PROVABLE REPRESENTATION WITH EFFICIENT PLANNING FOR PARTIALLY OBSERVABLE REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

In real-world reinforcement learning, state information is often only partially observable, which breaks the Markov decision process assumption and leads to inferior performance for algorithms that conflate observations with state. Partially Observable Markov Decision Processes (POMDPs), on the other hand, provide a general framework that allows for partial observability to be accounted for in *learning, exploration and planning*, but presents significant computational and statistical challenges. To address these difficulties, we develop a representation-based perspective that leads to a coherent framework and tractable algorithmic approach for practical reinforcement learning from partial observations.

1 INTRODUCTION

Reinforcement learning (RL) addresses the problem of making sequential decisions that maximize a cumulative reward through interaction and observation in an environment (Mnih et al., 2013; Levine et al., 2016). The Markov decision process (MDP) has been the standard mathematical model used for most RL algorithm design. However, the success of MDP-based RL algorithms (Uehara et al., 2021; Zhang et al., 2022; Ren et al., 2023c) relies on an assumption that state information is fully observable, which implies that the optimal policy is memoryless, *i.e.*, optimal actions can be selected based only on the current state (Puterman, 2014). However, such an assumption typically does not hold in practice. For example, in (Mnih et al., 2013; Jiang et al., 2021) only images and dialogues are observed, from which the state information only can be partially inferred. The violation of full observability can lead to significant performance degeneration of MDP-based RL algorithms.

The Partially Observed Markov Decision Process (POMDP) (Åström, 1965) has been proposed to extend the the classical MDP formulation by introducing observation variables that only give partial information about the underlying latent state (Hauskrecht & Fraser, 2000; Roy & Gordon, 2002; Chen et al., 2016). This extension greatly expands applicability of POMDPs over MDPs, but the additional uncertainty modeling creates a non-Markovian dependence between successive observations, even though Markovian dependence is preserved between latent states. Consequently, the optimal policy for a POMDP is no longer memoryless but *entire-history* dependent, expanding state complexity exponentially w.r.t. horizon length. Such a non-Markovian dependence creates a significant computational and statistical complexity challenges in *learning, exploration* and *planning*. In fact, without additional assumptions, computing an optimal policy for a given POMDP with known dynamics (*i.e.*, planning) is PSPACE-complete (Papadimitriou & Tsitsiklis, 1987), while the sample complexity of learning for POMDPs grows exponentially w.r.t. the horizon (Jin et al., 2020a).

Despite the worst case hardness of POMDPs, due to their importance there has been extensive work on developing practical RL algorithms that can cope with partial observations. One common heuristic is to extend MDP-based RL algorithms by maintaining a history window over observations to encode a policy or value function, *e.g.*, recurrent neural networks (Wierstra et al., 2007; Hausknecht & Stone, 2015; Zhu et al., 2017). Such algorithms have been applied to many real-world applications with image- or text-based observations (Berner et al., 2019; Jiang et al., 2021), sometimes even surpassing human-level performance (Mnih et al., 2013; Kaufmann et al., 2023).

Such empirical successes have motivated investigation into *structured* POMDPs that allow some of the core computational and statistical complexities to be overcome, which provides an improved understanding of exploitable structure, and practical new algorithms with rigorous justification. For example, the concept of *decodability* has been used to express POMDPs where the latent state can

be exactly recovered from a window of past observations (Efroni et al., 2022; Guo et al., 2023). *Observability* is another special structure, where the m-step emission model is assume to be full-rank, allowing the latent state to be identified from m future observation sequences (Jin et al., 2020a; Golowich et al., 2022; Liu et al., 2022; 2023). Such structures eliminate unbounded history dependence, and thus, reduce the computational and statistical complexity. However, most works rely on the existence of an ideal computational oracle for planning, which, unsurprisingly, is infeasible in practice. Although there have been a few attempts to overcome the computational complexity of POMDPS, these algorithms are either only applicable to the tabular setting (Golowich et al., 2022) or rely on integrations that quickly become intractable for large observation spaces (Guo et al., 2018).

How can we design an **efficient** *and* **practical** *RL algorithm for* **structured** *partial observations?*

By "efficient" we mean the statistical and computational complexity avoids an exponential dependence on history length, while the computational components of *learning, planning* and *exploration* are computationally feasible; while by "practical" we mean that every component of an algorithm can be easily implemented and applied to a real-world problem. In this paper, we provide a affirmative answer to these questions. More specifically,

- We show that a linear structured POMDP admits a sufficient representation, *Multi-step Latent Variable Representation (μLV-Rep)*, that supports exact and tractable representation of the value function (Section 4.1), breaking the computational barriers explained in Section 3.
- We design a computationally efficient planning algorithm that can implement both the principles of optimism and pessimism in the face of uncertainty for online and offline POMDPs, respectively, upon the learned sufficient representation μLV-Rep (Section 4.2).
- We provide a theoretical analysis of the sample complexity of the structured POMDP, justifying the efficiency in balancing exploitation versus exploration in Section 5.
- We conduct a comprehensive comparison to current existing RL algorithms for POMDPs on several benchmarks, demonstrating the superior performance of μLV-Rep (Section 7).

2 PRELIMINARIES

We follow the definition of partially observable Markov decision process (POMDP) in (Efroni et al., 2022; Liu et al., 2022; 2023), which can be formally denoted as a tuple $\mathcal{P} = (\mathcal{S}, \mathcal{A}, \mathcal{O}, r, H, \rho_0, \mathbb{P}, \mathbb{O})$, where $\mathcal{S}$ represents the state space, $\mathcal{A}$ represents the action space, and $\mathcal{O}$ represents the observation space. The positive integer H denotes the horizon length, ρ_0 is the initial state distribution, $r : \mathcal{O} \times \mathcal{A} \to [0, 1]$ is the reward function, $\mathbb{P}(\cdot|s, a) : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition kernel capturing state dynamics, and $\mathbb{O}(\cdot|s) : \mathcal{S} \to \Delta(\mathcal{O})$ is the emission kernel.

Initially, the agent starts at a state s_0 drawn from $\rho_0(s)$. At each step h, the agent selects an action a from $\mathcal{A}$. This leads to the generation of a new state s_{h+1} following the distribution $\mathbb{P}(\cdot|s_h, a_h)$, and the agent observes o_{h+1} according to $\mathbb{O}(\cdot|s_{h+1})$. The agent also receives a reward $r(o_{h+1}, a_{h+1})$. Since the observations are partially observable, the transition between observations is non-Markovian, which means we need to consider policies $\pi_h : \mathcal{O} \times (\mathcal{A} \times \mathcal{O})^h \to \Delta(\mathcal{A})$ that depend on the entire history, denoted by $\tau_h = \{o_0, a_0, \cdots, o_h\}$. The value associated with policy $\pi = \{\pi_h\}_{h\in[H]}$ with $[H] := \{0, \ldots, H\}$ is defined as $V^\pi = \mathbb{E}_\pi \left[\sum_{h\in[H]} r(o_h, a_h)\right]$, and the goal is to find the optimal policy $\pi^* = \arg\max_\pi V^\pi$. Note that, the Markov Decision Process (MDP), given by $\mathcal{M} = (\mathcal{S}, \mathcal{A}, r, H, \rho_0, \mathbb{P})$, is a special case of a POMDP, where the state space $\mathcal{S}$ is equivalent to the observation space $\mathcal{O}$, and the emission kernel $\mathbb{O}(o|s)$ is defined as $\delta(o = s)$.

Define the belief $b : \mathcal{O} \times (\mathcal{A} \times \mathcal{O})^h \to \Delta(\mathcal{S})$. Let $b(s_1|o_1) = \mathbb{P}(s_1|o_1)$. We can recursively compute

$$b(s_{h+1}|\tau_{h+1}) \propto \int_{\mathcal{S}} b(s_h|\tau_h)\mathbb{P}(s_{h+1}|s_h, a_h)\mathbb{O}(o_{h+1}|s_{h+1})ds_h.$$

With such definition, one can convert a POMDP to an equivalent belief MDP, denoted as $\mathcal{M}_b = (\mathcal{B}, \mathcal{A}, R_h, H, \mu_b, T_b)$, where $\mathcal{B} \subseteq \Delta(\mathcal{S})$ represents the set of possible beliefs, and

$$\mu_b = \int b(s|o_1)\mu(o_1)do_1, \ \mathbb{P}_b\left(b_{h+1}|b_h, a_h\right) = \int \mathbf{1}_{b_{h+1}=b(\tau_h,a_h,o_{h+1})}\mathbb{P}(o_{h+1}|b_h, a_h)do_{h+1}. \quad (1)$$

Here we use b with subscript as one element of $\mathcal{B}$ and use b itself as the mapping $b : \mathcal{O} \times (\mathcal{A} \times \mathcal{O})^h \to \Delta(\mathcal{S})$. We emphasize that each belief corresponds to a density measure defined over the state space. Therefore, $\mathbb{P}_b\left(\cdot|b_h, a_h\right)$ represents a conditional operator that characterizes the transition between belief distributions. Considering a policy $\pi : \mathcal{B} \to \Delta(\mathcal{A})$, we can define the state value function

$V_h^\pi(b_h)$ and state-action value function $Q_h^\pi(b_h, a_h)$ for the belief MDP:

$$V_h^\pi(b_h) = \mathbb{E}\left[\sum_{t=h}^H r(o_t, a_t)|b_h\right], \quad Q_h^\pi(b_h, a_h) = \mathbb{E}\left[\sum_{t=h}^H r(o_t, a_t)|b_h, a_h\right]. \tag{2}$$

Adopting the perspective of the equivalent belief MDPs, we can express the Bellman equation as:

$$V_h^\pi(b_h) = \mathbb{E}_\pi\left[Q_h^\pi(b_h, a_h)\right], \quad Q_h^\pi(b_h, a_h) = r(o_h, a_h) + \mathbb{E}_{\mathbb{P}_b}\left[V_{h+1}^\pi(b_{h+1})\right]. \tag{3}$$

Although it is still feasible to utilize a dynamic programming approach and apply the Bellman equation (3) to solve a POMDP, it is crucial to recognize that the dependence of the belief on either the *entire history* results in the possibility of a large or the infinite number of beliefs, even when the number of states is finite, hence leading to infeasible computational and statistical complexity (Papadimitriou & Tsitsiklis, 1987; Jin et al., 2020a). Incorporating function approximation into the learning and planning in a POMDP is significantly more involved than in an MDP.

Consequently, several special structures has been introduced to reduce the statistical complexity of a POMDP. Specifically, *L-decodability* and *γ-observability* have been introduced in (Du et al., 2019; Efroni et al., 2022) and (Golowich et al., 2022; Even-Dar et al., 2007), respectively.

Definition 1 (L-step decodability (Efroni et al., 2022)). *$\forall h \in [H]$, define*

$$x_h \in \mathcal{X} := (\mathcal{O} \times \mathcal{A})^{L-1} \times \mathcal{O}, \quad x_h = (o_{h-L+1}, a_{h-L+1}, \cdots, o_h), \tag{4}$$

and there exists a decoder $p^* : \mathcal{X} \to \Delta(\mathcal{S})$, such that $p^*(x_h) = b(\tau_h)$.

Definition 2 (γ-observability (Golowich et al., 2022; Even-Dar et al., 2007)). *Denote $\langle \mathbb{O}, b\rangle := \int \mathbb{O}_h(\cdot|s)\, b(s)\, ds$, for arbitrary beliefs b and b' over states, $\|\langle \mathbb{O}, b\rangle - \langle \mathbb{O}, b'\rangle\|_1 \geqslant \gamma \|b - b'\|_1$.*

Note that we slightly generalize decodability from Efroni et al. (2022), which assumes $b(\tau_h)$ is a Dirac measure on $\mathcal{S}$. It is worth noticing that γ-observability and L-decodability are highly related. Existing works have shown that a γ-observable POMDP can be well approximated by a decodable POMDP with a history of proper length L (Golowich et al., 2022; Uehara et al., 2022; Guo et al., 2023) (see Appendix B for a detailed discussion). Hence, in the main text, we focus on algorithm design for an L-decodable POMDP, which can be directly extended to a γ-observable POMDP.

3 Difficulties in POMDP Planning from a Representation View

Obviously, such structures for a POMDP reduce the history dependence, and thus reduce statistical complexity. However, the computational tractability of planning and exploration given such structures remains open. Before attempting a practical algorithm for a structured POMDP, we first explain the the benefits of a linear structure representation, and the challenges in being applied for POMDPs.

Linear Structure of MDPs. Linear structures for MDPs were introduced in (Jin et al., 2020b; Yang & Wang, 2020; Ren et al., 2022; Uehara et al., 2021) to enable effective function approximation and address the core computational challenges of planning and exploration in general nonlinear control. Such an approach leverages the spectral factorization of transition kernel and reward, given by:

$$\mathbb{P}(s'|s, a) = \langle \phi(s, a), \mu(s')\rangle_{\mathcal{H}}, \; r(s, a) = \langle \phi(s, a), \theta\rangle_{\mathcal{H}}, \tag{5}$$

where $\phi : \mathcal{S} \times \mathcal{A} \to \mathcal{H}$ and $\mu : \mathcal{S} \to \mathcal{H}$ are feature maps to a Hilbert space $\mathcal{H}$. Under this factorization, the state-action value function Q^π for an arbitrary policy π can be represented as:

$$Q_h^\pi(s, a) = r(s, a) + \int V_{h+1}^\pi(s')\mathbb{P}(s'|s, a)ds' = \left\langle \phi(s, a), \theta + \int_{\mathcal{S}} V_{h+1}^\pi(s')\mu(s')ds' \right\rangle_{\mathcal{H}}, \tag{6}$$

where the first equation comes from the Bellman recursion, and the second equation is obtained by plugging in (5). This result implies that instead of dealing with a complex function space defined on the raw state space, one can design computationally efficient planning and exploration algorithms in the space linearly spanned by ϕ. In fact, based on the correspondence between policies and Q-functions discussed in Ren et al. (2023c), ϕ can be interpreted as representing primitives for constructing a skill set. Zhang et al. (2022); Qiu et al. (2022) took initial strides in developing efficient and practical algorithms that harness the linear structure of MDPs. Building upon their work, Ren et al. (2023b) observed that the transition kernel can be formulated with a latent variable model:

$$\mathbb{P}(s'|s, a) = \int_{\mathcal{Z}} p(z|s, a)p(s'|z)d\mu = \langle p(\cdot|s, a), p(s'|\cdot)\rangle_{L_2(\mu)} \tag{7}$$

with the linear structure over conditional distributions $p(\cdot|s, a) \in L_2(\mu)$, $p(s'|\cdot) \in L_2(\mu)$ for the Lebesgue measure μ. This linear structure can be leveraged to design a practical representation learning algorithm similar to the Dreamer-style algorithm (e.g. Hafner et al., 2021), but with more theoretically sound planning and exploration mechanisms.

Difficulties for POMDPs. Inspired by the successes of linear structure for MDPs, it is natural to consider an extension to POMDPs. However, limited progress had been made in exploiting the linear

structure of POMDPs (Guo et al., 2023). Specifically, for an arbitrary policy π, consider the Bellman equation:

$$Q_h^\pi(b_h, a_h) = r(o_h, a_h) + \mathbb{E}_{\mathbb{P}(o_{h+1}|b_h,a_h)}\left[V_{h+1}^\pi(b(\tau_h, a_h, o_{h+1}))\right] \tag{8}$$

The second equation comes from the belief transition (1). Straightforwardly applying the linear structure in the latent state transition, we obtain

$$Q_h^\pi(b_h, a_h) = r(o_h, a_h) + \mathbb{E}_{b_h(s)}\left[\int \mathbb{P}(s_{h+1}|s_h, a_h)\mathbb{E}_{O(o_{h+1}|s_{h+1})}\left[V_{h+1}^\pi(b(\tau_h, a_h, o_{h+1}))\right]\right] \tag{9}$$

$$= r(o_h, a_h) + \left\langle \int b_h(s)\phi(s, a)\, ds, \int \mu(s_{h+1})\, \mathbb{E}_{O(o_{h+1}|s_{h+1})}\left[V_{h+1}^\pi(b(\tau_h, a_h, o_{h+1}))\right] ds_{h+1} \right\rangle.$$

From (9), one can see the major computational difficulties marked red as:

- **i)** the representation is $\int b_h(s)\phi(s, a)\, ds$, which requires not only belief, but also an *integration* with an unknown $\phi(s, a)$ upon the belief, therefore, is difficult to estimate; more importantly,
- **ii)** the $b(\tau_h, a_h, o_{h+1})$ inside the *nonlinear* $V_{h+1}^\pi(\cdot)$ depends on the history, hence the integration will be a function of history, rather than a vector, which breaks the linear structure of Q_h^π.

These difficulties have made the usage of linear structure in POMDPs highly non-trivial, even with L-step decodability (Guo et al., 2023).

4 Multi-step Latent Variable Representation

We now discuss how to leverage the structure of L-step decodable POMDPs to bypass the two difficulties revealed above, and eventually develop a **computationally efficient** planning algorithm. Our key observation is that, the value function for an L-step decodable POMDP only depends on the last L-step observations, and for a policy family, they can be *linearly* represented by an L-step *latent variable representation*, therefore, bypass the beliefs. We then show how to design a latent variable representation learning and planning algorithm upon (Ren et al., 2023b), which leads to *Multi-step Latent Variable Representation (μLV-Rep)*.

4.1 Key Observations

Although the equivalent belief MDP provides a Markovian Bellman recursion (8), as we discussed in Section 3, the linear structure does not directly introduce tractability in planning.

To resolve such difficulties in a belief MDP, we make a first key observation, *i.e.*, **an observation-based value function will bypass the necessity for belief computation.** Specifically, $b_h(\cdot)$ is a mapping from the history τ_h to the space of probability densities over state. Therefore, we rewrite $Q_h^\pi(b_h, a_h) = Q_h^\pi(\tau_h, a_h)$. With this simply reformulation, we avoid explicit dependence on beliefs in the Q^π-function, eliminating the first difficulty.

Next, we make a second key observation, *i.e.*, **by the definition of L-step decodability in a POMDP, it is sufficient to consider the L-step memory x_h, instead of the entire history τ_h.** This can be easily verified from the L-step decodability definition 1. Specifically, because of L-step decodability, the beliefs $b_h(s)$ can be represented with a decoder from L-step windows over x_h, although the decoder is unknown, leading to $Q_h^\pi(\tau_h, a_h) = Q_h^\pi(x_h, a_h)$. Directly inserting $Q_h^\pi(x_h, a_h)$ into the Bellman equation (8) yields

$$Q_h^\pi(x_h, a_h) = r(o_h, a_h) + \mathbb{E}_{\mathbb{P}^\pi(x_{h+1}|x_h)}\left[V_{h+1}^\pi(x_{h+1})\right]. \tag{10}$$

This outcome reduces the statistical complexity, as previously exploited in (Efroni et al., 2022; Guo et al., 2023). However, the second difficulty remains, *i.e.*, there is an additional dependence of $V_{h+1}^\pi(x_{h+1})$ on (x_h, a_h), since $x_{h+1} = (o_{h-L+2}, a_{h-L+2}, \cdots, o_{h+1})$ has an overlap with $(x_h, a_h) = (o_{h-L+1}, a_{h-L+1}, o_{h-L+2}, a_{h-L+2}, \cdots, o_h, a_h)$. This overlap between the successive x_h and x_{h+1}, which are also known as mega-states (Efroni et al., 2022), breaks the low-rank structure, and thus, impedes directly extending low-rank MDP for POMDPs.

Recall that by L-step decodability, $V_{h+L}^\pi(x_{h+L})$ will be independent of (x_h, a_h). Therefore, our first attempt is to consider the L-step Bellman equation for $Q_h^\pi(\tau_h, a_h)$, which can be easily derived by expanding (8)

$$Q_h^\pi(x_h, a_h) = \mathbb{E}_{\pi_{h+1:h+L}|x_h,a_h}\left[\sum_{i=h}^{h+L-1} r(o_i, a_i) + V_{h+L}^\pi(x_{h+L})\right]. \tag{11}$$

At first glance, the L-step forward expansion induces $V_{h+L}^\pi(x_{h+L})$, which eliminates the overlapping dependence of x_{h+L} on (x_h, a_h), due to the L-step decodability. However, the remaining issue is that the policy $\pi_{h+1:h+L-1}$ still depends on some part of (x_h, a_h), which retains a dependence on $\mathbb{E}_{\pi_{h+1:h+L}|x_h,a_h}\left[V_{h+L}^\pi(x_{h+L})\right]$, and thus still breaks the linear structure for the Q_h^π-function. To

recover a linear structure for the Q^π-function, we introduce our most important observation, *i.e.*, **if we consider a policy ν_π, that conditioned on the sufficient h-step latent variable induced by the observation dynamics, as well as the future action-observation sequences, the dependence of $\pi_{h+1:h+L-1}$ on (x_h, a_h) at L-step can be eliminated.** Specifically, consider

$$\mathbb{P}^\pi\left(x_{h+L}|x_h,a_h\right)=\int p(z_{h+1}|x_h,a_h)\mathbb{P}^{\nu_\pi}\left(x_{h+L}|z_{h+1}\right)dz_{h+1}=\left\langle p(\cdot|x_h,a_h),\mathbb{P}^{\nu_\pi}\left(x_{h+L}|\cdot\right)\right\rangle_{L_2(\mu)}, \tag{12}$$

where z denotes the latent variable. This observation to eliminate the policy dependence on (x_h, a_h) has been discussed in (Efroni et al., 2022) as the *"moment matching policy"*. The existence of such sufficient h-step latent variable is guaranteed by the low rank structure, while the existence of the equivalent moment matching policy ν^π is guaranteed from L-decodability. Note that, the idea of the moment matching policy was only considered as a proof trick, and not previously exploited to reveal linear structure for algorithm design. The concrete moment matching policy is discussed in details in Appendix C.

One key difference between the factorization in (12) and the linear structure in (5) or (7) for an MDP is that in a linear MDP, one obtains a *policy-independent* decomposition, where both the components $\phi(s,a)$ and $\mu(s')$ from the transition dynamics are invariant w.r.t. the policy. Clearly, in (12) for a POMDP, one component from the obtained factorization, $\mathbb{P}^{\nu_\pi}\left(x_{h+L}|\cdot\right)$, depends on the policy. However, we will see that this does not affect the linear representation ability of $p\left(\cdot|x_h,a_h\right)$ for Q^π.

Remark (Identifiability): It should be noted that we deliberately use z as the latent variable, rather than s, in (12) to emphasize the learned latent variable structure can be different from the groundtruth state, hence without an *idenfitiability* assumption. Nevertheless, the learned structure has the same effect in representing Q^π linearly.

Now, we have established every component needed to derive the linear representation by introducing (12) into (11). For the first term in (11), for $\forall k \in \{1,\ldots,L-1\}$, we have

$$\mathbb{E}_\pi\left[r\left(o_{h+k},a_{h+k}\right)\right]=\left\langle p(\cdot|x_h,a_h),\underbrace{\int\mathbb{P}^{\nu_\pi}\left(o_{h+k},a_{h+k}|\cdot\right)r\left(o_{h+k},a_{h+k}\right)do_{h+k}da_{h+k}}_{w_k^\pi(\cdot)}\right\rangle. \tag{13}$$

With the "moment matching policy" trick, the ν_π is independent w.r.t. (x_h, a_h). Then, $\mathbb{P}^{\nu_\pi}\left(o_{h+k},a_{h+k}|\cdot\right)$ is independent to history, which leads to the linear representation in (13).

For the second term in (11), similarly, we have

$$\begin{aligned}\mathbb{E}_\pi\left[V_{h+L}^\pi\left(x_{h+L}\right)\right]&=\int\mathbb{P}^\pi\left(x_{h+L}|x_h,a_h\right)V^\pi\left(x_{h+L}\right)dx_{h+L}\\&=\left\langle p(\cdot|x_h,a_h),\underbrace{\int\mathbb{P}^{\nu_\pi}\left(x_{h+L}|\cdot\right)V^\pi\left(x_{h+L}\right)dx_{h+L}}_{w_{h+L}^\pi(\cdot)}\right\rangle.\end{aligned} \tag{14}$$

Recall that x_{h+L} does not have overlap with x_h, with the same "moment match policy" trick, $w_{h+L}^\pi(\cdot)$ is indepenent w.r.t. (x_h, a_h).

Together with (13) and (14), define $w^\pi=\sum_{k=h}^{h+L}w_{h+k}^\pi$, we can justify that under our key observations, for an L-step decodable POMDP, Q^π can be represented linearly in $p\left(\cdot|x_h,a_h\right)$ as

$$Q_h^\pi\left(x_h,a_h\right)=\left\langle p\left(\cdot|x_h,a_h\right),w^\pi(\cdot)\right\rangle_{L_2(\mu)}. \tag{15}$$

With assumption that $r\left(o_h,a_h\right)=\left\langle p(\cdot|x_h,a_h),\omega^r(\cdot)\right\rangle$, which is easy to achieve by feature augmentation (Ren et al., 2023a).

Connection to PSR (Littman & Sutton, 2001). Both the proposed μLV-Rep and the predictive state representation (PSR) (Littman & Sutton, 2001) bypass the explicit belief calculation by factorizing the observation transition system. However, there are significant differences between the structures, and hence in planning and exploration. Specifically, the PSR is based on the assumption that, for any finite sequence of events $y_{h+1:k}=\left(o_{h+1:h+k},a_{h:h+k-1}\right)$ upon history x_h with $k\in\mathbb{N}_+$, the probability can be linearly factorized as $\mathbb{P}\left(o_{h+1:h+k}|x_h,a_{h:h+k-1}\right)=\left\langle\omega_{y_{h+1:k}},\mathbb{P}\left(U|x_h\right)\right\rangle$, where $\omega_{y_{h+1:k}}\in\mathbb{R}^d$, $U:=[u_i]_{i=1}^d$ is a set of core test events, and $\mathbb{P}\left(U|x_h\right)$ is referred as the predictive state representation at h-step. Then, the forward observation dynamics can be represented in PSR via

Algorithm 1 Online Exploration for L-step decodable POMDPs with Latent Variable Representation

Input: Model Class $\mathcal{M} = \{\{(p_h(z|x_h, a_h), p_h(o_{h+1}|z)\}_{h\in[H]}\}$, Variational Distribution Class $\mathcal{Q} = \{\{q_h(z|x_h, a_h, o_{h+1})\}_{h\in[H]}\}$, Episode Number K.
Initialize $\pi_0^h(s) = \mathcal{U}(\mathcal{A}), \forall h \in [H]$ where $\mathcal{U}(\mathcal{A})$ denotes the uniform distribution on $\mathcal{A}$; $\mathcal{D}_{0,h} = \emptyset, \mathcal{D}'_{0,h} = \emptyset, \forall h \in [H]$.
for episode $k = 1, \cdots, K$ **do**
    Initialize $\mathcal{D}_{k,h} = \mathcal{D}_{k-1,h}$, $\mathcal{D}'_{k,h} = \mathcal{D}'_{k-1,h}$
    for Step $h = 1, \cdots, H$ **do**
        Collect the transition $(x_h, a_h, o_{h+1}, a_{h+1}, \cdots, o_{h+L-1}, a_{h+L-1}, o_{h+L})$ where $x_h \sim d_{\mathcal{P}}^{\pi_k,h}$, $a_{h:h+L-1} \sim \mathcal{U}(\mathcal{A})$, $o_{h+i} \sim \mathbb{P}^{\mathcal{P}}(\cdot|x_{h+i-1}, a_{h+i-1}), \forall i \in [L]$.
        $\mathcal{D}_{k,h} = \mathcal{D}_{k,h} \cup \{x_h, a_h, o_{h+1}\}$, $\mathcal{D}'_{k,h+i} = \mathcal{D}'_{k,h+i} \cup \{x_{h+i}, a_{h+i}, o_{h+i+1}\}, \forall i \in [L]$.
    end for
    Learn the latent variable model $\hat{p}_k(z|x_h, a_h)$ with $\mathcal{D}_{k,h} \cup \mathcal{D}'_{k,h}$ via maximizing the ELBO, and obtain the learned model $\widehat{\mathcal{P}}_k = \{(\hat{p}_{h,k}(z|x_h, a_h), \hat{p}_{h,k}(o_{h+1}|z))\}_{h\in[H]}$.
    (Optional) Set the exploration bonus $\hat{b}_{k,h}(s, a)$ with $\mathcal{D}_{k,h}$.
    Update policy $\pi_k = \arg\max_\pi V^\pi_{\widehat{\mathcal{P}}_k, r+\hat{b}_k}$.
end for
Return $\pi^1, \cdots, \pi^K$.

Bayes' rule, *i.e.*, $\mathbb{P}\left(o_{h+2:k}|x_h, a_{h:h+k-1}, o_{h+1}\right) = \frac{\left\langle \omega_{y_{h+2:k}}, \mathbb{P}(U|x_h)\right\rangle}{\left\langle \omega_{y_{h+1}}, \mathbb{P}(U|x_h)\right\rangle}$, which introduces a nonlinear operation, making the planning and exploration difficult.

4.2 Main Algorithm

We have revealed the linear representation for Q_h^π. In this section, we will discuss how we can learn the representation, and the planning and exploration procedure upon the representation. The full algorithm is presented in Algorithm 1.

Variational Learning of μLV-Rep. As we generally do not have the latent variable representation $p(\cdot|x_h, a_h)$ a priori, it is essential to perform the representation learning with online collected data. One straightforward idea is to apply maximum likelihood estimation on $\mathbb{P}^\pi\left(x_{h+k}|x_h, a_h\right)$. Although this is theoretically correct, due to the overlap on x_{h+k} and x_h, there will be parametrization issue, with the waste of memory and computation cost. Recall the fact that we only need $p\left(z_h|x_h\right)$ for representing Q_h^π, and the observation that for $\forall k \in \mathbb{N}_+$,

$$p\left(o_{h+1:h+l}|x_h, a_h\right) = \int_{\mathcal{Z}} p(z_h|x_h, a_h) \underbrace{\prod_{i=1}^{l}\left[\int_{\mathcal{Z}} \mathbb{P}^\pi(z_{h+i}|z_{h+i-1}, a_i)p(o_{h+i}|z_{h+i})dz_{h+i}\right]}_{\mathbb{P}^\pi(o_{h+1:h+l}|z_h)} dz_h, \tag{16}$$

we can obtain $p(\cdot|x_h, a_h)$ by performing maximum likelihood estimation (MLE) on $p\left(o_{h+1:h+l}|x_h, a_h\right)$ for arbitrary $l \in \mathbb{N}_+$. We exploit the evidence lower bound (ELBO) (Ren et al., 2023b) for a tractable surrogate of MLE of the latent variable model (16), *i.e.*,

$$\begin{aligned}
&\log p\left(o_{h+1:h+l}|x_h, a_h\right) = \log \int_{\mathcal{Z}} p(z_h|x_h, a_h)\mathbb{P}^\pi\left(o_{h+1:h+l}|z_h\right) \\
&= \log \int_{\mathcal{Z}} \frac{p(z_h|x_h, a_h)\mathbb{P}^\pi\left(o_{h+1:h+l}|z_h\right)}{q(z|x_h, a_h, o_{h+1:h+l})} q(z|x_h, a_h, o_{h+1:h+l}) \\
&= \max_{q\in\Delta(\mathcal{Z})} \mathbb{E}_{q(\cdot|x_h, a_h, o_{h+1:h+l})}\left[\log \mathbb{P}^\pi\left(o_{h+1:h+l}|z_h\right)\right] - D_{KL}\left(q(z|x_h, a_h, o_{h+1:h+l})||p(z_h|x_h)\right),
\end{aligned} \tag{17}$$

where the last equation comes from Jensen's inequality, with the equality holds when $q(z|x_h, a_h, o_{h+1:h+l}) \propto p(z_h|x_h, a_h)\mathbb{P}^\pi\left(o_{h+1:h+l}|z_h\right)$. One can use (17) with data to fit the μLV-Rep. For the ease of the presentation, we choose $l = 1$ in Algorithm 1.

Practical Parametrization of Q^π with μLV-Rep. With μLV-Rep, we can represent $Q_h^\pi\left(x_h, a_h\right) = \left\langle p(z|x_h), w_h^\pi(z)\right\rangle_{L_2(\mu)}$. If the latent variable z in $p(z|x_h)$ is an enumeratable discrete variable, $Q^\pi\left(x_h, a_h\right) = \sum_{i=m} w^\pi\left(z_i\right) p(z_i|x_h)$, can be simply represented.

However, the discrete latent variable is not differentiable, which may lead to some difficulty in learning. Therefore, continuous latent variable z will be used, which induces infinite-dimensional $w(z)$. We follow the trick in LV-Rep (Ren et al., 2023b) that we $Q^\pi\left(x_h, z_h\right)$ as an expectation,

$$Q^\pi\left(x_h, a_h\right) = \left\langle p(z|x_h), w^\pi(z)\right\rangle = \mathbb{E}_{p(z|x_h)}\left[w^\pi(z)\right]$$

which can be either approximated by Monte-Carlo method or random feature quadrature (Ren et al., 2023b), respectively,

$$Q^\pi(x_h, a_h) \approx \frac{1}{m}\sum_{i=1}^m w^\pi(z_i) \quad \text{or} \quad Q^\pi(x_h, a_h) \approx \frac{1}{m}\sum_{i=1}^m \tilde{w}^\pi(\xi_i)\varphi(z_i, \xi_i) \tag{18}$$

with samples $z_i \sim p(z|x_h)$ and $\xi_i \sim P(\xi)$ as the random feature measure for the RKHS space of $w(z)$. Both of these two approximation can be implemented by a neural network. Due to space limitation, we omit the derivation of the random feature quadrature. Please refer to Appendix E.

Planning and Exploration with μLV-Rep. With an accurate estimation of Q function, we can perform planning with the standard dynamic programming approach (e.g. Munos & Szepesvári, 2008). However, dynamic programming involves an $\arg\max$ operations, which can only be possible when $|\mathcal{A}| < \infty$. To deal with the continuous action scenarios, we can leverage the popular policy gradient methods like SAC (Haarnoja et al., 2018), with the critic parameterized with μLV-Rep.

To improve the exploration, we can leverage the idea of Uehara et al. (2021); Ren et al. (2023b) and add an additional ellipsoid bonus to implement the optimism in the face of uncertainty principle. Specifically, if we use the random feature quadrature, we can compute such bonus via:

$$\hat{\psi}_{h,k}(x_h, a_h) = [\varphi(z_i; \xi_i)]_{i\in[m]}, \quad \text{where} \quad \{z_i\}_{i\in[m]} \sim \hat{p}_{k,h}(z|x_h, a_h), \quad \{\xi_i\}_{i\in[m]} \sim P(\xi),$$
$$\hat{b}_{k,h}(s, a) = \alpha_k \hat{\psi}_{h,k}(x_h, a_h)\hat{\Sigma}_{k,h}^{-1}\hat{\psi}_{h,k}(x_h, a_h),$$

with $\hat{\Sigma}_{k,h} = \sum_{(x_{h,i}, a_{h,i})\in\mathcal{D}_{k,h}} \hat{\psi}_{k,h}(x_{h,i}, a_{h,i})\hat{\psi}_{k,h}(x_{h,i}, a_{h,i})^\top + \lambda I$, and α_k, λ are user-specified constants. Similarly, the bonus can be used for implementing the pessimism in the face of uncertainty principle in the offline setting, as we discussed in Appendix D, due to space limitation.

5 Theoretical Analysis

In this section, we provide a formal sample complexity analysis of the proposed algorithm. We start from the following assumptions, that are commonly used in the literature (e.g. Agarwal et al., 2020; Uehara et al., 2021; Ren et al., 2023b).

Assumption 1 (Finite Candidate Class with Realizability). *$|\mathcal{M}| < \infty$ and $\{(p_h^*(z|x_h, a_h), p_h^*(o_{h+1}|z))\}_{h\in[H]} \in \mathcal{M}$. Meanwhile, for all $(p_h(z|x_h, a_h), p(o_{h+1}|z)) \in \mathcal{M}$, $p_h(z|x_h, a_h, o_{h+1}) \in \mathcal{Q}$.*

Assumption 2 (Normalization Conditions). *$\forall \mathcal{P} \in \mathcal{M}, (x_h, a_h) \in \mathcal{X}\times\mathcal{A}, \|p_h(\cdot|x_h, a_h)\|_{\mathcal{H}_K} \leqslant 1$ for some kernel K. Furthermore, $\forall g : \mathcal{X} \to \mathbb{R}$ such that $\|g\|_\infty \leqslant 1$, we have $\left\|\int_{\mathcal{X}} p(x_{h+L}|\cdot)g(x_{h+L})dx_{h+L}\right\|_{\mathcal{H}_K} \leqslant C$.*

Now we are able to provide the sample complexity of μLV-Rep.

Theorem 3 (PAC Guarantee, Informal version of Theorem 13). *Assume the kernel K satisfies the regularity conditions in Appendix F.1. If we properly choose the exploration bonus $\hat{b}_k(x, a)$, we can obtain an ε-optimal policy with probability at least $1-\delta$ after we interact with the environments for $N = \text{poly}\left(C, H, |\mathcal{A}|^L, L, \varepsilon, \log(|\mathcal{M}|/\delta)\right)$ episodes.*

6 Related Work

Representation has been exploited in partially observable reinforcement learning, but for different purposes. Vision-based representations (Yarats et al., 2020; Seo et al., 2023) have been designed to extract compact feature from raw pixel observations. We emphasize that this type of observation feature does not explicitly capture dynamics properties, and essentially orthogonal to but naturally compatible with the proposed representation. Many dynamic-aware representation methods have been developed, such as bi-simulation (Ferns et al., 2004; Gelada et al., 2019; Zhang et al., 2020), successor features (Dayan, 1993; Barreto et al., 2017; Kulkarni et al., 2016), spectral representation (Mahadevan & Maggioni, 2007; Wu et al., 2018; Duan et al., 2019), and contrastive representation (Oord et al., 2018; Nachum & Yang, 2021; Yang et al., 2021). The proposed representation for POMDPs is inspired by recent progress (Jin et al., 2020b; Yang & Wang, 2020; Agarwal et al., 2020; Uehara et al., 2022) in revealing low-rank structure in the transition kernel of MDPs, and inducing effective linear representations for the state-action value function for an arbitrary policy. This prior discovery has led to a series of practical and provable RL algorithms in the MDP setting, achieving a delicate balance between learning, planning and exploration (Ren et al., 2022; Zhang et al., 2022; Ren et al., 2023c;b). Although these algorithms demonstrate theoretical and empirical benefits, they rely on the Markovian assumption, hence are not applicable to the POMDP setting we consider here.

There have been several attempts to exploit the low-rank representation in POMDPs to reduce the statistical complexity (Efroni et al., 2022). Azizzadenesheli et al. (2016); Guo et al. (2016) exploits

spectral learning for model estimation without exploration; Jin et al. (2020a) explores within an spectral estimation set; Uehara et al. (2022) builds upon the Bellman error ball; Zhan et al. (2022); Liu et al. (2022) consider the MLE confidence set for low-rank structured models; and (Huang et al., 2023) construct a UCB-type algorithm upon the MLE ball of PSR. However, these algorithms rely on intractable oracles for planning, and there are fewer works that consider exploiting low-rank structure to achieve computationally tractable planning. One exception is (Zhang et al., 2023; Guo et al., 2023), which still includes intractable operations, *i.e.*, infinite-dimensional operations or integrals.

7 Experimental Evaluation

We evaluate the proposed approach on RL tasks with partial observations, constructed based on the OpenAI gym MuJoCo (Todorov et al., 2012) and DeepMind Control Suites (Tassa et al., 2018), as well as Meta-world (Yu et al., 2019). For our implementation, we employ a continuous latent variable model similar to (Hafner et al., 2020), approximating distributions with Gaussians parameterized by their mean and variance. We use $L = 3$, so the representation is learned by making L-step predictions. We apply Soft Actor-Critic (SAC) as the planner (Haarnoja et al., 2018), where a multi-step critic objective is also adopted to improve learning efficiency (Hafner et al., 2020; Feinberg et al., 2018).We first compared the proposed algorithm in partially observable setting and image-based setting. We also provided the ablation study in Appendix H.1.

7.1 Partially Observable Continuous Control

The standard MuJoCo tasks from the OpenAI gym and DeepMind Control Suites are not partially observable. To generate partially observable problems based on these tasks, we adopt a widely employed technique of masking velocities within the observations (Ni et al., 2021; Weigand et al., 2021; Gangwani et al., 2020). In this way, it becomes impossible to extract complete decision-making information from a single environment observation, yet the ability to reconstruct the missing observation remains achievable by aggregating past observations. We verify the hardness of this setting by using the LV-Rep algorithm (Ren et al., 2023b), which takes the raw observation (with masked velocities) as input to the networks. The algorithm fails to learn on all tasks, confirming the difficulty caused by partial observability. We also provide the best performance when using the original fully observable states (without velocity masking) as input, denoted by *Best-FO* (Best result with Full Observations). This gives a reference for the best an algorithm can achieve in our tests.

We consider four baselines in the experiments, including two model-based methods Dreamer (Hafner et al., 2020; 2021) and Stochastic Latent Actor-Critic (SLAC) (Lee et al., 2020), and a model-free baseline, SAC-MLP, that concatenates history sequences (past four observations) as input to an MLP layer for both the critic and policy. This simple baseline can be viewed as an analogue to how DQN processes observations in Atari games (Mnih et al., 2013) as a sanity check. We also compare to the neural PSR (Guo et al., 2018). We compare all algorithms after running 200K environment steps. This setup exactly follows the benchmark (Wang et al., 2019), which has been widely adopted in (Zhang et al., 2022; Ren et al., 2023c;b) for fairness. All results are averaged across four random seeds. Table 1 presents all the experimental results, averaged over four random seeds. The results clearly demonstrate that the proposed method consistently delivers either competitive or superior outcomes across all domains compared to both the model-based and model-free baselines. We note that in most domains, μLV-Rep nearly matches the performance of Best-FO, further confirming that the proposed method is able to extract useful representations for decision-making in partially observable environments.

7.2 Image-based Continuous Control

We then evaluate the proposed method on the DeepMind Control Suites and Meta-world to demonstrate capability in complex visual control tasks. We observe that directly learning a robust control representation by predicting future visual observations can be challenging, since images contain redundant information for effective decision-making. Consequently, a more advantageous approach is to first acquire an image representation then learn a latent representation based on this initial image representation. In particular, we employ visual observations with dimensions of $64 \times 64 \times 3$ and apply a masked autoencoder (MAE) with mask ratio 0.75 to learn a representation of these visual observations (He et al., 2022). The MAE is first pre-trained with random trajectories then fine-tuned by the online learning procedure. This produces compact vector representations for the images, which are then forwarded as input to the representation learning method. More implementation details, including network architectures and parameters, are provided in Appendix H.

Table 1: Performance on various continuous control problems with partial observation. All results are averaged across 4 random seeds and a window size of 10K. μLV-Rep achieves the best performance compared to the baselines. Here, Best-FO denotes the performance of LV-Rep using full observations as inputs, providing a reference on how well an algorithm can achieve most in our tests.

	HalfCheetah	Humanoid	Walker	Ant	Hopper
μLV-Rep	**3596.2 ± 874.5**	**806.7 ± 120.7**	**1298.1± 276.3**	**1621.4 ± 472.3**	**1096.4 ± 130.4**
Dreamer-v2	2863.8 ± 386	672.5 ± 36.6	**1305.8 ± 234.2**	1252.1 ± 284.2	758.3 ± 115.8
SAC-MLP	1612.0 ± 223	242.1 ± 43.6	736.5 ± 65.6	**1612.0 ± 223**	614.15 ± 67.6
SLAC	**3012.4** ± 724.6	387.4 ± 69.2	536.5 ± 123.2	1134.8 ± 326.2	739.3 ± 98.2
PSR	2679.75±386	534.4 ± 36.6	862.4 ± 355.3	1128.3 ± 166.6	818.8 ± 87.2
Best-FO	5557.6±439.5	1086±278.2	2523.5±333.9	2511.8±460.0	2204.8±496.0
	Cheetah-run	Walker-run	Hopper-run	Humanoid-run	Pendulum
μLV-Rep	525.3 ± 89.2	**702.3 ± 124.3**	**69.3± 12.8**	**9.8 ± 6.4**	**168.2 ± 5.3**
Dreamer-v2	**602.3 ± 48.5**	438.2 ± 78.2	**59.2 ± 15.9**	2.3 ± 0.4	**172.3 ± 8.0**
SAC-MLP	483.3 ± 77.2	279.8 ± 190.6	19.2 ± 2.3	1.2 ± 0.1	163.6 ± 9.3
SLAC	105.1 ± 30.1	139.2 ± 3.4	36.1 ± 15.3	0.9 ± 0.1	**167.3 ± 11.2**
PSR	173.7 ± 25.7	57.4 ± 7.4	23.2 ± 9.5	0.8 ± 0.1	159.4 ± 9.2
Best-FO	639.3±24.5	724.2±37.8	72.9±40.6	11.8±6.8	167.1±3.1

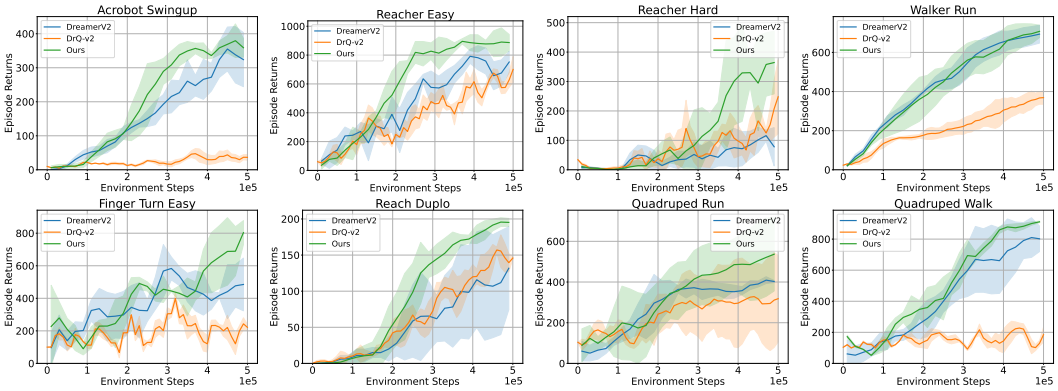


Figure 1: Learning curves on visual control tasks from DeepMind Control Suites measured by episodic return. Model-based methods outperform model-free method by a large margin on this domains. Our method demonstrates the best sample efficiency on most of the tasks.

Two baselines are used in this experiment: a model-based baseline Dreamer (Hafner et al., 2021), and DrQ-v2, a state-of-the-art model-free algorithm for visual continuous control (Yarats et al., 2021a). Figures 1 and 2 present the results. We compare all algorithms after running 500K environment steps on the DeepMind Control Suites and 1 million environment steps on Meta-world. All results are averaged across four random seeds. We observed that the proposed method achieves competitive or superior performance compared to both Dreamer-V2 and DrQ-v2 on all tested benchmarks, illustrating the versatility of our approach across a spectrum of complex visual control tasks.

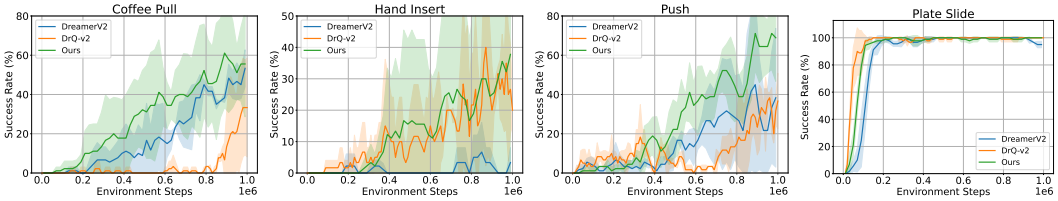


Figure 2: Learning curves on visual robotic manipulation tasks from Meta-world measured by success rate. Our method shows better or comparable sample efficiency to state-of-the-art model-free and model-based methods.

8 Conclusion

In this paper, we aimed to develop a practical RL algorithm for structured POMDPs that obtained efficiency in terms of both statistical and computational complexity. We revealed some of the challenges in computationally exploiting the low-rank structure of a POMDP, then derived a linear representation for the Q^π-function, which automatically implies a practical learning method, with tractable planning and exploration, as in μLV-Rep. We theoretically analyzed the sub-optimality of the proposed μLV-Rep, and empirically demonstrated its advantages on several benchmarks.

REFERENCES

Alekh Agarwal, Sham Kakade, Akshay Krishnamurthy, and Wen Sun. Flambe: Structural complexity and representation learning of low rank mdps. *Advances in neural information processing systems*, 33:20095–20107, 2020.

Nachman Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3):337–404, 1950.

Karl Johan Åström. Optimal control of markov processes with incomplete state information. *Journal of mathematical analysis and applications*, 10(1):174–205, 1965.

Kamyar Azizzadenesheli, Alessandro Lazaric, and Animashree Anandkumar. Reinforcement learning of pomdps using spectral methods. In *COLT*, 2016.

Bram Bakker. Reinforcement learning with long short-term memory. *NeurIPS*, 2001.

André Barreto, Will Dabney, Rémi Munos, Jonathan J Hunt, Tom Schaul, Hado P van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. *NeurIPS*, 2017.

Gabriel Barth-Maron, Matthew W Hoffman, David Budden, Will Dabney, Dan Horgan, Dhruva Tb, Alistair Muldal, Nicolas Heess, and Timothy Lillicrap. Distributed distributional deterministic policy gradients. *arXiv preprint arXiv:1804.08617*, 2018.

Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dębiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*, 2019.

Min Chen, Emilio Frazzoli, David Hsu, and Wee Sun Lee. Pomdp-lite for robust robot planning under uncertainty. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5427–5433. IEEE, 2016.

Peter Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural computation*, 5(4):613–624, 1993.

Marc Peter Deisenroth and Jan Peters. Solving nonlinear continuous state-action-observation pomdps for mechanical systems with gaussian noise. 2012.

Simon Du, Akshay Krishnamurthy, Nan Jiang, Alekh Agarwal, Miroslav Dudik, and John Langford. Provably efficient rl with rich observations via latent state decoding. In *International Conference on Machine Learning*, pp. 1665–1674. PMLR, 2019.

Yaqi Duan, Tracy Ke, and Mengdi Wang. State aggregation learning from markov transition data. *NeurIPS*, 2019.

Yonathan Efroni, Chi Jin, Akshay Krishnamurthy, and Sobhan Miryoosefi. Provable reinforcement learning with a short-term memory. In *International Conference on Machine Learning*, pp. 5832–5850. PMLR, 2022.

Eyal Even-Dar, Sham M Kakade, and Yishay Mansour. The value of observation for monitoring dynamic systems. In *IJCAI*, pp. 2474–2479, 2007.

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.

Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite markov decision processes. In *UAI*, 2004.

Tanmay Gangwani, Joel Lehman, Qiang Liu, and Jian Peng. Learning belief representations for imitation learning in pomdps. In *UAI*, 2020.

Carles Gelada, Saurabh Kumar, Jacob Buckman, Ofir Nachum, and Marc G Bellemare. Deepmdp: Learning continuous latent space models for representation learning. In *ICML*, 2019.

Noah Golowich, Ankur Moitra, and Dhruv Rohatgi. Learning in observable pomdps, without computationally intractable oracles. *Advances in Neural Information Processing Systems*, 35: 1458–1473, 2022.

Karol Gregor, Danilo Jimenez Rezende, Frederic Besse, Yan Wu, Hamza Merzic, and Aaron van den Oord. Shaping belief states with generative environment models for rl. *NeurIPS*, 2019.

Jiacheng Guo, Zihao Li, Huazheng Wang, Mengdi Wang, Zhuoran Yang, and Xuezhou Zhang. Provably efficient representation learning with tractable planning in low-rank pomdp. *arXiv preprint arXiv:2306.12356*, 2023.

Zhaohan Daniel Guo, Shayan Doroudi, and Emma Brunskill. A pac rl algorithm for episodic pomdps. In *Artificial Intelligence and Statistics*, pp. 510–518. PMLR, 2016.

Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Bernardo A Pires, and Rémi Munos. Neural predictive belief representations. *arXiv preprint arXiv:1811.06407*, 2018.

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

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=S1lOTC4tDS.

Danijar Hafner, Timothy P Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=0oabwyZbOu.

Matthew Hausknecht and Peter Stone. Deep recurrent q-learning for partially observable mdps. In *AAAI fall symposium series*, 2015.

Milos Hauskrecht and Hamish Fraser. Planning treatment of ischemic heart disease with partially observable markov decision processes. *Artificial intelligence in medicine*, 18(3):221–244, 2000.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.

Nicolas Heess, Jonathan J Hunt, Timothy P Lillicrap, and David Silver. Memory-based control with recurrent neural networks. *arXiv:1512.04455*, 2015.

Ruiquan Huang, Yingbin Liang, and Jing Yang. Provably efficient ucb-type algorithms for learning predictive state representations. *arXiv preprint arXiv:2307.00405*, 2023.

Maximilian Igl, Luisa Zintgraf, Tuan Anh Le, Frank Wood, and Shimon Whiteson. Deep variational reinforcement learning for pomdps. In *ICML*, 2018.

Haoming Jiang, Bo Dai, Mengjiao Yang, Tuo Zhao, and Wei Wei. Towards automatic evaluation of dialog systems: A model-free off-policy evaluation approach. *arXiv preprint arXiv:2102.10242*, 2021.

Chi Jin, Sham Kakade, Akshay Krishnamurthy, and Qinghua Liu. Sample-efficient reinforcement learning of undercomplete pomdps. *Advances in Neural Information Processing Systems*, 33: 18530–18539, 2020a.

Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. Provably efficient reinforcement learning with linear function approximation. In *COLT*, 2020b.

Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2):99–134, 1998.

Elia Kaufmann, Leonard Bauersfeld, Antonio Loquercio, Matthias Müller, Vladlen Koltun, and Davide Scaramuzza. Champion-level drone racing using deep reinforcement learning. *Nature*, 620 (7976):982–987, 2023.

Tejas D Kulkarni, Ardavan Saeedi, Simanta Gautam, and Samuel J Gershman. Deep successor reinforcement learning. *arXiv:1606.02396*, 2016.

Alex X Lee, Anusha Nagabandi, Pieter Abbeel, and Sergey Levine. Stochastic latent actor-critic: Deep reinforcement learning with a latent variable model. *NeurIPS*, 2020.

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.

Michael Littman and Richard S Sutton. Predictive representations of state. *Advances in neural information processing systems*, 14, 2001.

Qinghua Liu, Alan Chung, Csaba Szepesvári, and Chi Jin. When is partially observable reinforcement learning not scary? In *Conference on Learning Theory*, pp. 5175–5220. PMLR, 2022.

Qinghua Liu, Praneeth Netrapalli, Csaba Szepesvari, and Chi Jin. Optimistic mle: A generic model-based algorithm for partially observable sequential decision making. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pp. 363–376, 2023.

Sridhar Mahadevan and Mauro Maggioni. Proto-value functions: A laplacian framework for learning representation and control in markov decision processes. *JMLR*, 8(10), 2007.

Lingheng Meng, Rob Gorbet, and Dana Kulić. Memory-based deep reinforcement learning for pomdps. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5619–5626. IEEE, 2021.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

Rémi Munos and Csaba Szepesvári. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 9(5), 2008.

Ofir Nachum and Bo Dai. Reinforcement learning via fenchel-rockafellar duality. *arXiv preprint arXiv:2001.01866*, 2020.

Ofir Nachum and Mengjiao Yang. Provable representation learning for imitation with contrastive fourier features. *NeurIPS*, 2021.

Tianwei Ni, Benjamin Eysenbach, and Ruslan Salakhutdinov. Recurrent model-free rl is a strong baseline for many pomdps. *arXiv:2110.05038*, 2021.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018.

Christos H Papadimitriou and John N Tsitsiklis. The complexity of markov decision processes. *Mathematics of operations research*, 12(3):441–450, 1987.

Vern I Paulsen and Mrinal Raghupathi. *An introduction to the theory of reproducing kernel Hilbert spaces*, volume 152. Cambridge university press, 2016.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Shuang Qiu, Lingxiao Wang, Chenjia Bai, Zhuoran Yang, and Zhaoran Wang. Contrastive ucb: Provably efficient contrastive self-supervised learning in online reinforcement learning. In *ICML*, 2022.

Tongzheng Ren, Tianjun Zhang, Csaba Szepesvári, and Bo Dai. A free lunch from the noise: Provable and practical exploration for representation learning. In *Uncertainty in Artificial Intelligence*, pp. 1686–1696. PMLR, 2022.

Tongzheng Ren, Zhaolin Ren, Na Li, and Bo Dai. Stochastic nonlinear control via finite-dimensional spectral dynamic embedding. *arXiv preprint arXiv:2304.03907*, 2023a.

Tongzheng Ren, Chenjun Xiao, Tianjun Zhang, Na Li, Zhaoran Wang, Sujay Sanghavi, Dale Schuurmans, and Bo Dai. Latent variable representation for reinforcement learning. In *The Eleventh International Conference on Learning Representations*, 2023b.

Tongzheng Ren, Tianjun Zhang, Lisa Lee, Joseph E Gonzalez, Dale Schuurmans, and Bo Dai. Spectral decomposition representation for reinforcement learning. *ICLR*, 2023c.

Frigyes Riesz and Béla Sz Nagy. *Functional analysis*. Courier Corporation, 2012.

Nicholas Roy and Geoffrey J Gordon. Exponential family pca for belief compression in pomdps. *Advances in Neural Information Processing Systems*, 15, 2002.

Jürgen Schmidhuber. Reinforcement learning in markovian and non-markovian environments. *NeurIPS*, 3, 1990.

Younggyo Seo, Danijar Hafner, Hao Liu, Fangchen Liu, Stephen James, Kimin Lee, and Pieter Abbeel. Masked world models for visual control. In *Conference on Robot Learning*, pp. 1332–1344. PMLR, 2023.

Ingo Steinwart and Clint Scovel. Mercer's theorem on general domains: On the interaction between measures, kernels, and rkhss. *Constructive Approximation*, 35(3):363–417, 2012.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

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

Masatoshi Uehara, Xuezhou Zhang, and Wen Sun. Representation learning for online and offline rl in low-rank mdps. *arXiv preprint arXiv:2110.04652*, 2021.

Masatoshi Uehara, Ayush Sekhari, Jason D Lee, Nathan Kallus, and Wen Sun. Provably efficient reinforcement learning in partially observable dynamical systems. *Advances in Neural Information Processing Systems*, 35:578–592, 2022.

Tingwu Wang, Xuchan Bao, Ignasi Clavera, Jerrick Hoang, Yeming Wen, Eric Langlois, Shunshi Zhang, Guodong Zhang, Pieter Abbeel, and Jimmy Ba. Benchmarking model-based reinforcement learning. *arXiv preprint arXiv:1907.02057*, 2019.

Stephan Weigand, Pascal Klink, Jan Peters, and Joni Pajarinen. Reinforcement learning using guided observability. *arXiv:2104.10986*, 2021.

Daan Wierstra, Alexander Foerster, Jan Peters, and Juergen Schmidhuber. Solving deep memory pomdps with recurrent policy gradients. In *ICANN*, 2007.

Yifan Wu, George Tucker, and Ofir Nachum. The laplacian in rl: Learning representations with efficient approximations. *arXiv:1810.04586*, 2018.

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

Mengjiao Yang, Sergey Levine, and Ofir Nachum. Trail: Near-optimal imitation learning with suboptimal data. *arXiv:2110.14770*, 2021.

Denis Yarats, Ilya Kostrikov, and Rob Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. In *International conference on learning representations*, 2020.

Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Mastering visual continuous control: Improved data-augmented reinforcement learning. *arXiv preprint arXiv:2107.09645*, 2021a.

Denis Yarats, Amy Zhang, Ilya Kostrikov, Brandon Amos, Joelle Pineau, and Rob Fergus. Improving sample efficiency in model-free reinforcement learning from images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 10674–10681, 2021b.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2019. URL https://arxiv.org/abs/1910.10897.

Wenhao Zhan, Masatoshi Uehara, Wen Sun, and Jason D Lee. Pac reinforcement learning for predictive state representations. *arXiv preprint arXiv:2207.05738*, 2022.

Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. *arXiv:2006.10742*, 2020.

Marvin Zhang, Sharad Vikram, Laura Smith, Pieter Abbeel, Matthew Johnson, and Sergey Levine. Solar: Deep structured representations for model-based reinforcement learning. In *ICML*, 2019.

Tianjun Zhang, Tongzheng Ren, Mengjiao Yang, Joseph Gonzalez, Dale Schuurmans, and Bo Dai. Making linear mdps practical via contrastive representation learning. In *ICML*, 2022.

Tianjun Zhang, Tongzheng Ren, Chenjun Xiao, Wenli Xiao, Joseph Gonzalez, Dale Schuurmans, and Bo Dai. Energy-based predictive representations for partially observed reinforcement learning. In *The 39th Conference on Uncertainty in Artificial Intelligence*, 2023. URL https://openreview.net/forum?id=Z1BobPmCTy.

Pengfei Zhu, Xin Li, Pascal Poupart, and Guanghui Miao. On improving deep reinforcement learning for pomdps. *arXiv preprint arXiv:1704.07978*, 2017.

A More Related Work

Partially Observable RL. The majority of existing practical RL algorithms for partially observable settings can also be categorized into model-based *vs*. model-free.

The model-based algorithms (Kaelbling et al., 1998) for partially observed scenarios are naturally derived based on the definition of POMDPs, where both the emission and transition models are learned from data. The planning procedure for optimal policy is conducted over the posterior of latent state, *i.e.*, beliefs, which is approximately inferred based on learned dynamics and emission model. With different model parametrizations, (ranging from Gaussian processes to deep models), and different planning methods, a family of algorithms has been proposed (Deisenroth & Peters, 2012; Igl et al., 2018; Gregor et al., 2019; Zhang et al., 2019; Lee et al., 2020; Hafner et al., 2021). However, due to the compounding errors from **i)**, mismatch in model parametrization, **ii)**, inaccurate beliefs calculation, **iii)**, approximation in planning over nonlinear dynamics, and **iv)**, neglecting of exploration, such methods might suffer from sub-optimal performances in practice.

As we discussed in Section 1, the memory-based policy and value function have been exploited to extend the MDP-based model-free RL algorithms to handle the non-Markovian dependency induced by partial observations. For example, the value-based algorithms introduces memory-based neural networks to Bellman recursion, including temporal difference learning with explicit concatenation of 4 consecutive frames as input (Mnih et al., 2013) or recurrent neural networks for longer windows (Bakker, 2001; Hausknecht & Stone, 2015; Zhu et al., 2017), and DICE (Nachum & Dai, 2020) with features extracted from transformer (Jiang et al., 2021); the policy gradient-based algorithms have been extended to partially observable setting by introducing recurrent neural network for policy parametrization (Schmidhuber, 1990; Wierstra et al., 2007; Heess et al., 2015; Ni et al., 2021). The actor-critic approaches exploits memory-based value and policy together (Ni et al., 2021; Meng et al., 2021). Despite their simplicity in the algorithm extension, these algorithms demonstrate potentials in real-world applications. However, the it has been observed that the sample complexity for purely model-free RL with partial observations is very high (Mnih et al., 2013; Barth-Maron et al., 2018; Yarats et al., 2021b), and the exploration remains difficult, and thus, largely neglected.

B Observability Approximation

Although the proposed μLV-Rep is designed based on the L-step decodability in POMDPs, Golowich et al. (2022) shows that the γ-observable POMDPs can be ϵ-approximated with a $L = \tilde{O}\left(\gamma^{-4}\log\left(|S|/\epsilon\right)\right)$-step decodable POMDP. By exploiting the low-rank structure in the latent dynamics, this result has been extend with function approximator (Uehara et al., 2022). Specifically,

Theorem 4 (Proprosition 7 (Guo et al., 2023), Lemma 12 (Uehara et al., 2022))**.** *Given a γ-observable POMDP with d-rank latent transition, there exists an L-step decodable POMDP $\mathcal{M}$ with $L = \tilde{O}\left(\gamma^{-4}\log\left(d/\epsilon\right)\right)$, $\forall\epsilon > 0$, such that*

$$\mathbb{E}_{a_{1:h},o_{2:h}\sim\pi}\left[\left\|\mathbb{P}_h\left(o_{h+1}|o_{1:h},a_{1:h}\right) - \mathbb{P}_h^{\mathcal{M}}\left(o_{h+1}|x_h,a_h\right)\right\|_1\right] \leqslant \epsilon. \tag{19}$$

where $\pi_h \in \Delta\left(\prod_{h=1}^{H}\mathcal{A}^{\mathscr{H}_h}\right)$ with $\mathscr{H}_h := \mathcal{A}^{h-1}\times\mathcal{O}^h$, is mapping the whole history to a distribution of action.

With this understanding, the proposed μLV-Rep can be directly applied for γ-observable POMDPs, while still maintains theoretical guarantees. Due to the space limitation, please refer to Uehara et al. (2022); Guo et al. (2023) for the details of the proofs.

C Moment Matching Policy

We provide a formal definition of the moment matching policy here.

Definition 5 (Moment Matching Policy (Efroni et al., 2022))**.** *With the L-decodability assumption, for $h\in[H]$, $h'\in[h-L+1,h]$ and $l = h'-h+L-1$, we can define the moment matching policy $\nu^{\pi,h} = \{\nu_{h'}^{h,\pi} : \mathcal{S}^l\times\mathcal{O}^l\times\mathcal{A}^{l-1}\to\Delta(\mathcal{A})\}_{h'=h-L+1}^{h}$ introduced by Efroni et al. (2022) , such that*

$$\begin{aligned}&\nu_{h'}^{h,\pi}(a_{h'}|(s_{h-L+1:h'},o_{h-L+1:h'},a_{h-L+1:h'-1}))\\ &\quad:= \mathbb{E}_{\pi}^{\mathcal{P}}[\pi_{h'}(a_{h'}|x_{h'})|(s_{h-L+1:h'},o_{h-L+1:h'},a_{h-L+1:h'-1})],\quad \forall h'\leqslant h-1,\end{aligned}$$

Algorithm 2 Offline Learning for L-step decodable POMDPs with Latent Variable Representation

1: **Input:** Model Class $\mathcal{M} = \{\{(p_h(z|x_h, a_h), p_h(o_{h+1}|z)\}_{h\in[H]}\}$, Variational Distribution Class $\mathcal{Q} = \{\{q_h(z|x_h, a_h, o_{h+1})\}_{h\in[H]}\}$, Offline Dataset $\{\mathcal{D}_h\}_{h=1}^H$
2: Learn the latent variable model $\hat{p}(z|x_h, a_h)$ with $\mathcal{D}_h$ via maximizing the ELBO, and obtain the learned model $\widehat{\mathcal{P}} = \{(\hat{p}_h(z|x_h, a_h), \hat{p}_h(o_{h+1}|z))\}_{h\in[H]}$.
3: Set the exploitation penalty $\hat{b}_h(s, a)$ with $\mathcal{D}_k$.
4: Learn the policy $\hat{\pi}^* = \arg\max_\pi V^\pi_{\widehat{\mathcal{P}}, r-\hat{b}_k}$.
5: **Return** $\hat{\pi}^*$.

and $\nu_h^{\pi,h} = \pi_h$. We further define $\tilde{\pi}^h$, which takes first $h - L$ actions from π and the remaining L actions from $\nu^{\pi,h}$.

The main motivation to define such moment matching policy is that, we want to define a policy that is conditionally independent from the past history for theoretical justification while indistinguishable from the history dependent policy to match the practical algorithm. By Lemma B.2 in Efroni et al. (2022), under the L-decodability assumption, for a fixed $h \in [H]$, we have $d_h^{\mathcal{P},\pi}(x_h) = d_h^{\mathcal{P},\tilde{\pi}_h}(x_h)$, for all L-step policy π and $x_h \in \mathcal{X}_h$. As $\nu_h^{\pi,h} = \pi_h$. we have $d_h^{\mathcal{P},\pi}(x_h, a_h) = d_h^{\mathcal{P},\tilde{\pi}^h}(x_h, a_h)$, and hence $\mathbb{E}^{\mathcal{P}}_{\pi}(x_h, a_h) = \mathbb{E}^{\mathcal{P}}_{\tilde{\pi}^h}(x_h, a_h)$. This enables the factorization in (14) without the dependency of the overlap observation trajectory.

D Pessimistim in Offline Setting

Similar to Uehara et al. (2021); Ren et al. (2023b), the proposed algorithm can be directly extended to the offline setting by converting the optimism into the pessimism. Specifically, we can learn the latent variable model, set the penalty with the data and perform planning with the penalized reward. The whole algorithm is shown in Algorithm 2. Following the identical proof strategy from Uehara et al. (2021); Ren et al. (2023b), we can obtain a similar sub-optimal gap guarantee for $\hat{\pi}^*$.

E Technical Background

In this section, we revisit several core concepts of the kernel and the reproducing kernel Hilbert space (RKHS) that will be used in the theoretical analysis. For a complete introduction, we refer the reader to Ren et al. (2023b).

Definition 6 (Kernel and Reproducing Kernel Hilbert Space (RKHS) (Aronszajn, 1950; Paulsen & Raghupathi, 2016)). *The function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a kernel on $\mathcal{X}$ if there exists a Hilbert space $\mathcal{H}$ and a mapping $\phi : \mathcal{X} \to \mathcal{H}$ (termed as a feature map), such that $\forall x, x' \in \mathcal{X}$, $k(x, x') = \langle\phi(x), \phi(x')\rangle_{\mathcal{H}}$. The kernel k is said to be positive semi-definite if $\forall n \geqslant 1, \{a_i\}_{i\in[n]} \subset \mathbb{R}$ and mutually distinct $\{x_i\}_{i\in[n]}$, we have*

$$\sum_{i\in[n]}\sum_{j\in[n]} a_i a_j k(x_i, x_j) \geqslant 0.$$

The kernel k is said to be positive definite if the inequality is strict (which means we can replace $\geqslant$ with $>$).

With a given kernel k, we can define the Hilbert space $\mathcal{H}_k$ consists of $\mathbb{R}$-valued function on $\mathcal{X}$ as a reproducing kernel Hilbert space associated with k if both of the following conditions hold:

- $\forall x \in \mathcal{X}$, $k(x, \cdot) \in \mathcal{H}_k$.
- *Reproducing Property:* $\forall x \in \mathcal{X}, f \in \mathcal{H}_k, f(x) = \langle f, k(x, \cdot)\rangle_{\mathcal{H}_k}$.

The RKHS norm of $f \in \mathcal{H}_k$ is induced by the inner product, i.e. $\|f\|_{\mathcal{H}_k} := \sqrt{\langle f, f\rangle_{\mathcal{H}_k}}$.

Theorem 7 (Mercer's Theorem (Riesz & Nagy, 2012; Steinwart & Scovel, 2012)). *Let k be a continuous positive definite kernel defined on $\mathcal{X} \times \mathcal{X}$. There exists at most countable $\{\mu_i\}_{i\in I}$ such that $\mu_1 \geqslant \mu_2 \geqslant \cdots > 0$ and a set of orthonormal basis $\{e_i\}_{i\in I}$ on $L_2(\mu)$ where μ is a Borel measure*

on $\mathcal{X}$, such that

$$\forall x, x' \in \mathcal{X}, \quad k(x, x') = \sum_{i \in I} \mu_i e_i(x) e_i(x'),$$

where the convergence is absolute and uniform.

Definition 8 (Random Feature)**.** *The kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ has a random feature representation if there exists a function $\psi : \mathcal{X} \times \Xi \to \mathbb{R}$ and a probability measure P over Ξ such that*

$$k(x, x') = \int_{\Xi} \psi(x; \xi) \psi(x'; \xi) dP(\xi).$$

Remark (random feature quadrature): We here justify the random feature quadrature (Ren et al., 2023b) for completeness.

We can represent Q_h^π as an expectation,

$$Q_h^\pi(x_h, a_h) = \langle p(z|x_h), w_h^\pi(z) \rangle = \mathbb{E}_{p(z|x_h)} \left[w_h^\pi(z) \right]_{L_2(\mu)}$$

Under the assumption that $w_h^\pi(\cdot) \in \mathcal{H}_k$, where $\mathcal{H}_k$ denoting some RKHS with some kernel $k(\cdot, \cdot)$. When $k(\cdot, \cdot)$ can be represented through random feature, *i.e.*,

$$k(x, y) = \mathbb{E}_{P(\xi)} \left[\psi(x; \xi) \psi(y; \xi) \right],$$

the $w_h^\pi(z)$ admits a representation as

$$w_h^\pi(z) = \mathbb{E}_{P(\xi)} \left[\tilde{w}_h^\pi(\xi) \psi(z; \xi) \right].$$

Therefore, we plug this random feature representation of $w_h^\pi(z)$ to $Q_h^\pi(x_h, a_h)$, we obtain

$$Q_h^\pi(x_h, a_h) = \mathbb{E}_{p(z|x_h), P(\xi)} \left[\tilde{w}_h^\pi(\xi) \psi(z; \xi) \right]. \tag{20}$$

Applying Monte-Carlo approximation to (20), we obtain the random feature quadrature in (18).

F Theoretical Analysis

F.1 Technical Conditions

We adopt the following assumptions for the reproducing kernel, which have been used in Ren et al. (2023b) for the MDP setting.

Assumption 3 (Regularity Conditions)**.** *$\mathcal{Z}$ is a compact metric space with respect to the Lebesgue measure ν when $\mathcal{Z}$ is continuous. Furthermore, $\int_{\mathcal{Z}} k(z, z) d\nu \leqslant 1$.*

Assumption 4 (Eigendecay Conditions)**.** *Assume $\{\nu_i\}_{i \in I}$ defined in Theorem 7 satisfies one of the following conditions:*

- *β-finite spectrum: for some positive integer β, we have $\nu_i = 0, \forall i > \beta$.*
- *β-polynomial decay: $\nu_i \leqslant C_0 i^{-\beta}$ with absolute constant C_0 and $\beta > 1$.*
- *β-exponential decay: $\nu_i \leqslant C_1 \exp(-C_2 i^\beta)$, with absolute constants C_1, C_2 and $\beta > 0$.*

We will use C_{poly} to denote constants in the analysis of β-polynomial decay that only depends on C_0 and β, and $C_{\exp}$ to denote constants in the analysis of β-exponential decay that only depends on C_1, C_2 and β, to simplify the dependency of the constant terms. Both of C_{poly} and $C_{\exp}$ can be varied step by step.

F.2 Formal Proof

Before we proceed, we first define

$$\rho_{k,h} = \frac{1}{k} \sum_{i \in [K]} d_{\mathcal{P},h}^{\pi_k},$$

and $\circ^L \mathcal{U}(\mathcal{A})$ means uniformly taking actions in the consecutive L steps.

Lemma 9 (L-step back inequality for the true model)**.** *Given a set of functions $[g_h]_{h\in[H]}$, where $g_h : \mathcal{X}\times\mathcal{A}\to\mathbb{R}$, $\|g_h\|_\infty \leqslant B$, $\forall h\in[H]$, we have that $\forall\pi$,*

$$\sum_{h\in[H]}\mathbb{E}^{\mathcal{P}}_{\pi}[g(x_h,a_h)] \leqslant \sum_{h\in[H]}\mathbb{E}^{\mathcal{P}}_{(x_{h-L},a_{h-L})\sim d^{\pi}_{\mathcal{P},h-L}}\left[\|p^*(\cdot|x_{h-L},a_{h-L})\|_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},p^*}}\right] \cdot\sqrt{k|\mathcal{A}|^L\cdot\mathbb{E}_{(\tilde{x}_h,\tilde{a}_h)\sim\rho_{k,h-L}\circ^L\mathcal{U}(\mathcal{A})}\left[g(\tilde{x}_h,\tilde{a}_h)^2\right]+\lambda B^2C}$$

Proof. The proof can be adapted from the proof of Lemma 6 in Ren et al. (2023b), and we include it for the completeness. Recall the moment matching policy ν^π Since $\nu^{\pi,h}$ does not depend on (x_{h-L},a_{h-L}), we can make the following decomposition:

$$\begin{aligned}
&\mathbb{E}^{\mathcal{P}}_{\tilde{\pi}^h}(x_h,a_h)\\
=&\mathbb{E}^{\mathcal{P}}_{(x_{h-L},a_{h-L})\sim\pi}\left[\int_{s_{h-L+1}}\langle p^*(\cdot|x_{h-L},a_{h-L}),p^*(s_{h-L+1}|\cdot)\rangle_{L_2(\mu)}\cdot\mathbb{E}^{\mathcal{P}}_{a_{h-L+1:h}\sim\nu^{\pi,h}}[g(x_h,a_h)|s_{h-L+1}]ds_{h-L+1}\right]\\
\leqslant&\mathbb{E}^{\mathcal{P}}_{(x_{h-L},a_{h-L})\sim\pi}\|p^*(\cdot|x_{h-L},a_{h-L})\|_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},p^*}}\\
&\cdot\left\|\int_{s_{h-L+1}}p^*(s_{h-L+1}|\cdot)\mathbb{E}[g(x_h,a_h)|s_{h-L+1},\nu^{\pi,h}]ds_{h-L+1}\right\|_{L_2(\mu),\Sigma_{\rho_{k,h-L},p^*}}.
\end{aligned}$$

Direct computation shows that

$$\begin{aligned}
&\left\|\int_{s_{h-L+1}}p^*(s_{h-L+1}|\cdot)\mathbb{E}^{\mathcal{P}}[g(x_h,a_h)|s_{h-L+1},\nu^{\pi,h}]ds_{h-L+1}\right\|_{L_2(\mu),\Sigma_{\rho_{k,h-L},p^*}}\\
=&k\mathbb{E}_{(\tilde{x}_{h-L},\tilde{a}_{h-L})\sim\rho_{k,h-L}}\left[\mathbb{E}^{\mathcal{P}}_{s_{h-L+1}\sim\mathbb{P}^{\mathcal{P}}_{h-L}(\cdot|x_{h-L},a_{h-L})}[g(x_h,a_h)|s_{h-L+1},\nu^{\pi,h}]\right]^2\\
&+\left\|\int_{s_{h-L+1}}p^*(s_{h-L+1}|\cdot)\cdot\mathbb{E}^{\mathcal{P}}[g(x_h,a_h)|s_{h-L+1},\nu^{\pi,h}]ds_{h-L+1}\right\|^2_{\mathcal{H}}\\
\leqslant&k\mathbb{E}_{(\tilde{x}_{h-L},\tilde{a}_{h-L})\sim\rho_{k,h-L}}\mathbb{E}^{\mathcal{P}}_{s_{h-L+1}\sim\mathbb{P}^{\mathcal{P}}_{h-L}(\cdot|x_{h-L},a_{h-L}),a_{h-L+1:h}\sim\nu^{\pi,h}}\left[g(x_h,a_h)\right]^2+\lambda B^2C\\
\leqslant&k|\mathcal{A}|^L\mathbb{E}^{\mathcal{P}}_{(\tilde{x}_h,\tilde{a}_h)\sim\rho_{k,h-L}\circ^L\mathcal{U}(\mathcal{A})}[g(\tilde{x}_h,\tilde{a}_h)]^2+\lambda B^2C,
\end{aligned}$$

which finishes the proof. □

Lemma 10 (L-step back inequality for the learned model)**.** *Assume we have a set of functions $[g_h]_{h\in[H]}$, where $g_h:\mathcal{X}\times\mathcal{A}\to\mathbb{R}$, $\|g_h\|_\infty\leqslant B$, $\forall h\in[H]$. Given Lemma 15, we have that $\forall\pi$,*

$$\sum_{h\in[H]}\mathbb{E}^{\widehat{\mathcal{P}}_k}_{\pi}[g(x_h,a_h)] \leqslant \sum_{h\in[H]}\mathbb{E}^{\widehat{\mathcal{P}}_k}_{(x_{h-L},a_{h-L})\sim d^{\pi}_{\widehat{\mathcal{P}}_k,h-L}}\left[\|\hat{p}(\cdot|x_{h-L},a_{h-L})\|_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-2L}\circ^L\mathcal{U}(\mathcal{A}),\hat{p}}}\right] \cdot\sqrt{k|\mathcal{A}|^L\cdot\mathbb{E}_{(\tilde{x}_h,\tilde{a}_h)\sim\rho_{k,h-2L}\circ^{2L}\mathcal{U}(\mathcal{A})}\left[g(\tilde{x}_h,\tilde{a}_h)^2\right]+\lambda B^2C+kL|\mathcal{A}|^{L-1}B^2\zeta_k}$$

Proof. The proof can be adapted from the proof of Lemma 5 in Ren et al. (2023b), and we include it for the completeness. We define a similar moment matching policy and make the following decomposition:

$$\begin{aligned}
&\mathbb{E}^{\widehat{\mathcal{P}}_k}_{\pi}(x_h,a_h)\\
=&\mathbb{E}^{\widehat{\mathcal{P}}_k}_{(x_{h-L},a_{h-L})\sim\pi}\left[\int_{s_{h-L+1}}\langle\hat{p}(\cdot|x_{h-L},a_{h-L}),\hat{p}(s_{h-L+1}|\cdot)\rangle_{L_2(\mu)}\cdot\mathbb{E}^{\widehat{\mathcal{P}}_k}[g(x_h,a_h)|s_{h-L+1},\nu^{\pi,h}]ds_{h-L+1}\right]\\
\leqslant&\mathbb{E}^{\widehat{\mathcal{P}}_k}_{(x_{h-L},a_{h-L})\sim\pi}\|\hat{p}(\cdot|x_{h-L},a_{h-L})\|_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-2L}\circ^L\mathcal{U}(\mathcal{A}),\hat{p}}}\\
&\cdot\left\|\int_{s_{h-L+1}}\hat{p}(s_{h-L+1}|\cdot)\mathbb{E}^{\widehat{\mathcal{P}}_k}[g(x_h,a_h)|s_{h-L+1},\nu^{\pi,h}]ds_{h-L+1}\right\|_{L_2(\mu),\Sigma_{\rho_{k,h-2L}\circ^L\mathcal{U}(\mathcal{A}),\hat{p}}}.
\end{aligned}$$

Direct computation shows that

$$\begin{aligned}
&\left\|\int_{s_{h-L+1}} \hat{p}(s_{h-L+1}|\cdot)\mathbb{E}^{\widehat{\mathcal{P}}_k}[g(x_h,a_h)|s_{h-L+1},\nu^{\pi,h}]ds_{h-L+1}\right\|^2_{L_2(\mu),\Sigma_{\rho_{k,h-2L}\circ^L\mathcal{U}(\mathcal{A}),\hat{p}}}\\
=&k\mathbb{E}_{(\tilde{x}_{h-L},\tilde{a}_{h-L})\sim\rho_{k,h-2L}\circ^L\mathcal{U}(\mathcal{A})}\left[\mathbb{E}_{s_{h-L+1}\sim\mathbb{P}^{\widehat{\mathcal{P}}_k}_{h-L}(\cdot|\tilde{x}_{h-L},\tilde{a}_{h-L})}\mathbb{E}^{\widehat{\mathcal{P}}_k}[g(x_h,a_h)|s_{h-L+1},\nu^{\pi,h}]\right]^2\\
&+\left\|\int_{s_{h-L+1}} \hat{p}(s_{h-L+1}|\cdot)\mathbb{E}[g(x_h,a_h)|s_{h-L+1},\nu^{\pi,h}]ds_{h-L+1}\right\|^2_{\mathcal{H}}\\
\leqslant&k\mathbb{E}_{(\tilde{x}_{h-L},\tilde{a}_{h-L})\sim\rho_{k,h-2L}\circ^L\mathcal{U}(\mathcal{A})}\mathbb{E}^{\widehat{\mathcal{P}}_k}_{s_{h-L+1}\sim\mathbb{P}^{\widehat{\mathcal{P}}_k}_{h-L}(\cdot|\tilde{x}_{h-L},\tilde{a}_{h-L}),\nu^{\pi,h}}[g(x_h,a_h)]^2+\lambda B^2C\\
\leqslant&k|\mathcal{A}|^L\mathbb{E}_{(\tilde{x}_{h-L},\tilde{a}_{h-L})\sim\rho_{k,h-2L}\circ^L\mathcal{U}(\mathcal{A})}\mathbb{E}^{\widehat{\mathcal{P}}_k}_{a_{h-L+1:h}\sim\circ^L\mathcal{U}(\mathcal{A})}[g(x_h,a_h)]^2+\lambda B^2C\\
\leqslant&k|\mathcal{A}|^L\mathbb{E}_{(\tilde{x}_h,\tilde{a}_h)\sim\rho_{k,h-2L}\circ^{2L}\mathcal{U}(\mathcal{A})}[g(\tilde{x}_h,\tilde{a}_h)]^2+kL|\mathcal{A}|^{L-1}B^2\zeta_k+\lambda B^2C,
\end{aligned}$$

where we use the MLE guarantee for each individual step to obtain the last inequality. This finishes the proof. □

Lemma 11 (Almost Optimism)**.** *For episode $k\in[K]$, set*

$$\hat{b}_{k,h}=\min\left\{\alpha_k\|\hat{p}_k(\cdot|x_{h-L},a_{h-L})\|_{L_2(\mu),\hat{\Sigma}^{-1}_{k,h,\hat{p}_k}},2\right\},$$

with $\alpha_k=\frac{\sqrt{5kL|\mathcal{A}|^L\zeta_k+4\lambda d}}{c}$,

$\hat{\Sigma}_{k,h,\hat{p}_k}:L_2(\mu)\to L_2(\mu)$, $\quad\hat{\Sigma}_{k,h,\hat{p}_k}:=\sum_{(x_{h,i},a_{h,i})\in\mathcal{D}_{k,h}}\left[\hat{p}_k(z|x_{h,i},a_{h,i})\hat{p}_k(z|x_{h,i},a_{h,i})^\top\right]+\lambda T_K^{-1}$

where T_K is the integral operator associated with K (i.e. $T_Kf=\int f(x)K(x,\cdot)dx$) and λ is set for different eigendecay of K as follows:

- *β-finite spectrum:* $\lambda=\Theta(\beta\log K+\log(K|\mathcal{P}|/\delta))$
- *β-polynomial decay:* $\lambda=\Theta(C_{\mathrm{poly}}K^{1/(1+\beta)}+\log(K|\mathcal{P}|/\delta))$;
- *β-exponential decay:* $\lambda=\Theta(C_{\exp}(\log K)^{1/\beta}+\log(K|\mathcal{P}|/\delta))$;

c is an absolute constant, then with probability at least $1-\delta$, $\forall k\in[K]$ we have

$$V^{\pi^*,\widehat{\mathcal{P}}_k,r+\widehat{b}_k}-V^{\pi^*,\mathcal{P},r}\geqslant-\sqrt{|\mathcal{A}|^{L+1}\zeta_k}$$

Proof. With Lemma 14, we have that

$$\begin{aligned}
&V^{\pi^*,\widehat{\mathcal{P}}_k,r+\widehat{b}^k}-V^{\pi^*,\mathcal{P},r}\\
=&\sum_{h\in[H]}\mathbb{E}_{(x_h,a_h)\sim d^{\pi^*}_{\widehat{\mathcal{P}}_k,h}}\left[\widehat{b}^k_h(x_h,a_h)+\mathbb{E}_{o'\sim\mathbb{P}^{\widehat{\mathcal{P}}_k}_h(\cdot|x_h,a_h)}[V^{\pi^*,\mathcal{P},r}_{h+1}(x'_{h+1})]-\mathbb{E}_{o'\sim\mathbb{P}^{\mathcal{P}}_h(\cdot|x_h,a_h)}[V^{\pi^*,\mathcal{P},r}_{h+1}(x'_{h+1})]\right]\\
\geqslant&\sum_{h\in[H]}\mathbb{E}_{(x_h,a_h)\sim d^{\pi^*}_{\widehat{\mathcal{P}}_k,h}}\left[\min\left[c\alpha_k\left\|\hat{p}(\cdot|x_{h-L},a_{h-L})\right\|_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},\hat{p}}},2\right]+\mathbb{E}_{o'\sim\mathbb{P}^{\widehat{\mathcal{P}}_k}_h(\cdot|x_h,a_h)}[V^{\pi^*,\mathcal{P},r}_{h+1}(x'_{h+1})]\right.\\
&\left.-\mathbb{E}_{o'\sim\mathbb{P}^{\mathcal{P}}_h(\cdot|x_h,a_h)}[V^{\pi^*,\mathcal{P},r}_{h+1}(x'_{h+1})]\right],
\end{aligned}$$

where in the last step we replace the empirical covariance with the population counterpart thanks to Lemma 17 in Ren et al. (2023b). Define

$$g_h(z_h,a_h)=\mathbb{E}_{o'\sim\mathbb{P}^{\mathcal{P}}_h(\cdot|x_h,a_h)}[V^{\pi^*,\mathcal{P},r}_{h+1}(x'_{h+1})]-\mathbb{E}_{o'\sim\mathbb{P}^{\widehat{\mathcal{P}}_k}_h(\cdot|x_h,a_h)}[V^{\pi^*,\mathcal{P},r}_{h+1}(x'_{h+1})],$$

With Hölder's inequality, we have that $\|g_h\|_\infty\leqslant 2$.

Furthermore, with Lemma 10, we have that

$$\begin{aligned}
&\sum_{h\in[H]} \mathbb{E}_{(x_h,a_h)\sim d^{\pi^*}_{\widehat{\mathcal{P}}_k,h}}[g_h(x_h,a_h)] \\
\leqslant &\sum_{h\in[H]} \mathbb{E}_{(x_{h-L},a_{h-L})\sim d^{\pi^*}_{\widehat{\mathcal{P}}_k,h}} \left[\|\hat{p}(\cdot|x_{h-L},a_{h-L})\|_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},\hat{p}}}\right] \\
&\cdot \sqrt{k|\mathcal{A}|^L \cdot \mathbb{E}_{(\tilde{x}_h,\tilde{a}_h)\sim\rho_{h-2L}\circ^{2L}\mathcal{U}(\mathcal{A})}\left[g(\tilde{x}_h,\tilde{a}_h)^2\right] + 4\lambda C + 4kL|\mathcal{A}|^{L-1}\zeta_k} \\
\leqslant &\sum_{h\in[H]} \mathbb{E}_{(x_{h-L},a_{h-L})\sim d^{\pi^*}_{\widehat{\mathcal{P}}_k,h}} \left[c\alpha_k \|\hat{p}(\cdot|x_{h-L},a_{h-L})\|_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},\hat{p}}}\right],
\end{aligned}$$

where we use Lemma 15 in the last step. Now we deal with the case with $h \in [L]$. Note that, $\forall h \in [L]$

$$\begin{aligned}
&\mathbb{E}_{(x_h,a_h)\sim\pi}[g_h(x_h,a_h)] \\
\leqslant &|\mathcal{A}|^h \mathbb{E}_{x_1\sim d_1,a_{1:h}\sim\circ^h\mathcal{U}(\mathcal{A})} \left\|\mathbb{P}_h^{\widehat{\mathcal{P}}_k}(\cdot|x_h,a_h) - \mathbb{P}_h^{\mathcal{P}}(\cdot|x_h,a_h)\right\|_1 \\
\leqslant &\sqrt{\mathbb{E}_{x_1\sim d_1,a_{1:h}\circ^L\mathcal{U}(\mathcal{A})} \left\|\mathbb{P}_h^{\widehat{\mathcal{P}}_k}(\cdot|x_h,a_h) - \mathbb{P}_h^{\mathcal{P}}(\cdot|x_h,a_h)\right\|_1^2} \\
\leqslant &\sqrt{|\mathcal{A}|^h\zeta_k},
\end{aligned}$$

where in the last step we use Lemma 15. We finish the proof by summing over $h \in [L]$. □

Lemma 12 (Regret)**.** *With probability at least $1-\delta$, we have that*

- *For β-finite spectrum, we have*
  $\sum_{k=1}^{K} V^{\pi^*,\mathcal{P},r} - V^{\pi_k,\mathcal{P},r} \lesssim \sum_{k=1}^{K} V^{\pi^*,\mathcal{P},r} - V^{\pi_k,\mathcal{P},r} \lesssim H^2\beta^{3/2}|\mathcal{A}|^L \log K\sqrt{CLK\log(K|\mathcal{M}|/\delta)}$;
- *For β-polynomial decay, we have*
  $$\sum_{k=1}^{K} V^{\pi^*,\mathcal{P},r} - V^{\pi_k,\mathcal{P},r} \lesssim C_{\mathrm{poly}} H^2|\mathcal{A}|^L K^{\frac{1}{2}+\frac{1}{1+\beta}}\sqrt{CL\log(K|\mathcal{M}|/\delta)};$$
- *For β-exponential decay, we have*
  $$\sum_{k=1}^{K} V^{\pi^*,\mathcal{P},r} - V^{\pi_k,\mathcal{P},r} \lesssim C_{\exp} H^2|\mathcal{A}|^L (\log K)^{1+\frac{3}{2\beta}}\sqrt{CLK\log(K|\mathcal{M}|/\delta)};$$

Proof. With Lemma 11 and Lemma 14, we have

$$\begin{aligned}
&V^{\pi^*,\mathcal{P},r} - V^{\pi_k,\mathcal{P},r} \\
\leqslant &V^{\pi^*,\widehat{\mathcal{P}}_k,r+\widehat{b}^k} + \sqrt{|\mathcal{A}|^{L+1}\zeta_k} - V^{\pi_k,\mathcal{P},r} \\
\leqslant &V^{\pi^k,\widehat{\mathcal{P}}_k,r+\widehat{b}^k} + \sqrt{|\mathcal{A}|^{L+1}\zeta_k} - V^{\pi_k,\mathcal{P},r} \\
= &\sum_{h\in[H]} \mathbb{E}_{(x_h,a_h)\sim d^{\pi_k}_{\mathcal{P},h}} \left[\widehat{b}_h^k(x_h,a_h) + \mathbb{E}_{o'\sim\mathbb{P}_h^{\widehat{\mathcal{P}}_k}(\cdot|x_h,a_h)}\left[V_{h+1}^{\pi_k,\widehat{\mathcal{P}}_k,r+\widehat{b}_h^k}(x'_{h+1})\right] - \mathbb{E}_{o'\sim\mathbb{P}_h^{\mathcal{P}}(\cdot|x_h,a_h)}\left[V_{h+1}^{\pi_k,\widehat{\mathcal{P}}_k,r+\widehat{b}_h^k}(x'_{h+1})\right]\right], \\
&+ \sqrt{|\mathcal{A}|^{L+1}\zeta_k}.
\end{aligned}$$

Note that $\left\|\widehat{b}_h^k\right\|_\infty \leqslant 2$. Applying Lemma 9, we have that

$$\begin{aligned}
&\sum_{h\in[H]} \mathbb{E}_{(x_h,a_h)\sim d^{\pi_k}_{\mathcal{P},h}}\left[\widehat{b}_h^k(x_h,a_h)\right] \\
\leqslant &\sum_{h\in[H]} \mathbb{E}_{(\tilde{x}_{h-L},\tilde{a}_{h-L})\sim d^{\pi_k}_{\mathcal{P},h}} \left[\|p^*(\cdot|x_{h-L},a_{h-L})\|_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},p^*}}\right] \\
&\cdot \sqrt{k|\mathcal{A}|^L \cdot \mathbb{E}_{(\tilde{x}_h,\tilde{a}_h)\sim\rho_{k,h-L}\circ^L\mathcal{U}(\mathcal{A})}\left[\widehat{b}_h^k(\tilde{x}_h,\tilde{a}_h)^2\right] + 4\lambda C}
\end{aligned}$$

Following the proof of Lemma 8 in Ren et al. (2023b), we have that:

- for β-finite spectrum,
$$k\mathbb{E}_{(\tilde{x}_h,\tilde{a}_h)\sim\rho_{k,h-L}\circ^L\mathcal{U}(\mathcal{A})}\left[\widehat{b}_h^k(\tilde{x}_h,\tilde{a}_h)^2\right]=O(\beta\log K);$$
- for β-polynomial decay,
$$k\mathbb{E}_{(\tilde{x}_h,\tilde{a}_h)\sim\rho_{k,h-L}\circ^L\mathcal{U}(\mathcal{A})}\left[\widehat{b}_h^k(\tilde{x}_h,\tilde{a}_h)^2\right]=O\left(C_{\text{poly}}K^{\frac{1}{2(1+\beta)}}\log K\right);$$
- for β-exponential decay,
$$k\mathbb{E}_{(\tilde{x}_h,\tilde{a}_h)\sim\rho_{k,h-L}\circ^L\mathcal{U}(\mathcal{A})}\left[\widehat{b}_h^k(\tilde{x}_h,\tilde{a}_h)^2\right]=O\left(C_{\exp}(\log K)^{1+1/\beta}\right).$$

We then consider
$$\sum_{h\in[H]}\mathbb{E}_{(x_h,a_h)\sim d_{\mathcal{P},h}^{\pi_k}}\left[\mathbb{E}_{o'\sim\mathbb{P}_h^{\widehat{\mathcal{P}}_k}(\cdot|x_h,a_h)}\left[V_{h+1}^{\pi_k,\widehat{\mathcal{P}}_k,r+\widehat{b}_h^k}(x'_{h+1})\right]-\mathbb{E}_{o'\sim\mathbb{P}_h^{\mathcal{P}}(\cdot|x_h,a_h)}\left[V_{h+1}^{\pi_k,\widehat{\mathcal{P}}_k,r+\widehat{b}_h^k}(x'_{h+1})\right]\right].$$
Define
$$g(x_h,a_h)=\frac{1}{2H+1}\left[\mathbb{E}_{o'\sim\mathbb{P}_h^{\widehat{\mathcal{P}}_k}(\cdot|x_h,a_h)}\left[V_{h+1}^{\pi_k,\widehat{\mathcal{P}}_k,r+\widehat{b}_h^k}(x'_{h+1})\right]-\mathbb{E}_{o'\sim\mathbb{P}_h^{\mathcal{P}}(\cdot|x_h,a_h)}\left[V_{h+1}^{\pi_k,\widehat{\mathcal{P}}_k,r+\widehat{b}_h^k}(x'_{h+1})\right]\right].$$
With Hölder's inequality and note that $\left\|\widehat{b}_h^k\right\|\leqslant 2$, we have that $\|g\|_\infty\leqslant 2$. With Lemma 9, we have that
$$\begin{aligned}
&\sum_{h\in[H]}\mathbb{E}_{(x_h,a_h)\sim d_{\mathcal{P},h}^{\pi_k}}[g(x_h,a_h)]\\
\leqslant&\sum_{h\in[H]}\mathbb{E}^{\mathcal{P}}_{(x_{h-L},a_{h-L})\sim d_{\mathcal{P},h}^{\pi_k}}\left[\left\|p^*(\cdot|x_{h-L},a_{h-L})\right\|_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},p^*}}\right]\cdot\sqrt{k|\mathcal{A}|^L\mathbb{E}_{(\tilde{x}_h,\tilde{a}_h)\sim\rho_{k,h-L}\circ^L\mathcal{U}(\mathcal{A})}\left[g(\tilde{x}_h,\tilde{a}_h)^2\right]+4\lambda C}\\
\leqslant&\sum_{h\in[H]}\mathbb{E}^{\mathcal{P}}_{(x_{h-L},a_{h-L})\sim d_{\mathcal{P},h}^{\pi_k}}\left[\left\|p^*(\cdot|x_{h-L},a_{h-L})\right\|_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},p^*}}\right]\cdot\sqrt{k|\mathcal{A}|^L\zeta_k+4\lambda C}\\
\leqslant&c\alpha_k\sum_{h\in[H]}\mathbb{E}^{\mathcal{P}}_{(x_{h-L},a_{h-L})\sim d_{\mathcal{P},h}^{\pi}}\left[\left\|p^*(\cdot|x_{h-L},a_{h-L})\right\|_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},p^*}}\right]
\end{aligned}$$
With Cauchy-Schwartz inequality, we know that
$$\begin{aligned}
&\sum_{k\in[K]}\mathbb{E}^{\mathcal{P}}_{(x_{h-L},a_{h-L})\sim d_{\mathcal{P},h}^{\pi_k}}\left[\left\|p^*(\cdot|x_{h-L},a_{h-L})\right\|_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},p^*}}\right]\\
\leqslant&\sqrt{K\sum_{k\in[K]}\mathbb{E}^{\mathcal{P}}_{(x_{h-L},a_{h-L})\sim d_{\mathcal{P},h}^{\pi_k}}\left[\left\|p^*(\cdot|x_{h-L},a_{h-L})\right\|^2_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},p^*}}\right]}.
\end{aligned}$$
Following the proof of Lemma 8 in Ren et al. (2023b), we have that

- for β-finite spectrum,
$$\sum_{k\in[K]}\mathbb{E}^{\mathcal{P}}_{(x_{h-L},a_{h-L})\sim d_{\mathcal{P},h}^{\pi_k}}\left[\left\|p^*(\cdot|x_{h-L},a_{h-L})\right\|^2_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},p^*}}\right]=O(\beta\log K);$$
- for β-polynomial decay,
$$\sum_{k\in[K]}\mathbb{E}^{\mathcal{P}}_{(x_{h-L},a_{h-L})\sim d_{\mathcal{P},h}^{\pi_k}}\left[\left\|p^*(\cdot|x_{h-L},a_{h-L})\right\|^2_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},p^*}}\right]=O\left(C_{\text{poly}}K^{\frac{1}{2(1+\beta)}}\log K\right);$$
- for β-exponential decay,
$$\sum_{k\in[K]}\mathbb{E}^{\mathcal{P}}_{(x_{h-L},a_{h-L})\sim d_{\mathcal{P},h}^{\pi_k}}\left[\left\|p^*(\cdot|x_{h-L},a_{h-L})\right\|^2_{L_2(\mu),\Sigma^{-1}_{\rho_{k,h-L},p^*}}\right]=O\left(C_{\exp}(\log K)^{1+1/\beta}\right).$$

Combine the previous steps and take the dominating term out, we have that

- for β-finite spectrum,
$$\sum_{k=1}^{K} V^{\pi^*,\mathcal{P},r} - V^{\pi_k,\mathcal{P},r} \lesssim H^2\beta^{3/2}|\mathcal{A}|^L \log K\sqrt{CLK\log(K|\mathcal{M}|/\delta)};$$
- for β-polynomial decay,
$$\sum_{k=1}^{K} V^{\pi^*,\mathcal{P},r} - V^{\pi_k,\mathcal{P},r} \lesssim C_{\mathrm{poly}}H^2|\mathcal{A}|^L K^{\frac{1}{2}+\frac{1}{1+\beta}}\sqrt{CL\log(K|\mathcal{M}|/\delta)};$$
- for β-exponential decay,
$$\sum_{k=1}^{K} V^{\pi^*,\mathcal{P},r} - V^{\pi_k,\mathcal{P},r} \lesssim C_{\exp}H^2|\mathcal{A}|^L (\log K)^{1+\frac{3}{2\beta}}\sqrt{CLK\log(K|\mathcal{M}|/\delta)};$$

which finishes the proof. □

Theorem 13 (PAC Guarantee). *After interacting with the environments for KH episodes*

- $K = \Theta\left(\frac{CH^4L\beta^3|\mathcal{A}|^{2L}\log(|\mathcal{P}|/\delta)}{\varepsilon^2}\log^3\left(\frac{CH^4L\beta^3|\mathcal{A}|^{2L}\log(|\mathcal{P}|/\delta)}{\varepsilon^2}\right)\right)$ *for β-finite spectrum;*
- $K = \Theta\left(C_{\mathrm{poly}}\left(\frac{H^2L|\mathcal{A}|^L\sqrt{C\log(|\mathcal{P}|/\delta)}}{\varepsilon}\log^{3/2}\left(\frac{\sqrt{C}H^2L|\mathcal{A}|^L\log(|\mathcal{P}|/\delta)}{\varepsilon}\right)\right)^{\frac{2(1+\beta)}{\beta-1}}\right)$ *for β-polynomial decay;*
- $K = \Theta\left(\frac{C_{\exp}CH^4L|\mathcal{A}|^{2L}\log(|\mathcal{P}|/\delta)}{\varepsilon^2}\log^{\frac{3+2\beta}{\beta}}\left(\frac{CH^4L|\mathcal{A}|^{2L}\log(|\mathcal{P}|/\delta)}{\varepsilon^2}\right)\right)$ *for β-exponential decay;*

we can obtain an ε-optimal policy with high probability.

Proof. This is a direct extension of the proof of Theorem 9 in Ren et al. (2023b). □

G Technical Lemma

Lemma 14 (Simulation Lemma). *For two MDPs $\mathcal{M} = (P, r)$ and $\mathcal{M}' = (P', r + b)$, we have*
$$V^{\pi}_{P',r+b} - V^{\pi}_{P,r}$$
$$= \sum_{h\in[H]} \mathbb{E}_{(s_h,a_h)\sim d^h_{P,\pi}}\left[b_h(s_h,a_h) + \mathbb{E}_{s_{h+1}\sim P'(s_h,a_h)}\left[V^{\pi}_{P',r+b,h+1}(s_{h+1})\right] - \mathbb{E}_{s_{h+1}\sim P(s_h,a_h)}\left[V^{\pi}_{P',r+b,h+1}(s_{h+1})\right]\right],$$
and
$$V^{\pi}_{P',r+b} - V^{\pi}_{P,r}$$
$$= \sum_{h\in[H]} \mathbb{E}_{(s_h,a_h)\sim d^h_{P',\pi}}\left[b_h(s_h,a_h) + \mathbb{E}_{s_{h+1}\sim P'(s_h,a_h)}\left[V^{\pi}_{P,r,h+1}(s_{h+1})\right] - \mathbb{E}_{s_{h+1}\sim P(s_h,a_h)}\left[V^{\pi}_{P,r,h+1}(s_{h+1})\right]\right],$$

For the proof, see Uehara et al. (2021) for an example.

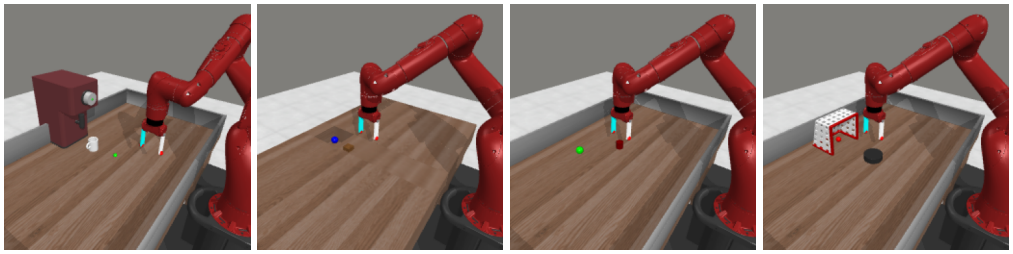
Lemma 15 (MLE Guarantee). *For any episoode $k \in [K]$, step $h \in [H]$, define ρ_h as the joint distribution of (x_h, a_h) in the dataset $\mathcal{D}_{h,k}$ at episode k. Then with probability at least $1 - \delta$, we have that*
$$\mathbb{E}_{(x_h,a_h)\sim\mathcal{D}_{h,k}}\left\|\mathbb{P}^{\mathcal{P}}_h(\cdot|x_h,a_h) - \mathbb{P}^{\widehat{\mathcal{P}}_k}_h(\cdot|x_h,a_h)\right\|^2_1 \leqslant \zeta_k,$$
where $\zeta_k = O(\log(Hk|\mathcal{M}|/\delta)/k)$

For the proof, see Agarwal et al. (2020).

H IMPLEMENTATION DETAILS ON IMAGE-BASED CONTINUOUS CONTROL

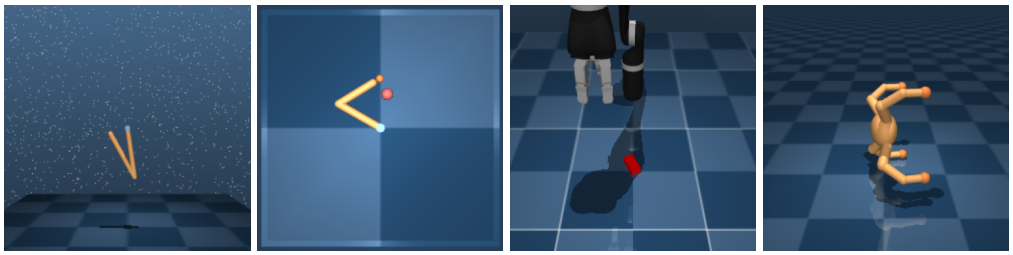
We evaluate our method on DeepMind Control Suites (Tassa et al., 2018) [1] and Meta-world (Yu et al., 2019) [2] to demonstrate its capability for complex visual control tasks. Meta-world is an open-source simulated benchmark consisting of 50 distinct robotic manipulation tasks. The DeepMind Control Suite is a set of continuous control tasks with a standardized structure and interpretable rewards, intended to serve as performance benchmarks for reinforcement learning agents. The visualization of the two domains is shown in Figures 3 and 4. With only one frame of the visual observation, we will miss some information related to the task, for example the speed, thus these tasks are partially observable.

In particular, we employ visual observations with dimensions of $64 \times 64 \times 3$ and apply a masked autoencoder (MAE) with a masking ratio of 0.75 to learn representations for these visual observations (He et al., 2022). The MAE is first pre-trained with random trajectories at the beginning and then fine-tuned during the online learning procedure. It produces compact vector representations for the images, which are then forwarded as input to our representation learning method. A Recurrent State-Space Model (RSSM) (Hafner et al., 2019) forms the dynamics of the world model. The RSSM uses a sequence of deterministic recurrent states, from which it computes two distributions over stochastic states at each step. The posterior latent state incorporates information about the current representation. The learning goal is to use the prior latent state to predict the posterior without access to the current representation. We incorporate multi-step prediction in RSSM with one starting state and a sequence of actions. Since the representation is learned from MAE, we reconstruct the representation after we get the representation predicted by RSSM. We apply actor-critic Learning based on the representation learned by MAE and the dynamics learned by RSSM. The configuration of used tasks are given in Table 2. The hyperparameters used in MAE, RSSM and the RL agent are shown in Tables 3 and 4.



(a) Coffee Pull (b) Hand Insert (c) Push (d) Plate Slide

Figure 3: Visualization of the visual robotic manipulation tasks in Meta-world.



(a) Acrobot Swingup (b) Reacher hard (c) Reach Duplo (d) Quadruped Run

Figure 4: Visualization of the visual control tasks in DeepMind Control Suites.

[1] https://github.com/google-deepmind/dm_control
[2] https://github.com/Farama-Foundation/Metaworld

Table 2: Configuration of environments.

Hyperparameter	Value
Image observation	$64\times64\times3$
Image normalization	Mean: $(0.485, 0.456, 0.406)$, Std: $(0.229, 0.224, 0.225)$
Action repeat	2
Episode length	500 (Meta-world), 1000 (DMC)
Normalize action	[-1,1]
Camera	corner2 (Meta-world), camera2 (DMC)
Total steps in environment	1M (Meta-world), 0.5M (DMC)

Table 3: Hyperparameters in world model.

Hyperparameter	Value
MAE	
ViT encoder size	depth: 4, heads: 4, embedding dim: 256
ViT decoder size	depth: 3, heads: 4, embedding dim: 128
Patch size	8×8
Mask ratio	0.75
Batch size	1024
Optimizer	Adam
Learning rate	0.0003
Pretrain step	5000
RSSM	
Deterministic state dim	1024
Stochastic state dim	32
Discrete latent dimensions	32
Batch size	50 (Meta-world), 16 (DMC)
Sequence length	50
KL balance	0.8
Optimizer	Adam
Learning rate	0.0003
Gradient clip	100

Table 4: Hyperparameters used in Actor Critic.

Hyperparameter	Value
Replay buffer	2,000,000
Batch size	50
Trajectory length	50
Network size	[512, 512, 512, 512]
Optimizer	Adam
Learning rate	0.0001
Gradient clip	100
Entropy weight	0.0001
Discount	0.99
λ return discount	0.95
Random steps	5000
Evaluate interval	10,000
Evaluate episodes	10 (Meta-world), 5 (DMC)

H.1 Ablation Studies

The importance of the exploration has been demonstrated in (Zhang et al., 2022). We perform ablation studies to demonstrate the effects of the major components, including representation dimension and window size, as illustrated below. Figure 5 presents an ablation study on representation dimension, where we compare μLV-Rep with latent representation dimensions 2048, 512, and 128. We also ablate the effect of window size L. In Figure 6, we compare μLV-Rep with window size $L = 1, 3, 5$. We also compare DrQ-v2 with $L = 1, 3, 5$ to show the effect of L on other algorithms. The results show that for $L = 1$, both μLV-Rep and DrQ-v2 struggle with learning, which confirms the Non-Markovian property of the DMC control problems. We can also find that $L = 3$ is sufficient for learning in both test domains.

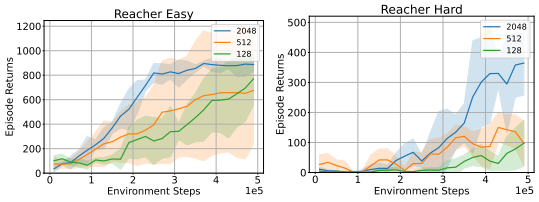


Figure 5: Ablation of feature dimension on visual control tasks from DeepMind Control Suites. Increasing the dimension of the feature gets better performance.

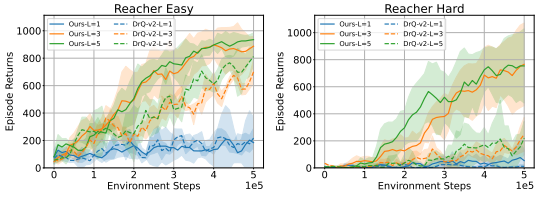


Figure 6: Ablation of window size L on visual control tasks from DeepMind Control Suites. $L = 3$ is sufficient for learning in both test domains.