Provable Reinforcement Learning with a Short-Term Memory

Yonathan Efroni [1] **Chi Jin** [2] **Akshay Krishnamurthy** [1] **Sobhan Miryoosefi** [2]

Abstract

Real-world sequential decision making problems commonly involve partial observability, which requires the agent to maintain a memory of history in order to infer the latent states, plan and make good decisions. Coping with partial observability in general is extremely challenging, as a number of worst-case statistical and computational barriers are known in learning Partially Observable Markov Decision Processes (POMDPs). Motivated by the problem structure in several physical applications, as well as a commonly used technique known as "frame stacking", this paper proposes to study a new subclass of POMDPs, whose *latent states can be decoded by the most recent history of a short length* m. We establish a set of upper and lower bounds on the sample complexity for learning near-optimal policies for this class of problems in both tabular and rich-observation settings (where the number of observations is enormous). In particular, in the rich-observation setting, we develop new algorithms using a novel "moment matching" approach with a sample complexity that scales exponentially with the short length m rather than the problem horizon, and is independent of the number of observations. Our results show that a short-term memory suffices for reinforcement learning in these environments.

1. Introduction

Reinforcement learning is a well-studied paradigm for sequential decision making, in which an agent learns to make decisions in a stateful environment to accumulate reward. The most common framework for reinforcement learning—particularly for theoretical analysis—is the Markov Decision Process (MDP), in which the environment is summarized by a state that is observable to the agent. One notable feature of the MDP is that the agent can be *memoryless*, meaning that it need not remember past states to make decisions in the present. However, many real world problems exhibit partial observability and require the agent to maintain a memory of the past to infer the latent states, plan, and make good decisions. These problems are best modeled via the framework of Partially Observable MDPs (POMDPs).

As a motivating example, consider a control task of navigating a robot that perceives the environment through a visual system like a first-person camera. Here, a single image may identify the agent's location, but it would not identify the agent's velocity, which is necessary for deciding how much force should be applied in order to accelerate or brake. For optimal control, the agent would have to maintain a memory of past images and infer its velocity from this historical information. This problem can be modeled as a POMDP where the system state is the position and velocity of the agent. However, the state cannot be inferred using a single image, hence it is *partially observable*.

Maintaining a memory and reasoning over histories in POMDPs is notoriously challenging, as evidenced by a number of complexity-theoretic barriers: computing the optimal policy (or planning) is computationally intractable (Papadimitriou & Tsitsiklis, 1987) and learning an unknown POMDP incurs a sample complexity that scales exponentially with the horizon (Mossel & Roch, 2005; Jin et al., 2020a). These lower bounds often involve constructions that require the agent to reason over very long histories. However, they are worst-case in nature, so they leave open the possibility of obtaining positive results for subclasses of POMDPs with special structure of practical interest.

One such structure concerns applications of POMDPs where the agent only needs a *short-term memory*. This structure holds in our motivating example, since the velocity can be recovered from just the most recent images. Short-term memory is also frequently used in the design of practical algorithms, which concatenate observations from the most recent time steps and use them to make decisions—a technique called "frame-stacking" (Mnih et al., 2013; 2015; Hessel et al., 2018). This gives rise to a natural question: *Can we develop a theoretical framework and design provably efficient algorithms for reinforcement learning with a short-term memory?*

[1]Microsoft Research, New York [2]Princeton University. Correspondence to: Sobhan Miryoosefi <miryoosefi@cs.princeton.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

Our contributions. In this paper, we address the question above by proposing a new class of models—m-step decodable POMDPs. This class is a subclass of general POMDPs where the latent state can be determined by the observations and actions of the m most recent time steps via an *unknown* decoding function $\phi^\star$ (see Assumption 2.2).

As a warm-up example, we first consider the tabular setting, where the number of states, observations, and actions, are all relatively small. Here a simple technique which stacks the observations and actions in the m most recent steps into a new "mega"-states yields an algorithm with sample complexity $\mathcal{O}(H(OA)^m)$ where O, A are the number of observations and actions respectively and H is the episode length. We also show an $\Omega(A^m)$ lower bound, establishing that an exponential dependence on m is indeed necessary.

Our main result concerns the rich-observation setting where the observation space can be arbitrarily complex (O is arbitrarily large) and one must use function approximation for generalization. We present a clean solution to this problem with a simple variant of the GOLF algorithm (Jin et al., 2021), which was originally proposed for RL with general function approximation in the observable/Markovian setting. We show that our algorithm finds a near-optimal policy within $\mathcal{O}(\text{poly}(H)A^m S \cdot \log|\mathcal{F}|)$ samples, where S is the number of latent states and $|\mathcal{F}|$ is cardinality of the function class. Most importantly, our sample complexity does not depend on the number of observations O. We further extend our result to the setting where the latent dynamics correspond to a linear MDP, with S in the sample complexity replaced by latent dimension d.

Our results in the rich observation setting crucially rely on a novel concept that we call the "moment matching policy," which breaks historical dependencies while matching the joint distribution of states, observations, and actions for a short time interval (See Section 5.2). These policies enable a low-rank or bilinear decomposition of the Bellman error of any value function in the POMDP, which is essential for obtaining sample efficient results in the rich observation setting (Jiang et al., 2017; Jin et al., 2021; Du et al., 2021). As such, the moment matching policies might be of independent interest for future research in partial observability.

1.1. Related Work

Partial observability is a central challenge in practical reinforcement learning settings and, as such, it has been the focus of a large body of empirical work. The two most popular high-level approaches are to use recurrent or other "temporally extended" neural architectures (Hausknecht & Stone, 2015; Zhu et al., 2017; Igl et al., 2018; Hafner et al., 2019), or to employ feature engineering (McCallum, 1993), for example by providing the most recent observations as input to the agent (Mnih et al., 2013; 2015; Hessel et al., 2018). However, we are not aware of any theoretical treatment of these methods in the RL context.

Turning to theoretical results, two lines of work are related to our own. The first addresses RL with partial observability. Kearns et al. (1999; 2002); Even-Dar et al. (2005) provide sparse sampling techniques that attain A^H-type sample complexity for various POMDP tasks, including without resets. These bounds have an undesirable exponential dependence on the horizon, which we show can be removed in some special cases. A more recent line of work (Azizzadenesheli et al., 2016; Guo et al., 2016; Jin et al., 2020a) use method of moment estimators (based on spectral methods for learning latent variable models (c.f., Anandkumar et al., 2014) to obtain guarantees in *undercomplete* tabular POMDPs. However, undercompleteness, which means that the emission matrix is robustly rank $|O|$, need not hold in our setting, so these results are orthogonal to ours.

The second line of work concerns rich observation RL, where the observation space can be infinite and arbitrarily complex, in (for the most part) *Markovian* environments. These works provide structural conditions that permit sample efficient RL with function approximation (Jiang et al., 2017; Sun et al., 2019; Jin et al., 2021; Du et al., 2021; Foster et al., 2021) as well as algorithms that are provably efficient in some special cases (Du et al., 2019; Misra et al., 2020; Agarwal et al., 2020; Uehara et al., 2021). However, as we will see, these structural conditions are not satisfied in our POMDP model so these results do not directly apply.

Outside of RL settings, the use of memory is prevalent in controls and time series prediction (Ljung, 1998; Box et al., 2015; Hamilton, 1994), dating back to the seminal work of Kalman (1960). Short-term memory is explicit in several autoregressive models, such as the AR and ARMA models. It is also classical to leverage memory in many control-theoretic settings. More recently, short-term memory has been employed in control settings, where one can use stability arguments to show that a short memory window suffices to approximate the optimal policy (Verhaegen, 1993; Arora et al., 2018; Agarwal et al., 2019; Oymak & Ozay, 2019; Simchowitz et al., 2019). These ideas provide further motivation for our study but the techniques developed in these continuous settings do not seem useful for discrete RL problems where exploration is challenging.

2. Preliminaries

Notation. We use $[H]$ to denote the set $\{1, \dots, H\}$. For any indexed sequence $a_1, a_2, \dots$, we use $a_{i:j}$ to denote the subsequence $(a_{\max\{1,i\}}, \dots, a_{\max\{1,j\}})$ for any $i, j \in \mathbb{Z}$ with $i \le j$. We adopt the standard big-oh notation and write $f = \tilde{\mathcal{O}}(g)$ to denote that $f = \mathcal{O}(g \cdot \max\{1, \text{polylog}(g)\})$.

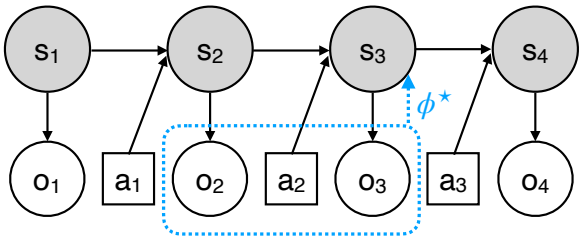

Figure 1. A schematic of a 2-step decodable POMDP. The latent state s_h can be recovered using only o_{h-1}, a_{h-1}, o_h, so a short-term memory suffices for decision making.

POMDPs. We consider an episodic Partially Observable Markov Decision Process (POMDP), which can be specified by $\mathcal{M} = (\mathcal{S}, \mathcal{O}, \mathcal{A}, H, \mathbb{P}, \mathbb{O}, r)$. Here $\mathcal{S}$ is the *unobservable* state space, $\mathcal{O}$ is the observation space, $\mathcal{A}$ is the action space, and H is the horizon. $\mathbb{P} = \{\mathbb{P}_h\}_{h=1}^H$ is a collection of *unknown* transition probabilities with $\mathbb{P}_h(s' \mid s, a)$ equal to the probability of transitioning to s' after taking action a in state s at the h^{th} step. $\mathbb{O} = \{\mathbb{O}_h\}_{h=1}^H$ are the *unknown* emissions with $\mathbb{O}_h(o \mid s)$ equal to probability that the environment emits observation o when in state s at the h^{th} step. $r = \{r_h : \mathcal{O} \to [0,1]\}_{h=1}^H$ are the deterministic reward functions.[1] Throughout the paper, we assume that $\sum_{h=1}^H r_h(o_h) \le 1$ almost surely. We assume our action space is finite, $|\mathcal{A}| \le A$, and in all sections except Section 4.1, we assume our state space is also finite, $|\mathcal{S}| \le S$.

Interaction protocol. In a POMDP, the states are hidden and unobservable; i.e., the agent is only able to see the observations and its own actions. Each episode starts with initial state s_1 which is sampled from some *unknown* initial distribution. Then, at each step $h \in [H]$, the environment emits observation $o_h \sim \mathbb{O}_h(\cdot \mid s_h)$, the agent observes $o_h \in \mathcal{O}$, receives reward $r_h(o_h)$, and takes action $a_h \in \mathcal{A}$ causing the environment to transition to $s_{h+1} \sim \mathbb{P}(\cdot \mid s_h, a_h)$.

Multi-step decodability. We first define the notion of reachable trajectories.

Definition 2.1 (Reachable trajectories)**.** We say a trajectory $\tau = (s_1, o_1, a_1, r_1, s_2, \ldots, s_H, o_H, a_H, r_H)$ is *reachable* if the probability $P((s,o)_{1:H} | a_{1:H}) = (\prod_{h=1}^H \mathbb{O}(o_h|s_h)) \cdot (\prod_{h=1}^{H-1} \mathbb{P}(s_{h+1}|s_h, a_h))$ is strictly positive.

Now we present the key structural assumption of this paper, which assumes that a suffix of length m of the history suffices to decode the latent state. We use $\mathcal{Z}_h$ to denote the set of suffixes at step h, given by $\mathcal{Z}_h = (\mathcal{O} \times \mathcal{A})^{\min\{h-1, m-1\}} \times \mathcal{O}$.[2] Additionally, since it will appear frequently in subscripts in the sequel, let $m(h) = \min\{h - m + 1, 1\}$.

Assumption 2.2 (m-step decodability)**.** There exists an *unknown* decoder $\phi^\star = \{\phi_h^\star : \mathcal{Z}_h \to \mathcal{S}\}_{h=1}^H$ such that for every *reachable* trajectory $\tau = (s,o,a)_{1:H}$, we have $s_h = \phi_h^\star(z_h)$ for all $h \in [H]$, where $z_h = ((o,a)_{m(h):h-1}, o_h)$.

We call a POMDP satisfying Assumption 2.2 an **m-step decodable POMDP**. An example with $m = 2$ is illustrated in Figure 1. Note that restricting decodability to only hold on reachable sequences results in a weaker assumption, which can include more practical settings.

Our model is a generalization of the block Markov decision process (BMDP) (Jiang et al., 2017; Du et al., 2019), which corresponds to the case where $m = 1$. However, we emphasize that when $m = 1$ there is no partial observability since the current observation suffices for decoding the hidden state. Thus the BMDP model does not require memory while, for $m > 1$, our model does.

Policies and value functions. For m-step decodable POMDPs, we consider the class of m-step policies. An m-step policy π is a collection $\pi = \{\pi_h : \mathcal{Z}_h \to \mathcal{A}\}$ that maps suffixes of length m of the history to actions. The agent follows policy π by choosing action $a_h = \pi_h(z_h)$ at the h^{th} step, where $z_h = ((o,a)_{m(h):h-1}, o_h) \in \mathcal{Z}_h$. We denote V^π as the value for policy π, defined as the expected total reward obtained when following policy π, that is $V^\pi = \mathbb{E}_\pi[\sum_{h=1}^H r_h(o_h)]$.

We can similarly define the value at step h to be the expected future reward when starting from step h. While this value may depend on the entire history in general, it is not hard to show that in m-step decodable POMDPs with an m-step policy π, this value only depends on the suffix of length m. Mathematically, we can define $V_h^\pi : \mathcal{Z}_h \to [0,1]$ to be the value function at step h for (the m-step) policy π as

$$V_h^\pi(z) := E_\pi\Big[\sum_{h'=h+1}^H r_{h'}(o_{h'}) \mid z_h = z\Big].$$

Similarly we define $Q_h^\pi : \mathcal{Z}_h \times \mathcal{A} \to [0,1]$ to be the Q-value function at step h for (the m-step) policy π as

$$Q_h^\pi(z,a) := E_\pi\Big[\sum_{h'=h+1}^H r_{h'}(o_{h'}) \mid z_h = z, a_h = a\Big].$$

Furthermore, Assumption 2.2 guarantees that there exists an m-step policy $\pi^\star$ which is optimal in the sense $V^{\pi^\star} = \max_{\pi \in \Pi} V^\pi$ where Π is the class of all policies, which may depend on the entire history. We use $V^\star$, $V_h^\star$, and $Q_h^\star$ to denote $V^{\pi^\star}$, $V_h^{\pi^\star}$, and $Q_h^{\pi^\star}$ respectively.

[1] We study deterministic reward for simplicity. Our results readily generalize to random rewards.

[2] When $h \le m$, this suffix includes the entire history starting from time step 1.

We define the *Bellman operator* $\mathcal{T}_h$ at step h as

$$(\mathcal{T}_h g)(z,a) := \mathbb{E}\big[r_{h+1}(o_{h+1}) + \max_{a_{h+1}\in\mathcal{A}} g(z_{h+1}, a_{h+1}) \mid z_h = z, a_h = a\big],$$

for any function $g : \mathcal{Z}_{h+1} \times \mathcal{A} \to [0,1]$ that depends on m-step suffix. It is not hard to check that $Q^\star$ satisfies the Bellman optimality equation $Q^\star_h(z,a) = (\mathcal{T}_h Q^\star_{h+1})(z,a)$ for all $h \in [H]$ and $(z,a) \in \mathcal{Z}_h \times \mathcal{A}$.

Finally, for two non-stationary policies π_1, π_2 we use the notation $\pi_1 \circ_t \pi_2$ be a non-stationary policy that executes π_1 for $t-1$ time steps and then, starting from the t^{th} time step, executes π_2.

Learning objective. Our objective is to learn an ϵ-optimal policy $\widehat{\pi}$, which satisfies $V^{\widehat{\pi}} \geq V^\star - \epsilon$.

2.1. Function approximation

In the function approximation setting, the learner is given a function class $\mathcal{F} = \mathcal{F}_1 \times \cdots \times \mathcal{F}_H$, where $\mathcal{F}_h \subseteq (\mathcal{Z}_h \times \mathcal{A} \to [0,1])$ consists of candidate functions to approximate $Q^\star_h$—the optimal Q-value function at step h. Without loss of generality we assume that $f_{H+1} \equiv 0$. We present two assumptions that are commonly adopted in the literature to avoid challenges associated with reinforcement learning with function approximation (e.g., the hardness results in Krishnamurthy et al. 2016; Weisz et al. 2021).

Assumption 2.3 (Realizability)**.** $Q^\star_h \in \mathcal{F}_h$ for all $h \in [H]$.

This assumption requires that our function class $\mathcal{F}$ in fact contains the the optimal Q-value function, $Q^\star$.

Assumption 2.4 (Generalized Completeness)**.** $\mathcal{T}_h f_{h+1} \in \mathcal{G}_h$ for all $h \in [H]$ and $f_{h+1} \in \mathcal{F}_{h+1}$, where $\mathcal{G} = \mathcal{G}_1 \times \cdots \times \mathcal{G}_H$ is an auxiliary function class provided to the learner, with $\mathcal{F}_h \subseteq \mathcal{G}_h \subseteq (\mathcal{Z}_h \times \mathcal{A} \to [0,1])$.

The *generalized completeness* (Antos et al., 2008; Chen & Jiang, 2019) assumption requires the auxiliary function class $\mathcal{G}$ to be rich enough so that applying the Bellman operator on any function in the original class $\mathcal{F}$ results in a function in $\mathcal{G}$. If we choose $\mathcal{G} = \mathcal{F}$, Assumption 2.4 reduces to the standard completeness assumption, but separating the two classes provides more flexibility.

We use covering numbers to capture the statistical complexity, or effective size, of the classes $\mathcal{F}$ and $\mathcal{G}$.

Definition 2.5 (ϵ-cover)**.** The ϵ-covering of a set $\mathcal{X}$ under a metric ρ, denoted by $\mathcal{N}(\mathcal{X}, \epsilon, \rho)$ is the minimum integer n such that there exists a subset $\mathcal{X}_0 \subseteq \mathcal{X}$ with $|\mathcal{X}_0| = n$ and for any $x \in \mathcal{X}$ there exists $y \in \mathcal{X}_0$ such that $\rho(x,y) \leq \epsilon$.

In this work, for the function class $\mathcal{F} = \mathcal{F}_1 \times \cdots \times \mathcal{F}_H$, we use the metric $\rho(f^{(1)} - f^{(2)}) = \max_{h\in H} ||f^{(1)}_h - f^{(2)}_h||_\infty$ where $f^{(1)}, f^{(2)} \in \mathcal{F}$. Since this metric is fixed throughout the paper, we use a simpler notation of $\mathcal{N}_{\mathcal{F}}(\epsilon)$ to denote the ϵ-covering number of $\mathcal{F}$.

Finally, let $\pi_f = \{z_h \mapsto \arg\max_{a\in\mathcal{A}} f_h(z_h, a)\}_{h=1}^H$ denote the greedy policy with respect to $f \in \mathcal{F}$, where ties are broken in a canonical fashion.

3. Warmup: Tabular Case

We start by considering a basic setting where the numbers of states, actions, and observations are all finite and small, so we additionally have $|\mathcal{O}| \leq O$. In this setting, we describe a simple reduction from an m-step decodable POMDP to a new MDP with augmented states. With this reduction at hand, we can to apply any RL algorithms designed for the fully observable setting to learn a near optimal policy.

In the reduction to an MDP, instead of using only the current observation o_h as the state at time h, we use the m-length suffix of observations and actions z_h. We refer to such a suffix as a *megastate*. Formally, the reduction uses a time-dependent extended state space $\mathcal{S}^{m,h} = \mathcal{Z}_h$, and the next result establishes that $\mathcal{S}^{m,h}$ induces Markovian dynamics. Additionally, an optimal policy of this MDP is also an optimal policy of the original m-step decodable POMDP.[3]

Proposition 3.1 (Megastate MDP)**.** *The state space $\mathcal{S}^{m,h}$ induces Markovian dynamics $\mathbb{P}^m$ and reward r^m. Let this MDP be $\mathcal{M}^m = \big(\mathcal{S}^{m,h}, \mathcal{A}, H, \mathbb{P}, r\big)$. An optimal policy of $\mathcal{M}^m$ is an optimal policy of the m-step decodable POMDP.*

We refer to $\mathcal{M}^m$ as the megastate MDP. With this proposition, we can apply any RL algorithm (e.g., UCB-VI by Azar et al. 2017) to the megastate MDP to learn a near optimal policy for the original POMDP. Since the cardinality of the state space of $\mathcal{M}^m$ at each step is $\max_{h\in[H]} \big|\mathcal{S}^{m,h}\big| \leq O^m A^{m-1}$. We immediately obtain the following result.

Corollary 3.2 (Upper bound, tabular setting)**.** *For any $\epsilon, \delta \in (0,1)$, UCB-VI applied on to the megastate-MDP $\mathcal{M}^m$ learns an ϵ-optimal policy for the original m-step decodable POMDP with probability greater than $1-\delta$ given $O\left(O^m A^m poly(H) \log(1/\delta)/\epsilon^2\right)$ samples.*

We remark that the sample complexity scales exponentially with the decoding length m. The next lower-bound verifies the necessity of the $O(A^m)$ term in the upper bound, so some exponential dependence is required. It follows by a reduction to the lower bound of Krishnamurthy et al. (2016); we show that their construction is, in fact, an m-step decodable POMDP. This yields the following result.

Proposition 3.3 (Lower bound, tabular setting)**.** *There exists an m-step decodable MDP that requires at least $\Omega(A^m/\epsilon^2)$ samples to find an ϵ-optimal policy.*

Thus the A^m dependence in the megastate reduction is opti-

[3] All proofs are deferred to the appendices.

mal, although it is not clear whether the O^m dependence is necessary, which we discuss in more detail in Section 6. Regardless, the megastate reduction is a reasonable approach for m-step decodable POMDPs when the observation space is small, but, in many applications, the observations represent complex objects (like images or high-dimensional data) so that even linear in O dependence is unsatisfactory. Such problems lie outside the scope of tabular methods, and a fundamentally different approach is required.

4. Main results

In this section we present our main results which address the *rich observation* setting, where the number of observation O is extremely large or infinite. The standard approach to tackle such problems is via *value function approximation*: we assume access to a function class $\mathcal{F}$ of candidate Q-value functions. Given such a class, the goal is to learn a near-optimal policy with sample complexity scaling with the statistical complexity of $\mathcal{F}$—in our case the log covering number $\log \mathcal{N}_{\mathcal{F}}$—but independent of the size of the observation space. In this section, we develop an algorithm for rich observation m-step decodable POMDPs and analyze its sample complexity.

Our algorithm, which we call m-GOLF, is displayed in Algorithm 1. It is an adaptation of the GOLF algorithm, developed by Jin et al. (2021), for the rich observation MDP setting. m-GOLF itself differs from GOLF only in one seemingly minor way, although this is quite critical for our analysis. Before turning to this difference, let us review the high-level algorithmic approach.

GOLF, and m-GOLF, are optimistic algorithms that maintain a confidence-set of plausible Q-value functions, and act optimistically with respect to this set. Given a function class $\mathcal{F}$, we first collect a few observations o_1 and estimate the predicted initial value, i.e., $\mathbb{E}\left[f(o_1, \pi_f(o_1))\right]$, for each $f \in \mathcal{F}$. Then, we initialize the confidence set $\mathcal{B}^0 \leftarrow \mathcal{F}$ and empty datasets $\{\mathcal{D}_h\}_{h=1}^H$, one for each time step. Then for each epoch $k \in [K]$ we follow three steps:

1. *Optimistic planning*. Compute the function $f \in \mathcal{B}^{k-1}$ with largest predicted initial value.

2. *Data collection.* Collect one trajectory by following $\pi_{f_k} \circ_{m(h)} \mathrm{Uniform}(\mathcal{A})$ for each $h \in [H]$. That is we collect h trajectories total, rolling in with the greedy policy π_{f_k} until time $h - m$ and rolling out randomly.

3. *Refine the confidence set.* Update the confidence set to $\mathcal{B}^k$ using the newly collected trajectories. The confidence set is designed so that $Q^\star \in \mathcal{B}^k$ for all $k \in [K]$ and that all functions in $\mathcal{B}^k$ have low squared Bellman error on the data collected in the previous episodes.

After iterating through these steps for several epochs, m-GOLF outputs uniform mixture over all previous policies $\{\pi^k\}_{k=1}^K$.

The main difference between GOLF and m-GOLF is in the data collection procedure. Instead of collecting H trajectories per epoch, GOLF collets a single trajectory where all actions are taken by the greedy policy π_{f_k}. On the other hand, in m-GOLF, we interrupt the greedy policy and execute random actions so that the tuple z_h that is added to $\mathcal{D}^h$ is collected from $\pi_{f_k} \circ_{m(h)} \mathrm{Uniform}$. At face value, this modification is relatively benign, but we will see how interrupting the greedy policy is critical to establishing sample complexity guarantees in the m-step decodable POMDP.

We analyze m-GOLF in two settings. The first is where the underlying/latent MDP is tabular, meaning that S and A are small. The second setting is where the latent MDP has a linear or low rank structure. Our first theorem provides a sample complexity guarantee for m-GOLF when the latent dynamics are tabular.

Theorem 4.1. *Under Assumptions 2.2, 2.3, and 2.4, there exists an absolute constant c such that for any $\delta \in (0, 1]$ and $\epsilon > 0$, if we choose*

$$
\begin{aligned}
K_{\mathrm{est}} &= c \cdot \left(\log[\mathcal{N}_{\mathcal{F}}(\epsilon)/\delta]/\epsilon^2 \right) \\
\beta &= c \cdot \left(\log\left[\mathcal{N}_{\mathcal{G}}(\rho) K H/\delta\right] + K\rho \right) \\
\rho &= \epsilon^2 \cdot \left[H^2 A^m S \log[S/\epsilon]\right]^{-1}
\end{aligned}
$$

in m-GOLF (Algorithm 1), then the output policy π^{out} is $\mathcal{O}(\epsilon)$-optimal with probability at least $1 - \delta$ if

$$
K \geq \tilde{\Omega}\left(\frac{H^2 A^m S}{\epsilon^2} \cdot \log\left[\frac{\mathcal{N}_{\mathcal{G}}(\rho)}{\delta} \right] \right).
$$

Theorem 4.1 establishes a sample complexity bound for m-GOLF scaling as $\mathrm{poly}(S, A^m, H, \mathrm{comp}(\mathcal{F}, \mathcal{G}), 1/\epsilon)$ where $\mathrm{comp}(\cdot)$ is our measure of statistical complexity, namely, the covering number of the function class. Unlike the megastate reduction, there is no explicit dependence on the size of the observation space O; instead the bound scales with the complexity of the function class, which allows us to exploit domain knowledge and inductive biases when deploying the algorithm. In addition, the bound exhibits a linear dependence on S, the cardinality of the latent state space. This dependence matches GOLF (Jin et al., 2021) and improves over previous upper bounds for the Block MDP case (Jiang et al., 2017; Du et al., 2021), which we recall is a special case with $m = 1$. We emphasize that these previous analyses do not seem to yield guarantees when $m > 2$, as we will see in Section 5.

Algorithm 1 m-GOLF: GOLF for m-step decodable POMDP

Initialize: $\mathcal{D}_1, \dots, \mathcal{D}_H \leftarrow \emptyset$, $\mathcal{B}^0 \leftarrow \mathcal{F}$.
Estimate value of initial state by collecting K_{est} episodes and only keeping their first observations, denoted by $\hat{o}_1^1, \dots, \hat{o}_1^{K_{\mathrm{est}}}$. For $f \in \mathcal{F}$, define

$$\hat{f}_1 = (1/K_{\mathrm{est}}) \sum_{i=1}^{K_{\mathrm{est}}} f(\hat{o}_1^i, \pi_f(\hat{o}_1^i))$$

for epoch k from 1 to K **do**
  Choose policy $\pi^k = \pi_{f^k}$, where $f^k = \mathrm{argmax}_{f \in \mathcal{B}^{k-1}} \hat{f}_1$.
  for step h from 1 to H **do**
    Collect $z_h = (o_{h-m+1}, a_{h-m+1}, \dots, o_h)$, a_h, r_h, and o_{h+1} by executing π^k at step $1, \dots, h-m$ and taking action uniformly at random at step $h-m+1, \dots, h$.
    Augment $\mathcal{D}_h = \mathcal{D}_h \cup (z_h, a_h, r_h, o_{h+1})$ for all $h \in [H]$.
  end for
  Update

$$\mathcal{B}^k = \left\{ f \in \mathcal{F} : \mathcal{L}_{\mathcal{D}_h}(f_h, f_{h+1}) \le \inf_{g \in \mathcal{G}_h} \mathcal{L}_{\mathcal{D}_h}(g, f_{h+1}) + \beta \text{ for all } h \in [H] \right\},$$

$$\text{where } \mathcal{L}_{\mathcal{D}_h}(\xi_h, \zeta_{h+1}) = \sum_{(z_h, a_h, r_h, o_{h+1}) \in \mathcal{D}_h} [\xi_h(z_h, a_h) - r_h - \max_{a' \in \mathcal{A}} \zeta_{h+1}(z_{h+1}, a')]^2.$$

end for
Output π^{out} uniform mixture policy over $\{\pi^k\}_{k=1}^K$.

4.1. Linear m-step Decodable POMDP

In this subsection, we show that m-GOLF extends to the setting where the number of state S is also large. Specifically, we consider the case where the latent MDP is a linear MDP (Jin et al., 2020b)—there exists an unknown feature map $\psi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^{d_{lin}}$ such that the transition dynamics are linear in ψ. Interestingly, we show that m-GOLF is still applicable without change. It retains a similar sample complexity guarantee where we replace the dependence on S with a dependence on the latent dimensionality d_{lin}.

Formally, a linear MDP is defined as follows:

Definition 4.2 (Linear MDP)**.** An MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, \mathbb{P}, r)$ is said to be a linear with a feature map $\psi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^{d_{\mathrm{lin}}}$, if for any $h \in [H]$: There exists d_{lin} unknown (signed) measures $\boldsymbol{\mu}_h = \{\mu_h^{(1)}, \dots, \mu_h^{(d_{\mathrm{lin}})}\}$ over $\mathcal{S}$ such that for any $(s, a) \in \mathcal{S} \times \mathcal{A}$ we have

$$\mathbb{P}_h(\cdot \mid s, a) = \langle \boldsymbol{\mu}_h(\cdot), \psi(s, a) \rangle$$

We assume the standard normalization: $||\psi(s, a)|| \le 1$ for all $(s, a) \in \mathcal{S} \times \mathcal{A}$, $||\int v(s) \boldsymbol{\mu}_h(s)||_2 \le \sqrt{d_{\mathrm{lin}}}$ for all $h \in [H]$ and v with $\|v\|_\infty \le 1$.

The following result gives a sample complexity guarantee for m-GOLF in the more general linear m-step decodable POMDP model.

Theorem 4.3. *Under Assumptions 2.2, 2.3, and 2.4 and assuming linear latent MDP; there exists an absolute constant c such that for any $\delta \in (0, 1]$ and $\epsilon > 0$, if we choose*

$$K_{\mathrm{est}} = c \cdot \left(\log[\mathcal{N}_{\mathcal{F}}(\epsilon)/\delta]/\epsilon^2 \right)$$
$$\beta = c \cdot \left(\log \left[\mathcal{N}_{\mathcal{G}}(\rho) K H / \delta \right] + K\rho \right)$$
$$\rho = \epsilon^2 \cdot \left[H^2 A^m d_{lin} \log[d_{lin}/\epsilon] \right]^{-1}$$

in m-GOLF (Algorithm 1), then the output policy π^{out} is $\mathcal{O}(\epsilon)$-optimal with probability at least $1 - \delta$ if

$$K \ge \tilde{\Omega} \left(\frac{H^2 A^m d_{lin}}{\epsilon^2} \cdot \log \left[\frac{\mathcal{N}_{\mathcal{G}}(\rho)}{\delta} \right] \right).$$

Theorem 4.3 is almost the same as Theorem 4.1 with the dependency on the number of latent state S replaced by the ambient dimensionality d_{lin}. As a result, Theorem 4.3 can apply to the case where the number of state S is extremely large or even infinite, as long as the underlying MDP has a linear structure.

5. Challenges and Proof Overview

In this section we elaborate on the main challenges in analysis, explain our main technique and provide a proof overview for Theorem 4.1. For clarity, we will focus on the special case of 2-step decodable POMDP in this setting. We refer reader to Appendix B for cases where $m > 2$.

5.1. Challenges: Bellman Rank is Prohibitively Large

We first note that existing postive results for RL algorithms with general function approximation such as OLIVE (Jiang et al., 2017), GOLF (Jin et al., 2021) all rely on the structural properties that certain complexity measure on the Bellman error is small. One such complexity is the Bellman rank (Jiang et al., 2017), which explains the tractability of block MDP (the special case of m-step decodable POMDP with $m = 1$).

Consider the Bellman error at the h^{th} time step of a function $f \in \mathcal{F}$ when executing roll-in policy π, given by

$$\mathcal{E}_h(\pi, f) = \mathbb{E}[(f_h - \mathcal{T}_h f_{h+1})(z_h, \pi_f(z_h)) \mid a_{1:h-1} \sim \pi].$$

Bellman rank is defined as the smallest integer M such that the Bellman error can be factorized as inner product in M dimensional linear space. That is, there exists $\zeta, \xi \in \mathbb{R}^M$ such that $\mathcal{E}_h(\pi, f) = \langle \zeta(\pi), \xi(f) \rangle$.

Intuitively, Bellman rank describes how much information is shared among past (roll-in policy π) and future (value function f) at step h. In the special case of 1-step decodable POMDP, it suffices to consider 1-step policy where the choice of action a_h only depends on the current observation o_h. In this case, given the state s_h at the current step h, the past—$(s, o, a)_{1:h-1}$ (which only depends on roll-in policy π) is completely independent of the future—$(o_h, a_h, (s, o, a)_{h+1:H})$ (which only depends on function f). Therefore, it can be shown the Bellman rank of 1-step decodable POMDP (i.e. block MDP) is upper bounded by the number of states S (Jiang et al., 2017).

However, such independent structure completely collapses in 2-step decodable POMDP, where we must consider 2-step policy. Due to the nature of such policies, the choice of action a_h not only depends on the current observation o_h, but also the observation and action in the previous step o_{h-1}, a_{h-1} (as shown in Figure 2 blue box). Therefore, conditioning s_h, the past is no longer independent of the future. This can potentially lead to very large Bellman rank.

Formally, our next result shows that the Bellman rank in 2-step decodable POMDP can be prohibitively large—there exists examples where the Bellman rank can be lower bounded by the cardinality of the observation space $\Omega(O)$. This is highly undesirable in the rich observation setting where O can be even infinite. Furthermore, we also show that OLIVE algorithm—which was proposed in (Jiang et al., 2017) to solve all RL problems with small Bellman rank—needs at least $\Omega(O)$ samples to find an $O(1)$ optimal policy.

Proposition 5.1 (Bellman rank of m-step decodable POMDP is large)**.** *There exists a 2-step decodable POMDP $\mathcal{M}$ and a function class $\mathcal{F}$ such that the Bellman rank of $(\mathcal{M}, \mathcal{F})$ is $\Omega(O)$. Additionally,* OLIVE *instantiated with $\mathcal{F}$ requires $\Omega(O)$ samples to find an $o(1)$ optimal policy.*

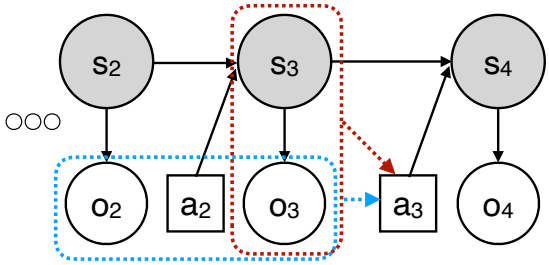

Figure 2. An illustration of the dependency structure of a moment matching policy, depicted in red, and a regular policy, depicted in blue, in a 2-step decodable POMDP. The moment matching policy $\mu^{\pi,h+1}$ selects action a_h based on the state s_h and observation o_h to match the distribution $\mathbb{P}^\pi[a_h \mid s_h, o_h]$. It breaks the dependence on the history by marginalizing out (o_{h-1}, a_{h-1}), but correctly matches the distribution $\mathbb{P}^\pi[o_{h+1}, a_h, o_h]$.

This highlights the challenge on directly applying existing results or techniques to solve m-step decodable POMDPs. Although OLIVE solves a 1-step decodable POMDP—namely, a block MDP—it fails in solving an m-step decodable POMDP for $m \geq 2$.

5.2. Proof Overview & Moment Matching Policy

Our main proof idea revolves around breaking the complicated dependencies introduced by multiple-step policies, which requires a number of crucial observations.

Our first key observation is that, in order to establish the sample complexity for GOLF algorithm, we don't necessarily need to prove the low rank structure of the Bellman error. We only need to alternatively identify an auxiliary function $\mathcal{E}_h^\star(\pi, f)$ which satisfies the following two properties (see formal statement in Lemma B.9):

1. *Matches with standard bellman error when $\pi = \pi_f$*:
$$\mathcal{E}_h^\star(\pi_f, f) = \mathcal{E}_h(\pi_f, f).$$
2. *Has a low-rank decomposition*:
$$\mathcal{E}_h^\star(\pi, f) = \langle \zeta(\pi), \xi(f) \rangle .$$
for some $\zeta(\cdot), \xi(\cdot) \in \mathbb{R}^M$ with small M,

This discovery gives us a lot extra freedom in designing the functional form of the $\mathcal{E}_h^\star$. In particular, for 2-step decodable POMDP, we define $\mathcal{E}_h^\star$ to be the normal Bellman error but with the policy at step $h-1$ changed from roll-in policy π to a new policy μ_f which depends only on f instead of π.

$$\mathcal{E}_h^\star(\pi, f)$$
$$\equiv \mathbb{E}[(f_h - \mathcal{T}_h f_{h+1})(z_h, \pi_f(z_h)) \mid a_{1:h-1} \sim \pi \circ_{h-1} \mu_f].$$

The second key observation is that we can choose μ_f in a form which breaks the dependency and allows low-rank dependency. Concretely, instead of choosing μ_f to be standard 2-step policy where a_{h-1} will then depend on $(o_{h-2}, a_{h-2}, o_{h-1})$, we choose μ_f to be the policy that only depends on (s_{h-1}, o_{h-1}) (See Figure 2 red box). The benefit of considering such policy is that now conditioned on s_{h-1} at step $h-1$, the past—$(s,o,a)_{1:h-2}$ (which only depends on roll-in policy π) is now independent of the future—$(o_{h-1}, a_{h-1}, (s,o,a)_{h:H})$ (which only depends on function f). This immediately leads to a low-rank decomposition of $\mathcal{E}_h^\star(\pi, f)$ with rank $M = S$.

Our third key observation is that we can carefully choose the value of μ_f within the form specified above, so that $\mathcal{E}_h^\star(\pi, f)$ matches the Bellman error $\mathcal{E}_h(\pi, f)$ when roll-in policy is the greedy policy of f, i.e. $\pi = \pi_f$. This is done by the idea of "*moment-matching*", which is the reason we call policy μ_f the "moment matching policy". Specifically, we choose policy μ_f such that

$$\mu_f(a_{h-1}|(o,s)_{h-1}) = \mathbb{E}_{\pi_f}[\pi_f(a_{h-1}|z_{h-1})|(o,s)_{h-1}]$$

which is policy of π_f averaging over all trajectories with $(o,s)_{h-1}$ fixed. The most important property of this policy is that the joint distributions over z_h for policy π_f and policy $\pi_f \circ_{h-1} \mu_f$ (which switches at time step $h-1$) are the same. In symbol:

$$P_{\pi_f}(z_h) = P_{\pi_f \circ_{h-1} \mu_f}(z_h)$$

This directly leads to the matching in the Bellman error. This finishes our construction of $\mathcal{E}_h^\star(\pi, f)$ satisfying the two properties mentioned earlier and the main part of proof overview.

Finally, we comment that our construction of μ_f depends on the latent state s which can not be observed in POMDP. Nevertheless, m-GOLF bypasses this problem by executing a uniform action for m time steps, instead of executing μ_f; taking the uniform action for the last m time steps allows us to upper bound $\mathcal{E}_h^\star(\pi, f)$ using the importance sampling trick, while only suffering an A^m degradation in the sample complexity. Such factor is necessary according to Proposition 3.3.

6. Conclusion

In this paper, we initiate the study of m-step decodable POMDPs as a model for understanding the role of short-term memory in sequential decision making. We consider both the tabular and function approximation setting and obtain results that scale exponential with the memory window rather than the horizon, which could be much larger. In the function approximation case, our techniques rely crucially on the moment matching policy to break dependency on the history, and we hope this concept may be useful in other settings with partial observability.

We believe our progress on understanding short-term memory is just scratching the surface and there are many questions that remain open even in the m-step decodable POMDP model. The most basic question pertains to the tabular setting, where the upper bound in Corollary 3.2 and the lower bound in Proposition 3.3 differ by an O^m factor. Instantiating m-GOLF in the tabular setting also incurs an O^m factor. On the other hand, the next result shows that by using a carefully constructed policy class in an importance sampling approach, we can avoid the O^m factor in exchange for an A^H factor, which could be more favorable in some settings. See Appendix C for details and the proof.

Proposition 6.1. *There exists an algorithm such that for any $m \le H$ and any m-step decodable POMDP, the algorithm returns an ϵ-optimal policy with probability greater than $1-\delta$ given $poly(A^H, O, S, H, \log(1/\delta))/\epsilon^2$ samples.*

Based on this result, we conjecture that the O^m factor can be avoided and that $A^m \mathrm{poly}(H, S, O, A)$ is the optimal sample complexity for m-step decodable POMDPs. However, this question remains open.

The second question concerns whether we can avoid completeness, as defined in Assumption 2.4, in the rich observation setting. Intuition from prior works suggests that if we could replace the squared bellman error constraint with one on the average Bellman errors, then an algorithm and analysis similar to OLIVE would successfully do this. However, when working with average Bellman errors, introducing the moment matching policy requires explicitly importance weighting with them, meaning that we must use these moment matching policies in the algorithm and not just the analysis. Unfortunately since we do not know the moment matching policies (or a small class containing them), this approach seems to fail.

We believe that characterizing the optimal sample complexity (in the tabular setting) or removing the completeness assumption (in the rich observation setting) will require new techniques and be a mark of significant progress toward expanding our understanding of decision making with short-term memory. We look forward to studying these questions in future work.

References

Agarwal, A., Kakade, S., Krishnamurthy, A., and Sun, W. Flambe: Structural complexity and representation learning of low rank mdps. *Advances in Neural Information Processing Systems*, 33, 2020.

Agarwal, N., Bullins, B., Hazan, E., Kakade, S., and Singh, K. Online control with adversarial disturbances. In *International Conference on Machine Learning*, pp. 111–119. PMLR, 2019.

Anandkumar, A., Ge, R., Hsu, D., Kakade, S. M., and Telgarsky, M. Tensor decompositions for learning latent variable models. *Journal of machine learning research*, 15:2773–2832, 2014.

Antos, A., Szepesvári, C., and Munos, R. Learning near-optimal policies with bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 71(1):89–129, 2008.

Arora, S., Hazan, E., Lee, H., Singh, K., Zhang, C., and Zhang, Y. Towards provable control for unknown linear dynamical systems. In *International Conference on Learning Representations, Workshop Track*, 2018.

Azar, M. G., Osband, I., and Munos, R. Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*, pp. 263–272. PMLR, 2017.

Azizzadenesheli, K., Lazaric, A., and Anandkumar, A. Reinforcement learning of pomdps using spectral methods. In *Conference on Learning Theory*, pp. 193–256. PMLR, 2016.

Box, G. E., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

Chen, J. and Jiang, N. Information-theoretic considerations in batch reinforcement learning. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1042–1051. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/chen19e.html.

Du, S., Krishnamurthy, A., Jiang, N., Agarwal, A., Dudik, M., and Langford, J. Provably efficient rl with rich observations via latent state decoding. In *International Conference on Machine Learning*, pp. 1665–1674. PMLR, 2019.

Du, S. S., Kakade, S. M., Lee, J. D., Lovett, S., Mahajan, G., Sun, W., and Wang, R. Bilinear classes: A structural framework for provable generalization in rl. *arXiv preprint arXiv:2103.10897*, 2021.

Even-Dar, E., Kakade, S. M., and Mansour, Y. Reinforcement learning in pomdps without resets. In *International Joint Conference on Artificial Intelligence*, 2005.

Foster, D. J., Kakade, S. M., Qian, J., and Rakhlin, A. The statistical complexity of interactive decision making. *arXiv preprint arXiv:2112.13487*, 2021.

Guo, Z. D., Doroudi, S., and Brunskill, E. A pac rl algorithm for episodic pomdps. In *Artificial Intelligence and Statistics*, pp. 510–518. PMLR, 2016.

Hafner, D., Lillicrap, T., Ba, J., and Norouzi, M. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

Hamilton, J. D. *Time series analysis*. Princeton university press, 1994.

Hausknecht, M. and Stone, P. Deep recurrent q-learning for partially observable mdps. In *2015 aaai fall symposium series*, 2015.

Hessel, M., Modayil, J., Van Hasselt, H., Schaul, T., Ostrovski, G., Dabney, W., Horgan, D., Piot, B., Azar, M., and Silver, D. Rainbow: Combining improvements in deep reinforcement learning. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

Igl, M., Zintgraf, L., Le, T. A., Wood, F., and Whiteson, S. Deep variational reinforcement learning for pomdps. In *International Conference on Machine Learning*, pp. 2117–2126. PMLR, 2018.

Jiang, N., Krishnamurthy, A., Agarwal, A., Langford, J., and Schapire, R. E. Contextual decision processes with low bellman rank are pac-learnable. In *International Conference on Machine Learning*, pp. 1704–1713. PMLR, 2017.

Jin, C., Kakade, S. M., Krishnamurthy, A., and Liu, Q. Sample-efficient reinforcement learning of undercomplete pomdps. *arXiv:2006.12484*, 2020a.

Jin, C., Yang, Z., Wang, Z., and Jordan, M. I. Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, pp. 2137–2143. PMLR, 2020b.

Jin, C., Liu, Q., and Miryoosefi, S. Bellman eluder dimension: New rich classes of rl problems, and sample-efficient algorithms. *arXiv preprint arXiv:2102.00815*, 2021.

Kalman, R. E. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 1960.

Kearns, M., Mansour, Y., and Ng, A. Y. A sparse sampling algorithm for near-optimal planning in large markov decision processes. *Machine learning*, 49(2):193–208, 2002.

Kearns, M. J., Mansour, Y., and Ng, A. Y. Approximate planning in large pomdps via reusable trajectories. In *NIPS*, pp. 1001–1007. Citeseer, 1999.

Krishnamurthy, A., Agarwal, A., and Langford, J. Pac reinforcement learning with rich observations. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates,

Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/2387337ba1e0b0249ba90f55b2ba2521-Paper.pdf.

Ljung, L. *System Identification: Theory for the User*. Pearson Education, 1998.

McCallum, R. A. Overcoming incomplete perception with utile distinction memory. In *Proceedings of the Tenth International Conference on Machine Learning*, pp. 190–196, 1993.

Misra, D., Henaff, M., Krishnamurthy, A., and Langford, J. Kinematic state abstraction and provably efficient rich-observation reinforcement learning. In *International conference on machine learning*, pp. 6961–6971. PMLR, 2020.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *nature*, 518(7540): 529–533, 2015.

Mossel, E. and Roch, S. Learning nonsingular phylogenies and hidden markov models. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pp. 366–375, 2005.

Oymak, S. and Ozay, N. Non-asymptotic identification of lti systems from a single trajectory. In *2019 American control conference (ACC)*, pp. 5655–5661. IEEE, 2019.

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

Russo, D. and Van Roy, B. Eluder dimension and the sample complexity of optimistic exploration. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper/2013/file/41bfd20a38bb1b0bec75acf0845530a7-Paper.pdf.

Simchowitz, M., Boczar, R., and Recht, B. Learning linear dynamical systems with semi-parametric least squares. In *Conference on Learning Theory*, pp. 2714–2802. PMLR, 2019.

Sun, W., Jiang, N., Krishnamurthy, A., Agarwal, A., and Langford, J. Model-based rl in contextual decision processes: Pac bounds and exponential improvements over model-free approaches. In *Conference on learning theory*, pp. 2898–2933. PMLR, 2019.

Uehara, M., Zhang, X., and Sun, W. Representation learning for online and offline rl in low-rank mdps. *arXiv preprint arXiv:2110.04652*, 2021.

Verhaegen, M. Subspace model identification part 3. analysis of the ordinary output-error state-space model identification algorithm. *International Journal of control*, 58(3): 555–586, 1993.

Weisz, G., Amortila, P., and Szepesvári, C. Exponential lower bounds for planning in mdps with linearly-realizable optimal action-value functions. In *Algorithmic Learning Theory*, pp. 1237–1264. PMLR, 2021.

Zhu, P., Li, X., Poupart, P., and Miao, G. On improving deep reinforcement learning for pomdps. *arXiv preprint arXiv:1704.07978*, 2017.

A. Proof for Section 3

In this section we provide formal proofs for the results stated in Section 3.

Proof of Proposition 3.1. We need to verify that $\mathcal{M}^m$ is an MDP. To do so, we check that the state space induces a Markovian dynamics and that the expected reward is also a function of the state. These two properties follow from the m-step decodability assumption.

- *Reward depends on the states.* This holds since the reward is assumed to depend only on the current observation o_h and the current observation is included in the megastate. Formally, for any $s^{m,h} = \left(o_h, o_{h-1}, a_{h-1}, \ldots, o_{\min(h-m,1)}, a_{\min(h-m,1)}\right) \in \mathcal{S}^{m,h}$, any history $\mathcal{H}$, and policy π, it holds that

$$\mathbb{E}_\pi\left[r|s^{m,h}, \mathcal{H}\right] = \mathbb{E}_\pi\left[r|s^{m,h}\right] = r(o_h)$$

  where $o_h \in s^{m,h}$ due to the assumption on the reward generation process of m-step decodable POMDP.

- *Transition model is Markov.* For any $s^{m,h} = \left(o_h^h, o_{h-1}^h, a_{h-1}^h, \ldots, o_{\min(h-m,1)}^h, a_{\min(h-m,1)}^h\right) \in \mathcal{S}^{m,h}$ and $s_{h+1}^n = \left(o_{h+1}^{h+1}, o_h^{h+1}, a_h^{h+1}, \ldots, o_{\min(h+1-m,1)}^{h+1}, a_{\min(h+1-m,1)}^{h+1+1}\right) \in \mathcal{S}_{h+1}^m$, any action $a_h^h \in \mathcal{A}$, any history $\mathcal{H}$ and any policy π it holds that

$$\mathbb{P}_\pi\left(s^{m,h+1} \mid s^{m,h}, a_h, \mathcal{H}\right) = \mathbb{P}_\pi\left(o_{h+1} \mid s^{m,h}, a_h, \mathcal{H}\right) \cdot \prod_{j=\min(h+1-m,1)}^{h} \delta\left(o_j^{h+1} = o_j^h, a_j^{h+1} = a_j^h\right)$$

  Finally, observe that by the m-step decodability assumption it holds that

$$\mathbb{P}_\pi\left(o_{h+1} \mid \phi^\star(s^{m,h}) = s, a_h, \mathcal{H}\right) = \mathbb{O}_{h+1}\left(o_{h+1} \mid s_{h+1}\right) \mathbb{P}\left(s_{h+1} \mid \phi^\star(s^{m,h}) = s, a_h\right),$$

  where the last relation holds by the Markov assumption of the latent model. This shows that

$$\mathbb{P}_\pi\left(s^{m,h+1} \mid s^{m,h}, a_h, \mathcal{H}\right) = \mathbb{P}\left(s^{m,h+1} \mid s^{m,h}, a_h\right),$$

  and hence the dynamics are Markovian.

Lastly, we elaborate on the optimality of any optimal policy of $\mathcal{M}^m$; that is, any optimal policy of $\mathcal{M}^m$ is an optimal policy of the m-step decodable POMDP. First, observe that the optimal policy of the latent MDP that underlies the m-step decodable POMDP is also the optimal policy of the m-step decodable POMDP.

Further, since the latent state is decodable from a suffix of length m of the history, any state in $\mathcal{S}^{m,h}$ (that represents a reachable suffix) can decode the latent state. Hence, the optimal policy on the latent MDP can be executed based on the states in $\mathcal{S}^{m,h}$. Thus, an optimal policy of $\mathcal{M}^m$ is also an optimal policy of the m-step decodable POMDP; otherwise, an optimal policy of the latent MDP is not optimal for the m-step decodable POMDP.

□

Proof of Corollary 3.2. The sample complexity follows immediately from a standard online-to-batch conversion of the minimax optimal regret bound in (Azar et al., 2017), combined with Proposition 3.1. In particular, the online-to-batch conversion gives $\tilde{O}(HSA\log^2(1/\delta)/\epsilon^2)$ sample complexity in an MDP with S states and A actions. By Proposition 3.1 we have an MDP with $O^m A^{m-1}$ states, so the result follows. □

Proof sketch of Proposition 3.3. We construct a simple m-step decodable POMDP with horizon m, two states per layer and two actions. The construction and argument are identical to the one in (Krishnamurthy et al., 2016), so we only sketch the construction here. It is a standard "combination lock" construction, with A actions and no observations, but where the state is decodable from the past actions.

In particular, the agent starts in the "good state" g_1 and at each time step h can be either in the good state g_h or the "bad state" b_h. From the good state, a special action $a_h^\star$ transits to the next good state, while all other actions (from both good or

bad state) transit to the next bad state b_{h+1}. At the last time step the agent gets reward for being in state g_m. There are no observations (or there is a trivial observation), but note that the latent state is decodable using the history of actions. Thus provided the horizon $H \leq m$ the process is m-step decodable.

Intuitively, the construction requires the agent to try all A^m action sequences before finding the reward. More formally this construction embeds an $\Omega(A^m)$ armed bandit problem resulting in a sample complexity lower bound of $\Omega(A^m/\epsilon^2)$. We refer the reader to (Krishnamurthy et al., 2016) for more details.

□

B. Proof for Section 4 and 5

In this section we provide formal proofs for the results stated in Section 4 and 5.

B.1. Properties of Moment Matching Policy

We start with formal definition of *moment matching policy*. For a policy π, we construct $\nu^{\pi,h}_{h'}$ for $h' \geq h - m$ such that it matches the distribution of the action $a_{h'}$ conditioning on latent states and observations from time step $h - m + 1$ to time step h under the sampling process of π. For this reason we refer to ν^{π} as the *moment matching policy* for π (see Figure 2 for illustration). Formally, we define it as follows:

Definition B.1 (Moment-Matching Policy for π)**.** Denote $m(h) = \max\{h - m + 1, 1\}$; Fix $h \in [H]$ and for $h' \in [m(h), h]$ we define

$$x_{h'} = \left(s_{m(h):h'}, o_{m(h):h'}, a_{m(h):h'-1}\right) \in \mathcal{X}_l,$$

where $\mathcal{X}_l = \mathcal{S}^l \times \mathcal{O}^l \times \mathcal{A}^{l-1}$ and $l = h' - m(h) + 1$. For a m-step policy π and $h \in [H]$, we define the moment matching policy $\mu^{\pi,h} = \{\mu^{\pi,h}_{h'} : \mathcal{X}_l \to \Delta(\mathcal{A})\}^{h}_{h'=m(h)}$ as following:

$$\mu^{\pi,h}_{h'}(a_{h'} \mid x_{h'}) := \mathbb{E}_{\pi}[\pi_{h'}(a_{h'} \mid z_{h'}) \mid x_{h'}].$$

By Assumption 2.2, states and therefore x'_h is decodable by the history of actions and observations, therefore we let

$$\nu^{\pi,h}_{h'}(a_{h'} \mid o_{1:h'}, a_{1:h'-1}) := \mu^{\pi,h}_{h'}(a_{h'} \mid x_{h'}).$$

As we discussed in Section 5, we prove the following lemma that establishes two important properties of the moment matching policy.

Lemma B.2. *For a fixed $h \in [H]$ and fixed m-step policies $\pi, \bar{\pi}$, define policy $\tilde{\pi}^h$ which takes first $m(h) - 1$ actions from π and remaining actions from $\nu^{\bar{\pi},h}$, i.e. $\tilde{\pi}^h = \pi \circ_{m(h)} \nu^{\bar{\pi},h}$. Then we have,*

1. *If $\pi = \bar{\pi}$, for any $z_h \in \mathcal{Z}_h$, $P_{\pi}(z_h) = P_{\tilde{\pi}^h}(z_h)$*
2. *For any function $g : \mathcal{Z}_h \to [0, 1]$,*

$$\mathbb{E}_{\tilde{\pi}^h}[g(z_h)] = \langle \zeta_h(\pi), \xi_h(g, \bar{\pi}) \rangle,$$

where $\zeta_h(\pi), \xi_h(g, \bar{\pi}) \in \mathbb{R}^S$ satisfying $||\zeta_h(\pi)|| \leq 1$ and $||\xi_h(g, \bar{\pi})|| \leq \sqrt{S}$.

Recall that we use the notation $m(h) = \max\{h - m + 1, 1\}$ and that we define $x_{h'} = (s_{m(h):h'}, o_{m(h):h'}, a_{m(h):h'-1})$ for $h' \in [m(h), h]$. By definition of $\mu^{\pi,h}$ (as in Definition B.1), for $h' \in [m(h), h]$ we have

$$\mu^{\pi,h}_{h'}(a_{h'} \mid x_{h'}) P_{\pi}\left[x_{h'}\right] = \sum_{(o,a)_{m(h'):m(h)-1}} \pi(a_{h'} \mid z_{h'}) P_{\pi}\left[(o, a)_{m(h'):m(h)-1}, x_{h'}\right] \tag{1}$$

We will use this identity below.

Proof of Lemma B.2. Recall that we define $\tilde{\pi}^h$ to take actions $a_{1:m(h)-1}$ according to π and take actions $a_{m(h):h-1}$ according to the moment matching policy $\nu^{\pi,h}$.

Item 1. We prove the first item by induction on $h' \in \{m(h), \dots, h\}$, where the induction hypothesis is

$$\forall x_{h'} : \quad P_\pi\big[x_{h'}\big] = P_{\tilde{\pi}^h}\big[x_{h'}\big]$$

- **Base case:** The base case is when $h' = m(h)$. In this case, $P_\pi\big[(s,o)_{m(h)}\big] = P_{\tilde{\pi}^h}\big[(s,o)_{m(h)}\big]$ since all actions up to $a_{m(h)-1}$ are taken by the same policy.
- **Induction step**: Let $h' \in \{m(h), \dots, h\}$ and assume $P_\pi\big[x_{h'-1}\big] = P_{\tilde{\pi}^h}\big[x_{h'-1}\big]$. We have

$$\begin{aligned}
P_\pi(x_{h'+1}) &= P_\pi\big[(s,o,a)_{m(h):h'}, (s,o)_{h'+1}\big] \\
&= \sum_{(o,a)_{m(h'):m(h)-1}} P_\pi\big[(o,a)_{m(h'):m(h)-1}, x_{h'}, a_{h'}, (s,o)_{h'+1}\big] \\
&= \sum_{(o,a)_{m(h'):m(h)-1}} \mathbb{O}(o_{h'+1} \mid s_{h'+1})\mathbb{P}(s_{h'+1} \mid s_{h'}, a_{h'})\pi(a_{h'} \mid z_{h'}) P_\pi\big[(o,a)_{m(h'):m(h)-1}, x_{h'}\big]
\end{aligned}$$

  Similarly we have,

$$\begin{aligned}
P_{\tilde{\pi}^h}(x_{h'+1}) &= P_{\tilde{\pi}^h}\big[(s,o,a)_{m(h):h'}, (s,o)_{h'+1}\big] \\
&= P_{\tilde{\pi}^h}\big[x_{h'}, a_{h'}, (s,o)_{h'+1}\big] \\
&= \mathbb{O}(o_{h'+1} \mid s_{h'+1})\mathbb{P}(s_{h'+1} \mid s_{h'}, a_{h'})\mu_{h'}^{\pi,h}(a_{h'} \mid x_{h'}) P_{\tilde{\pi}^h}\big[x_{h'}\big] \\
&\overset{(i)}{=} \mathbb{O}(o_{h'+1} \mid s_{h'+1})\mathbb{P}(s_{h'+1} \mid s_{h'}, a_{h'})\mu_{h'}^{\pi,h}(a_{h'} \mid x_{h'}) P_\pi\big[x_{h'}\big],
\end{aligned}$$

  where (i) uses the induction hypothesis. Equation (1) implies that right-hand side of the two above expressions are equal, which completes the proof of induction step.

Now item 1 is immediate since the variables in z_h are contained within x_h, in particular

$$P_\pi(z_h) = \sum_{s_{m(h):h}} P_\pi(x_h) = \sum_{s_{m(h):h}} P_{\tilde{\pi}^h}(x_h) = P_{\tilde{\pi}^h}(z_h).$$

Item 2. Recall that here $\tilde{\pi}^h$ is defined to take actions $a_{1:m(h)-1} \sim \pi$ and $a_{m(h):h-1} \sim \nu^{\bar{\pi},h}$ where π and $\bar{\pi}$ may not be equal. Since $\mu^{\bar{\pi},h}$ is defined to be independent of the past given $s_{m(h)}$ we have the factorization

$$\mathbb{E}_{\tilde{\pi}^h}[g(z_h)] = \sum_{s_{m(h)} \in \mathcal{S}} P_\pi(s_{m(h)}) \cdot \mathbb{E}_{a_{m(h):h-1} \sim \mu^{\bar{\pi},h}}[g(z_h) \mid s_{m(h)}].$$

We note that $\mu^{\bar{\pi},h}$ only depends on $(s,o)_{m(h):h-1}$ and $a_{m(h):h-2}$, thus the second term is independent of π and only depends g and $\bar{\pi}$. Defining

$$\zeta_h(\pi) := \big(P_\pi(s_{m(h)})\big)_{s_{m(h)} \in \mathcal{S}} \in \mathbb{R}^S \quad \text{and} \quad \xi_h(g, \bar{\pi}) = \big(\mathbb{E}_{a_{m(h):h-1} \sim \mu^{\bar{\pi},h}}[g(z_h) \mid s_{m(h)}]\big)_{s_{m(h)} \in \mathcal{S}} \in \mathbb{R}^S,$$

completes the proof. □

B.2. Concentration lemmas

We start with the following lemma, which is quite similar to Lemmas 39 and 40 in Jin et al. 2021. The lemma shows that: (1) with high probability any function in the confidence set at the k^{th} iteration has low Bellman error over the data distributions from visited in the previous iterations at all layers $h \in [H]$ and (2) the optimal value function is inside the confidence set with high probability.

Lemma B.3. *For any $\rho > 0$ and $\delta \in (0,1)$, if we run Algorithm 1 with $\beta = c\Big(\log\big[KH\mathcal{N}_{\mathcal{G}}(\rho)/\delta\big] + K\rho\Big)$ where $c > 0$ is an absolute constant, then with probability at least $1-\delta$, we have*

1. $\sum_{i=1}^{k-1} \mathbb{E}\Big[\big(f_h^k(z_h,a_h) - (\mathcal{T}_h f_h^k)(z_h,a_h)\big)^2 \mid a_{1:h-m} \sim \pi^i, a_{h-m+1:h} \sim \mathrm{unif}(\mathcal{A})\Big] \le \mathcal{O}(\beta)$ *for all* $(k,h) \in [K] \times [H]$,
2. $Q^\star \in \mathcal{B}^k$ *for all* $k \in [K]$.

Proof of Lemma B.3. The proof relies on a standard martingale concentration inequality (e.g., Freedman's inequality), the construction of our confidence set, and our generalized completeness assumption (Assumption 2.4). The argument is almost identical to the proofs of Lemma 39 and 40 in Jin et al. 2021 and therefore omitted for brevity. □

Lemma B.4. *For any* $\delta \in (0,1)$*, if we choose* $K_{\mathrm{est}} = c \cdot \big(\log[\mathcal{N}_{\mathcal{F}}(\rho_{\mathrm{est}})/\delta]/\rho_{\mathrm{est}}^2\big)$ *where* $c > 0$ *is some absolute constant; then, with probability at least* $1-\delta$ *for any* $f \in \mathcal{F}$*, we have*

$$|\hat{f}_1 - \mathbb{E}_{s_1}\big[f_1(o_1, \pi_f(o_1))\big]| \le \mathcal{O}(\rho_{\mathrm{est}}).$$

Proof. The proof follows from applying uniform concentration argument over a ρ_{est}-cover of $\mathcal{F}$; then, a covering argument finishes the proof. □

B.3. Eluder Dimension

In this section we describe complexity measure *Eluder dimension* proposed by Russo & Van Roy (2013) since it has been used in the analysis of the original GOLF algorithm (Jin et al., 2021).

Definition B.5 (ϵ-Independence)**.** Let $\mathcal{W}$ be a function class defined over domain $\mathcal{Y}$ and $y^1, \dots, y^n, \bar{y}$ be elements in $\mathcal{Y}$. We say $\bar{y}$ is ϵ-independent with respect to $\mathcal{W}$, if there exists $w \in \mathcal{W}$ such that $\sqrt{\sum_{i=1}^n [w(y^i)]^2} \le \epsilon$, but $|w(\bar{y})| > \epsilon$.

Definition B.6 (Eluder Dimension)**.** The Eluder dimension $\dim_{\mathrm{E}}(\mathcal{W}, \epsilon)$, is the length of the longest sequence of $\{y^1, \dots, y^n\}$ in $\mathcal{Y}$, such that there exists $\epsilon' \ge \epsilon$ where y^i is ϵ'-independent of $\{y^i, \dots, y^{i-1}\}$ with respect to $\mathcal{W}$ for all $i \in [n]$.

The following proposition shows that if $\mathcal{W}$ has a low rank structure with rank d, then the Eluder dimension can be upper bounded by $\widetilde{\mathcal{O}}(d)$.

Proposition B.7 (Proposition 6 in Russo & Van Roy 2013)**.** *Suppose for any* $w \in \mathcal{W}$ *and any* $y \in \mathcal{Y}$*, we have* $w(y) = \langle \zeta(y), \xi(w) \rangle$*, where* $\zeta(y), \xi(w) \in \mathbb{R}^d$ *satisfying* $||\zeta(y)|| \cdot ||\xi(w)|| \le \gamma$*. Then we have,*

$$\dim_{\mathrm{E}}(\mathcal{W}, \epsilon) \le \mathcal{O}\big(1 + d \log[1 + \gamma/\epsilon^2]\big).$$

The following lemma could be seen as an analogue to the standard elliptical potential argument for Eluder dimension that was proposed by Russo & Van Roy (2013) and been used in analysis of GOLF. The following lemma could be obtained from Lemma 41 in Jin et al. (2021) by setting the family of probability measures used in that lemma to be $\{\delta_y \mid y \in \mathcal{Y}\}$, where δ_y is the dirac measure centered at y.

Lemma B.8 (Simplification of Lemma 41 in Jin et al. 2021)**.** *Given a function class* $\mathcal{W}$ *defined over* $\mathcal{Y}$ *with* $w(y) \le C$ *for all* $(w,y) \in \mathcal{W} \times \mathcal{Y}$*; Suppose* $\{y^i\}_{i=1}^K \subseteq \mathcal{Y}$ *and* $\{w^i\}_{i=1}^K \subseteq \mathcal{W}$ *satisfy that for all* $k \in K$*,* $\sum_{i=1}^{k-1} [w^k(y^i)]^2 \le \alpha$*. Then for all* $k \in [K]$ *and* $\omega > 0$*, we have*

$$\sum_{i=1}^{k} |w^i(y^i)| \le \mathcal{O}\Big(\sqrt{\dim_{\mathrm{E}}(\mathcal{W}, \omega)\alpha k} + \min\{k, \dim_{\mathrm{E}}(\mathcal{W}, \omega)\} \cdot C + k\omega\Big).$$

B.4. Proof of Theorem 4.1

We use $\mathcal{E}_h(\pi, f)$ to denote the Bellman error of function $f \in \mathcal{F}$ at step h using roll-in policy π, which is defined as

$$\mathcal{E}_h(\pi, f) = \mathbb{E}[(f_h - \mathcal{T}_h f_{h+1})(z_h, \pi_f(z_h)) \mid a_{1:h-1} \sim \pi].$$

In addition, we use $\mathcal{E}_h^\star(\pi, f)$ to denote the Bellman error of function f at step h using roll-in policy π for the first $h-m$ steps and $\nu^{\pi_f, h}$ (the moment matching policy for π_f) for $a_{m(h):h-1}$; namely,

$$\mathcal{E}_h^\star(\pi, f) = \mathbb{E}[(f_h - \mathcal{T}_h f_{h+1})(z_h, \pi_f(z_h)) \mid a_{1:h-m} \sim \pi, a_{m(h):h-1} \sim \nu^{\pi_f, h}].$$

The next lemma shows that $\mathcal{E}_h^\star$ satisfies two important properties that are critical to the rest of the proof. The first property is that when $\pi = \pi_f$, $\mathcal{E}_h$ and $\mathcal{E}^\star$ coincide. The second property shows that $\mathcal{E}_h^\star$ has low rank or bilinear structure.

Lemma B.9. *For any policy π, any function $f \in \mathcal{F}$, and any $h \in [H]$, we have*

1. $\mathcal{E}_h(\pi_f, f) = \mathcal{E}_h^\star(\pi_f, f)$
2. $\mathcal{E}_h^\star(\pi, f) = \langle \zeta_h(\pi), \xi_h(f) \rangle$ *where* $\zeta_h(\pi), \xi_h(f) \in \mathbb{R}^S$ *satisfy* $||\zeta_h(\pi)|| \le 1$ *and* $||\xi_h(f)|| \le 2\sqrt{S}$.

Proof of Lemma B.9. For item (1) define $\tilde{\pi}_f^h$ to be the policy that takes actions $a_{1:h-m} \sim \pi_f$ and $a_{m(h):h-1} \sim \nu^{\pi_f, h}$, and let $g : \mathcal{Z}_h \to [0, 2]$ be defined as $g(z_h) = (f_h - \mathcal{T}_h f_{h+1})(z_h, \pi_f(z_h))$. Then by item (1) of Lemma B.2 we have

$$\begin{aligned}
\mathcal{E}_h^\star(\pi, f) &= \mathbb{E}[(f_h - \mathcal{T}_h f_{h+1})(z_h, \pi_f(z_h))) \mid a_{1:h-m} \sim \pi, a_{m(h):h} \sim \nu^{\pi_f, h}] \\
&= \sum_{z_h \in \mathcal{Z}_h} P_{\tilde{\pi}_f^h}(z_h) \cdot g(z_h) = \sum_{z_h \in \mathcal{Z}_h} P_{\pi_f}(z_h) \cdot g(z_h) \\
&= \mathbb{E}[(f_h - \mathcal{T}_h f_{h+1})(z_h, \pi_f(z_h))) \mid a_{1:h-1} \sim \pi] = \mathcal{E}_h(\pi_f, f),
\end{aligned}$$

Item (2) immediately follows from item (2) of Lemma B.2 by selecting g as $g(z_h) = (f_h - \mathcal{T}_h f_{h+1})(z_h, \pi_f(z_h))$ and $\bar{\pi} = \pi_g$. □

The following corollary shows that Eluder dimension with respect to $\mathcal{E}^\star$ is upper bounded by $\tilde{\mathcal{O}}(S)$. The proof immediately follows from Lemma B.9 and Proposition B.7.

Corollary B.10. *Let Π to be set of all m-step policies, and define $\mathcal{W}_{\mathcal{F}}^\star = \{\mathcal{E}^\star(\cdot, f) : \Pi \to [0, 2] \mid f \in \mathcal{F}\}$, then*

$$\dim_{\mathrm{E}}(\mathcal{W}_{\mathcal{F}}^\star, e) \le \mathcal{O}\big(S \log[S/\epsilon]\big).$$

Now we are ready to prove Theorem 4.3.

Proof of Theorem 4.1. With probability at least $1 - 2\delta$ the events in Lemma B.3 and Lemma B.4 holds. Under this good event, we proceed in several steps.

Step 1. Bounding the optimality gap by the Bellman error. Lemma B.3 guarantees that $\forall k \in [K] : \ Q^\star \in \mathcal{B}^k$, this together with optimistic choice of f^k (Line 4 in Algorithm 1), for all $k \in [K]$, we have:

$$V^\star \le \hat{Q}_1^\star + \mathcal{O}(\rho_{\mathrm{est}}) \le \hat{f}_1^k + \mathcal{O}(\rho_{\mathrm{est}}) \le \mathbb{E}_{s_1}\big[f_1^k(o_1, \pi_{f^k}(o_1))\big] + 2 \cdot \mathcal{O}(\rho_{\mathrm{est}}).$$

It implies that $\sum_{k=1}^K \big(V^\star - V^{\pi^k}\big) \le \sum_{k=1}^K \mathbb{E}_{s_1}\big[f_1^k(o_1, \pi_{f^k}(o_1))\big] - V^{\pi^k} + \mathcal{O}(K\rho_{\mathrm{est}})$. We also have

$$\mathbb{E}_{s_1}\big[f_1^k(o_1, \pi_{f^k}(o_1))\big] - V^{\pi^k} \overset{(i)}{=} \sum_{k=1}^K \sum_{h=1}^H \mathcal{E}_h(\pi^k, f^k) \overset{(ii)}{=} \sum_{h=1}^H \sum_{k=1}^K \mathcal{E}_h^\star(\pi^k, f^k),$$

where (i) is by standard policy loss decomposition (e.g., Lemma 1 in Jiang et al. 2017) and (ii) is due to part (1) of Lemma B.9 since we have $\pi^k = \pi_{f^k}$. Therefore, we showed

$$\sum_{k=1}^K \big(V^\star - V^{\pi^k}\big) \le \sum_{h=1}^H \sum_{k=1}^K \mathcal{E}_h^\star(\pi^k, f^k) + \mathcal{O}(K\rho_{\mathrm{est}})$$

Step 2: Utilizing the confidence set. By Lemma B.3, we have

$$\sum_{i=1}^{k-1} \mathbb{E}\Big[\big((f_h^k - \mathcal{T}_h f_{h+1}^k)(z_h, a_h)\big)^2 \mid a_{1:h-m} \sim \pi^i, a_{h-m+1:h} \sim \mathrm{unif}(\mathcal{A})\Big] \le \mathcal{O}(\beta) \quad \forall (k, h) \in [K] \times [H].$$

It implies that

$$\begin{aligned}\sum_{i=1}^{k-1}[\mathcal{E}_h^\star(\pi^i, f^k)]^2 &\le \sum_{i=1}^{k-1}\mathbb{E}\Big[\big((f_h^k - \mathcal{T}_h f_{h+1}^k)(z_h, \pi_f(z_h))\big)^2 \mid a_{1:h-m} \sim \pi^i, a_{h-m+1:h} \sim \nu^{\pi_{f^k},h}\Big] \\ &\le A^m \sum_{i=1}^{k-1}\mathbb{E}\Big[\big((f_h^k - \mathcal{T}_h f_{h+1}^k)(z_h, a_h)\big)^2 \mid a_{1:h-m} \sim \pi^i, a_{h-m+1:h} \sim \mathrm{unif}(\mathcal{A})\Big] \\ &\le \mathcal{O}(A^m\beta).\end{aligned}$$

Here the A^m factor arises to change measure from $\nu^{\pi_{f^k},h}$ to the uniform distribution over actions $a_{h-m+1:h}$.

Step 3: Utilizing Low-rank Structure. From previous step, we know that $\sum_{i=1}^{k-1}[\mathcal{E}_h^\star(\pi^i, f^k)]^2 \le A^m\beta$, Therefore if we invoke Lemma B.8 and Corollary B.10 with

$$\begin{cases}\mathcal{Y} = \Pi, & \mathcal{W} = \mathcal{W}_{\mathcal{F}}^\star = \{\mathcal{E}^\star(\cdot, f) : \Pi \to [0,2] \mid f \in \mathcal{F}\}, \\ \omega = \epsilon/H, & \alpha = \mathcal{O}(A^m\beta), \quad C = 2,\end{cases}$$

we obtain

$$\frac{1}{K}\sum_{k=1}^{K}\mathcal{E}_h^\star(\pi^k, f^k) \le \mathcal{O}\big(\sqrt{\frac{A^m S\log[S/\epsilon]\beta}{K}} + \epsilon/H\big)$$

Step 4: Putting everything together Choosing $\rho_{\mathrm{est}} = \mathcal{O}(\epsilon)$ and combining the conclusion of step 1 and step 3, we have

$$\frac{1}{K}\sum_{k=1}^{K}\big(V^\star - V^{\pi^k}\big) \le \frac{1}{K}\sum_{h=1}^{H}\sum_{k=1}^{K}\mathcal{E}_h^\star(\pi^k, f^k) \le \mathcal{O}\big(\sqrt{\frac{H^2A^mS\log[S/\epsilon]\beta}{K}} + \epsilon\big) + \mathcal{O}(\epsilon).$$

By definition of π^{out}, we have

$$\begin{aligned}V^\star - V^{\pi^{\mathrm{out}}} &= \frac{1}{K}\sum_{k=1}^{K}\big(V^\star - V^{\pi^k}\big) \le \mathcal{O}\big(\sqrt{\frac{H^2A^mS\log[S/\epsilon]\beta}{K}}\big) + \mathcal{O}(\epsilon) \\ &\overset{(i)}{\le} \mathcal{O}\Big(\sqrt{\frac{H^2A^mS\log[S/\epsilon]\log[KH\mathcal{N}_{\mathcal{G}}(\rho)/\delta]}{K}} + H^2A^mS\log[S/\epsilon]\rho\Big)\Big) + \mathcal{O}(\epsilon) \\ &\overset{(ii)}{\le} \mathcal{O}\Big(\sqrt{\frac{H^2A^mS\log[S/\epsilon]\log[KH\mathcal{N}_{\mathcal{G}}(\rho)/\delta]}{K}}\Big) + \mathcal{O}(\epsilon)\end{aligned}$$

,

where (i) is follows from $\beta = c\Big(\log\big[KH\mathcal{N}_{\mathcal{G}}(\rho)/\delta\big] + K\rho\Big)$ as in Lemma B.3 and (ii) is by picking

$$\rho = \frac{\epsilon^2}{(H^2A^mS\log[S/\epsilon])}.$$

We need to pick K such that

$$\sqrt{\frac{H^2A^mS\log[S/\epsilon]\log[KH\mathcal{N}_{\mathcal{G}}(\rho)/\delta]}{K}} \le \mathcal{O}(\epsilon).$$

By simple calculations, one can verify that it suffices to pick

$$K \ge \Omega(\frac{H^2SA^m}{\epsilon^2}\cdot\log[HSA^m\mathcal{N}_{\mathcal{G}}(\rho)/(\delta\epsilon)]\cdot\log[S/\epsilon]),$$

which completes the proof. □

Algorithm 2 IS-RL: Importance sampling for Reinforcement Learning

1: **Initialize:** N number of samples, policy class Π,
2: **Collect:** N trajectories $\{o_h^{(t)}, a_h^{(t)}, r_h^{(t)}\}_{h=1}^H$ for $t \in [N]$ by executing the uniform policy $a_h^{(t)} \sim \mathrm{Uniform}(\mathcal{A})$.
3: For any $\pi \in \Pi$ calculate its empirical value

$$\widehat{V}^\pi = \frac{1}{N}\sum_{t=1}^{N}\prod_{h=1}^{H}\left(\frac{\pi(a_h^{(t)} \mid z_h^{(t)})}{1/A}\right)\cdot\left(\sum_{h=1}^{H} r_h^{(t)}\right)$$

4: **Output** $\widehat{\pi} \in \arg\max_{\pi\in\Pi} \widehat{V}^\pi$.

B.5. Proof for Theorem 4.3

The following lemma (akin to part (2) of Lemma B.9) shows that $\mathcal{E}^*$ has low rank structure with rank d_{lin}. The proof of Theorem 4.3 is almost identical to proof of Theorem 4.1 where the only difference is to use Lemma B.11 instead of part (2) of Lemma B.9 resulting in S being replaced by d_{lin} wherever it has been used.

Lemma B.11 (akin to part (2) of Lemma B.9)**.** *Under Definition 4.2; for any policy π and any function $f \in \mathcal{F}$, and any $h \in [H]$, we have $\mathcal{E}_h^\star(\pi, f) = \langle \zeta_h(\pi), \xi_h(f)\rangle$ where $\zeta_h(\pi), \xi_h(f) \in \mathbb{R}^{d_{lin}}$ satisfy $||\zeta_h(\pi)|| \le 1$ and $||\xi_h(f)|| \le 2\sqrt{d_{lin}}$.*

Proof of Lemma B.11. Let g be a function $g : \mathcal{Z}_h \to [0,1]$ and $\tilde{\pi}^h = \pi \circ_{m(h)} \bar{\pi}$. Recall that here $\tilde{\pi}^h$ is defined to take actions $a_{1:m(h)-1} \sim \pi$ and $a_{m(h):h-1} \sim \nu^{\bar{\pi},h}$ where π and $\bar{\pi}$ may not be equal. Since $\mu^{\bar{\pi},h}$ is defined to be independent of the past given $s_{m(h)}$ we have the factorization

$$\begin{aligned}\mathbb{E}_{\tilde{\pi}^h}[g(z_h)] &= \mathbb{E}_\pi\Big[\int_{s_{m(h)}\in\mathcal{S}} \langle \psi_\pi(s_{m(h)-1}, a_{m(h)-1}), \boldsymbol{\mu}(s_{m(h)})\cdot \mathbb{E}_{a_{m(h):h-1}\sim\mu^{\bar{\pi},h}}[g(z_h) \mid s_{m(h)}]\Big] \\ &= \Big\langle \mathbb{E}_\pi \psi_\pi(s_{m(h)-1}, a_{m(h)-1}), \int_{s_{m(h)}\in\mathcal{S}} \boldsymbol{\mu}(s_{m(h)})\cdot \mathbb{E}_{a_{m(h):h-1}\sim\mu^{\bar{\pi},h}}[g(z_h)\mid s_{m(h)}]\Big\rangle\end{aligned}$$

We note that $\mu^{\bar{\pi},h}$ only depends on $(s,o)_{m(h):h-1}$ and $a_{m(h):h-2}$, thus the second term is independent of π and only depends g and $\bar{\pi}$. Define

$$\begin{cases} \zeta_h(\pi) := \mathbb{E}_\pi \psi_\pi(s_{m(h)-1}, a_{m(h)-1}) \in \mathbb{R}^{d_{\text{lin}}} \\ \xi_h(g,\bar{\pi}) = \int_{s_{m(h)}\in\mathcal{S}} \boldsymbol{\mu}(s_{m(h)})\cdot \mathbb{E}_{a_{m(h):h-1}\sim\mu^{\bar{\pi},h}}[g(z_h)\mid s_{m(h)}] \in \mathbb{R}^{d_{\text{lin}}}. \end{cases}$$

Picking g as $g(z_h) = (f_h - \mathcal{T}_h f_{h+1})(z_h, \pi_f(z_h))$ and $\bar{\pi} = \pi_g$ completes the proof. □

C. On H-Step Decodable POMDPs

In this section, we show that there exists an algorithm that returns an ϵ optimal policy for any H-step decodable POMDP with sample complexity which is only polynomial in $|\mathcal{O}|$, the cardinality of the observation space. To do so, we construct a policy class Π that contains the optimal policy and has cardinality bounded by $|\Pi| \le O\left(H(SA)^{2HSOA}\right)$ and we use this policy class in a standard importance-sampling procedure. The procedure is formally specified Algorithm 2, and Proposition 6.1 follows immediately from Corollary C.2 and Lemma C.3.

Constructing the policy class Π via recurrent function class. Let $\mathcal{B}_h$ denote the set of all mappings of the form $b_h : \mathcal{S}_{h-1} \times \mathcal{A}_{h-1} \times \mathcal{O}_h \to \mathcal{S}_h$. This class represents all mappings from the latent state at the previous time step, action at the previous time step, and current observation to the latent state at the current time step. We call them *belief operators*.

We show that the latent state at time step h is decodable from the tuple (o_h, s_{h-1}, a_{h-1}). In other words, we can write $\phi^\star(z_h) = b_h^\star(\phi^\star(z_{h-1}), a_{h-1}, o_h)$ for some belief operator $b_h^\star \in \mathcal{B}_h$. This relation is established in the following lemma.

Lemma C.1. *For each $h \in [H]$ there exists $b_h^\star \in \mathcal{B}_h$ such that for all reachable histories z_h we have $\phi^\star(z_h) = b_h^\star(\phi^\star(z_{h-1}), a_{h-1}, o_h)$.*

Using the belief operator class we can design a policy class that contains the optimal policy for any H-step decodable POMDP. Given a decoder $\vec{b} := (b_1, \ldots, b_H) \in \mathcal{B}_1 \times \ldots \times \mathcal{B}_H$ and a trajectory z_H (or a partial trajectory z_h), the predicted state is

updated recursively as $\hat{s}_1 = b_1(o_1)$, $\hat{s}_h = b_h(\hat{s}_{h-1}, a_{h-1}, o_h)$. Then we can define $\Pi_{\vec{b}} := \{\pi : \pi(a_h \mid z_h) = \pi_h(a_h \mid \hat{s}_h)\}$, where here implicitly we are updated $\hat{s}_h$ using $\vec{b}$. Then we can take $\Pi = \bigcup_{\vec{b}\in\vec{\mathcal{B}}} \Pi_b$. For this class we have the following corollary.

Corollary C.2. *We have* $|\Pi| \leq (SA)^{2SHOA}$ *and for any* H*-step decodable POMDP* $\pi^\star \in \Pi$.

Importance Sampling Procedure for H-step POMDPs. Algorithm 2 describes a standard importance sampling approach for policy learning in POMDPs, which is essentially the same as the trajectory tree method of Kearns et al. (1999). A standard analysis of importance weighting using Bernstein's inequality and a uniform convergence argument yield the following lemma. As the result is quite standard, we omit the proof here.

Lemma C.3. *Fix any* $\epsilon, \delta > 0$ *and let* $N = \Omega\left(HA^H \log\left(|\Pi|/\delta\right)/\epsilon^2\right)$. *Then with probability at least* $1-\delta$, *Algorithm 2 returns a policy* $\widehat{\pi} \in \Pi$ *such that*

$$\max_{\pi\in\Pi} V^\pi \leq V^{\widehat{\pi}} + \epsilon.$$

C.1. Proofs

We now turn to the proofs of Lemma C.1 and Corollary C.2.

Proof of Lemma C.1. By the decodability assumption, for any $z_h = (o_{1:h}, a_{1:h-1})$ such that $\sup_\pi \mathbb{P}^\pi[z_h] > 0$, it holds that

$$\mathbb{P}(s_h \mid z_h) = \delta\left(\phi^\star(z_h)\right).$$

On the other hand, it holds that

$$\mathbb{P}(s_h \mid z_h) = \frac{\sum_{s_{h-1}} \mathbb{P}(s_h, o_h, s_{h-1} \mid o_{h-1:1}, a_{h-1:1})}{\sum_{s_{h-1}} \mathbb{P}(o_h, s_{h-1} \mid o_{h-1:1}, a_{h-1:1})}. \tag{2}$$

By the POMDP model assumption and decodability the numerator is also given by,

$$\mathbb{P}(s_h, o_h, s_{h-1} \mid o_{h-1:1}, a_{h-1:1}) = \mathbb{P}(s_h, o_h \mid s_{h-1}, a_{h-1})\delta(s_{h-1} = \phi^\star(z_{h-1})).$$

Similarly, the denominator is given by

$$\mathbb{P}(o_h, s_{h-1} \mid o_{h-1:1}, a_{h-1:1})\mathbb{P}(s_h \mid s_{h-1}, a_{h-1}) = \sum_{\bar{s}_h} \mathbb{P}(\bar{s}_h, o_h \mid s_{h-1}, a_{h-1})\delta(s_{h-1} = \phi^\star(z_{h-1})).$$

Plugging this back into equation (2) we obtain

$$\begin{aligned}
\mathbb{P}(s_h \mid z_h) &= \frac{\sum_{s_{h-1}} \mathbb{P}(s_h, o_h \mid s_{h-1}, a_{h-1})\delta(s_{h-1} = \phi^\star(z_{h-1}))}{\sum_{s_{h-1}}\sum_{\bar{s}_h} \mathbb{P}(\bar{s}_h, o_h \mid s_{h-1}, a_{h-1})\delta(s_{h-1} = \phi^\star(z_{h-1}))} \\
&= \frac{\mathbb{P}(s_h, o_h \mid \phi^\star(z_{h-1}), a_{h-1})}{\sum_{\bar{s}_h} \mathbb{P}(\bar{s}_h, o_h \mid \phi^\star(z_{h-1}), a_{h-1})} \\
&= \frac{\mathbb{P}(s_h \mid o_h, \phi^\star(z_{h-1}), a_{h-1})\mathbb{P}(o_h \mid \phi^\star(z_{h-1}), a_{h-1})}{\sum_{\bar{s}_h} \mathbb{P}(\bar{s}_h \mid \phi^\star(z_{h-1}), a_{h-1})\mathbb{P}(o_h \mid \phi^\star(z_{h-1}), a_{h-1})} \\
&= \frac{\mathbb{P}(s_h \mid o_h, \phi^\star(z_{h-1}), a_{h-1})}{\sum_{\bar{s}_h} \mathbb{P}(\bar{s}_h \mid \phi^\star(z_{h-1}), a_{h-1})} \\
&= \mathbb{P}(s_h \mid o_h, \phi^\star(z_{h-1}), a_{h-1}).
\end{aligned}$$

Recall that $\mathbb{P}(s_h \mid z_h) = \delta\left(s_h = \phi^\star(z_h)\right)$ by the decodability assumption. Hence, it holds that

$$\mathbb{P}(s_h \mid o_h, \phi^\star(z_{h-1}), a_{h-1}) = \delta(s_h = \phi^\star(z_h)).$$

Therefore for any reachable z_h, with $s_{h-1} = \phi^\star(z_{h-1})$ we take $b_h^\star(s_{h-1}, a_{h-1}, o_h)$ to be the unique s_h for which $\mathbb{P}(s_h \mid o_h, s_{h-1}, a_{h-1}) \neq 0$ and if this does not completely specify $b_h^\star$, we complete can complete it arbitrarily. □

Proof of Corollary C.2. The fact that $\pi^\star \in \Pi$ follows directly from Lemma C.1, since $\vec{b}^\star \in \mathcal{B}$ and for any H-step POMDP the optimal action depends only on the state. As for the size of Π observe that for each h we have $|\mathcal{B}_h| \leq S^{SOA}$ and so $|\vec{\mathcal{B}}| \leq S^{HSOA}$. Finally, for each $\vec{b} \in \vec{\mathcal{B}}$ we have $|\Pi_{\vec{b}}| = A^{SH}$. Taken together we have $|\Pi| \leq (SA)^{HSOA}$ as desired. □

D. Proof for Proposition 5.1

Here we construct an instance of a 2-step decodable POMDP in which the bellmank rank scales with the number of observations O. We further show that the OLIVE algorithm has sample complexity that scales polynomially with O, thus motivating our new algorithmic techniques. We believe a similer construction will also show that this model does not fall into either the bilinear class or Bellman-Eluder frameworks (Du et al., 2021; Jin et al., 2021).

The key idea is to use a construction inspired by the Hadamard matrix. Let $O = 2^s$ for some natural number s and $\mathcal{O} = \{1, \dots, O\}$. Then, there exist sets $S_1, \dots, S_{O-1} \subset \mathcal{O}$ such that:

$$\forall i : |S_i| = O/2, \quad \text{and} \quad \forall i \neq j : |S_i \cap S_j| = |S_i \cap \bar{S}_j| = O/4 \tag{3}$$

The existence of these can be verified by the existence and orthogonality of Hadamard matrices in dimension $O = 2^s$. Indeed, if we define $\{v_i\}_{i=0}^{O} \subset \{\pm 1\}^O$ such that $v_0 = \mathbf{1}$ and v_i is the ± 1 indicator vector for set S_i. Then the first property above is equivalent to $v_i^\top v_0 = 0$ for all $i \neq 0$ while the second property is equivalent to

$$\forall i \neq j \in \{1, \dots, O\} \sum_k \mathbf{1}\{v_i[k] = +1\} v_j[k] = 0$$

We claim that these two properties are satisfied if the vectors v are the columns of a Hadamard matrix. The first follows directly from orthogonality. For the second, since $v_i^\top v_j = 0$ and $v_j^\top v_0 = 0$ both by orthogonality, we have

$$v_i^\top v_j = 0 \Rightarrow \underbrace{\sum_k \mathbf{1}\{v_i[k] = +1\} v_j[k]}_{=:A_{ij}} - \underbrace{\sum_k \mathbf{1}\{v_i[k] = -1\} v_j[k]}_{=:B_{ij}} = 0$$

$$v_j^\top v_0 = 0 \Rightarrow \sum_k \mathbf{1}\{v_i[k] = +1\} v_j[k] + \sum_k \mathbf{1}\{v_i[k] = -1\} v_j[k] = 0.$$

Thus we have $A_{ij} + B_{ij} = A_{ij} - B_{ij} = 0$ which implies that $A_{ij} = 0$. So we have established the existence of $O - 1$ sets satisfying (3).

Let us now put this construction to use in a 2-step decodable POMDP. We consider a $H = 2$, three state POMDP with initial state s_0 and two states s_1, s_2 reachable at time $h = 2$. We have: $\mathbb{O}(\cdot \mid s_0) = \text{Unif}(\{1, \dots, O\})$ while $\mathbb{O}(\cdot \mid s_1) = \mathbb{O}(\cdot \mid s_2) = \delta(\{\perp\})$. In words, from the initial state we see an observation uniformly at random, while from s_1 or s_2 we see no observation. The dynamics are such that taking a_1 from s_0 reaches s_1 and taking a_2 from s_0 reaches s_2. Only a single action a_1 is available from s_1 or s_2 and it enjoys reward $R(s_1, a_1) = 1/2$, $R(s_2, a_1) = 3/4$. Clearly this POMDP is 2-step decodable since the first state is always decodable and the previous action uniquely determines the second state.

We have a function class $\mathcal{F}$ of 2-step candidate Q functions. The functions are $\mathcal{F} := \{Q^\star\} \cup \{f_i\}_{i=1}^{O-1}$ where each f_i is associated with a set S_i from the above Hadamard construction. These functions are defined as

$$f_i(oa_1) = \mathbf{1}\{o \in S_i\}, \quad f_i(oa_2) = 3/4, \quad f_i(oa_1 \perp a_1) = \mathbf{1}\{o \in S_i\}, \quad f_i(oa_2 \perp a_1) = 3/4$$

It is easy to very that these functions have zero bellman error at the first time step, that is

$$\forall (o, a) : f_i(oa) = f_i(oa \perp a_1)$$

On the other hand, f_i has very high bellman error at the second time step, since it never correctly predicts the reward for state s_1. In particular we have $\mathbb{E}_{d_2^{\pi_{f_i}}}[f_i(oa \perp a_1) - r] = 1/4$, since π_{f_i} visits states s_1 on half of the observations and every time it does it overpredicts the reward by $1/2$. However, observe that

$$\mathbb{E}_{d_2^{\pi_{f_i}}}[f_j(oa \perp a_1) - r] = \frac{1}{O} \sum_{o \in S_i} \mathbf{1}\{o \in S_j\}(1 - 1/2) + \mathbf{1}\{o \notin S_j\}(0 - 1/2) = 0,$$

where the last identity uses (3). Thus we see that we have embedded an $(O-1) \times (O-1)$-sized identity matrix inside of the Bellman error matrix at time 2, which shows that the Bellman rank is $\Omega(O)$.

Note that the OLIVE algorithm itself will also incur $\text{poly}(O)$ sample complexity in this instance. This is because the value predicted by f_i at the starting state, namely $\mathbb{E}[\max_a f(oa)]$, is $1/2 + 3/8$ which is greater than $V^\star = 3/4$. Thus OLIVE will enumerate over the f_i functions, eliminating one at a time and incurring a $\text{poly}(O)$ sample complexity.