we obtain

$$\begin{aligned}\sum_{t=1}^T \mathbb{E}\big[V_\pi^{f_t} - V_\pi^{\widetilde{f}_t}\big] &\leq \frac{\gamma LT}{T-1}\sum_{t=1}^T \mathbb{E}_{\widetilde{\rho}_\pi}[\omega_t(s,a)] + \gamma C \\ &\leq \frac{\gamma LT}{T-1}\cdot\Big(1 + \frac{1}{1-\gamma}Cd_E + 4\sqrt{Td_E\beta_T(1/(2T),\alpha)}\Big) + \gamma C, \end{aligned} \tag{A.16}$$

where the last inequality follows from Lemma B.6 to bound the sum of the set width. We denote $d_E := \dim_E(\mathcal{F}, T^{-1})$ for notation simplicity.

Since (A.16) holds for all policy π, we have the bound for $\mathbb{E}[V_{\pi_{f_t}}^{f_t} - V_{\pi_{f_t}}^{\widetilde{f}_t}]$ and the bound for $\mathbb{E}[V_{q_t}^{\widetilde{f}_t} - V_{q_t}^{f_t}]$. What remains in the expected regret (A.19) is the $\mathbb{E}[V_{\pi_t}^{f_t} - V_{\pi_t}^{f^*}]$ term, which can be bounded similarly.

Specifically, we define another event $B = \left\{ f^* \in \bigcap_t \mathcal{F}_t, f_t \in \bigcap_t \mathcal{F}_t \right\}$. Since by construction $\mathbb{P}\Big(f^* \in \bigcap_t \mathcal{F}_t\Big) \geq 1-2\delta$ and $\mathbb{P}\Big(f_t \in \bigcap_t \mathcal{F}_t\Big) \geq 1-2\delta$, we have $\mathbb{P}(B) \geq 1-4\delta$ via a union bound. This implies the following bound

$$\begin{aligned}\sum_{t=1}^T \mathbb{E}\big[V_\pi^{f_t} - V_\pi^{f^*}\big] &\leq \gamma \sum_{t=1}^T \mathbb{E}\Big[\frac{L_t}{1-4\delta}\cdot\Big|\mathbb{E}\big[\omega_t(s,a)\big]\Big|\Big] + 4\gamma\delta CT \\ &\leq \frac{\gamma LT}{T-2}\sum_{t=1}^T \mathbb{E}_{\rho_\pi}[\omega_t(s,a)] + 2\gamma C \\ &\leq \frac{\gamma LT}{T-2}\cdot\Big(1 + \frac{1}{1-\gamma}Cd_E + 4\sqrt{Td_E\beta_T(1/(2T),\alpha)}\Big) + 2\gamma C, \end{aligned} \tag{A.17}$$

where the second inequality follows from the choice of δ, i.e., $\delta = 1/(2T)$.

Plugging (A.16) and (A.17) into (A.19), we obtain the expected regret as

$$\begin{aligned}
\text{BayesRegret}(T, \pi, \phi) &\leq \sum_{t=1}^{T} \mathbb{E}[V_{\pi_{f_t}}^{f_t} - V_{\pi_{f_t}}^{\widetilde{f}_t} + V_{q_t}^{\widetilde{f}_t} - V_{q_t}^{f_t} + V_{\pi_t}^{f_t} - V_{\pi_t}^{f^*}] \\
&\leq \left(\frac{2\gamma LT}{T-1} + \frac{\gamma LT}{T-2}\right) \cdot \left(1 + \frac{1}{1-\gamma} C d_E + 4\sqrt{T d_E \beta_T(1/(2T), \alpha)}\right) + 4\gamma C \\
&= \frac{\gamma T(3T-5)L}{(T-1)(T-2)} \cdot \left(1 + \frac{1}{1-\gamma} C d_E + 4\sqrt{T d_E \beta_T(1/(2T), \alpha)}\right) + 4\gamma C.
\end{aligned} \tag{A.18}$$

By setting $\alpha = 1/(T^2)$ and $\delta = 1/(2T)$ in Lemma B.5, we have the following confidence parameter that can guarantee that f^* is contained in the confidence set with high probability:

$$\beta_T(1/(2T), 1/(T^2)) = 8\sigma^2 \log\Big(2N\big(\mathcal{F}, 1/(T^2), \|\cdot\|_2\big)T\Big) + 2\big(8C + \sqrt{8\sigma^2 \log(8T^3)}\,\big)/T,$$

where recall that $N\big(\mathcal{F}, \alpha, \|\cdot\|_2\big)$ is the α-covering number of $\mathcal{F}$ with respect to the $\|\cdot\|_2$-norm. □

A.3 Proof of Theorem 5.1

Proof. Denote the *imagined* optimal policy π_{f_t} under a sampled model f_t as $\pi_{f_t} = \max_\pi V_\pi^{f_t}$. For PSRL, its expected regret can be decomposed as

$$\begin{aligned}
\text{BayesRegret}(T, \pi^{\text{PSRL}}, \phi) &= \sum_{t=1}^{T} \mathbb{E}[V_{\pi^*}^{f^*} - V_{\pi_t}^{f^*}] \\
&= \sum_{t=1}^{T} \mathbb{E}[V_{\pi^*}^{f^*} - V_{\pi_{f_t}}^{f^*}] \\
&= \sum_{t=1}^{T} \mathbb{E}[V_{\pi_{f_t}}^{f_t} - V_{\pi_{f_t}}^{f^*}],
\end{aligned} \tag{A.19}$$

where the second equality holds since the PSRL policy $\pi_t := \pi_{f_t}$ for a sampled f_t. The third equality follows from (A.9), obtained by the Posterior Sampling Lemma and the law of total expectation.

Similar with the proof in A.2, we obtain from the Simulation Lemma B.1 that

$$\begin{aligned}
\mathbb{E}\Big[\big|V_{\pi_{f_t}}^{f_t} - V_{\pi_{f_t}}^{f^*}\big|\Big] &= \gamma \mathbb{E}\Big[\big|\mathbb{E}_{(s,a)\sim\rho_\pi}[(f_t(\cdot|s,a) - f^*(\cdot|s,a)) \cdot V^\pi(s,a)]\big|\Big] \\
&\leq \gamma \mathbb{E}\Big[\Big|\mathbb{E}_{(s,a)\sim\rho_\pi}\big[L_t \cdot \|f_t(s,a) - f^*(s,a)\|_2\big]\Big|\Big],
\end{aligned} \tag{A.20}$$

where the equality follows from Lemma B.1 and the inequality follows the simulation property of continuous distributions and the Lipschitz value function assumption.

Define the event $E = \left\{f^* \in \bigcap_t \mathcal{F}_t, f_t \in \bigcap_t \mathcal{F}_t\right\}$. The expected regret can be reduced to the sum of set widths:

$$\begin{aligned}
\mathbb{E}\big[V_\pi^{f_t} - V_\pi^{\widetilde{f}_t}\big] &\leq \gamma \mathbb{E}\Big[\Big|\mathbb{E}_{(s,a)\sim\rho_\pi}\big[\mathbb{E}[L_t|E]\omega_t(s,a) + \big(1 - \mathbb{P}(E)\big)C\big]\Big|\Big] \\
&\leq \gamma \mathbb{E}\Big[\Big|\mathbb{E}_{(s,a)\sim\rho_\pi}\big[L_t/(1-4\delta)\omega_t(s,a) + 4\delta C\big]\Big|\Big] \\
&\leq \gamma \mathbb{E}\Big[\frac{L_t}{1-4\delta} \cdot \Big|\mathbb{E}_{(s,a)\sim\rho_\pi}\big[\omega_t(s,a)\big]\Big|\Big] + 4\gamma\delta C,
\end{aligned} \tag{A.21}$$

where the second inequality follows from the construction of confidence set that $\mathbb{P}\Big(f^* \in \bigcap_t \mathcal{F}_t\Big) \geq 1 - 2\delta$ and thus $\mathbb{P}(E) \geq 1 - 4\delta$.

Therefore, the PSRL expected regret can be bounded by

$$\text{BayesRegret}(T, \pi^{\text{PSRL}}, \phi) \leq \gamma \frac{L}{1-4\delta} \sum_{t=1}^{T} \mathbb{E}\big[\omega_t\big] + 4T\gamma\delta C, \tag{A.22}$$

From the proof in A.2, the expected regret of CDPO is bounded by

$$\text{BayesRegret}(T, \pi^{\text{CDPO}}, \phi) \leq \gamma \frac{L}{1-4\delta} \sum_{t=1}^{T} 3\mathbb{E}\big[\omega_t\big] + 8T\gamma\delta C, \tag{A.23}$$

The claim is thus established. □

B Useful Lemmas

Lemma B.1 (Simulation Lemma)**.** For any policy π and transition f_1, f_2, we have

$$V_\pi^{f_1} - V_\pi^{f_2} = \gamma(I - \gamma f_2^\pi)^{-1}(f_1 - f_2)V_\pi^{f_1}. \tag{B.1}$$

Proof. Denote the expected reward under policy π as r_π. Let f^π be the transition matrix on state-action pairs induced by policy π, defined as $f^\pi_{(s,a),(s',a')} := P(s'|s,a)\pi(a'|s')$.

Then we have

$$V_\pi = r_\pi + \gamma f^\pi V_\pi.$$

Since $\gamma < 1$, it is easy to verify that $I - \gamma f^\pi$ is full rank and thus invertible. Therefore, we can write

$$V_\pi = (I - \gamma f^\pi)^{-1} r_\pi. \tag{B.2}$$

Therefore, we conclude the proof by

$$\begin{aligned}
V_\pi^{f_1} - V_\pi^{f_2} &= V_\pi^{f_1} - (I - \gamma f_2^\pi)^{-1} r_\pi \\
&= (I - \gamma f_2^\pi)^{-1} \cdot \big((I - \gamma f_2^\pi) - (I - \gamma f_1^\pi)\big) V_\pi^{f_1} \\
&= \gamma (I - \gamma f_2^\pi)^{-1} (f_1^\pi - f_2^\pi) V_\pi^{f_1} \\
&= \gamma (I - \gamma f_2^\pi)^{-1} (f_1 - f_2) V_\pi^{f_1},
\end{aligned}$$

where the second equality follows from the Bellman equation. □

Lemma B.2 (Performance Difference Lemma)**.** For all policies π, π^* and distribution μ over $\mathcal{S}$, we have

$$J(\pi) - J(\pi') = \frac{1}{1-\gamma} \cdot \mathbb{E}_{(s,a)\sim\sigma_\pi}[A^{\pi'}(s,a)]. \tag{B.3}$$

Proof. This lemma is widely adopted in RL. Proof can be found in various previous works, e.g. Lemma 1.16 in [3].

Let $\mathbb{P}^\pi(\tau|s_0 = s)$ denote the probability of observing trajectory τ starting at state s_0 and then following π. Then the value difference can be written as

$$\begin{aligned}
V_\pi^{f^*}(s) - V_{\pi'}^{f^*}(s) &= \mathbb{E}_{\tau\sim\mathbb{P}^\pi(\cdot|s_0=s)}\Big[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h)\Big] - V_{\pi'}^{f^*}(s) \\
&= \mathbb{E}_{\tau\sim\mathbb{P}^\pi(\cdot|s_0=s)}\Big[\sum_{h=0}^{\infty} \gamma^h \big(r(s_h, a_h) + V_{\pi'}^{f^*}(s_h) - V_{\pi'}^{f^*}(s_h)\big)\Big] - V_{\pi'}^{f^*}(s) \\
&= \mathbb{E}_{\tau\sim\mathbb{P}^\pi(\cdot|s_0=s)}\Big[\sum_{h=0}^{\infty} \gamma^h \big(r(s_h, a_h) + \gamma V_{\pi'}^{f^*}(s_{h+1}) - V_{\pi'}^{f^*}(s_h)\big)\Big]
\end{aligned}$$

Following the law of iterated expectations, we obtain

$$\begin{aligned} V_{\pi}^{f^*}(s) - V_{\pi'}^{f^*}(s) &= \mathbb{E}_{\tau\sim\mathbb{P}^{\pi}(\cdot|s_0=s)}\Big[\sum_{h=0}^{\infty}\gamma^h\big(r(s_h,a_h)+\gamma\mathbb{E}[V_{\pi'}^{f^*}(s_{h+1})|s_h,a_h]-V_{\pi'}^{f^*}(s_h)\big)\Big] \\ &= \mathbb{E}_{\tau\sim\mathbb{P}^{\pi}(\cdot|s_0=s)}\Big[\sum_{h=0}^{\infty}\gamma^h\big(Q_{\pi'}^{f^*}(s_h,a_h)-V_{\pi'}^{f^*}(s_h)\big)\Big] \\ &= \mathbb{E}_{\tau\sim\mathbb{P}^{\pi}(\cdot|s_0=s)}\Big[\sum_{h=0}^{\infty}\gamma^h A_{\pi'}^{f^*}(s_h,a_h)\Big], \end{aligned} \tag{B.4}$$

where the third equation rearranges terms in the summation via telescoping, and the fourth equality follows from the law of total expectation.

From the definition of objective $J(\pi)$ in (2.3), we obtain

$$\begin{aligned} J(\pi) - J(\pi') &= \mathbb{E}_{s_0\sim\zeta}[V_{\pi}^{f^*}(s_0) - V_{\pi'}^{f^*}(s_0)] \\ &= \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim\sigma_\pi}[A^{\pi'}(s,a)]. \end{aligned} \tag{B.5}$$

□

Lemma B.3 (Performance Difference and Model Error)**.** For any two policies π_1 and π_2, it holds that

$$\begin{aligned} J(\pi_1) - J(\pi_2) &= \mathbb{E}_{s\sim\zeta}\big[V_{\pi_1}^{f}(s) - V_{\pi_2}^{f}(s)\big] \\ &\quad - \frac{\gamma}{2(1-\gamma)}\left(\mathbb{E}_{\rho_{\pi_1}}\Big[\big\|f(\cdot|s,a)-f^*(\cdot|s,a)\big\|_1\Big] + \mathbb{E}_{\rho_{\pi_2}}\Big[\big\|f(\cdot|s,a)-f^*(\cdot|s,a)\big\|_1\Big]\right). \end{aligned}$$

Proof. The proof can be established by combining the Performance Difference Lemma and the Simulation Lemma. We refer to Corollary 3.1 in [48] or Lemma A.3 in [59] for a detailed proof. □

Lemma B.4 (Least Squares Generalization Bound)**.** Given a dataset $\mathcal{H} = \{x_i, y_i\}_{i=1}^n$ where $x_i \in \mathcal{X}$ and $x_i, y_i \sim \nu$, and $y_i = f^*(x_i) + \epsilon_i$. Suppose $|y_i| \le Y$ and ϵ_i is independently sampled noise. Given a function class $\mathcal{F} : \mathcal{X} \to [0, Y]$, we assume approximate realizable, i.e., $\min_{f\in\mathcal{F}} \mathbb{E}_{x\sim\nu}\big[|f^*(x) - f(x)|^2\big] \le \epsilon_{\text{approx}}$. Denote $\widehat{f}$ as the least square solution, i.e., $\widehat{f} = \operatorname{argmin}_{f\in\mathcal{F}} \sum_{i=1}^n \big(f(x_i) - y_i\big)^2$. With probability at least $1-\delta$, we have

$$\mathbb{E}_{x\sim\nu}\Big[\big(\widehat{f}(x) - f^*(x)\big)^2\Big] \le \frac{22Y^2\ln(|\mathcal{F}|/\delta)}{n} + 20\epsilon_{\text{approx}}. \tag{B.6}$$

Proof. The result is standard and can be proved by using the Bernstein's inequality and union bound. Detailed proof can be found at Lemma A.11 in [3]. □

Lemma B.5 (Confidence sets with high probability)**.** If the control parameter $\beta_t(\delta,\alpha)$ is set to

$$\beta_t(\delta,\alpha) = 8\sigma^2\log(N(\mathcal{F},\alpha,\|\cdot\|_2)/\delta) + 2\alpha t\left(8C + \sqrt{8\sigma^2\log(4t^2/\delta)}\right), \tag{B.7}$$

then for all $\delta > 0$, $\alpha > 0$ and $t \in \mathbb{N}$, the confidence set $\mathcal{F}_t = \mathcal{F}_t(\beta_t(\delta,\alpha))$ satisfies:

$$P\Big(f^* \in \bigcap_t \mathcal{F}_t\Big) \ge 1 - 2\delta. \tag{B.8}$$

Proof. See [43] Proposition 5 for a detailed proof. □

Lemma B.6 (Bound of Set Width Sum)**.** If $\{\beta_t | t \in \mathbb{N}\}$ is nondecreasing with $\mathcal{F}_t = \mathcal{F}_t(\beta_t)$ and $\|f\|_2 \le C$ for all $f \in \mathcal{F}$, then finite-horizon MDP we have

$$\sum_{t=1}^{T}\sum_{h=1}^{H}\omega_t(s_h,a_h) \le 1 + HC\dim_E(\mathcal{F},T^{-1}) + 4\sqrt{\dim_E(\mathcal{F},T^{-1})\beta_T T}, \tag{B.9}$$

where $\omega_t(s,a) = \sup_{\underline{f},\overline{f}\sim\mathcal{F}_t}\|\overline{f}(s,a) - \underline{f}(s,a)\|_2$.

Proof. See [43] Proposition 6 for a detailed proof. □

C Limitations of Eluder Dimension

In Theorem 5.8, the *eluder dimension* d_E appears in the Bayes expected regret bound to capture how effectively the observed samples can extrapolate to unobserved transitions.

For some specific function classes, Osband et al. [43] provide the corresponding eluder dimension bound, e.g., for (generalized) linear function classes, quadratic function class, and for finite MDPs, c.f. Proposition 1-4 in [43].

However, for non-linear models, Dong et al. [13] show that the ε-eluder dimension of one-layer neural networks is *at least* exponential in model dimension. Similar results are also established in [33]. We refer to Section 5 in [13] or Section 4 in [33] for details and more explanations.

D Additional Related Work

Some MBRL work also concerns iterative policy improvement. SLBO [35] provides a trust-region policy optimization framework based on OFU. However, the conditions for monotonic improvement cannot be satisfied by most parameterized models [35, 13], which leads to a greedy algorithm in practice. Prior work that shares similarities with ours contains DPI [59] and GPS [31, 39] as dual policy optimization procedures are adopted. Both DPI and GPS leverage a *locally* accurate model and use different objectives for imitating the intermediate policy within a trust-region. However, the policy imitation procedure updates the policy parameter in a *supervised* manner, which poses additional challenges for effective exploration, resulting in unknown convergence results even with a simple model class. In contrast, CDPO by taking the epistemic uncertainty into consideration can be shown to achieve global optimality. In fact, greedy model exploitation is provably optimal only in very limited cases, e.g., linear-quadratic regulator (LQR) settings [36].

OFU-RL has shown to achieve an optimal sublinear regret when applied to online LQR [1], tabular MDPs [19] and linear MDPs [22]. Among them, HUCRL [10] is a deep algorithm proposed to deal with the joint optimization intractability in (3.1). Besides, Russo and Van Roy [49, 50] unify the bounds in various settings (e.g., finite or linear MDPs) by introducing an additional model complexity measure — eluder dimension. Other complexity measure include witness rank [60], linear dimensionality [66] and sequential Rademacher complexity [13].

E Algorithm Instantiations

The model-based policy optimization solver $\texttt{MBPO}(\pi, \{f\}, \mathcal{J})$ in Algorithm 1 can be instantiated as one of the following algorithms, Dyna-style policy optimization in Algorithm 2, model-based back-propagation in Algorithm 3, and model predictive control policy optimization in Algorithm 4. By default, MBPO is instantiated as the Dyna solver (i.e. Algorithm 2) in our MuJoCo experiments and as the policy iteration solver in our N-Chain MDPs experiments. We note that the instantiations are not restricted to the listed algorithms, and many other MBPO algorithms that augment policy learning with a predictive model can also be leveraged, e.g., model-based value expansion [15, 6]. In the Referential Update step where no input policy exists in $\texttt{MBPO}(\cdot, \widehat{f}_t^{LS}, (4.1)$, we initialize policy $\pi = \pi_{t-1}$, i.e. the reactive policy from the last iteration.

Dyna. Dyna involves model-generated data and optimizing the policy with any model-free RL method, e.g., REINFORCE or actor-critic [28]. The state-action value can be estimated by learning a critic function or unrolling the model. In Constrained Conservative Update, the input objective function $\mathcal{J}$ is (4.2), which is with constraints. Thus, the Lagrangian multiplier is introduced, similar to the model-free trust-region algorithms [53, 54, 2].

Back-Propagation Through Time. BPTT [30, 64] is a first-order model-based policy optimization framework based on pathwise gradient (or reparameterization gradient) [58]. There are also several variants including Stochastic Value Gradients (SVG) [18], Model-Augmented Actor-Critic (MAAC) [9], and Probabilistic Inference for Learning COntrol (PILCO) [12]. Specifically, the policy parameters are updated by directly computing the derivatives of the performance with respect to the parameters. When the optimization of objective function is constrained, the accumulating step (Algorithm 3

Algorithm 2 Dyna Model-Based Policy Optimization

Input: Policy π, model set $\{f\}$, objective function $\mathcal{J}$.

1: Initialize a simulation data buffer $\widehat{\mathcal{D}}$
2: Sample a batch of initial states from the initial distribution ζ
3: ▷ Data simulation
4: **for** initial state sample s_0 **do**
5: **for** model f in model set $\{f\}$ **do**
6: **for** timestep $h = 1, ..., H$ **do**
7: Sample action $\widehat{a}_h \sim \pi(\cdot|\widehat{s}_h)$
8: Sample simulation state $\widehat{s}_{h+1} \sim f(\widehat{s}_h, \widehat{a}_h)$
9: Append simulation data to buffer $\widehat{\mathcal{D}} = \widehat{\mathcal{D}} \cup (\widehat{s}_h, \widehat{a}_h, r_h, \widehat{s}_{h+1})$
10: **end for**
11: **end for**
12: **end for**
13: ▷ Policy optimization with any model-free algorithm `ModelFree`
14: Objective optimization of policy on the simulated data $\pi \leftarrow$ `ModelFree`$(\widehat{\mathcal{D}}, \pi)$

Line 9) can be $L \leftarrow L + \gamma^h r(\widehat{s}_h, \widehat{a}_h) - \lambda D_{\text{KL}}$, where λ is the Lagrangian multiplier and D_{KL} is the corresponding KL constraint.

Algorithm 3 Model-Based Back-Propagation Policy Optimization

Input: Policy π, model set $\{f\}$, objective function $\mathcal{J}$.

1: Initialize a simulation data buffer $\widehat{\mathcal{D}}$
2: Start from initial state s_0
3: Reset $L \leftarrow 0$
4: ▷ Data simulation
5: **for** model f in model set $\{f\}$ **do**
6: **for** timestep $h = 1, ..., H$ **do**
7: Sample action $\widehat{a}_h \sim \pi(\cdot|\widehat{s}_h)$
8: Sample simulation state $\widehat{s}_{h+1} \sim f(\widehat{s}_h, \widehat{a}_h)$
9: Accumulate reward and constraint to L
10: **end for**
11: **end for**
12: ▷ Policy optimization
13: Compute policy gradient with back-propagation through time
14: Objective optimization of policy $\pi \leftarrow$ `PolicyGradient`

Model Predictive Control Policy Optimization. MPC is a *planning* framework that directly generates optimal action sequences under the model. Different from the above model-augmented policy optimization methods, MPC policy optimization directly generates optimal action sequences under the model and then distills the policy. Specifically, the pseudocode in Algorithm 4 begins with initial actions generated by the policy. Then with a shooting method, e.g., the cross-entropy method (CEM), the actions are refined and the policy that generates these optimal actions are distilled. Below, the algorithm to obtain the refined actions `EliteActions` can be CEM with action noise added to the action or policy parameter, i.e., POPLIN-A and POPLIN-P in [63]. The policy can be updated by `UpdatePolicy` using behavior cloning.

Policy Iteration for Tabular MDPs. In tabular settings where the state space $\mathcal{S}$ and action space $\mathcal{A}$ are discrete and countable, we can perform policy iteration under each model in the model set $\{f\}$. Here, the model is the tabular representation instead of function approximators. Based on the state-action values under various models, the optimal action at each state is the one that maximizes the weighted average of the values within the constraint of total variation distance.

Algorithm 4 Model Predictive Control Policy Optimization

Input: Policy π, model set $\{f\}$, objective function $\mathcal{J}$, algorithm to update actions `EliteActions`, algorithm to update policy `UpdatePolicy`.

Start from initial state s_0
Reset $J \leftarrow 0$
▷ Model-based planning
for model f in model set $\{f\}$ **do**
  for timestep $h = 1, ..., H$ **do**
    Sample action $\widehat{a}_h \sim \pi(\cdot|\widehat{s}_h)$
    Sample simulation state $\widehat{s}_{h+1} \sim f(\widehat{s}_h, \widehat{a}_h)$
    Accumulate reward and constraint to J
  end for
end for
$\mathbf{a} \leftarrow$ `EliteActions`$(J, \widehat{a}_{1:N})$
▷ Policy distillation
$\pi \leftarrow$ `UpdatePolicy`$(\mathbf{a})$

F Experimental Settings and Results in N-Chain MDPs

F.1 Settings of MuJoCo Experiments

In the MuJoCo experiments, we use a 5-layer neural network to approximate the dynamical model. We use deterministic ensembles [8] to capture the model epistemic uncertainty. Specifically, different ensembles are learned with independent transition data to construct the 1-step ahead confidence interval at every timestep. Each ensemble is separately trained using Adam [26]. And the number of ensemble heads can be set to 3, 4, or, 5, each of which is shown to be able to provide considerable performance in our experiments. All the experiments are repeated with 6 random seeds.

Since neural networks are not calibrated in general, i.e., the model uncertainty set is not guaranteed to contain the real dynamics, we follow HUCRL [10] to re-calibrate [29] the model. Our MuJoCo code is also built upon the HUCRL GitHub repository.

When using the Dyna model-based policy optimization, the number of gradient steps for each optimization procedure in an iteration is set to 20. And we empirically find that the KL divergence (or total variance) constraint makes the algorithm more efficient when computing the argmax in the optimization step, since optimizing from π_{t-1} at iteration t needs fewer policy gradient steps if the policy update is constrained within a certain trust region.

The task-specific and task-common settings and parameters are listed below in Table 1.

Table 1: Experimental parameters.

	Inverted Pendulum	Pusher	Half-Cheetah
episode length H	200	150	1000
dimension of state	4	23	18
dimension of action	1	7	6
action penalty	0.001	0.1	0.1
hidden nodes	(200, 200, 200, 200, 200)		
activation function	Swish		
optimizer	Adam		
learning rate	10^{-3}		

F.2 Experiments in N-Chain MDPs

Besides the experiments in MuJoCo, we also conduct tabular experiments in the N-Chain environment that is proposed in [37]. Specifically, there are in total 2 actions and N states in an MDP. The initial state is s_1 and the agent can choose to go *left* or *right* at each of the N states. The *left* action always succeeds and moves the agent to the left state, giving reward $r \sim \mathcal{N}(0, \delta^2)$. Taking the *right* action at

state $s_1, \ldots, s_{N-1}$ gives reward $r \sim \mathcal{N}(-\delta, \delta^2)$ and succeeds with probability $1 - 1/N$, moving the agent to the right state and otherwise moving the agent to the left state. Taking the *right* action at s_N gives reward $r \sim \mathcal{N}(1, \delta^2)$ and moves the agent back to s_1 with probability $1 - 1/N$.

We set $\delta = 0.1 \exp(-N/4)$, such that going *right* is the optimal action at least up to $N = 40$. As the number of states N is increasing, the agent needs *deep* exploration (e.g. guided by uncertainty) instead of *dithering* exploration (e.g. epsilon-greedy exploration), such that the agent can keep exploring despite receiving negative rewards [45].

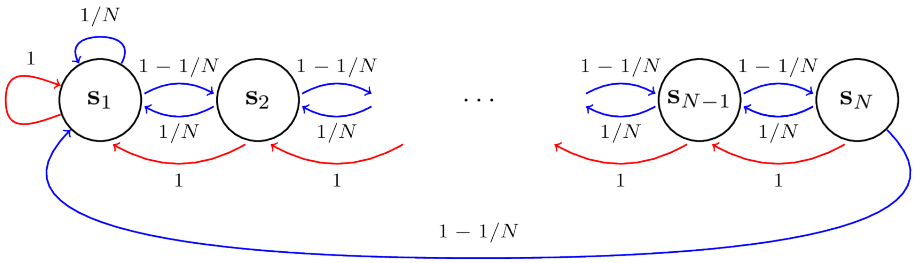


Figure 6: Illustration of the N-Chain MDP. Blue arrows correspond to action *right* (optimal) and red arrows correspond to action *left* (suboptimal). The figure is copied from [37].

For this reason, we evaluate the proposed algorithm CDPO and compare it with other Bayesian RL algorithms, including Bayesian Q-Learning (BQL) [11], Posterior Sampling for RL (PSRL) [42], the Uncertainty Bellman Equation (UBE) [46] and Moment Matching (MM) approach [37]. For CDPO, the dual optimization steps are solved by policy iteration, and the conservative update is performed within the total variation distance $\eta = 0.2$ (c.f. Policy Iteration for Tabular MDPs in Appendix E). We choose conjugate priors to represent the posterior distribution: we use a Categorical-Dirichlet model for discrete transition distribution at each (s, a), and a Normal-Gamma (NG) model for continuous reward distribution at each (s, a, s').

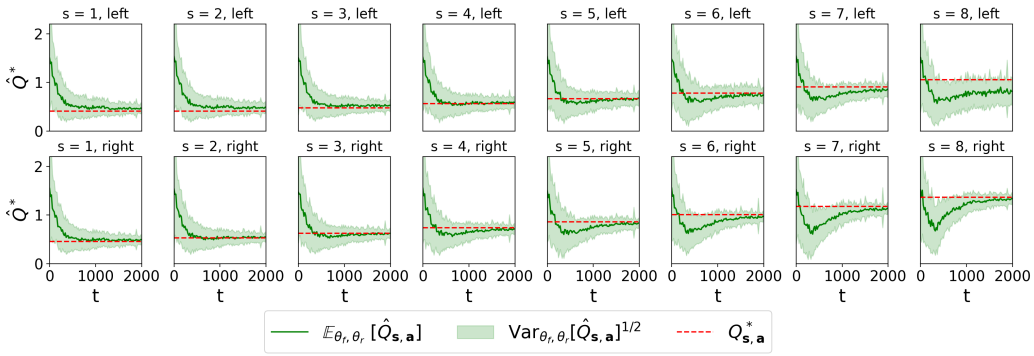


Figure 7: Posterior evolution of CDPO algorithm in the 8-Chain MDP.

Evolution of Posterior. Figure 7 demonstrates the evolution of the posterior of the CDPO algorithm in an 8-Chain MDP. As training progresses the posteriors concentrate on the true optimal state-action values and the behavior policy converges on the optimal one. The fast reduction of uncertainty is central to achieving principled and efficient exploration.

Compared to the posterior evolution of the PSRL algorithm corresponding to the optimal actions, i.e. the bottom row of curves in Figure 8, the expected value estimates of CDPO are closer to the ground-truth, and the variance is also smaller. Notably, the variance of CDPO might be higher for suboptimal actions, e.g., $s = 8, a =$ left (the last image of the first row in Figure 7). It is due to the conservative nature of CDPO that it only cares about the *expected* value, instead of the value of a sampled (imperfect) model as in PSRL. In other words, as long as the uncertainty is large, the PSRL agents can take suboptimal actions to explore the uninformative regions, which causes the inefficient over-exploration issue.

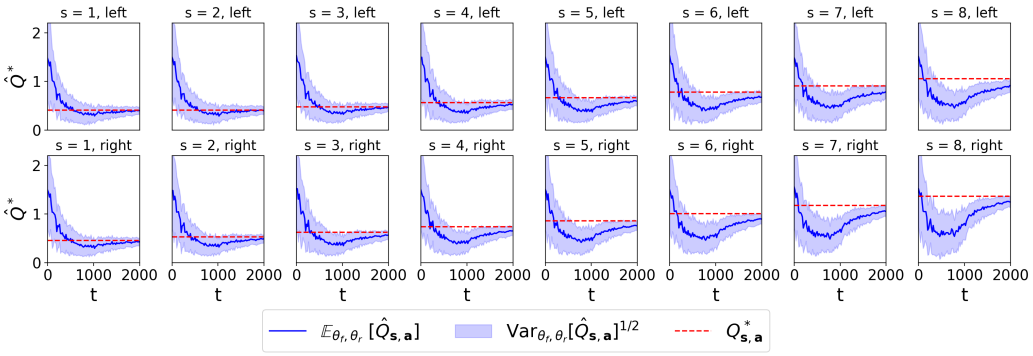
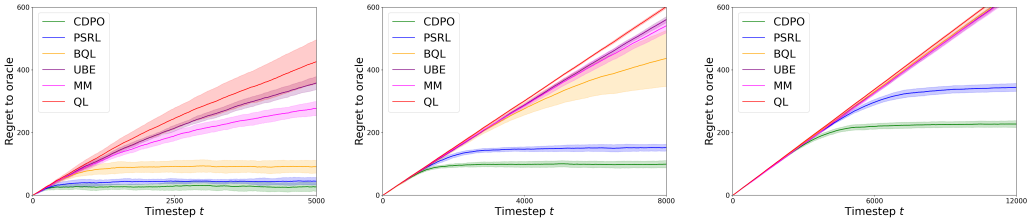


Figure 8: Posterior evolution of PSRL algorithm in the 8-Chain MDP.

Cumulative Regret. We compare CDPO and previous algorithms on the N-Chain MDPs with various state sizes N by measuring the cumulative regret of an oracle agent following the optimal policy. The results are shown in Figure 9. To make the performances comparable on the same scale, we also provide the normalized regret in Figure 10.

We observe that when the size of state space N is relatively smaller, e.g. $N \leq 5$, CDPO, PSRL, BQL, and MM algorithms achieve sublinear regret. The performances of these algorithms are also comparable, showing the necessity of deep exploration. On the contrary, Q-Learning which only relies on dithering exploration mechanisms fail to find the optimal strategy. However, as N is increasing, where the exploration must be effective for the agent to continually explore despite receiving negative rewards, the CDPO agents offer significantly lower cumulative regret and faster convergence.



(a) 5-Chain. (b) 10-Chain. (c) 15-Chain.

Figure 9: Comparison of cumulative regret.

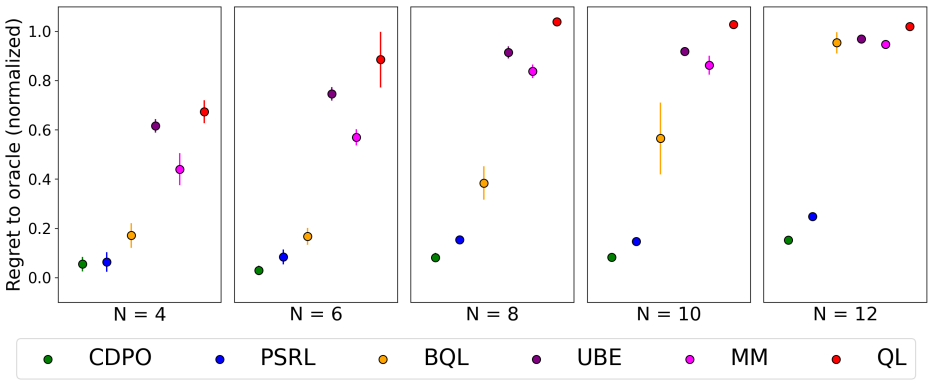


Figure 10: Performance comparison in terms of regret to the oracle.

G Algorithmic Comparisons between MBRL Algorithms

We provide algorithmic comparisons of four MBRL frameworks, including greedy model exploitation algorithms, OFU-RL, PSRL, and the proposed CDPO algorithm.

The differences mainly lie in the model selection and policy update procedures. The high-level pseudocode is given in Algorithm 5, 6, 7 and 8. Among them, the greedy model exploitation algorithm is a naive instantiation, where other instantiations can include the ones that augment Algorithm 5 with e.g., a dual framework that involves a locally accurate model and a supervised imitating procedure [59, 31]. In Algorithm 5, $\widetilde{f}_t$ can either be a probabilistic model or a deterministic model (with additive noise), which can be estimated via Maximum Likelihood Estimation (MLE) or minimizing the Mean Squared Error (MSE), respectively.

Algorithm 5 Naive Greedy Model Exploitation

for iteration $t = 1, ..., T$ **do**
  Estimate model $\widetilde{f}_t$ via MLE or MSE
  Compute $\pi_t = \operatorname{argmax}_\pi V_\pi^{\widetilde{f}_t}$
  Execute π_t in the real MDP
  $\mathcal{H}_{t+1} = \mathcal{H}_t \cup \{s_{h,t}, a_{h,t}, s_{h+1,t}\}_h$
end for
return policy π_T

Algorithm 6 OFU-RL Algorithm

for iteration $t = 1, ..., T$ **do**
  Construct confidence set $\mathcal{F}_t$
  Compute $\pi_t = \operatorname{argmax}_{\pi, f \sim \mathcal{F}_t} V_\pi^{f_t}$
  Execute π_t in the real MDP
  $\mathcal{H}_{t+1} = \mathcal{H}_t \cup \{s_{h,t}, a_{h,t}, s_{h+1,t}\}_h$
end for
return policy π_T

Algorithm 7 PSRL Algorithm

for iteration $t = 1, ..., T$ **do**
  Sample $f_t \sim \phi(\cdot \mid \mathcal{H}_t)$
  Compute $\pi_t = \operatorname{argmax}_\pi V_\pi^{f_t}$
  Execute π_t in the real MDP
  $\mathcal{H}_{t+1} = \mathcal{H}_t \cup \{s_{h,t}, a_{h,t}, s_{h+1,t}\}_h$
end for
return policy π_T

Algorithm 8 CDPO Algorithm

for iteration $t = 1, ..., T$ **do**
  Referential Update q_t following (4.1)
  Conservative Update π_t following (4.2)
  Execute π_t in the real MDP
  $\mathcal{H}_{t+1} = \mathcal{H}_t \cup \{s_{h,t}, a_{h,t}, s_{h+1,t}\}_h$
end for
return policy π_T

H Societal Impact

For real-world applications, interactions with the system imply energy or economic costs. With practical efficiency, CDPO reduces the training investment and is aligned with the principle of responsible AI. However, as an RL algorithm, CDPO is unavoidable to introduce safety concerns, e.g., self-driving cars make mistakes during RL training. Although CDPO does not explicitly address them, it may be used in conjunction with safety controllers to minimize negative impacts, while drawing on its powerful MBRL roots to enable efficient learning.