Representation Learning with Multi-Step Inverse Kinematics: An Efficient and Optimal Approach to Rich-Observation RL

Zakaria Mhammedi [1] **Dylan J. Foster** [2] **Alexander Rakhlin** [1]

Abstract

We study the design of sample-efficient algorithms for reinforcement learning in the presence of rich, high-dimensional observations, formalized via the *Block MDP* problem. Existing algorithms suffer from either 1) computational intractability, 2) strong statistical assumptions that are not necessarily satisfied in practice, or 3) suboptimal sample complexity. We address these issues by providing the first computationally efficient algorithm that attains rate-optimal sample complexity with respect to the desired accuracy level, with minimal statistical assumptions. Our algorithm, `MusIK`, combines systematic exploration with representation learning based on *multi-step inverse kinematics*, a learning objective in which the aim is to predict the learner's own action from the current observation and observations in the (potentially distant) future. `MusIK` is simple and flexible, and can efficiently take advantage of general-purpose function approximation. Our analysis leverages several new techniques tailored to non-optimistic exploration algorithms, which we anticipate will find broader use.

1. Introduction

Many of the most promising application domains for reinforcement learning entail navigating unknown environments in the presence of complex, high-dimensional sensory inputs. For example, a challenging task in robotic control is to navigate to a goal state in a new, unmapped environment using only raw pixels from a camera as feedback (Baker et al., 2022; Bharadhwaj et al., 2022). Such tasks demand reinforcement learning agents capable of both 1) deliberate exploration, and 2) representation learning, as a means to learn from high-dimensional ("rich") observations. In this context, a major challenge—in theory and practice—is to develop algorithms that are practical and sample-efficient, yet require minimal prior knowledge.

We study the design of sample-efficient algorithms for rich-observation reinforcement learning through a canonical model known as the *Block MDP* (Jiang et al., 2017; Du et al., 2019a). The Block MDP is a setting in which the *observed* state space $\mathcal{X}$ is high-dimensional (e.g., pixels from a camera), but the dynamics are governed by a small, finite *latent* state space (e.g., a robot's actuator configuration). The key structural property of the Block MDP model, which makes the problem tractable statistically, is that the latent states can be uniquely *decoded* from observations (avoiding issues of partial observability). However, the mapping from observations to latent states is not known in advance, necessitating the use of representation learning in tandem with exploration. As such, the Block MDP is appealing as a stylized testbed in which to study design of sample-efficient algorithms based on representation learning.

Algorithm design for the Block MDP is particularly challenging because representation learning and exploration are not only required, but must be *interleaved*: learning a good representation is necessary to effectively control the agent and explore, but it is difficult to learn such a representation without exploring and gathering diverse feedback. In spite of extensive research into the design of algorithms with provable guarantees (Jiang et al., 2017; Du et al., 2019a; Misra et al., 2020; Zhang et al., 2022b; Uehara et al., 2022), all existing algorithms suffer from one or more of the following drawbacks:

1. Computational intractability.
2. Strong statistical assumptions that are not necessarily satisfied in practice.
3. Suboptimal sample complexity.

In more detail, computationally efficient algorithms can be split into two families. The first achieves rate-optimal sample complexity with respect to the desired accuracy level (Misra et al., 2020; Modi et al., 2021), but their guarantees scale inversely proportional to a *reachability* parameter

[1]Massachusetts Institute of Technology [2]Microsoft Research. Correspondence to: Zakaria Mhammedi <mhammedi@mit.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

which captures the minimum probability with which any state can be reached by a policy targeting it; when reachability is violated, these results give no guarantees. More recent approaches dispense with the reachability assumption (Zhang et al., 2022b), but do not attain rate-optimal sample complexity.

Our contributions. We address issues (1), (2), and (3) by providing the first computationally efficient algorithm that attains rate-optimal sample complexity[1] without reachability or other strong statistical assumptions (Table 1 in Appendix A). Our algorithm, `MusIK` ("`Mu`lti-step `s`ensing `I`nverse `K`inematics"), interleaves exploration with representation learning based on *multi-step inverse kinematics* (Lamb et al., 2022; Islam et al., 2022), a learning objective in which the aim is to predict the learner's own action from the current observation and observations in the (potentially distant) future. `MusIK` is simple and flexible: it can take advantage of general-purpose function approximation, and is computationally efficient whenever a standard supervised regression objective for the function class of interest can be solved efficiently. In a validation experiment, we find that it obtains comparable or superior performance to other provably efficient methods (Misra et al., 2020; Zhang et al., 2022b).

Organization. Section 2 introduces the Block MDP setting and the online reinforcement learning framework, as well as necessary notation. In Section 3, we present our main algorithm, `MusIK`, formally state its main guarantee for reward-free exploration, and discuss some of its implications. In Section 4, we give an overview of the main analysis ideas behind `MusIK`, with a more thorough overview deferred to Appendix E. Lastly, in Section 5 we present an experimental validation. All proofs are deferred to the appendix unless otherwise stated.

2. Problem Setting

We consider an episodic finite-horizon reinforcement learning framework, with $H \in \mathbb{N}$ denoting the horizon. A Block MDP $\mathcal{M} = (\mathcal{X}, \mathcal{S}, \mathcal{A}, T, q)$ consists of an *observation space* $\mathcal{X}$, *latent state space* $\mathcal{S}$, *action space* $\mathcal{A}$, *latent space transition kernel* $T : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$, and *emission distribution* $q : \mathcal{S} \to \Delta(\mathcal{X})$ (Du et al., 2019a). For each layer $h \in [H]$, the *latent state* $\boldsymbol{s}_h \in \mathcal{S}$ evolves in a Markovian fashion based on the agent's action $\boldsymbol{a}_h \in \mathcal{A}$ via $\boldsymbol{s}_{h+1} \sim T(\cdot \mid \boldsymbol{s}_h, \boldsymbol{a}_h)$, with $\boldsymbol{s}_1 \sim T(\cdot \mid \varnothing)$, where $T(\cdot \mid \varnothing)$ denotes the initial state distribution. The latent state is not observed directly. Instead, we observe *observations* $\boldsymbol{x}_h \in \mathcal{X}$ generated by the emission process

$$\boldsymbol{x}_h \sim q(\cdot \mid \boldsymbol{s}_h).$$

We assume that the latent space $\mathcal{S}$ and action space $\mathcal{A}$ are finite, with $S := |\mathcal{S}|$ and $A := |\mathcal{A}|$, but the observation space $\mathcal{X}$ may be large (with $|\mathcal{X}| \gg |\mathcal{S}|$) or potentially infinite. The most important property of the BMDP model, which facilitates sample-efficient learning, is *decodability*:

$$\operatorname{supp} q(\cdot \mid s) \cap \operatorname{supp} q(\cdot \mid s') = \varnothing, \quad \forall s' \neq s \in \mathcal{S}.$$

Decodability implies that latent states can be uniquely recovered from observations. In particular, there exists a (unknown to the agent) *decoder* $\phi_\star : \mathcal{X} \to \mathcal{S}$ such that $\phi_\star(\boldsymbol{x}_h) = \boldsymbol{s}_h$ a.s. for all $h \in [H]$.

To simplify presentation and keep notation compact, we assume that the BMDP $\mathcal{M}$ is *layered* in the sense that $\mathcal{S} = \mathcal{S}_1 \cup \cdots \cup \mathcal{S}_H$ for $\mathcal{S}_i \cap \mathcal{S}_j = \varnothing$ for all $i \neq j$, where $\mathcal{S}_h \subseteq \mathcal{S}$ is the subset of states in $\mathcal{S}$ that are reachable at layer $h \in [H]$. This comes with no loss of generality (up to dependence on H), as one can always augment the state space to include the layer index. We also define $\mathcal{X}_h := \bigcup_{s \in \mathcal{S}_h} \operatorname{supp} q(\cdot \mid s)$, for all $h \in [H]$, and note that by decodability, we have that $\mathcal{X}_i \cap \mathcal{X}_j = \varnothing$, for all $i \neq j$.

Online reinforcement learning and reward-free exploration. We consider the standard *online reinforcement learning* framework in which the underlying BMDP $\mathcal{M}$ is unknown, but the learning agent can interact with it by repeatedly executing a policy $\pi : \mathcal{X} \to \mathcal{S}$ (or, a potentially non-Markovian policy, as we will consider in the sequel) and observing the resulting trajectory $(\boldsymbol{x}_1, \boldsymbol{a}_1), \ldots, (\boldsymbol{x}_H, \boldsymbol{a}_H)$. We do not assume that a reward function is given. Instead, we aim to perform the more general problem of *reward-free exploration*, which entails learning a collection of policies that covers the latent state space to the greatest extent possible (Du et al., 2019a; Misra et al., 2020; Efroni et al., 2021).

In more detail, we consider the reward-free exploration task of learning an *approximate policy cover*, which is a collection of policies which can reach any latent state with near-optimal probability. To formalize this notion, for $s \in \mathcal{S}_h$, we let $d^\pi(s) := \mathbb{P}^\pi[\boldsymbol{s}_h = s]$ denote the probability of reaching state s when executing a policy π, and let $\Pi_{\mathsf{M}} := \left\{\pi : \bigcup_{h=1}^H \mathcal{X}_h \to \mathcal{A}\right\}$ be the set of all Markovian policies.

Definition 2.1 (Approximate policy cover)**.** *A collection of policies Ψ is an (α, ε)-policy cover for layer h if for all $s \in \mathcal{S}_h$ such that $\max_{\pi \in \Pi_{\mathsf{M}}} d^\pi(s) \geq \varepsilon$, we have*

$$\max_{\pi \in \Psi} d^\pi(s) \geq \alpha \cdot \max_{\pi' \in \Pi_{\mathsf{M}}} d^{\pi'}(s).$$

Informally, an (α, ε)-policy cover Ψ has the property that for every state $s \in \mathcal{S}$ that can be reached with probability

[1] We use the term "rate-optimal" to refer to optimality of the rate with respect to the accuracy parameter ε, but not necessarily with respect to other parameters.

at least ε, there exists a policy in Ψ that reaches it with probability at least $\alpha \cdot \varepsilon$. For our results, α will be a numeric constant (say, $1/2$), and ε will be a parameter to the algorithm. We show (Appendix B) that given access to such a policy cover, it is possible to optimize any downstream reward function to precision $O(\varepsilon)$.

When $\varepsilon = 0$, Definition 2.1 recovers the policy cover definition used in (Misra et al., 2020). The relaxed notion of a policy cover in Definition 2.1, which allows one to "sacrifice" states that are hard to reach with any policy (i.e. those states s for which $\max_{\pi \in \Pi_{\mathsf{M}}} d^{\pi}(s) < \varepsilon$), is natural in our setting, as we do not assume that all states can be reached with some minimum probability.

Function approximation. To provide sample-efficient learning guarantees, we make use of function approximation. In particular, we do not assume that the true decoder $\phi_\star : \mathcal{X} \to \mathcal{S}$ is known to the learner and, as in prior work (Du et al., 2019a; Misra et al., 2020; Zhang et al., 2022b), assume access to a *decoder class* $\Phi \subseteq (\mathcal{X} \to \mathcal{S})$ that contains $\phi_\star$.

Assumption 2.1 (Realizability). *The decoder class $\Phi \subseteq \{\phi : \mathcal{X} \to \mathcal{S}\}$ contains the true decoder $\phi_\star$.*

The class Φ captures the learner's prior knowledge about the environment, and may consist of neural networks or other flexible function approximators. To simplify presentation, we assume that Φ is finite; as our results only invoke standard uniform convergence arguments, extension to infinite classes and other notions of statistical capacity is straightforward (Misra et al., 2020). We aim to learn an (α, ε)-policy cover (for constant α) using a number of episodes/trajectories ("sample complexity") that scales with $\mathrm{poly}(H, S, A, \log|\Phi|) \cdot 1/\varepsilon^2$. Notably, this guarantee depends on the number of latent states S and the complexity $\log|\Phi|$ for the decoder class, but does not explicitly depend on the size of the observation space $\mathcal{X}$.

2.1. Preliminaries

We proceed to introduce additional notation required to present our main results. Most important will be the notion of *partial policies*, both Markovian and non-Markovian. For any $n, m \in \mathbb{N}$, we denote by $[m \,..\, n]$ the integer interval $\{m, \dots, n\}$. We also let $[n] \coloneqq [1 \,..\, n]$. Further, for any sequence of objects $o_1, o_2, \dots$, we define $o_{m:n} \coloneqq (o_i)_{i \in [m..n]}$.

Partial policies. A *partial policy* is a policy that is defined only over a contiguous subset of layers $[l \,..\, r] \subseteq [H]$. We let $\Pi_{\mathsf{M}}^{l:r} \coloneqq \{\pi : \bigcup_{h=l}^{r} \mathcal{X}_h \to \mathcal{A}\}$ be the set of *Markovian partial* policies that are defined over layers l to r. For a policy $\pi \in \Pi_{\mathsf{M}}^{l:r}$ and layer $h \in [l \,..\, r]$, $\pi(x_h)$ denotes the action taken by the policy at layer h when $x_h \in \mathcal{X}_h$ is the current observation. We will use the notation $\Pi_{\mathsf{M}} \equiv \Pi_{\mathsf{M}}^{1:H}$.

We also consider *non-Markov* (history-dependent) partial policies. For $1 \le l \le r \le H$, we let

$$\Pi_{\mathsf{NM}}^{l:r} \coloneqq \left\{ \pi : \bigcup_{h=l}^{r} (\mathcal{X}_l \times \cdots \times \mathcal{X}_h) \to \mathcal{A} \right\}$$

denote the set of non-Markovian partial policies that are defined over layers l to r. The action of a partial policy $\pi \in \Pi_{\mathsf{NM}}^{l:r}$ is only defined for layers $h \in [l \,..\, r]$, but may depend on the entire history of observations $x_{l:h} = x_l, \dots, x_h$ beginning from layer l. In particular, for layer $h \in [l \,..\, r]$, $\pi(x_{l:h})$ denotes the policy's action when $x_{l:h} \in \mathcal{X}_l \times \cdots \times \mathcal{X}_h$ is the history. For any $1 \le t \le h \le H$, and any pair of partial policies $\pi \in \Pi_{\mathsf{NM}}^{1:t-1}, \pi' \in \Pi_{\mathsf{NM}}^{t:h}$, we let $\pi \circ_t \pi'$ be the partial policy in $\Pi_{\mathsf{NM}}^{1:h}$ that satisfies $(\pi \circ_t \pi')(x_{1:\tau}) = \pi(x_{1:\tau})$ for all $\tau < t$ and $(\pi \circ_t \pi')(x_{1:\tau}) = \pi'(x_{t:\tau})$ for all $\tau \in [t \,..\, h]$. We define $\pi \circ_t \pi'$ similarly when $\pi \in \Pi_{\mathsf{NM}}^{1:\tau}$ for $\tau \ge t$.

Further notation. Given any policy π and BMDP $\mathcal{M}$, we denote by $\mathbb{P}^{\mathcal{M},\pi}$ the probability law over $\{(\boldsymbol{s}_h, \boldsymbol{x}_h, \boldsymbol{a}_h) : h \in [H]\}$ under the process induced by executing π in $\mathcal{M}$. We let $\mathbb{E}^{\mathcal{M},\pi}$ denote the corresponding expectation. For any $h \in [H]$ and $s \in \mathcal{S}_h$, we denote by $d^{\mathcal{M},\pi}(s) \coloneqq \mathbb{P}^{\mathcal{M},\pi}[\boldsymbol{s}_h = s]$ the *occupancy* of s under π. We drop the $\mathcal{M}$ superscript when clear from the context. Given a set of partial policies $\Psi \coloneqq \{\pi^{(i)} : i \in [N]\}$, we denote by $\mathtt{unif}(\Psi)$ the random partial policy obtained by sampling $\boldsymbol{i} \sim \mathtt{unif}([N])$ and playing $\pi^{(\boldsymbol{i})}$. We overload notation slightly and denote by $\pi_{\mathtt{unif}}$ the random policy that plays actions in $\mathcal{A}$ uniformly at random at all layers. We use the notation $\widetilde{O}(1)$ to hide poly-logarithmic factors in $H, S, A, \log|\Phi|$, and ε^{-1}.

3. Algorithm and Main Results

We now present our algorithm, MusIK, and prove that it efficiently learns a policy cover with rate-optimal $\mathrm{poly}(H, S, A, \log|\Phi|) \cdot 1/\varepsilon^2$ sample complexity. First, in Section 3.1, we highlight the challenges faced in achieving similar guarantees with existing approaches, with an emphasis on difficulties removing a statistical assumption known as *reachability*. With this out of the way, we introduce the MusIK algorithm (Section 3.2) and give an overview of its main performance guarantee and key features (Section 3.3). Extensions to reward-based reinforcement learning are deferred to Appendix B.

3.1. Challenges and Related Work

For the Block MDP model, the optimal sample complexity to learn an ε-optimal policy or learn an (α, ε)-approximate policy cover for constant α scales with $1/\varepsilon^2$.[2] Previous approaches—both for reward-free and reward-based

[2] An $\Omega(1/\varepsilon^2)$ lower bound on the sample complexity follows from standard lower bounds for tabular RL (Jin et al., 2020).

exploration—either achieve this rate, but are not computationally efficient, or only achieve it under additional statistical assumptions that may not be satisfied in general. To motivate the need for new algorithm design and analysis ideas, let us highlight where these challenges arise.

Existing algorithms can be broken into two families, *optimistic algorithms*, and algorithms that are not optimistic, but require *reachability* conditions. Optimistic algorithms use the principle of *optimism in the face of uncertainty* to drive exploration. Implementing optimism in the BMDP setting is challenging because the latent states are not observed, which prevents the naive application of state-action exploration bonuses found in tabular RL (Azar et al., 2017; Jin et al., 2018). An alternative is to appeal to *global optimism*, which computes an optimistic policy by optimizing over a *version space* of plausibly-optimal value functions. This approach enjoys rate-optimal sample complexity (Jiang et al., 2017; Du et al., 2021; Jin et al., 2021), but cannot be implemented efficiently in general because it requires searching for value functions that satisfy non-convex constraints at all layers $h \in [H]$ simultaneously ("globally") (Dann et al., 2018).

As a tractable replacement for global optimism, a more recent line of algorithms implement optimism using a *plug-in* approach which computes layer-wise bonuses with respect to an *estimated decoder*. First, Uehara et al. (2022) show that under the stronger assumption that the learner has access to a realizable *model class*, it is possible to learn a decoder for which the plug-in approach attains rate-optimal sample complexity; this observation, while interesting, falls short of a model-free guarantee that scales only with $\log|\Phi|$. More recently, Zhang et al. (2022b) observed that similar results can be achieved with only decoder realizability by appealing to a certain min-max representation learning objective.[3] However, this objective involves a form of adversarial training that increases the sample complexity, leading to a final guarantee that scales with $1/\varepsilon^4$ instead of $1/\varepsilon^2$.

Given the challenges faced by optimistic approaches, an alternative is to do away with optimism entirely. Algorithms from this family (Du et al., 2019b; Misra et al., 2020) proceed in a forward fashion: They first solve a representation learning objective which enables building a policy cover for layer 2. Then, using this policy cover, they explore to collect data that can be used to solve a similar representation learning objective for layer 3, then use this to build a policy cover for layer 3, and so on. A-priori, a natural concern is that the myopic nature of these step-by-step approaches might lead to approximation errors that compound exponentially as a function of the horizon H. To avoid, this, existing work (Du et al., 2019b; Misra et al., 2020) makes a *minimum reachability assumption*.

[3] Modi et al. (2021) employ a similar representation learning objective, but require a minimum reachability assumption.

Definition 3.1 (Minimum reachability). *There exists $\eta_{\min} > 0$ such that for all $h \in [H]$ and $s \in \mathcal{S}_h$, there exists $\pi \in \Pi_{\mathsf{M}}$ such that $d^{\pi}(s) \geq \eta_{\min}$.*

Reachability is a useful assumption because it ensures that for every possible state s in the latent space, we can learn a policy that can reach s with sufficiently high probability (say, with probability at least $\eta_{\min}/2$), which prevents errors from cascading as one moves forward from layer h to layer $h+1$. The best algorithm from this family, HOMER, attains sample complexity that is proportional to $1/\varepsilon^2$, but scales inversely proportional to the reachability parameter $\eta_{\min}$, and provides no guarantees when $\eta_{\min} = 0$. Prior to our work, it was not known whether any algorithm based on the non-optimistic layer-by-layer approach could succeed at all in the absence of reachability, let alone achieve rate-optimal sample complexity. We refer to Table 1 in Appendix A for a summary.

3.2. The MusIK Algorithm

Algorithm 1 MusIK: Multi-Step Inverse Kinematics

Require: Decoder class Φ. Number of samples n.

1: Set $\Psi^{(1)} = \varnothing$.
2: **for** $h = 2 \dots, H$ **do**
3: Let $\Psi^{(h)} = \texttt{IKDP}(\Psi^{(1)}, \dots, \Psi^{(h-1)}, \Phi, n)$ // Alg. 2
4: **Return:** Policy covers $\Psi^{(1)}, \dots, \Psi^{(H)}$.

Our main algorithm, MusIK, is presented in Algorithm 1. MusIK performs reward-free exploration, iteratively building approximate policy covers $\Psi^{(1)}, \dots, \Psi^{(H)}$ for layers $h = 1, \dots, H$. The algorithm first gathers data from the initial state distribution, and uses this to learn a policy cover $\Psi^{(2)}$ for layer 2 (we adopt the convention that $\Psi^{(1)} = \varnothing$). The algorithm then collects data using $\Psi^{(2)}$, and uses this to build an approximate policy cover $\Psi^{(3)}$ for layer 3, and so on. Once layer H is reached, the algorithm returns $\Psi^{(1)}, \dots, \Psi^{(H)}$. The crux of the MusIK algorithm is a subroutine, IKDP (Inverse Kinematics for Dynamics Programming, Algorithm 2) which, at each step h, makes use of the previous policy covers $\Psi^{(1)}, \dots, \Psi^{(h-1)}$ to compute the policy cover $\Psi^{(h)}$. In what follows, we give a detailed overview of IKDP.

The IKDP subroutine. For each $h \in [H]$, the IKDP subroutine (Algorithm 2) uses the policy covers $\Psi^{(1)}, \dots, \Psi^{(h-1)}$ to construct the policy cover $\Psi^{(h)}$ for layer h in a backwards fashion inspired by dynamic programming: Beginning from layer $h-1$, the algorithm builds a collection of partial policies $\{\hat{\pi}^{(i,h-1)}\}_{i \in [S]} \in \Pi_{\mathsf{M}}^{h-1:h-1}$ using data collected by rolling in with $\Psi^{(h-1)}$; each policy $\hat{\pi}^{(i,h-1)}$ is responsible for targeting a single latent state in layer h.

The algorithm then moves back one layer, and constructs a collection $\{\hat{\pi}^{(i,h-2)}\}_{i\in[S]} \in \Pi_{\mathsf{M}}^{h-2:h-1}$ using data collected by rolling in with $\Psi^{(h-2)}$ and rolling out using the collection $\{\hat{\pi}^{(i,h-1)}\}_{i\in[S]}$. This process is repeated until the first layer is reached, and the final collection of policies $\Psi^{(h)} = \{\hat{\pi}^{(i,1)}\}_{i\in[S]}$ is returned. The key invariant maintained throughout this process is that for all layers $t \in [h-1]$, for every latent state $s \in \mathcal{S}_h$, there exists a partial policy in the set $\{\hat{\pi}^{(i,t)}\}_{i\in[S]}$ that reaches s with near-optimal probability starting from layer t (in a certain average-case sense).

Multi-step inverse kinematics objective. For each layer $t \in [h-1]$, given the partial policies $\{\hat{\pi}^{(i,t+1)}\}_{i\in[S]}$ from the previous backward step, IKDP computes the collection $\{\hat{\pi}^{(i,t)}\}_{i\in[S]}$ by appealing to a regression objective (Line 7) based on *multi-step inverse kinematics* (Lamb et al., 2022). To motivate the approach, we recall that a significant challenge faced in the BMDP setting is that the latent states are not directly observed. Were not the case, it would be possible to build a policy cover by directly optimizing "visitation" reward functions of the form $\boldsymbol{r}_h^{(s)} \coloneqq \mathbb{I}\{\boldsymbol{s}_h = s\}$ for each $s \in \mathcal{S}_h$ (this can be accomplished using standard methods such as PSDP (Bagnell et al., 2003; Misra et al., 2020)). As an alternative, one can think of IKDP as constructing proxies for the state-action value functions (Q-functions) associated with the (unobserved) reward functions $\boldsymbol{r}_h^{(s)}$ for each $s \in \mathcal{S}_h$. These proxies are constructed using the objective in Line 7, which involves predicting actions from observations at different layers (multi-step inverse kinematics).

In more detail, for each backward iteration $t \in [h-1]$, IKDP samples $\pi \sim \Psi^{(t)}$, executes π up to layer t, plays a random action $\boldsymbol{a}_t \sim \pi_{\mathsf{unif}}$, then selects a random index $\boldsymbol{i}_t \sim \mathsf{unif}([S])$ and executes $\hat{\pi}^{(\boldsymbol{i}_t,t+1)} \in \Pi_{\mathsf{NM}}^{t+1:h-1}$ from layer $t+1$ onward. The regression objective in Line 7 then uses this data to estimate the conditional density for the pair $(\boldsymbol{a}_t, \boldsymbol{i}_t)$, conditioned on the observations $\boldsymbol{x}_t$ and $\boldsymbol{x}_h$. This estimate for the conditional density acts as a proxy for the Q-functions associated with the unobserved visitation reward functions $\boldsymbol{r}_h^{(s)}$ described above. Thanks to the decodability property of the BMDP model, it can be shown that the Bayes-optimal solution to the regression objective in Line 7 depends on observations only through latent states. This allows us to parameterize the objective using the decoder class Φ, which is key to achieving low sample complexity.

Policy composition. After solving the multi-step inverse kinematics objective in Line 7, IKDP uses the resulting decoder $\hat{\phi}^{(t)}$ and function $\hat{f}^{(t)}$ to build the set of partial policies $\{\hat{\pi}^{(i,t)}\}_{i\in[S]}$ from the set $\{\hat{\pi}^{(i,t+1)}\}_{i\in[S]}$ produced at the previous backward step (Lines 8 and 9). Here, the challenge is that there is no way to know which policy $\{\hat{\pi}^{(i,t+1)}\}_{i\in[S]}$ is responsible for targeting a given state $s \in \mathcal{S}_h$ due to non-identifiability. We address this using a *non-Markovian* policy construction in Lines 8 and 9, which we now describe.

Recall that the objective in Line 7 predicts both actions and *indices of roll-out policies*. Predicting the indices of roll-out policies offers a mechanism to associate partial policies at successive layers. To do so, Line 8 of IKDP defines

$$(\hat{a}^{(i,t)}(x), \hat{\imath}^{(i,t)}(x)) = \arg\max_{(a,j)} \hat{f}^{(t)}((a,j) \mid \hat{\phi}^{(t)}(x), i),$$

for $x \in \mathcal{X}_t$. One should interpret $j = \hat{\imath}^{(i,t)}(x)$ as the *most likely* (or most closely associated) roll-out policy $\hat{\pi}^{(j,t+1)}$ when the (decoded) latent state at layer h is $i \in [S]$ and $x \in \mathcal{X}_t$ is the current observation at layer t. Meanwhile, the action $\hat{a}^{(i,t)}(x)$ (approximately) maximizes the probability of reaching i if we roll out with $\hat{\pi}^{(j,t+1)}$. With this in mind, the composition rule in Line 9 constructs $\hat{\pi}^{(i,t)}$ via

$$\hat{\pi}^{(i,t)}(x_{t:\tau}) \coloneqq \hat{a}^{(i,t)}(x_t)$$

for $\tau = t$ and $x_t \in \mathcal{X}_t$, and

$$\hat{\pi}^{(i,t)}(x_{t:\tau}) \coloneqq \hat{\pi}^{(\hat{\imath}^{(i,t)}(x_t),t+1)}(x_{t+1:\tau}),$$

for $\tau \in [t+1\,..\,h-1]$ and $x_{t:\tau} \in \mathcal{X}_t \times \cdots \times \mathcal{X}_\tau$. That is, for layers $t+1, \ldots, h-1$, this construction follows the policy $\hat{\pi}^{(\hat{\imath}^{(i,t)}(\boldsymbol{x}_t),t+1)}$ which—per the discussion above—is most associated with the decoded state $i \in [S]$. At layer t, we select $\boldsymbol{a}_t = \hat{a}^{(i,t)}(\boldsymbol{x}_t)$, maximizing the probability of reaching the decoded latent state $i \in [S]$ when we roll-out with $\hat{\pi}^{(\hat{\imath}^{(i,t)}(\boldsymbol{x}_t),t+1)}$. This construction, while intuitive, is non-Markovian, since for layers $t+1$ and onward the policy depends on $\boldsymbol{x}_t$ through $\hat{\imath}^{(i,t)}(\boldsymbol{x}_t)$.

We refer to Appendix E for a detailed overview of the analysis ideas behind IKDP, as well as further intuition.

On inverse kinematics. MusIK can be viewed as generalizing the notion of *one-step inverse kinematics* to multiple steps. One-step inverse kinematics, which aims to predict the action $\boldsymbol{a}_h$ from $\boldsymbol{x}_h$ and $\boldsymbol{x}_{h+1}$, has been explored in a number of empirical works (Pathak et al., 2017; Badia et al., 2020; Baker et al., 2022; Bharadhwaj et al., 2022). In theory, however, it can be shown that this approach can fail to meaningfully recover latent state information (Misra et al., 2020; Efroni et al., 2021). In particular, it is prone to incorrectly merging latent states with different dynamics. Multi-step inverse kinematics generalizes one-step inverse kinematics by predicting $\boldsymbol{a}_h$ from $\boldsymbol{x}_h$ and $\boldsymbol{x}_{h'}$ for all possible choices for $h' > h$. Recent work of Lamb et al. (2022) observed that—in the infinite-data limit—multi-step inverse kinematics can rectify the issues with one-step IK, and enjoys other benefits including robustness to exogenous information. Our work is the first to provably combine multi-step inverse kinematics with systematic exploration to derive finite-sample

Algorithm 2 IKDP: Inverse Kinematics for Dynamic Programming

Require: Approximate covers $\Psi^{(1)}, \dots, \Psi^{(h-1)}$ for layers 1 to $h-1$, where $\Psi^{(t)} \subseteq \Pi_{\mathsf{NM}}^{1:t-1}$. Decoder class Φ. Number of samples n.

1: **for** $t = h-1, \dots, 1$ **do**
2: $\quad \mathcal{D}^{(t)} \leftarrow \varnothing$.
3: $\quad$ **for** n times **do**
4: $\quad\quad$ Sample $\boldsymbol{i}_t \sim \mathtt{unif}([S])$ and set $\hat{\pi} = \hat{\pi}^{(\boldsymbol{i}_t, t+1)}$.
5: $\quad\quad$ Sample $(\boldsymbol{x}_t, \boldsymbol{a}_t, \boldsymbol{x}_h) \sim \mathbb{P}^{\mathtt{unif}(\Psi^{(t)}) \circ_t \pi_{\mathtt{unif}} \circ_{t+1} \hat{\pi}}$.
6: $\quad\quad$ $\mathcal{D}^{(t)} \leftarrow \mathcal{D}^{(t)} \cup \{(\boldsymbol{i}_t, \boldsymbol{a}_t, \boldsymbol{x}_t, \boldsymbol{x}_h)\}$.

`/* Inverse kinematics */`

7: $\quad$ Compute the solution $(\hat{f}^{(t)}, \hat{\phi}^{(t)})$ of the problem

$$\max_{f \in \mathcal{F}, \phi \in \Phi} \sum_{(j,a,x,x') \in \mathcal{D}^{(t)}} \log f((a,j) \mid \phi(x), \phi(x')), \quad (1)$$

$\quad$ where $\mathcal{F} \coloneqq [S]^2 \to \Delta(\mathcal{A} \times [S])$.

`/* Update partial policy cover */`

8: $\quad$ For each $i \in [S]$ and $x \in \mathcal{X}_t$ define

$$(\hat{a}^{(i,t)}(x), \hat{\iota}^{(i,t)}(x)) = \arg\max_{(a,j)} \hat{f}^{(t)}((a,j) \mid \hat{\phi}^{(t)}(x), i). \quad (2)$$

9: $\quad$ For $i \in [S]$, $\tau \in [t+1 \,..\, h-1]$, $x_{t:\tau} \in \bigtimes_{k=t}^{\tau} \mathcal{X}_k$, define $\hat{\pi}^{(i,t)} \in \Pi_{\mathsf{NM}}^{t:h-1}$ via $\hat{\pi}^{(i,t)}(x_t) = \hat{a}^{(i,t)}(x_t)$ and $\hat{\pi}^{(i,t)}(x_{t:\tau}) = \hat{\pi}^{(\hat{\iota}^{(i,t)}(x_t), t+1)}(x_{t+1:\tau})$.

10: **Return:** Layer h cover $\Psi^{(h)} = \{\hat{\pi}^{(i,1)}\}_{i \in [S]} \subseteq \Pi_{\mathsf{NM}}^{1:h-1}$.

guarantees.[4]

Remark 3.1. IKDP *also bears some similarity to the* PSDP *algorithm (see Bagnell et al. (2003); Misra et al. (2020) and Algorithm 4), and uses the principle of dynamic programming in a similar fashion. Unlike* PSDP, IKDP *does not require feedback from an external reward function, and can be thought of as automatically discovering its own reward function to drive exploration.*

Efficient implementation. MusIK is practical, and is computationally efficient whenever the standard log-loss conditional density estimation problem on Line 7 of IKDP can be solved efficiently for the decoder class Φ of interest. In practice, Φ and $\mathcal{F} \coloneqq [S]^2 \to \Delta(\mathcal{A} \times [S])$ can both be approximated with neural networks or other flexible function classes, and the conditional density estimation problem in Line 7 can be solved by appealing to stochastic gradient descent or other off-the-shelf training procedures; this is the approach taken in our experiments (Section 5).

Let us also remark on the complexity of representing and executing the partial policies $\{\hat{\pi}^{(i,t)} : i \in [S], t \in [h-1]\}$ computed in Line 9 of IKDP. These policies are non-Markovian, which presents a problem at first glance, since general non-Markovian policies in a horizon-H MDP with S states require a table of size S^H to represent. Fortunately, the non-Markovian policies in IKDP are quite structured, and can be represented and executed with runtime and memory complexity that is polynomial in H instead of exponential; see Algorithm 3 in Appendix A for pseudocode. In particular, the partial policies for layer h can be fully represented using $O(H)$ memory via the collection of functions $\{(\hat{f}^{(t)}, \hat{\phi}^{(t)}) : t \in [h-1]\}$ learned in Line 7 of Algorithm 2 (assuming that, for $t \in [h-1]$, storing $(\hat{f}^{(t)}, \hat{\phi}^{(t)})$ requires $O(1)$ memory). One can then execute the partial policies to generate a trajectory using $O(HSA)$ runtime, (assuming that evaluating $\hat{\phi}^{(t)}(x)$ costs $O(1)$ units of time).

3.3. Main Result

We now state the main guarantee for MusIK (proven in Appendix H.2) and discuss some of its implications.

Theorem 3.2 (Main theorem for MusIK). *Let $\varepsilon, \delta \in (0,1)$ be given. Suppose that Assumption 2.1 holds, and that n is chosen such that*

$$n \geq \frac{cA^2S^{10}H^2\left(S^3A\log n + \log(|\Phi|H^2/\delta)\right)}{\varepsilon^2},$$

for some absolute constant $c > 0$ independent of all problem parameters. Then, with probability at least $1-\delta$, the policies $\Psi^{(1)}, \dots, \Psi^{(H)}$ produced by MusIK *(Algorithm 1) are $(1/4, \varepsilon)$-policy covers for layers 1 to H. The total number of trajectories used by the algorithm is at most*

$$\widetilde{O}(1) \cdot \frac{A^2S^{10}H^4\left(AS^3 + \log(|\Phi|H^2/\delta)\right)}{\varepsilon^2}. \quad (3)$$

Theorem 3.2 is the first sample complexity guarantee for the BMDP setting that 1) is attained by an efficient algorithm, 2) does not scale with the reachability parameter $\eta_{\min}$, and 3) attains rate-optimal $1/\varepsilon^2$ sample complexity. Previous efficient BMDP algorithms such as MOFFLE or HOMER have sample complexity scaling with $1/\varepsilon^2 \cdot \mathrm{poly}(1/\eta_{\min})$, where $\eta_{\min} \coloneqq \min_{s \in \mathcal{S}} \sup_{\pi \in \Pi_{\mathsf{M}}} d^{\pi}(s)$ is the reachability parameter, and do not provide guarantees if $\eta_{\min} = 0$. More recents results (Zhang et al., 2022b) do not require $\eta_{\min} > 0$, but have suboptimal dependence on ε. We remark that the dependence on the problem-dependent parameters S, A, and H in our result is loose, and improving this with an

[4] The work of (Efroni et al., 2021) also makes use of multi-step inverse models, but is limited to deterministic systems. Mhammedi et al. (2020) also uses a form of multi-step inverse kinematics in the context of linear control with rich observations, but their approach is specialized to the linear setting.

efficient algorithm is an interesting open question; other efficient algorithms have similarly loose dependence, per Table 1.

Practicality. As discussed in the prequel, MusIK is computationally efficient whenever the standard conditional density estimation problem in Line 7 of IKDP can be solved efficiently for the decoder class Φ of interest, allowing for the use of off-the-shelf models and estimation algorithms; in experiments (Section 5), we appeal to deep neural networks and stochastic gradient descent.

From prior work, the only other computationally-efficient (and model-free) algorithm that does not require minimum reachability in BMDPs is BRIEE (Zhang et al., 2022b). The log-loss conditional density estimation objective in MusIK is somewhat simpler than the min-max representation learning objective in BRIEE, with the latter necessitating adversarial training.

4. Proof Techniques

We find it somewhat surprising that MusIK attains rate-optimal sample complexity in spite of forgoing optimism. The proof of Theorem 3.2, which we sketch in Appendix E, has two main components. For the first component, we prove that the multi-step inverse kinematics objective learns a decoder that can be used to drive exploration; this formalizes the intuition in Section 3.2. With this established, proving that MusIK succeeds under minimum reachability (Definition 3.1) is somewhat straightforward, but proving that the algorithm 1) succeeds in absence of this assumption, and 2) achieves optimal sample complexity is more involved. For this component of the proof, we use a new analysis tool we refer to as an *extended BMDP* which, in tandem with another tool we refer to as the *truncated policy class*, allows one to emulate certain consequences of reachability even when the condition does not hold. These techniques, which we anticipate will find broader use in the context of non-optimistic algorithms based on policy covers, appear to be new even for tabular reinforcement learning.

Due to space limitations, an in-depth overview of the analysis of* MusIK *is deferred to Appendix E.

As a teaser, in this section we introduce the most important technical tools used in the proof of Theorem 3.2, the extended BMDP and truncated policy class, which, play a key role in providing tight guarantees for MusIK (and more broadly, non-optimistic algorithms) in the absence of minimum reachability. *We recommend reading the full overview in Appendix E before diving into the full proof of Theorem 3.2 (Appendix H).*

Analysis in Extended BMDP. MusIK proceeds by inductively building a sequence of policy covers $\Psi^{(1)}, \dots, \Psi^{(H)}$. A key invariant maintained by the algorithm is that for each layer h, $\Psi^{(1)}, \dots, \Psi^{(h-1)}$ provide good coverage for layers $1, \dots, h-1$, and thus can be used to gather data which will allow us to efficiently learn $\Psi^{(h)}$. Prior approaches that build policy covers in a similar fashion (Du et al., 2019b; Misra et al., 2020) require the minimum reachability assumption (Definition 3.1) to ensure that for each h, $\Psi^{(h)}$ *uniformly* covers all states in $\mathcal{S}_h$. In the absence of reachability, we inevitably must sacrifice certain hard-to-reach states, which requires a more refined analysis. In particular, we must show that the effect of ignoring hard-to-reach states at earlier layers do not compound as the algorithm progresses.

To provide such an analysis, we make use of an extended BMDP $\overline{\mathcal{M}}$. The extended BMDP $\overline{\mathcal{M}}$ augments $\mathcal{M}$ by adding H "terminal" states $\mathfrak{t}_{1:H}$ and one additional *terminal* action $\mathfrak{a}$ as follows: **I)** The latent state space is $\overline{\mathcal{S}} := \bigcup_{h=1}^{H} \overline{\mathcal{S}}_h$, where $\overline{\mathcal{S}}_h := \mathcal{S}_h \cup \{\mathfrak{t}_h\}$; **II)** the action space is $\overline{\mathcal{A}} := \mathcal{A} \cup \{\mathfrak{a}\}$, where $\mathfrak{a}$ is an action that deterministically transitions to $\mathfrak{t}_{h+1}$ from every state at layer $h \in [H-1]$; and **III)** For $h \in [H-1]$, taking any action in $\overline{\mathcal{A}}$ at state $\mathfrak{t}_h$ transitions to $\mathfrak{t}_{h+1}$ deterministically. We assume the state $\mathfrak{t}_h$ emits itself as an observation and we write $\overline{\mathcal{X}}_h := \mathcal{X}_h \cup \{\mathfrak{t}_h\}$, for all $h \in [H]$. The dynamics of $\overline{\mathcal{M}}$ are otherwise identical to $\mathcal{M}$, and for any policy $\pi \in \overline{\Pi}_{\mathsf{NM}} := \left\{\pi : \bigcup_{h=1}^{H} (\overline{\mathcal{X}}_1 \times \cdots \times \overline{\mathcal{X}}_h) \to \overline{\mathcal{A}}\right\}$, we define $\overline{\mathbb{P}}^{\pi} := \mathbb{P}^{\overline{\mathcal{M}},\pi}$, $\overline{\mathbb{E}}^{\pi} := \mathbb{E}^{\overline{\mathcal{M}},\pi}$, and $\bar{d}^{\pi}(s) := \mathbb{P}^{\overline{\mathcal{M}},\pi}[\boldsymbol{s}_h = s]$, for all $s \in \mathcal{S}_h$ and $h \in [H]$.

Truncated policy class. On its own, the extended BMDP is not immediately useful. The most important idea behind our analysis is to combine it with a restricted sub-class of policies we refer to as the truncated policy class. Define $\overline{\Pi}_{\mathsf{M}} := \{\pi : \bigcup_{h=1}^{H} \overline{\mathcal{X}}_h \to \overline{\mathcal{A}}\}$. For $\epsilon \in (0,1)$, we define a sequence of policy classes $\overline{\Pi}_{0,\epsilon}, \dots, \overline{\Pi}_{H,\epsilon}$, inductively, starting from $\overline{\Pi}_{0,\epsilon} = \overline{\Pi}_{\mathsf{M}}$ and letting $\overline{\Pi}_{t,\epsilon}$ be the set for which $\pi \in \overline{\Pi}_{t,\epsilon}$ if and only if

$$\exists \pi' \in \overline{\Pi}_{t-1,\epsilon} \text{ such that } \forall h \in [H], \forall s \in \overline{\mathcal{S}}_h, \forall x \in \phi_\star^{-1}(s),$$
$$\pi(x) = \begin{cases} \mathfrak{a}, & \text{if } h = t \text{ and } \max_{\tilde{\pi} \in \overline{\Pi}_{t-1,\epsilon}} \bar{d}^{\tilde{\pi}}(s) < \epsilon, \\ \pi'(x), & \text{otherwise.} \end{cases}$$

Restated informally, the class $\overline{\Pi}_{t,\epsilon}$ is identical to $\overline{\Pi}_{t-1,\epsilon}$, except that at layer t, all policies in the class take the terminal action $\mathfrak{a}$ in latent states s for which $\max_{\tilde{\pi} \in \overline{\Pi}_{t-1,\epsilon}} \bar{d}^{\tilde{\pi}}(s) < \epsilon$.

We define the *truncated policy class* as $\overline{\Pi}_{\epsilon} := \overline{\Pi}_{H,\epsilon}$. The truncated policy class satisfies two fundamental technical properties. First, by construction, all policies in the class take the terminal action $\mathfrak{a}$ when they encounter states that are not ϵ-reachable by $\overline{\Pi}_{\epsilon}$. The next lemma formalizes this.

Lemma 4.1. *Let $\epsilon \in (0,1)$ be given, and define $\mathcal{S}_{h,\epsilon} := \{s \in \mathcal{S}_h : \max_{\pi \in \overline{\Pi}_{\epsilon}} \bar{d}^{\pi}(s) \geq \epsilon\}$. Then for all $h \in [H]$ if $s \in \mathcal{S}_h \setminus \mathcal{S}_{h,\epsilon}$, then $\pi(x) = \mathfrak{a}$, for all $x \in \phi_\star^{-1}(s)$ and $\pi \in \overline{\Pi}_{\epsilon}$.*

Second, in spite of the fact that policies in $\overline{\Pi}_\epsilon$ always take the terminal action on states with low visitation probability, they can still achieve near-optimal visitation probability for all states in $\overline{\mathcal{M}}$ (up to additive error).

Lemma 4.2 (Approximation for truncated policies)**.** *Let $\epsilon \in (0,1)$ be given. For all $h \in [H]$ and $s \in \mathcal{S}_h$,*

$$\max_{\pi \in \overline{\Pi}_{\mathtt{M}}} \bar{d}^{\pi}(s) \leq \max_{\pi \in \overline{\Pi}_{\epsilon}} \bar{d}^{\pi}(s) + S\epsilon.$$

The proofs for these results (and other results in this subsection) are elementary, and are given in Appendix F. Building on these properties, our proof of Theorem 3.2 makes use of two key ideas:

1. Even though the extended BMDP $\overline{\mathcal{M}}$ does not necessarily enjoy minimum reachability (Definition 3.1), if we restrict ourselves to competing against policies in $\overline{\Pi}_\epsilon$, Lemma 4.1 will allow us to "emulate" certain properties enjoyed by ϵ-reachable MDPs. This in turn will imply that if we only wish to learn a policy cover that has good coverage relative to $\overline{\Pi}_\epsilon$, Algorithm 1 will succeed.
2. By Lemma 4.2, we lose little by restricting our attention to the class $\overline{\Pi}_\epsilon$. This will allow us to transfer any guarantees we achieve with respect to the extended BMDP $\overline{\mathcal{M}}$ and truncated policy class $\overline{\Pi}_\epsilon$ back to the original BMDP $\mathcal{M}$ and unrestricted policy class $\Pi_{\mathtt{M}}$.

We will make the first point precise in Appendices E.1 and E.2. For now, we formalize the second point via another technical result, Lemma 4.3. To do so, we introduce the notion of a *relative policy cover* (generalizing Definition 2.1).

Definition 4.1 (Relative policy cover)**.** *Let $\alpha, \varepsilon \in [0,1)$ be given. Consider a BMDP $\mathcal{M}'$, and let Π and Ψ be two sets of policies. We say that Ψ is an (α, ε)-policy cover relative to Π for layer h in $\mathcal{M}'$, if*

$$\max_{\pi \in \Psi} d^{\mathcal{M}', \pi}(s) \geq \alpha \cdot \max_{\pi \in \Pi} d^{\mathcal{M}', \pi}(s),$$

for all $s \in \mathcal{S}_h$ such that $\max_{\pi \in \Pi} d^{\mathcal{M}', \pi}(s) \geq \varepsilon$.

Lemma 4.3 (Policy cover transfer)**.** *Let $\varepsilon \in (0,1)$ be given, and define $\epsilon := \varepsilon/(2S)$. Let Ψ be a set of policies for $\overline{\mathcal{M}}$ that never take the terminal action $\mathfrak{a}$. If Ψ is a $(1/2, \epsilon)$-policy cover relative to $\overline{\Pi}_\epsilon$ in $\overline{\mathcal{M}}$ for all layers, then Ψ is a $(1/4, \varepsilon)$-policy cover relative to $\Pi_{\mathtt{M}}$ in the $\mathcal{M}$ for all layers.*

Lemma 4.3 implies that for any ε, letting $\epsilon := \varepsilon/2S$, if we can construct a set Ψ that acts as a $(1/2, \epsilon)$-policy cover relative to $\overline{\Pi}_\epsilon$ in $\overline{\mathcal{M}}$, then Ψ will also be a $(1/4, \varepsilon)$-policy cover relative to $\Pi_{\mathtt{M}}$ in the original BMDP $\mathcal{M}$, which is ultimately what we wish to accomplish. This allows us to restrict our attention to the former goal in the analysis.

We refer the reader to Appendix E for the overview of the analysis of Theorem 3.2, which builds on the tools presented in this section, and to Appendix H for the full proof. We anticipate that these techniques will find broader use in RL.

5. Experiments

As a validation experiment, we evaluate the performance of MusIK on the challenging "diabolical combination lock" ("CombLock") environment (Misra et al., 2020; Zhang et al., 2022b), which combines high-dimensional observations with anti-shaped, sparse rewards, necessitating representation learning and systematic exploration.

Environment. We adopt the diabolical combination lock (CombLock) environment from Misra et al. (2020); Zhang et al. (2022b), which is parameterized by the horizon H and number of actions $A = 10$. At each layer h, there are $N = 3$ states $s_{h,1}, s_{h,2}, s_{h,3} \in \mathcal{S}_h$, where $s_{h,1}, s_{h,2}$ are "good" states and $s_{h,3}$ is a "bad" terminal state. For each layer h, there exists a pair of "good" actions $u_{h,1}, u_{h,2} \in \mathcal{A}$ such that taking action $u_{h,j}$ in state $s_{h,j}$, for $j \in \{1,2\}$, leads to one of the good states $\{s_{h+1,1}, s_{h+1,2}\}$ at next layer with equal probability. All actions $a_h \notin \{u_{h,1}, u_{h,2}\}$ lead to the bad state $s_{h+1,3}$ deterministically. The sequences of good actions $u_{1:H,1}$ and $u_{1:H,2}$ are sampled uniformly at random from the set of actions $\mathcal{A}$ when the environment is initialized and are unknown to the learner. The optimal reward of 1 can only be achieved at the states $s_{H,1}$ and $s_{H,2}$ (we postpone the details of the reward and observation processes to Appendix C).

Since the good actions $\{u_{h,1}, u_{h,2}\}_{h \in [H]}$ are not known to the agent, deliberate exploration is required to learn a policy that maximizes the reward function; note it is only possible achieve reward 1 if the agent selects a good action for all $h \in [H]$. For example, when the horizon is set to $H = 100$, the probability of finding the optimal policy through naive uniform exploration is 10^{-100}. In addition, representation learning is required to recover the latent state $\boldsymbol{s}_h$ from the observation $\boldsymbol{x}_h$ at each layer, with the best decoder depending on the layer h.

Evaluation and results. We compare MusIK to HOMER (Misra et al., 2020) and BRIEE (Zhang et al., 2022b) which, amongst provably efficient algorithms, have the best known empirical performance.[5] For MusIK, we adopt the decoder class $\Phi := \{\phi_W : x \mapsto \arg\max_{i \in [N]} Wx \mid W \in \mathbb{R}^{N \times d}\}$, which is the same as that used in (Misra et al., 2020). See Appendix C for implementation details. We do not reproduce BRIEE and HOMER, and instead report the results from

[5] We compare only against other model-free methods, and do not consider model-based approaches (Uehara et al., 2022; Zhang et al., 2022a; Ren et al., 2022).

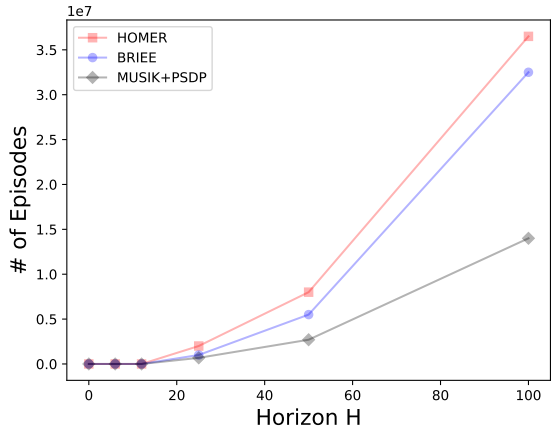

Figure 1. Number of episodes required to identy the optimal policy, as a function of the horizon H for the CombLock experiment.

Zhang et al. (2022b).

Figure 1 reports the number of episodes (or, number of sampled trajectories) required for each method to identify the optimal policy, as a function of the horizon H; we declare the returned policy $\hat{\pi}$ to be optimal if the average reward over 50 trajectories is 1. For MusIK, we plot the *worst-case* number of episodes across 5 different initialization seeds. For BRIEE and HOMER, we only report the median (instead of the worst-case) number of trajectories over 5 different seeds required to find the optimal policy; note that this only improves the results for the baseline methods compared to MusIK + PSDP. We find that for small values of H, all methods have similar performance, but for large horizon, MusIK outperforms the baselines. In particular, for $H = 100$, MusIK is able to find the optimal policy using almost three times fewer episodes than HOMER and BRIEE. This suggests that the multi-step inverse kinematics objective in MusIK may indeed carry practical (as opposed to just theoretical) benefits over the alternative representation learning approaches used in HOMER and BRIEE. Performing a large scale evaluation is a promising direction for future research.

References

Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*, pages 263–272, 2017.

Adrià Puigdomènech Badia, Bilal Piot, Steven Kapturowski, Pablo Sprechmann, Alex Vitvitskyi, Zhaohan Daniel Guo, and Charles Blundell. Agent57: Outperforming the atari human benchmark. In *International Conference on Machine Learning*, pages 507–517. PMLR, 2020.

James Bagnell, Sham M Kakade, Jeff Schneider, and Andrew Ng. Policy search by dynamic programming. *Advances in neural information processing systems*, 16, 2003.

Bowen Baker, Ilge Akkaya, Peter Zhokhov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. Video pretraining (vpt): Learning to act by watching unlabeled online videos. *arXiv preprint arXiv:2206.11795*, 2022.

Homanga Bharadhwaj, Mohammad Babaeizadeh, Dumitru Erhan, and Sergey Levine. Information prioritization through empowerment in visual model-based rl. *arXiv preprint arXiv:2204.08585*, 2022.

Fan Chen, Yu Bai, and Song Mei. Partially observable rl with b-stability: Unified structural condition and sharp sample-efficient algorithms. *arXiv preprint arXiv:2209.14990*, 2022.

Christoph Dann, Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. On oracle-efficient PAC RL with rich observations. In *Advances in neural information processing systems*, pages 1422–1432, 2018.

Simon S Du, Akshay Krishnamurthy, Nan Jiang, Alekh Agarwal, Miroslav Dudik, and John Langford. Provably efficient RL with rich observations via latent state decoding. *arXiv preprint arXiv:1901.09018*, 2019a.

Simon S Du, Yuping Luo, Ruosong Wang, and Hanrui Zhang. Provably efficient Q-learning with function approximation via distribution shift error checking oracle. In *Advances in Neural Information Processing Systems*, pages 8060–8070, 2019b.

Simon S Du, Sham M Kakade, Jason D Lee, Shachar Lovett, Gaurav Mahajan, Wen Sun, and Ruosong Wang. Bilinear classes: A structural framework for provable generalization in RL. *arXiv preprint arXiv:2103.10897*, 2021.

Yonathan Efroni, Dipendra Misra, Akshay Krishnamurthy, Alekh Agarwal, and John Langford. Provably filtering exogenous distractors using multistep inverse dynamics. In *International Conference on Learning Representations*, 2021.

Riashat Islam, Manan Tomar, Alex Lamb, Yonathan Efroni, Hongyu Zang, Aniket Didolkar, Dipendra Misra, Xin Li, Harm van Seijen, Remi Tachet des Combes, et al. Agent-controller representations: Principled offline rl with rich exogenous information. *arXiv preprint arXiv:2211.00164*, 2022.

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

Chi Jin, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan. Is q-learning provably efficient? *Advances in neural information processing systems*, 31, 2018.

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

Chi Jin, Qinghua Liu, and Sobhan Miryoosefi. Bellman eluder dimension: New rich classes of RL problems, and sample-efficient algorithms. *arXiv preprint arXiv:2102.00815*, 2021.

Sham Machandranath Kakade. *On the sample complexity of reinforcement learning*. University of London, University College London (United Kingdom), 2003.

Alex Lamb, Riashat Islam, Yonathan Efroni, Aniket Didolkar, Dipendra Misra, Dylan Foster, Lekan Molu, Rajan Chari, Akshay Krishnamurthy, and John Langford. Guaranteed discovery of controllable latent states with multi-step inverse models. *arXiv preprint arXiv:2207.08229*, 2022.

Zakaria Mhammedi, Dylan J Foster, Max Simchowitz, Dipendra Misra, Wen Sun, Akshay Krishnamurthy, Alexander Rakhlin, and John Langford. Learning the linear quadratic regulator from nonlinear observations. *Advances in Neural Information Processing Systems*, 33: 14532–14543, 2020.

Dipendra Misra, Mikael Henaff, Akshay Krishnamurthy, and John Langford. Kinematic state abstraction and provably efficient rich-observation reinforcement learning. In *International conference on machine learning*, pages 6961–6971. PMLR, 2020.

Aditya Modi, Jinglin Chen, Akshay Krishnamurthy, Nan Jiang, and Alekh Agarwal. Model-free representation learning and exploration in low-rank mdps. *CoRR*, abs/2102.07035, 2021.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pages 2778–2787. PMLR, 2017.

Tongzheng Ren, Chenjun Xiao, Tianjun Zhang, Na Li, Zhaoran Wang, Sujay Sanghavi, Dale Schuurmans, and Bo Dai. Latent variable representation for reinforcement learning. *arXiv preprint arXiv:2212.08765*, 2022.

Masatoshi Uehara, Xuezhou Zhang, and Wen Sun. Representation learning for online and offline RL in low-rank mdps. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*, 2022.

Sara A Van de Geer and Sara van de Geer. *Empirical Processes in M-estimation*, volume 6. Cambridge university press, 2000.

Tianjun Zhang, Tongzheng Ren, Mengjiao Yang, Joseph Gonzalez, Dale Schuurmans, and Bo Dai. Making linear mdps practical via contrastive representation learning. In *International Conference on Machine Learning*, pages 26447–26466. PMLR, 2022a.

Xuezhou Zhang, Yuda Song, Masatoshi Uehara, Mengdi Wang, Alekh Agarwal, and Wen Sun. Efficient reinforcement learning in block mdps: A model-free representation learning approach. In *International Conference on Machine Learning*, pages 26517–26547. PMLR, 2022b.

Contents of Appendix

Table 1. Comparison of sample complexity required learn an ε-optimal policy. For approaches that require a minimum reachability assumption $\eta_{\min} \coloneqq \min_{s \in \mathcal{S}} \max_{\pi \in \Pi_{\mathsf{M}}} d^{\pi}(s)$ denotes the reachability parameter. Φ and Ψ denote the decoder and model classes, respectively.

	Sample complexity	Model-free	Comp. efficient	Rate-optimal $1/\varepsilon^2$-sample comp.
`OLIVE` (Jiang et al., 2017)	$\frac{A^2 H^3 S^3 \log\lvert\Phi\rvert}{\varepsilon^2}$	Yes	No	Yes
`MOFFLE` (Modi et al., 2021)	$\frac{A^{13} H^8 S^7 \log\lvert\Phi\rvert}{(\varepsilon^2 \eta_{\min} \wedge \eta_{\min}^5)}$	Yes	Yes	No
`HOMER` (Misra et al., 2020)	$\frac{AHS^6(S^2A^3+\log\lvert\Phi\rvert)}{(\varepsilon^2 \wedge \eta_{\min}^3)}$	Yes	Yes	No
`Rep-UCB` (Uehara et al., 2022)	$\frac{A^2 H^5 S^4 \log(\lvert\Phi\rvert\lvert\Psi\rvert)}{\varepsilon^2}$	No	Yes	Yes
`BRIEE` (Zhang et al., 2022b)	$\frac{A^{14} H^9 S^8 \log\lvert\Phi\rvert}{\varepsilon^4}$	Yes	Yes	No
`MusIK` (this paper)	$\frac{A^2 H^4 S^{10}(AS^3+\log\lvert\Phi\rvert)}{\varepsilon^2}$	**Yes**	**Yes**	**Yes**

Part I

Additional Details and Results

A. Omitted Tables and Pseudocode

Algorithm 3 Execute non-Markov partial policy produced by `MusIK`.

Require: Indices $t, h \in [H]$ and $i \in [S]$ (Indexes policy $\hat{\pi}^{(i,t)} \in \Pi_{\mathsf{NM}}^{t:h-1}$ produced in Line 9 of Algorithm 2). Initial observation $x_t \in \mathcal{X}_t$. Functions $(\hat{f}^{(t)}, \hat{\phi}^{(t)}), \dots, (\hat{f}^{(h-1)}, \hat{\phi}^{(h-1)})$ produced in Line 7.

Set $\boldsymbol{j}_{t-1} = i$.
for $\tau = t, \dots, h-1$ **do**
  $(\boldsymbol{a}_\tau, \boldsymbol{j}_\tau) \leftarrow \arg\max_{(a,i)} \hat{f}^{(\tau)}((a,i) \mid \hat{\phi}^{(\tau)}(\boldsymbol{x}_\tau), \boldsymbol{j}_{\tau-1})$
  Play action $\boldsymbol{a}_\tau$ at layer τ and observe $\boldsymbol{x}_{\tau+1}$.
Return: Partial trajectory $(\boldsymbol{a}_{t:h-1}, \boldsymbol{x}_{t:h})$ generated by $\hat{\pi}^{(i,t)} \in \Pi_{\mathsf{NM}}^{t:h-1}$ (Line 9 of Algorithm 2).

B. Application to Reward-Based RL: Planning with an Approximate Cover

In this section, we show how the policy cover learned by `MusIK` can be used to optimize any downstream reward function of interest. For the results that follow, we assume that at each layer $h \in [H]$, the learner observes a reward $\boldsymbol{r}_h \in [0,1]$ in addition to the observation $\boldsymbol{x}_h \in \mathcal{X}$, so that trajectories take the form $(\boldsymbol{x}_1, \boldsymbol{a}_1, \boldsymbol{r}_1), \dots, (\boldsymbol{x}_H, \boldsymbol{a}_H, \boldsymbol{r}_H)$. We will make the following standard BMDP assumption (Misra et al., 2020; Zhang et al., 2022b), which asserts that the mean reward function depends only on the latent state, not the full observation.

Assumption B.1 (Realizability)**.** *For all $h \in [H]$, there exists $\bar{r}_h : \mathcal{S} \times \mathcal{A} \to [0,1]$ such that $\mathbb{E}[\boldsymbol{r}_h \mid \boldsymbol{x}_h = x, \boldsymbol{a}_h = a] = \bar{r}_h(\phi_\star(x), a)$.*

The PSDP algorithm. To optimize rewards, we take a somewhat standard approach and appeal to a variant of the Policy Search by Dynamic Programming (`PSDP`) algorithm of Bagnell et al. (2003); Misra et al. (2020). `PSDP` uses the approximate policy cover produced by `MusIK` as part of a dynamic programming scheme, which constructs a near-optimal policy in a layer-by-layer fashion. In particular, starting from layer H, `PSDP` first constructs a partial policy $\hat{\pi}^{(H)} \in \Pi_{\mathsf{M}}^{H:H}$ using data collected with $\Psi^{(H)}$, then moves back a layer and constructs a partial policy $\hat{\pi}^{(H-1)} \in \Pi_{\mathsf{M}}^{H-1:H}$ using data collected with $\Psi^{(H-1)}$ and $\hat{\pi}^{(H)}$, and so on, until the first layer is reached. The variant of `PSDP` we present here differs slightly from the original version in (Bagnell et al., 2003; Misra et al., 2020), with the main difference being that instead of using a policy optimization sub-routine to compute the policy for each layer, we appeal to least-squares regression (see Line 6 of Algorithm 4) to estimate a Q-function, and then select the greedy policy this function induces.

Algorithm 4 PSDP: Policy Search by Dynamic Programming (variant of Bagnell et al. (2003))

Require: Policy cover $\Psi^{(1)}, \ldots, \Psi^{(H)}$. Decoder class Φ. Number of samples n.

for $h = H, \ldots, 1$ **do**

  $\mathcal{D}^{(h)} \leftarrow \varnothing$.

  for n times **do**

    Sample $(\boldsymbol{x}_h, \boldsymbol{a}_h, \boldsymbol{r}_{h:H}) \sim \mathtt{unif}(\Psi^{(h)}) \circ_h \mathtt{unif}(\mathcal{A}) \circ_{h+1} \hat{\pi}^{(h+1)}$.

    Update dataset: $\mathcal{D}^{(h)} \leftarrow \mathcal{D}^{(h)} \cup \left\{\left(\boldsymbol{x}_h, \boldsymbol{a}_h, \sum_{t=h}^{H} \boldsymbol{r}_t\right)\right\}$.

  Solve regression:

$$(\hat{f}^{(h)}, \hat{\phi}^{(h)}) \leftarrow \operatorname*{arg\,min}_{f:[S]\times\mathcal{A}\to[0,H-h+1], \phi\in\Phi} \sum_{(x,a,R)\in\mathcal{D}} (f(\phi(x), a) - R)^2.$$

  Define $\hat{\pi}^{(h)} \in \Pi_{\mathsf{M}}^{h:H}$ via

$$\hat{\pi}^{(h)}(x) = \begin{cases} \arg\max_{a\in\mathcal{A}} \hat{f}^{(h)}(\hat{\phi}^{(h)}(x), a), & x \in \mathcal{X}_h, \\ \hat{\pi}^{(h)}(x), & x \in \mathcal{X}_t, \ \ t \in [h+1 \,..\, H]. \end{cases}$$

Return: Near-optimal policy $\hat{\pi}^{(1)} \in \Pi_{\mathsf{M}}$.

The following result, proven in Appendix I, provides the main sample complexity guarantee for PSDP.[6]

Theorem B.1. *Let $\alpha, \varepsilon, \delta \in (0,1)$ be given. Suppose that Assumptions 2.1 and B.1 hold, and that for all $h \in [H]$:*

1. *$\Psi^{(h)}$ is a (α, ϵ)-approximate cover for layer h, where $\epsilon := \varepsilon/(2SH^2)$.*
2. *$|\Psi^{(h)}| \le S$.*

Then, for appropriately chosen $n \in \mathbb{N}$, the policy $\hat{\pi}^{(1)}$ returned by Algorithm 4 satisfies

$$\mathbb{E}^{\hat{\pi}^{(1)}}\left[\sum_{h=1}^{H} \boldsymbol{r}_h\right] \ge \max_{\pi\in\Pi_{\mathsf{M}}} \mathbb{E}^{\pi}\left[\sum_{h=1}^{H} \boldsymbol{r}_h\right] - \varepsilon$$

with probability at least $1-\delta$. Furthermore, the total number of sampled trajectories used by the algorithm is bounded by

$$\widetilde{O}(1) \cdot \frac{H^5 S^6 (SA + \log(|\Phi|/\delta))}{\alpha^2 \varepsilon^2}.$$

Sample complexity to find an ε-suboptial policy with MusIK + PSDP. From Theorem B.1, to find an ε-suboptimal policy, PSDP requires an (α, ϵ)-approximate cover for all layers, where $\epsilon := \varepsilon/(2SH^2)$. Focusing only on dependence on the accuracy parameter ε, it follows from the results in Section 3.3 that MusIK can generate an $(1/4, \epsilon)$-approximate cover using $\widetilde{O}(1/\varepsilon^2)$ trajectories (see (5)). Thus, the total number of trajectories required to find an ε-suboptimal policy in reward-based RL using MusIK + PSDP scales with $\widetilde{O}(1/\varepsilon^2)$. To the best of our knowledge, this is the first computationally efficient approach that gives $\widetilde{O}(1/\varepsilon^2)$ sample complexity for reward-based reinforcement learning in BMDPs (without reachability).

C. Details for Experiments

In this section, we give the details of the experiments. We provide the full code in the supplementary material.

[6] This result does not immediately follow from prior work (Misra et al., 2020) because it allows for an (α, ε)-policy cover with $\varepsilon > 0$; previous work only handles the case where $\varepsilon = 0$.

Environment. We adopt the CombLock environment from Misra et al. (2020); Zhang et al. (2022b), which is parameterized by the horizon H and number of actions $A = 10$. At each layer $h \in [H]$, there are $N = 3$ states $s_{h,1}, s_{h,2}, s_{h,3} \in \mathcal{S}_h$, where $s_{h,1}, s_{h,2}$ are "good" states and $s_{h,3}$ is a "bad" terminal state. For each layer $h \in [H]$, there exists a pair of "good" actions $u_{h,1}, u_{h,2} \in \mathcal{A}$ such that taking action $u_{h,j}$ in state $s_{h,j}$ (for $j \in \{1,2\}$) leads to one of the good states $\{s_{h+1,1}, s_{h+1,2}\}$ at next layer with equal probability. All actions $a_h \notin \{u_{h,1}, u_{h,2}\}$ lead to the bad state $s_{h+1,3}$ deterministically. The sequences of good actions $u_{1:H,1}$ and $u_{1:H,2}$ are sampled uniformly at random from the set of actions $\mathcal{A}$ when the environment is initialized and are unknown to the learner.

For $h = H$, the agent receives a reward of 1 if action $u_{H,j}$ is taken in state $s_{H,j}$ (for $j \in \{1,2\}$), and receives reward of 0 otherwise. For $h < H$, the agent receives an anti-shaped reward of 0.1 for choosing any action $a_h \neq u_{h,j}$ in state $s_{h,j}$, for $j \in \{1,2\}$, and receives a reward of 0 otherwise (in particular, the agent never receives a reward in the bad state $s_{h,3}$). This anti-shaped reward encourages the agent to take actions that lead to the bad state $s_{h+1,3}$, from which it is not possible to reach the good states $\{s_{H,1}, s_{H,2}\}$ at layer H and achieve the optimal reward of 1.

The agent does not observe the states $\{\boldsymbol{s}_h\}$ directly, and instead receives observations $\{\boldsymbol{x}_h\}$. For each h, the observation $\boldsymbol{x}_h$ is a d-dimensional vector, where $d := 2^{\lceil \log_2(H+N+1) \rceil}$, obtained by concatenating the one-hot vector of the latent state $\boldsymbol{s}_h$ and the one hot vector of the layer index h, followed by adding noise sampled from $\mathcal{N}(0, 0.1)$ in one dimension, padding with zeros if necessary, and multiplying with a Hadamard matrix. Strictly speaking, the CombLock environment is more challenging than a Block MDP, since two latent states can emit the same observation due to the addition of the Gaussian noise in the observation process.

Since the good actions $\{u_{h,1}, u_{h,2}\}_{h \in [H]}$ are not known to the agent, deliberate exploration is required to learn a policy that that maximizes the reward function (note it is only possible achieve reward 1 if the agent selects a good action for all $h \in [H]$). For example, when the horizon is set to $H = 100$, the probability of finding the optimal policy through naive uniform exploration is 10^{-100}. In addition, representation learning is required to recover the latent state $\boldsymbol{s}_h$ from the observation $\boldsymbol{x}_h$ at each layer, with the best decoder depending on the layer h.

Implementation of `MusIK`. We use `MusIK` to learn a policy cover that we then use in `PSDP` to find a near-optimal policy in the CombLock environment. In this environment, the optimal policy cover can be learned by composing optimal policy covers at each layer (though this is not true in general, many problems share this property). We follow an approach taken with `HOMER` in (Misra et al., 2020), and take advantage of this composability property to implement a more sample-efficient version of `MusIK`, where during the call to the `IKDP` subroutine at layer h, we only learn $\hat{f}^{(h-1)}, \hat{\phi}^{(h-1)}$ (i.e. the `IKDP` for-loop stops at $t = h - 1$); this is exactly what was done in (Misra et al., 2020). This version of `MusIK`, which we name `MusIK.comp`, is displayed in Algorithm 5 (we write the full algorithm without a reference to an external (`IKDP`) subroutine).

We use $\Phi := \{\phi_W : x \mapsto \arg\max_{i \in [N]} [Wx]_i \mid W \in \mathbb{R}^{N \times d}\}$ for the decoder class, where we recall that N is the number of latent states per layer in the CombLock environment—this is exactly the same decoder class as the one used by (Misra et al., 2020) for `HOMER`. Given the observation process in the CombLock environment, there exists a matrix $W_\star \in \mathbb{R}^{N \times d}$ such that the true decoder $\phi_\star$ is given by $\phi_{W_\star}$. To learn $W_\star$, we use the set of differentiable maps $\Phi' := \{x \mapsto \mathrm{softmax}(Wx) \mid W \in \mathbb{R}^{N \times d}\}$ during training (this is reflected in the objective in the next display). Further, we make a slight (empirically-motivated) modification to the conditional density estimation problem in Line 7 of `IKDP`, where we instead solve

$$\hat{f}^{(h-1)}, \hat{\psi}^{(h-1)} \leftarrow \operatorname*{arg\,max}_{f : \mathcal{X} \times [N] \to \Delta(\mathcal{A}), \psi \in \Phi'} \sum_{(a_h, x_{h-1}, x_h) \in \mathcal{D}^{(h-1)}} \log \left(\sum_{i=1}^{N} f(a_{h-1} \mid x_{h-1}, i) \cdot [\psi(x_h)]_i \right). \tag{4}$$

Compared to the original objective of `IKDP` in Line 7 of Algorithm 2, we no longer need to predict the index i_{h-1} of the future roll-out policies (since the for-loop of `IKDP` now stops at $t = h - 1$, there are no future roll-outs). Another difference is that we do not use a decoder at layer $h - 1$; we use $f(a \mid x, j)$ instead of $f(a \mid \phi(x), j)$ (this helps with the training). For each $j \in [N]$, we instantiate $f(\cdot \mid \cdot, j)$ with a two-layer neural network with $\tanh$ activation, input dimension d and output dimension A, where the output is pushed through a softmax so that $f(\cdot \mid x, j)$ is a distribution over actions for any $x \in \mathcal{X}$. We use `Adam` for the optimization problem in (6). We specify the choices of hyperparameters in the sequel.

With $\hat{\psi}^{(h-1)}$ as in (6), the learned decoder is given by $\hat{\phi}^{(h-1)}(x) := \arg\max_{i \in [N]} [\hat{\psi}^{(h-1)}(x)]_i$. Further, for $\hat{f}^{(h-1)}$ as in (6), the hth layer policy cover $\Psi^{(h)} = \{\hat{\pi}^{(j,h)}\}_{j \in [N]}$ constructed by `MusIK.comp` is essentially given by:

$$\hat{\pi}^{(j,h)} = \hat{\pi} \circ_{h-1} \hat{a}^{(j,h-1)}, \quad \text{where} \quad \hat{\pi} \in \operatorname*{arg\,max}_{\pi \in \Psi^{(h-1)}} \mathbb{P}^{\pi \circ_{h-1} \pi_{\mathsf{unif}}} \left[\hat{\phi}^{(h-1)}(\boldsymbol{x}_h) = j \right], \tag{5}$$

and $\hat{a}^{(j,h-1)}(x) \coloneqq \arg\max_{a \in \mathcal{A}} \hat{f}^{(h-1)}(a \mid x, j)$. That is, the policy $\hat{\pi}^{(j,h)}$ is the composition of the best partial policy $\hat{\pi}$ among the partial policies in $\Psi^{(h-1)}$ (the policy cover at the previous layer) and the best action at layer $h-1$ to maximize to probability of reaching the 'abstract state' $j \in [N]$. Technically, computing $\hat{\pi}$ requires estimating $\mathbb{P}^{\pi \circ_{h-1} \pi_{\mathsf{unif}}}\left[\hat{\phi}^{(h-1)}(\boldsymbol{x}_h) = j\right]$, for all $\pi \in \Psi^{(h-1)}$. For this, we reuse the dataset $\mathcal{D}^{(h-1)}$ from (8) and solve another conditional density estimation problem—see (9) in Algorithm 5[7].

PSDP implementation. The only modification we make to the PSDP algorithm is that we use $f(a \mid x)$ instead of $f(a \mid \phi(x))$ in the objective (6) (i.e. we do not use a decoder). We instantiate $f(\cdot \mid \cdot)$ with a two-layer neural network with input dimension d, hidden dimension of 400, and output dimension of 1. We use the $\tanh$ activation function at all layers.

Hyper-parameters. For each $j \in [N]$, we instantiate $f(\cdot \mid \cdot, j)$ in (8) with a two-layer neural network with $\tanh$ activation, input dimension d, hidden dimension of size N_{hidden}, and output dimension A, where the output is run through the softmax activation function (with temperature 1) so that $f(\cdot \mid x, j)$ is a distribution over actions for any $x \in \mathcal{X}$. We also instantiate $g(\cdot \mid \cdot)$ in (9) with a two-layer neural network with $\tanh$ activation. input dimension N, hidden dimension of size 400, and output dimension N, where the output is pushed through a softmax (with temperature 1) so that $g(\cdot \mid j)$ is a distribution over $[N]$ for any $j \in [N]$. For the choice of hidden size N_{hidden}, we searched over the grid $\{100, 200, 400\}$. The results reported in Figure 1 are for $N_{\mathsf{hidden}} = 400$.

We optimize the parameters of (f, θ) [resp. g] in (8) [resp. (9)] using Adam with the default parameters in PyTorch. We use a batch size of $\min(n, N_{\mathsf{batch}})$, where n is as in Algorithm 5, and perform N_{update} gradient updates. For the batch size N_{batch} and number of updates N_{updates}, we searched over the girds $\{512, 1024, 2048, 4096, 8196\}$ and $\{64, 128, 256\}$, respectively. The results reported in Figure 1 are for $N_{\mathsf{batch}} = 8196$ and $N_{\mathsf{updates}} = 128$.

Baselines. As baselines, we use HOMER (Misra et al., 2020) and BRIEE (Zhang et al., 2022b). Amongst provably efficient algorithms, these methods are known to have the best empirical performance (Zhang et al., 2022b) on the CombLock environment. The HOMER algorithm has the same structure as MusIK: it first learns a policy cover, then uses the cover within PSDP to learn a near-optimal policy. The BRIEE algorithm does not explicitly learn a policy cover, but rather interleaves exploration and exploitation using optimism. We do not reproduce BRIEE and HOMER, and instead report the results from Zhang et al. (2022b).

[7] Technically, the solution of the conditional estimation problem in (9) does not yield an estimator of $\mathbb{P}^{\pi \circ_{h-1} \pi_{\mathsf{unif}}}\left[\hat{\phi}^{(h-1)}(\boldsymbol{x}_h) = j\right]$ per se. But it gives us a proxy for a function whose argmax $\hat{\pi}$ in (7).

Algorithm 5 MusIK.comp: Variant of MusIK for composable policy covers (version of MusIK used in the experiments).

Require:

- Dimension of the observation space d.
- Number of latent states per layer N.
- Number of samples n.

Set $\Psi^{(1)} = \{\pi_{\mathsf{unif}}, \dots, \pi_{\mathsf{unif}}\}$ with $|\Psi^{(1)}| = N$.

for $h = 2, \dots, H$ **do**

  $\mathcal{D}^{(h)} \leftarrow \varnothing$.

  Let $\iota^{(h-1)} : \Psi^{(h-1)} \to [N]$ be any one-to-one mapping.

  /* Collect data by rolling in with policy cover */

  for n times **do**

    Sample $\hat{\pi} \sim \mathsf{unif}(\Psi^{(h-1)})$.

    Sample $(\boldsymbol{x}_{h-1}, \boldsymbol{a}_{h-1}, \boldsymbol{x}_h) \sim \hat{\pi} \circ_{h-1} \pi_{\mathsf{unif}}$.

    $\mathcal{D}^{(h-1)} \leftarrow \mathcal{D}^{(h-1)} \cup \{(\iota^{(h-1)}(\hat{\pi}), \boldsymbol{a}_{h-1}, \boldsymbol{x}_{h-1}, \boldsymbol{x}_h)\}$.

  /* Inverse kinematics */

  For $\Phi' \coloneqq \{x \mapsto \mathrm{softmax}(Wx) \mid W \in \mathbb{R}^{N \times d}\}$, solve

$$\hat{f}^{(h-1)}, \hat{\psi}^{(h-1)} \leftarrow \operatorname*{arg\,max}_{f:\mathcal{X}\times[N]\to\Delta(\mathcal{A}),\psi\in\Phi'} \sum_{(-,a,x,x')\in\mathcal{D}^{(h-1)}} \log\left(\sum_{j\in[N]} f(a \mid x, j) \cdot [\psi(x')]_j\right). \tag{6}$$

  /* Inverse Kinematics to learn associations between policies at subsequent layers */

  Solve

$$\hat{g}^{(h-1)} \leftarrow \operatorname*{arg\,max}_{g:[N]\to\Delta([N])} \sum_{(i,-,-,x')\in\mathcal{D}^{(h-1)}} \log g\left(i \,\middle|\, \operatorname*{arg\,max}_{j\in[N]} [\hat{\psi}^{(h-1)}(x')]_j\right). \tag{7}$$

  /* Update partial policy cover */

  For each $j \in [S]$, define

$$\hat{a}^{(j,h-1)}(x) = \operatorname*{arg\,max}_{a\in\mathcal{A}} \hat{f}^{(h-1)}(a \mid x, j), \quad x \in \mathcal{X}_t.$$
$$\hat{\iota}^{(j,h-1)}(x) = \operatorname*{arg\,max}_{i\in[N]} \hat{g}^{(h-1)}(i \mid j).$$

  For each $j \in [S]$, define $\hat{\pi}^{(j,h)} \in \Pi_{\mathsf{M}}^{1:h-1}$ via

$$\hat{\pi}^{(j,h)}(x_\tau) \coloneqq \begin{cases} \hat{a}^{(j,h-1)}(x_\tau), & \tau = h-1, \\ \hat{\pi}^{(\hat{\iota}^{(j,h-1)},h-1)}(x_\tau), & \tau \in [h-2], \end{cases} \qquad x_\tau \in \mathcal{X}_\tau.$$

  Define $\Psi^{(h)} = \{\hat{\pi}^{(j,h)} : j \in [N]\}$ /* Policy cover for layer h. */

Return: Policy covers $\Psi^{(1)}, \dots, \Psi^{(H)}$.

Part II

Analysis

D. Organization

Part II of the appendix contains the proof of our main result, Theorem 3.2, as well as other proofs. This section is organized as follows.

- First, in Appendix E, we give an informal overview of the analysis of Theorem 3.2, using the tools introduced in Section 4 as a starting point. In particular:
  - Appendix E.1 introduces and analyzes a simplified version of MusIK intended for tabular reinforcement learning as a warm-up exercise.
  - Appendix E.2 builds on this development to showcase the main ideas behind the proof of Theorem 3.2.
- Appendix F provides proofs for the structural results introduced in Section 4.
- Appendix G contains proofs for the tabular warm-up exercise in Appendix E
- Appendix H contains the proof of our main result, Theorem 3.2. For background on the key ideas, we recommend reading the overview in Appendix E.
- Appendix I contains proofs for the extensions to reward-based RL in Appendix B.

E. Overview of Analysis

In this section, we give an overview of the analysis of our main result, Theorem 3.2, with the full proof deferred to Appendix H. First, in Appendix E.1 we show how to analyze a simplified version of MusIK for the *tabular* setting in which the state $\boldsymbol{s}_h$ is directly observed. Then, in Appendix E.2, we build on these developments to give a proof sketch for the full Block MDP setting.

E.1. Warm-Up: Multi-Step Inverse Kinematics for Tabular MDPs

Algorithm 6 MusIK.Tab: Multi-Step Inverse Kinematics (tabular variant)

Require: Number of samples n.

1: Set $\Psi^{(1)} = \varnothing$.
2: **for** $h = 2 \dots, H$ **do**
3: Let $\Psi^{(h)} = \texttt{IKDP.Tab}(\Psi^{(1)}, \dots, \Psi^{(h-1)}, n)$. // Algorithm 7.
4: **Return:** Policy covers $\Psi^{(1)}, \dots, \Psi^{(H)}$.

In this section, we use the extended BMDP, truncated policy class, and relevant structural results introduced in prequel to analyze a simplified version of MusIK for the *tabular* setting in which the state $\boldsymbol{s}_h$ is directly observed (a special case of the BMDP in which $\mathcal{X} = \mathcal{S}$ and $\boldsymbol{x}_h = \boldsymbol{s}_h$ almost surely). The tabular setting preserves the most important challenges in removing reachability, and will serve as a useful warm-up exercise for the full BMDP setting. Our analysis will also give a taste for how the multi-step inverse kinematics objective in IKDP (Line 7) allows one to approximately implement dynamic programming.

MusIK and IKDP for tabular MDPs. Algorithm 6 (MusIK.Tab) and Algorithm 7 (IKDP.Tab) are simplified variants of MusIK and IKDP tailored to the tabular setting. MusIK.Tab is identical to MusIK, except that the subroutine IKDP is replaced by IKDP.Tab. IKDP.Tab has the same structure as IKDP, but does not require access to a decoder class Φ, since the states are observed directly. The algorithm takes advantage of a slightly simplified multi-step inverse kinematics objective (Line 7 of Algorithm 7) which involves directly predicting actions based on the latent states. Recall that for iteration $t \in [h-1]$,

the full version of IKDP uses observations to predict *pairs* $(\boldsymbol{a}_t, \boldsymbol{i}_t)$, where $\boldsymbol{a}_t$ is the action played at layer t and $\boldsymbol{i}_t \in [S]$ is the (random) index of the partial policy executed after layer t. IKDP.Tab does not require randomizing over the index $\boldsymbol{i}_t$, and instead solves a separate regression problem for each state $i \in [S]$ (representing the state being targeted at layer h), predicting only the action $\boldsymbol{a}_t$; we will highlight the need for the randomization over indices $\boldsymbol{i}_t$ when we return to the BMDP setting in the sequel (Appendix E.2).

The following theorem, an analogue of Theorem 3.2 for tabular MDPs, provides the main guarantee for MusIK.Tab.

Theorem E.1 (Main theorem for MusIK.Tab)**.** *Let $\varepsilon, \delta \in (0,1)$ be given, and let $n \geq 1$ be chosen such that*

$$n \geq \frac{cA^2S^6H^2\left(S^2A\log n + \log(SH^2/\delta)\right)}{\varepsilon^2}, \tag{8}$$

for some absolute constant $c > 0$ independent of all problem parameters. Then, with probability at least $1-\delta$, the collections $\Psi^{(1)}, \dots, \Psi^{(H)}$ produced by MusIK.Tab *are $(1/4, \varepsilon)$-policy covers for layers 1 through H.*

Analysis by induction. To prove Theorem E.1, we proceed by induction over the layers $h = 1, \dots, H$. Leveraging the extended MDP and truncated policy class, we will show that for each layer $h \in [H]$, if the collections $\Psi^{(1)}, \dots, \Psi^{(h-1)}$ produced by IKDP.Tab have the property that

$$\Psi^{(1)}, \dots, \Psi^{(h-1)} \text{ are } (1/2, \epsilon)\text{-policy covers relative to } \overline{\Pi}_\epsilon \text{ in } \overline{\mathcal{M}} \text{ for layers 1 through } h-1, \tag{9}$$

then with high probability, the collection Ψ_h produced by $\texttt{IKDP.Tab}(\Psi_{1:h-1}, n)$ will be a $(1/2, \epsilon)$-policy cover relative to $\overline{\Pi}_\epsilon$ in $\overline{\mathcal{M}}$ for layer h. Formally, we will prove the following result.

Theorem E.2 (Main theorem for IKDP.Tab)**.** *Let $\epsilon, \delta \in (0,1)$ and $h \in [H]$ be given and define $\varepsilon_{\texttt{stat}}(n, \delta') := \sqrt{n^{-1}(S^2A\log n + \log(1/\delta'))}$. Assume that:*

1. IKDP.Tab *is invoked with $\Psi^{(1)}, \dots, \Psi^{(h-1)}$ satisfying Eq. (11).*
2. *The policies in $\Psi^{(1)}, \dots, \Psi^{(h-1)}$ never take the terminal action* $\mathfrak{a}$.
3. *The parameter n is chosen such that $8AS^2HC \cdot \varepsilon_{\texttt{stat}}(n, \frac{\delta}{SH^2}) \leq \epsilon$ for some absolute constant $C > 0$ independent of all problem parameters.*

Then, with probability at least $1 - \frac{\delta}{H}$, the collection $\Psi^{(h)}$ produced by $\texttt{IKDP.Tab}(\Psi^{(1)}, \dots, \Psi^{(h-1)}, n)$ is an $(1/2, \epsilon)$-policy cover relative to $\overline{\Pi}_\epsilon$ in $\overline{\mathcal{M}}$ for layer h. In addition, $\Psi^{(h)} \subseteq \Pi_{\mathsf{M}}^{1:h-1}$.

With this result in hand, the proof of Theorem E.1 follows swiftly.

Proof of Theorem E.1. Let $\delta, \varepsilon \in (0,1)$ be given and let $\epsilon := \varepsilon/(2S)$. Let $\varepsilon_{\texttt{stat}}(\cdot,\cdot)$ and C be as in Theorem E.2; here C is an absolute constant independent of all problem parameters. Let $\mathcal{E}_h$ denote the event that IKDP.Tab succeeds as in Theorem E.2 for layer $h \in [H]$ with parameters δ and ϵ, and define $\mathcal{E} := \bigcap_{h \in [H]} \mathcal{E}_h$. Observe that by Theorem E.2 and the union bound, we have $\mathbb{P}[\mathcal{E}] \geq 1 - \delta$. For n large enough such that $8AS^2HC \cdot \varepsilon_{\texttt{stat}}(n, \frac{\delta}{SH^2}) \leq \epsilon$ (which is implied by the condition on n in the theorem's statement for $c = 2^5C$), Theorem E.2 implies that under $\mathcal{E}$, the output $\Psi^{(1)}, \dots, \Psi^{(H)}$ of MusIK are $(1/2, \epsilon)$-policy covers relative to $\overline{\Pi}_\epsilon$ in $\overline{\mathcal{M}}$ for layers 1 to H, respectively. We conclude by appealing to Lemma 4.3, which now implies that $\Psi^{(1)}, \dots, \Psi^{(H)}$ are $(1/4, \varepsilon)$-policy covers relative to Π_{M} in $\mathcal{M}$.

We now compute the total number of trajectories used by the algorithm. Recall that when invoked with parameter $n \in \mathbb{N}$, MusIK.Tab invokes $H-1$ instances of IKDP.Tab, each with parameter n. Each instance of IKDP.Tab uses n trajectories for each layer $t \in [h-1]$ and $i \in [S]$ (see Lines 1 and 3 of Algorithm 7), so the total number of trajectories used by MusIK.Tab is at most

$$\widetilde{O}(1) \cdot \frac{A^2S^7H^4\left(S^2A + \log(|\Phi|SH^2/\delta)\right)}{\varepsilon^2}.$$

□

Algorithm 7 `IKDP.Tab` : Inverse Kinematics for Dynamic Programming (tabular variant)

Require:

- Approximate covers $\Psi^{(1)}, \dots, \Psi^{(h-1)}$ for layers 1 to $h-1$, where $\Psi^{(t)} \subseteq \Pi_{\mathsf{M}}^{1:t-1}$.
- Number of samples n.

for $t = h-1, \dots, 1$ **do**

  $\mathcal{D}^{(t)} \leftarrow \varnothing$.

  `/* Collect data by rolling in with policy cover and rolling out with partial policy */`

  for $i \in [S]$ **do**

    for n times **do**

      Sample $(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_h) \sim \mathtt{unif}(\Psi^{(t)}) \circ_t \pi_{\mathtt{unif}} \circ_{t+1} \hat{\pi}^{(i,t+1)}$.

      $\mathcal{D}^{(t)} \leftarrow \mathcal{D}^{(t)} \cup \{(\boldsymbol{a}_t, \boldsymbol{s}_t, \boldsymbol{s}_h)\}$.

  `/* Inverse kinematics */`

$$\hat{f}^{(i,t)} \in \operatorname{argmax}_{f:[S]^2 \to \Delta_A} \sum_{(a,s,s') \in \mathcal{D}^{(t)}} \log f(a \mid s, s'). \tag{10}$$

  `/* Update partial policy cover */`

  Define $\hat{a}^{(i,t)}(s) \in \arg\max_{a \in \mathcal{A}} \hat{f}^{(i,t)}(a \mid s, i)$.

  Define $\hat{\pi}^{(i,t)} \in \Pi_{\mathsf{M}}^{t:h-1}$ via

$$\hat{\pi}^{(i,t)}(s) := \begin{cases} \hat{a}^{(i,t)}(s), & s \in \mathcal{S}_t, \\ \hat{\pi}^{(i,\tau)}(s), & s \in \mathcal{S}_\tau, \ \ \tau \in [t+1 \,..\, h-1]. \end{cases} \tag{11}$$

Return: Policy cover $\Psi^{(h)} = \{\hat{\pi}^{(i,1)} : i \in [S]\} \subseteq \Pi_{\mathsf{M}}^{1:h-1}$ for layer h.

E.1.1. Proof Sketch for Theorem E.2

We now sketch the proof of Theorem E.2. The most important feature of the proof is that the guarantee on which we induct, Eq. (11), is stated with respect to the extended MDP and truncated policy class. We work in the extended MDP throughout the proof, and only pass back to the original MDP $\mathcal{M}$ and full policy class Π_{M} in the proof of Theorem E.1 (see above) *once the induction is completed*.

Let $h \in [H]$ and $\epsilon > 0$ be fixed, and assume that Eq. (11) holds (that is, $\Psi^{(1)}, \dots, \Psi^{(h-1)}$ are $(1/2, \epsilon)$-policy covers relative to $\overline{\Pi}_\epsilon$ in $\overline{\mathcal{M}}$ for layers 1 through $h-1$). We will prove that the collection $\Psi^{(h)}$ produced by $\mathtt{IKDP.Tab}(\Psi^{(1)}, \dots, \Psi^{(h-1)}, n)$ is an $(1/2, \epsilon)$-policy cover relative to $\overline{\Pi}_\epsilon$ in $\overline{\mathcal{M}}$ for layer h. We first argue that proving Theorem E.2 reduces to showing the following lemma. To state the result, recall that $\mathcal{S}_{h,\epsilon} := \left\{s \in \mathcal{S}_h : \max_{\pi \in \overline{\Pi}_\epsilon} \bar{d}^\pi(s) \geq \epsilon\right\}$ is the set of states that are ϵ-reachable by $\overline{\Pi}_\epsilon$ in $\overline{\mathcal{M}}$.

Lemma E.1. *Assuming points 1. and 2. in Theorem E.2 hold, and if n is chosen large enough such that $8AS^2HC \cdot \varepsilon_{\mathtt{stat}}(n, \frac{\delta}{SH^2}) \leq \epsilon$ for some absolute constant $C > 0$ independent of all problem parameters, then for all $t \in [h-1]$, with probability at least $1 - \delta/H^2$, the learned partial policies $\left\{\hat{\pi}^{(i,t)}\right\}_{i \in [S]}$ in `IKDP.Tab` have the property that for all $i \in \mathcal{S}_{h,\epsilon}$,*

$$\bar{d}^{\pi_\star^{(i)} \circ_{t+1} \hat{\pi}^{(i,t+1)}}(i) - \bar{d}^{\pi_\star^{(i)} \circ_t \hat{\pi}^{(i,t)}}(i) \leq \frac{\epsilon}{2H}, \quad \textit{where} \quad \pi_\star^{(i)} \in \arg\max_{\pi \in \overline{\Pi}_\epsilon} \bar{d}^\pi(i). \tag{12}$$

For each $i \in \mathcal{S}_{h,\epsilon}$, $\pi_\star^{(i)}$ in Eq. (14) denotes the policy in the truncated class $\overline{\Pi}_\epsilon$ that maximizes the probability of visiting i at layer h. Informally, Eq. (14) states that if we execute $\pi_\star^{(i)}$ up to layer $t-1$ (inclusive), then switch to the learned partial policy $\hat{\pi}^{(i,t)}$ for the remaining steps (i.e. execute $\pi_\star^{(i)} \circ_t \hat{\pi}^{(i,t)}$), then the probability of reaching state i in layer h is close to what is achieved by running $\pi_\star^{(i)} \circ_{t+1} \hat{\pi}^{(i,t)}$. In other words, $\hat{\pi}^{(i,t)}$ is near-optimal in an average-case sense. We now show

that Theorem E.2 follows from Lemma E.1.

Proof of Theorem E.2. For $t \in [h-1]$, let $\mathcal{E}_t$ denote the success event of Lemma E.1. Let us condition on the event $\mathcal{E} := \bigcap_{t\in[h-1]} \mathcal{E}_t$. Fix $\mathfrak{i} \in \mathcal{S}_{h,\epsilon}$. Summing the left-hand side of Eq. (14) over $t = 1, \dots, h-1$ for $i = \mathfrak{i}$ and telescoping, we have that

$$\bar{d}^{\hat{\pi}^{(\mathfrak{i},1)}}(\mathfrak{i}) \geq \max_{\pi\in\overline{\Pi}_\epsilon} \bar{d}^{\pi}(\mathfrak{i}) - \frac{\epsilon}{2} \geq \frac{1}{2}\max_{\pi\in\overline{\Pi}_\epsilon} \bar{d}^{\pi}(\mathfrak{i}), \tag{13}$$

where the last inequality follows by the fact that $\max_{\pi\in\overline{\Pi}_\epsilon} \bar{d}^{\pi}(\mathfrak{i}) \geq \epsilon$ (since $\mathfrak{i} \in \mathcal{S}_{h,\epsilon}$). Since this conclusion holds uniformly for all $\mathfrak{i} \in \mathcal{S}_{h,\epsilon}$, we have that under the event $\mathcal{E}$, the output $\Psi^{(h)} := \{\hat{\pi}^{(i,1)} : i \in [S]\}$ of Algorithm 7 is a $(1/2, \epsilon)$-policy cover relative to $\overline{\Pi}_\epsilon$ for layer h. Finally, by a union bound, we have $\mathbb{P}[\mathcal{E}] \geq 1 - \mathbb{P}[\mathcal{E}^c] \geq 1 - \sum_{t\in[h-1]}\sum_{i\in[S]} \mathbb{P}[(\mathcal{E}_t^{(i)})^c] \geq 1 - \delta/H$, which completes the proof. □

Remark E.1. *It is also possible to derive Eq. (15) from Lemma E.1 using the performance difference lemma (Kakade, 2003) with a specific state-action value function; this perspective will be useful when we generalize our analysis from the tabular to the BMDP setting. To see how the performance difference lemma can be applied to obtain Eq. (15), fix $i \in [S]$ and consider the* state-action value function *(Q-function) at layer t with respect to the partial policy $\hat{\pi}^{(i,t)} \in \Pi_{\mathsf{M}}^{t:h-1}$ for the MDP $\overline{\mathcal{M}}$ with rewards $r_\tau^{(i)}(s) = \mathbf{1}\{s = i\} \cdot \mathbf{1}\{\tau = h\}$, for $\tau \in [h]$; that is,*

$$Q_t^{\hat{\pi}^{(i,t)}}(s, a; i) = r_t^{(i)}(s) + \overline{\mathbb{E}}^{\hat{\pi}^{(i,t)}}\left[\sum_{\tau=t+1}^{h} r_\tau^{(i)}(\boldsymbol{s}_\tau) \;\middle|\; \boldsymbol{s}_t = s, \boldsymbol{a}_t = a\right]. \tag{14}$$

Thanks to the choice of reward functions, we have

$$Q_t^{\hat{\pi}^{(i,t)}}(s, a; i) = \overline{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{s}_h = i \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a], \tag{15}$$

and thus

$$\bar{d}^{\hat{\pi}^{(i,1)}}(i) - \bar{d}^{\pi_\star^{(i)}}(i) = \mathbb{E}\left[Q_1^{\hat{\pi}^{(i,t)}}(\boldsymbol{s}_1, \hat{\pi}^{(i,t)}(\boldsymbol{s}_1); i) - Q_1^{\pi_\star^{(i)}}(\boldsymbol{s}_1, \pi_\star^{(i)}(\boldsymbol{s}_1); i)\right]. \tag{16}$$

Thus, by the performance difference lemma, the right-hand side of (18) can be bounded by

$$\sum_{t=1}^{h-1} \overline{\mathbb{E}}^{\pi_\star^{(i)}}\left[Q_t^{\hat{\pi}^{(i,t)}}(\boldsymbol{s}_t, \pi_\star^{(i)}(\boldsymbol{s}_t); i) - Q_t^{\hat{\pi}^{(i,t)}}(\boldsymbol{s}_t, \hat{\pi}^{(i,t)}(\boldsymbol{s}_t); i)\right]. \tag{17}$$

Thanks to Eq. (17), the quantity in (19) is simply $\sum_{t=1}^{h-1}\left(\bar{d}^{\pi_\star^{(i)}\circ_{t+1}\hat{\pi}^{(i,t+1)}}(i) - \bar{d}^{\pi_\star^{(i)}\circ_t\hat{\pi}^{(i,t)}}(i)\right)$, which can directly be bounded using Lemma E.1 to arrive at the conclusion in Eq. (15).

It remains to prove Lemma E.1. To prove the result, we first use the multi-step inverse kinematics objective to establish a certain "local" optimality guarantee. We combine this with the assumption that $\Psi^{(1)}, \dots, \Psi^{(h-1)}$ are policy covers, along with certain structural properties of the extended MDP $\overline{\mathcal{M}}$, to conclude the result.

A local optimality guarantee from multi-step inverse kinematics. Fix $1 \leq t < h$ and a state $i \in \mathcal{S}_{h,\epsilon}$, and let $\{\hat{\pi}^{(i,t+1)} : i \in [S]\}$ be the partial policies constructed by `IKDP.Tab` at layer $t+1$. As the first step toward constructing the policy $\hat{\pi}^{(i,t)}$, `IKDP.Tab` computes an estimator $\hat{f}^{(i,t)} : [S]^2 \to \Delta(\mathcal{A})$ by solving the multi-step inverse kinematics objective in Line 7. This entails predicting the probability of the action $\boldsymbol{a}_t$ conditioned on the states $\boldsymbol{s}_t$ and $\boldsymbol{s}_h$, under the process $(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_h) \sim \mathbb{P}^{\mathsf{unif}(\Psi^{(t)})\circ_t\pi_{\mathsf{unif}}\circ_{t+1}\hat{\pi}^{(i,t+1)}}$.[8] The following result gives a generalization guarantee for $\hat{f}^{(i,t)}$ under this process.

Lemma E.2 (Conditional density estimation guarantee)**.** *Fix $t \in [h-1]$. Let $n \geq 1$ and $\delta \in (0,1)$ be given, and define $\varepsilon_{\mathsf{stat}}(n,\delta) := n^{-1/2} \cdot \sqrt{S^2 A \log n + \log(1/\delta)}$. Assume that the policies in $\Psi^{(t)}$ never take the terminal action $\mathfrak{a}$. Then, there exists an absolute constant $C > 0$ (independent of t, h, and other problem parameters) such that for all $i \in [S]$ the solution $\hat{f}^{(i,t)}$ to the conditional density estimation problem in Line 7 of Algorithm 7 has that with probability at least $1-\delta$,*

$$\overline{\mathbb{E}}^{\mathsf{unif}(\Psi^{(t)})\circ_t\pi_{\mathsf{unif}}\circ_{t+1}\hat{\pi}^{(i,t+1)}}\left[\sum_{a\in\mathcal{A}}\left(\hat{f}^{(i,t)}(a \mid \boldsymbol{s}_t, \boldsymbol{s}_h) - P_{\mathsf{bayes}}^{(i,t)}(a \mid \boldsymbol{s}_t, \boldsymbol{s}_h)\right)^2\right] \leq C^2 \cdot \varepsilon_{\mathsf{stat}}^2(n,\delta), \tag{18}$$

[8] Note that π_{unif} denotes the policy that samples $\boldsymbol{a}_t$ uniformly from $\mathcal{A}$, not $\bar{\mathcal{A}}$.

where

$$P^{(i,t)}_{\mathsf{bayes}}(a \mid s, s') := \frac{\bar{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a]}{Z^{(i,t)}(s,s')}, \text{ for } Z^{(i,t)}(s,s') := \sum_{a' \in \mathcal{A}} \bar{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a']. \quad (19)$$

Lemma E.2 is a consequence of a standard generalization bound for conditional density estimation. The *Bayes-optimal regression function* $P^{(i,t)}_{\mathsf{bayes}}$ represents the true conditional probability for $\boldsymbol{a}_t$ under the process $(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_h) \sim \mathbb{P}^{\mathsf{unif}(\Psi^{(t)}) \circ_t \pi_{\mathsf{unif}} \circ_{t+1} \hat{\pi}^{(i,t+1)}}$. This quantity is useful as a proxy for another quantity we refer to as *forward kinematics*:

$$P^{(t)}_{\mathsf{FK}}(i \mid s, a) := \bar{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{s}_h = i \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a]. \quad (20)$$

The utility of forward kinematics is somewhat more immediate: It represents the probability that we reach state i at layer h if we start from $\boldsymbol{s}_t = s$, take action $\boldsymbol{a}_t = a$, and then roll out with $\hat{\pi}^{(i,t+1)}$; equivalently $P^{(t)}_{\mathsf{FK}}(i \mid s, a)$ is the Q-function for the reward function $\mathbb{I}\{\boldsymbol{s}_h = i\}$—see Eq. (17). Hence, by the principle of dynamic programming, it is natural to choose

$$\hat{\pi}^{(i,t)}(s) = \arg\max_{a \in \mathcal{A}} P^{(t)}_{\mathsf{FK}}(i \mid s, a). \quad (21)$$

`IKDP.Tab` does not directly compute the forward kinematics, and hence cannot directly define $\hat{\pi}^{(i,t)}$ based on Eq. (23). Instead, we compute

$$\hat{\pi}^{(i,t)}(s) = \arg\max_{a \in \mathcal{A}} P^{(i,t)}_{\mathsf{bayes}}(a \mid s, i). \quad (22)$$

To see that this is equivalent, observe that $P^{(i,t)}_{\mathsf{bayes}}(a \mid s, i)$ is a ratio of two quantities: The numerator is exactly $P^{(t)}_{\mathsf{FK}}(i \mid s, a)$, and the denominator is a "constant" whose value does not depend on a. With some manipulation, we can use this fact to relate suboptimality with respect to $P^{(t)}_{\mathsf{FK}}$ to the regression error in Eq. (20), leading to the following "local" optimality guarantee for $\hat{\pi}^{(i,t)}$ (see Appendix G.2 for a proof).

Lemma E.3 (Local optimality guarantee)**.** *Consider the setting of Theorem E.2 and let $t \in [h-1]$. Then, there is an event $\mathcal{E}_t$ of probability at least $1 - \delta/H^2$ under which the learned partial policies $\{\hat{\pi}^{(i,t)}\}_{i \in [S]}$ and $\{\hat{\pi}^{(i,t+1)}\}_{i \in [S]}$ in `IKDP.Tab` have the property that for all $i \in \mathcal{S}_h$,*

$$\sum_{\pi \in \Psi^{(t)}} \bar{d}^{\pi}(s_t) \left(\max_{a \in \mathcal{A}} P^{(t)}_{\mathsf{FK}}(i \mid s_t, a) - P^{(t)}_{\mathsf{FK}}(i \mid s_t, \hat{\pi}^{(i,t)}(s_t)) \right) \le 2SAC\varepsilon_{\mathtt{stat}}(n, \delta/(SH^2)), \quad \forall s_t \in \mathcal{S}_t, \quad (23)$$

where $\varepsilon_{\mathtt{stat}}(\cdot,\cdot)$ and $C > 0$ are as in Lemma E.2; here $C > 0$ is an absolute constant independent of problem parameters.

Remark E.2. *For the tabular setting where $\boldsymbol{s}_h$ is observed, it is also possible to estimate the function $P^{(t)}_{\mathsf{FK}}(i \mid s, a)$ directly. However, in the BDMP setting, estimating forward kinematics is* not possible *because states are not observed. We will see that in spite of this, the multi-step inverse kinematics objective used in `IKDP` still serves as a useful proxy for the forward kinematics.*

We now use Lemma E.3 to prove Lemma E.1.

Proof of Lemma E.1. To prove Lemma E.1, we translate the local suboptimality guarantee in Eq. (25) to the global guarantee in Eq. (14). Fix $\mathfrak{i} \in \mathcal{S}_{h,\epsilon}$ and let us abbreviate $\varepsilon'_{\mathtt{stat}} \equiv C \cdot \varepsilon_{\mathtt{stat}}(n, \delta/(SH^2))$, where $\varepsilon_{\mathtt{stat}}(\cdot,\cdot)$ and $C > 0$ are as in Lemma E.2. Condition on the event $\mathcal{E}_t$ of Lemma E.3. We begin by writing the left-hand side of Eq. (14) in a form that is closer to the left-hand side of Eq. (25):

$$\bar{d}^{\pi^{(\mathfrak{i})}_\star \circ_{t+1} \hat{\pi}^{(\mathfrak{i},t+1)}}(\mathfrak{i}) - \bar{d}^{\pi^{(\mathfrak{i})}_\star \circ_t \hat{\pi}^{(\mathfrak{i},t)}}(\mathfrak{i}) = \sum_{s \in \mathfrak{S}_t \cup \{\mathfrak{t}_t\}} \bar{d}^{\pi^{(\mathfrak{i})}_\star}(s) \cdot \left(P^{(t)}_{\mathsf{FK}}(\mathfrak{i} \mid s, \pi^{(\mathfrak{i})}_\star(s)) - P^{(t)}_{\mathsf{FK}}(\mathfrak{i} \mid s, \hat{\pi}^{(\mathfrak{i},t)}(s)) \right), \quad (24)$$

where we use the convention that $\hat{\pi}^{(\mathfrak{i},t)}(\mathfrak{t}_t) = \mathfrak{a}$; this equality follows by the definition of $P^{(t)}_{\mathsf{FK}}$ in Eq. (22). Now, we bound the right-hand side of Eq. (26) in terms of the left-hand side of Eq. (25) by using that $\Psi^{(t)}$ is a relative policy cover. In particular, since $\Psi^{(t)}$ is an $(1/2, \epsilon)$-policy cover relative to $\overline{\Pi}_\epsilon$ at layer t, and since $\pi^{(\mathfrak{i})}_\star \in \overline{\Pi}_\epsilon$, we have that for all $s_t \in \mathcal{S}_{t,\epsilon}$,

$$\bar{d}^{\pi^{(\mathfrak{i})}_\star}(s_t) \left(P^{(t)}_{\mathsf{FK}}(\mathfrak{i} \mid s_t, \pi^{(\mathfrak{i})}_\star(s_t)) - P^{(t)}_{\mathsf{FK}}(\mathfrak{i} \mid s_t, \hat{\pi}^{(\mathfrak{i},t)}(s_t)) \right)$$

$$
\begin{aligned}
&\leq \bar{d}^{\pi^{(\mathrm{i})}_\star}(s_t)\left(\max_{a\in\bar{\mathfrak{A}}} P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,a) - P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,\hat{\pi}^{(\mathrm{i},t)}(s_t))\right),\\
&= \bar{d}^{\pi^{(\mathrm{i})}_\star}(s_t)\left(\max_{a\in\mathcal{A}} P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,a) - P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,\hat{\pi}^{(\mathrm{i},t)}(s_t))\right), \qquad (25)\\
&\leq 2\sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t)\left(\max_{a\in\bar{\mathfrak{A}}} P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,a) - P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,\hat{\pi}^{(\mathrm{i},t)}(s_t))\right),\\
&\leq 4SA\varepsilon'_{\mathtt{stat}}, \qquad (26)
\end{aligned}
$$

where Eq. (27) follows from the fact $P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,\mathfrak{a}) = 0$ (since $\mathfrak{a}$ is the action leading to the terminal state $\mathfrak{t}_{t+1}$ from any state at layer t), so that $\max_{a\in\bar{\mathfrak{A}}} P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,a) = \max_{a\in\mathcal{A}} P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,a)$; Eq. (28) follows from Eq. (25) in Lemma E.3. On the other hand, by Lemma 4.1, we have that for all $s_t\in\mathcal{S}_t\setminus\mathcal{S}_{t,\epsilon}$, $\pi^{(\mathrm{i})}_\star(s_t) = \mathfrak{a}$. Therefore,

$$
\forall s_t\in\mathfrak{S}_t\setminus\mathfrak{S}_{t,\epsilon},\quad P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,\pi^{(\mathrm{i})}_\star(s_t)) = P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,\mathfrak{a}) = 0. \qquad (27)
$$

Using this together with the fact that $P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,\hat{\pi}^{(\mathrm{i},t)}(s_t))\geq 0$ and Eq. (28) implies that

$$
\forall s_t\in\mathfrak{S}_t,\quad \bar{d}^{\pi^{(\mathrm{i})}_\star}(s_t)\left(P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,\pi^{(\mathrm{i})}_\star(s_t)) - P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s_t,\hat{\pi}^{(\mathrm{i},t)}(s_t))\right)\leq 4SA\varepsilon'_{\mathtt{stat}}. \qquad (28)
$$

Now, by choosing n large enough such that $8HS^2AC\varepsilon_{\mathtt{stat}}(n,\frac{\delta}{SH^2})\leq\epsilon$ (as in the lemma's statement), we have $8HS^2A\varepsilon'_{\mathtt{stat}}\leq\epsilon$ by definition of $\varepsilon'_{\mathtt{stat}}$. Using this and summing (30) over $s_t\in\mathcal{S}_t$ in (30) we have that

$$
\sum_{s\in\mathfrak{S}_t\cup\{\mathfrak{t}_t\}} \bar{d}^{\pi^{(\mathrm{i})}_\star}(s)\cdot\left(P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s,\pi^{(\mathrm{i})}_\star(s)) - P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid s,\hat{\pi}^{(\mathrm{i},t)}(s))\right)\leq\frac{\epsilon}{2H}, \qquad (29)
$$

where we have used that $P^{(t)}_{\mathsf{FK}}(\mathfrak{i}\mid\mathfrak{t}_t,\cdot) = 0$. □

E.2. From Tabular MDPs to Block MDPs

We now give an overview of the proof of Theorem 3.2. The proof builds on the techniques in Appendix E.1 and follows the same structure, but requires non-trivial changes to accommodate the general BMDP setting. We highlight the most important similarities and differences below, with the full proof deferred to Appendix H.

Recall that on the algorithmic side, the main change in moving from the tabular setting to the general BMDP setting is that the latent states $\boldsymbol{s}_h$ are unobserved. To address this, the multi-step inverse kinematics objective in `IKDP` (Line 7) differs from the simplified version in `IKDP.Tab` by incorporating estimation of a decoder $\hat{\phi}^{(t)}\in\Phi$ at each step $t\in[h-1]$. Here, a critical property of the multi-step inverse kinematics objective is that the Bayes-optimal regression function (the BMDP analogue of Eq. (21)) only depends on the observations $\boldsymbol{x}_t$ and $\boldsymbol{x}_h$ through $\phi_\star(\boldsymbol{x}_t)$ and $\phi_\star(\boldsymbol{x}_h)$, which ensures that the conditional density estimation problem in Line 7 is always well-specified.

The need for non-Markovian policies. `IKDP` also differs from `IKDP.Tab` in how we construct the partial policy collection $\{\hat{\pi}^{(i,t)}\}_{i\in[S]}$ for layer $t\in[h-1]$ from the collection $\{\hat{\pi}^{(i,t+1)}\}_{i\in[S]}$ learned at layer $t+1$. The construction in Line 9 of `IKDP`, as discussed in Section 3.2, leads to policies that are *non-Markovian* (that is, history-dependent). This complicates the analysis because we cannot appeal to the performance difference lemma in the same fashion Appendix E.1 (see Remark E.1), where it was used to relate the local suboptimality for each policy to global suboptimality. Before giving an overview for how we overcome this challenge, we first give a more detailed explanation as to *why* `IKDP` builds non-Markovian policies.

Fix $h\in[H]$. Recall that in the tabular setting, for each backward step $t\in[h-1]$, each partial policy $\hat{\pi}^{(i,t)}$ constructed in `IKDP.Tab` is designed to target the state $i\in\mathcal{S}_h$. In the BMDP setting, the states $\boldsymbol{s}_h$ are unobserved, and it is no longer the case that the partial policy $\hat{\pi}^{(i,t)}\in\Pi^{t:h-1}_{\mathsf{NM}}$ constructed in `IKDP` targets the state $i\in\mathcal{S}_h$. Indeed, while we will show that each partial policy $\hat{\pi}^{(i,t)}$ (approximately) targets *some* state in $\mathcal{S}_h$, the algorithm has no way of knowing which one.[9] An additional challenge, which motivates the composition rule in Line 9 of `IKDP`, is that for each $i\in[S]$, the suffix policy $\hat{\pi}^{(i,t+1)}$ and the one-step policy $\hat{a}^{(i,t)}$ learned in Line 8 may target *different latent states*, so it does not suffice to simply

[9] Unless additional assumptions are added, the latent representation may only be learned up to an unknown permutation.

construct $\hat{\pi}^{(i,t)}$ by composing them. This motivates the second key difference between the multi-step inverse kinematics objectives used in IKDP versus IKDP.Tab. The objective in IKDP predicts both actions and *indices of roll-out policies* (instead of just actions, as in the tabular case) in order to learn to associate partial policies at successive layers. In particular, recall that Line 8 of IKDP defines

$$(\hat{a}^{(i,t)}(x), \hat{\imath}^{(i,t)}(x)) = \arg\max_{(a,j)} \hat{f}^{(t)}((a,j) \mid \hat{\phi}^{(t)}(x), i), \quad x \in \mathcal{X}_t.$$

As described in Section 3.2, one should interpret $j = \hat{\imath}^{(i,t)}(x)$ as the *most likely* (or most closely associated) roll-out policy $\hat{\pi}^{(j,t+1)}$ given that the (decoded) latent state at layer h is $i \in [S]$ and $x \in \mathcal{X}_t$ is the current observation at layer t. With this in mind, the composition rule in Line 9 constructs $\hat{\pi}^{(i,t)}$ via

$$\hat{\pi}^{(i,t)}(x_{t:\tau}) \coloneqq \begin{cases} \hat{a}^{(i,t)}(x_t), & \tau = t, \ \ x_t \in \mathcal{X}_t, \\ \hat{\pi}^{(\hat{\imath}^{(i,t)}(x_t),t+1)}(x_{t+1:\tau}), & \tau \in [t+1\,..\,h-1], \ \ x_{t:\tau} \in \mathcal{X}_t \times \cdots \times \mathcal{X}_\tau. \end{cases}$$

For layers $t+1, \ldots, h-1$, this construction follows the policy $\hat{\pi}^{(\hat{\imath}^{(i,t)}(x_t),t+1)}$ which—per the discussion above—is most associated with the decoded state $i \in [S]$. At layer t, we select $a_t = \hat{a}^{(i,t)}(x_t)$, which maximizes the probability of reaching the decoded latent state $i \in [S]$ when we roll-out with $\hat{\pi}^{(\hat{\imath}^{(i,t)}(x_t),t+1)}$. The construction, while intuitive, is non-Markovian, since for layers $t+1$ and onward the policy depends on x_t through $\hat{\imath}^{(i,t)}(x_t)$.

Analysis by induction. The proof of Theorem 3.2 follows the same high-level structure as Theorem E.1 (MusIK.Tab), and we use the same induction strategy: For each layer $h \in [H]$, we assume that $\Psi^{(1)}, \ldots, \Psi^{(h-1)}$ are approximate policy covers relative to $\overline{\Pi}_\epsilon$ for $\overline{\mathcal{M}}$, then show that the collection $\Psi^{(h)}$ produced by $\texttt{IKDP}(\Psi^{(1)}, \ldots, \Psi^{(h-1)}, \Phi, n)$ is an approximate cover with high probability whenever this holds. As with the tabular setting, a key component in our proof is to work with the extended BMDP and truncated policy class throughout the induction, and only pass back to the original BMDP at the end.

The following result (proven in Appendix H.2) is our main theorem concerning the performance of IKDP, and serves as the BMDP analogue of Theorem E.2.

Theorem E.3 (Main Theorem for IKDP)**.** *Let $\epsilon, \delta \in (0,1)$ and $h \in [H]$ be given, and define $\varepsilon_{\texttt{stat}}(n,\delta) \coloneqq n^{-1/2}\sqrt{S^3 A \log n + \log(|\Phi|/\delta)}$. Assume that:*

1. IKDP *is invoked with $\Psi^{(1)}, \ldots, \Psi^{(h-1)}$ satisfying Eq. (11).*
2. *The policies in $\Psi^{(1)}, \ldots, \Psi^{(h-1)}$ never take the terminal action $\mathfrak{a}$.*
3. *The parameter n is chosen such that $8AS^4HC\varepsilon_{\texttt{stat}}(n, \frac{\delta}{H^2}) \le \epsilon$, for some absolute constant $C > 0$ independent of h and other problem parameters.*

Then, with probability at least $1 - \frac{\delta}{H}$, the collection $\Psi^{(h)}$ produced by $\texttt{IKDP}(\Psi^{(1)}, \ldots, \Psi^{(h-1)}, \Phi, n)$ is an $(1/2, \epsilon)$-policy cover relative to $\overline{\Pi}_\epsilon$ in $\overline{\mathcal{M}}$ for layer h. In addition, $\Psi^{(h)} \subseteq \Pi_{\texttt{NM}}^{1:h-1}$.

We close the section by highlighting some key differences between the proof of this result and its tabular counterpart (Theorem E.2).

An alternative to Lemma E.1. Recall that in the tabular setting, the proof of Theorem E.2 relied on Lemma E.1 and the performance difference lemma (see Remark E.1). In the BMDP setting, Lemma E.1 does not necessarily hold since, unlike in the tabular setting, successive partial policies $\hat{\pi}^{(i,t)} \in \Pi_{\texttt{NM}}^{t:h-1}$ and $\hat{\pi}^{(i,t+1)} \in \Pi_{\texttt{NM}}^{t+1:h-1}$ may target different states at layer h despite sharing the same index $i \in [S]$. For this reason, we use a modified version of Lemma E.1, together with a generalized version of the performance difference lemma.

Lemma E.4 (BMDP counterpart to Lemma E.1)**.** *There is an absolute constant $C > 0$ such that for all $t \in [h-1]$, with probability at least $1 - \delta/H^2$, the learned partial policies $\{\hat{\pi}^{(i,t)}\}_{i\in[S]}$ and $\{\hat{\pi}^{(i,t+1)}\}_{i\in[S]}$ in* IKDP *have the property that for all $s_h \in \mathcal{S}_{h,\epsilon}$, there exists $\mathfrak{i} \in [S]$ such that*

$$0 \le \sum_{\pi \in \Psi^{(t)}} \bar{d}^{\pi}(s_t) \overline{\mathbb{E}}_{\boldsymbol{x}_t \sim q(\cdot|s_t)} \left[\max_{a \in \mathcal{A}, j \in [S]} Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t, a; s_h) - V_t^{\hat{\pi}^{(\cdot,t)}}(\boldsymbol{x}_t; s_h) \right] \le 2S^3AC\varepsilon_{\texttt{stat}}(n, \tfrac{\delta}{H^2}), \quad \forall s_t \in \mathcal{S}_t, \tag{30}$$

where $Q_t^{\hat{\pi}^{(j,t+1)}}(x_t, a; s_h) \coloneqq \mathbb{P}^{\hat{\pi}^{(j,t)}}[\boldsymbol{s}_h = s_h \mid \boldsymbol{x}_t = x_t, \boldsymbol{a}_t = a]$, $V_t^{\hat{\pi}^{(i,t)}}(x_t; s_h) \coloneqq Q_t^{\hat{\pi}^{(i,t+1)}}(x_t, \hat{\pi}^{(i,t)}(x_t); s_h)$, *and* $\varepsilon_{\mathtt{stat}}(n, \delta') \coloneqq n^{-1/2}\sqrt{S^3 A \log n + \log(|\Phi|/\delta')}$.

This result is proven in Appendix H.3.3. To see the similarity between Lemma E.4 and Lemma E.1, note that the main quantity that the latter bounds (i.e. the quantity on the right-hand side of Eq. (14)) can also be written as a difference between Q; see Remark E.1. Once Lemma E.4 is established, it can be shown to imply Theorem E.3 using a generalized variant of the performance difference lemma (Lemma H.5).

Establishing Eq. (32) using multi-step inverse kinematics. To show that Eq. (32) holds, we use the structure of the multi-step inverse kinematics objective in Line 7 of IKDP, as well as the non-Markov policy construction outlined in the prequel. In particular, we show that the multi-step inverse kinematics objective acts as a proxy for the forward kinematics given by

$$\mathbb{P}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{s}_h = \phi_\star(x_h) \mid \boldsymbol{s}_t = \phi_\star(x_t), \boldsymbol{a}_t = a],$$

for $i \in [S]$, $x_t \in \mathcal{X}_t$ and $x_h \in \mathcal{X}_h$. We use this to show that up to statistical error, the partial policies $\{\hat{\pi}^{(i,t)}\}_{i\in[S]}$ constructed from $\{\hat{\pi}^{(i,t+1)}\}_{i\in[S]}$ i) identify (using observations at layer t) the best action at layer t, and ii) identify the best partial policy from $\{\hat{\pi}^{(j,t+1)}\}_{j\in[S]}$ to switch to from layer $t+1$ onwards.

Beyond the multi-step inverse kinematics objective and non-Markov policy construction, the proof of Theorem E.3 uses the extended BMDP in a similar fashion to the tabular setting. We make use of the fact that for each layer $t \in [h-1]$, the policies in $\overline{\Pi}_\epsilon$ always play the terminal action $\mathfrak{a}$ on observations emitted from states in $\mathcal{S}_{t,\epsilon}$, and the generalized performance difference lemma (Lemma H.5) is specifically designed to take advantage of this. This allows us to "write off" these states (analogous to Eq. (29) in the proof of Lemma E.1), and use the policy cover property for $\Psi^{(1)}, \dots, \Psi^{(h-1)}$ to control the error for states in $\mathcal{S}_{t,\epsilon}$; see Appendix H.3.4 for details. However, there is some added complexity stemming from the non-Markovian nature of $\{\hat{\pi}^{(i,t)}\}_{i\in[S]}$.

F. Proofs for Structural Results for Extended BMDP

In this section, we prove the main structural results concerning the extended BMDP and truncated policy class introduced in **??**. We first recall the definition of the truncated policy class. For $\epsilon \in (0,1)$, let $\overline{\Pi}_{0,\epsilon}, \dots, \overline{\Pi}_{H,\epsilon}$ be the policies defined recursively as follows: $\overline{\Pi}_{0,\epsilon} = \overline{\Pi}_{\mathsf{M}}$ and for all $t \in [H]$, $\pi \in \overline{\Pi}_{t,\epsilon}$ if and only if there exists $\pi' \in \overline{\Pi}_{t-1,\epsilon}$ such that for all $h \in [H]$, $s \in \overline{\mathcal{S}}_h$, and $x \in \phi_\star^{-1}(s)$,

$$\pi(x) \coloneqq \begin{cases} \mathfrak{a}, & \text{if } h = t \text{ and } \max_{\tilde{\pi}\in\overline{\Pi}_{t-1,\epsilon}} \bar{d}^{\tilde{\pi}}(s) < \epsilon, \\ \pi'(x), & \text{otherwise.} \end{cases} \tag{31}$$

Finally, we let $\overline{\Pi}_\epsilon \coloneqq \overline{\Pi}_{H,\epsilon}$.

The proofs in this section make use of the following lemma.

Lemma F.1. *For all $h \in [H]$, it holds that*

$$\forall s \in \mathcal{S}_h, \quad \max_{\pi\in\overline{\Pi}_{h-1,\epsilon}} \bar{d}^{\pi}(s) = \max_{\pi\in\overline{\Pi}_{\epsilon}} \bar{d}^{\pi}(s). \tag{32}$$

Proof of Lemma F.1. We will show that for all $t \in [h\,..\,H]$,

$$\forall s \in \mathcal{S}_h, \quad \max_{\pi\in\overline{\Pi}_{t-1,\epsilon}} \bar{d}(s) = \max_{\pi\in\overline{\Pi}_{t,\epsilon}} \bar{d}(s). \tag{33}$$

This implies Eq. (34) by summing both sides of Eq. (35) over $t = h, \dots, H$, telescoping, and using that $\overline{\Pi}_\epsilon = \overline{\Pi}_{H,\epsilon}$. To prove the result, let $t \in [h\,..\,H]$, $s \in \mathcal{S}_h$, and $\tilde{\pi} \in \arg\max_{\pi'\in\overline{\Pi}_{t-1,\epsilon}} \bar{d}^{\pi'}(s)$. Further, let $\pi \in \overline{\Pi}_{t,\epsilon}$ be as in Eq. (33) with $\pi' = \tilde{\pi}$. In this case, by Eq. (33), we have $\tilde{\pi}(x) = \pi(x)$, for all $x \in \phi^{-1}(s')$, $s' \in \mathcal{S}_\tau$, and $\tau \le [t-1]$. Using this and the fact that $s \in \mathcal{S}_h$ and $t \ge h$, we have

$$\max_{\breve{\pi}\in\overline{\Pi}_{t-1,\epsilon}} \bar{d}^{\breve{\pi}}(s) = \bar{d}^{\tilde{\pi}}(s) = \bar{d}^{\pi}(s) \le \max_{\breve{\pi}\in\overline{\Pi}_{t,\epsilon}} \bar{d}^{\breve{\pi}}(s).$$

We now show the inequality in the other direction. Let $t \in [h\,..\,H]$, $s \in \mathcal{S}_h$, and $\tilde{\pi} \in \arg\max_{\breve{\pi} \in \overline{\Pi}_{t,\epsilon}} \bar{d}^{\breve{\pi}}(s)$. Further, let $\pi' \in \overline{\Pi}_{t-1,\epsilon}$ be as in Eq. (33) for $\pi = \tilde{\pi}$. In this case, by Eq. (33), we have $\tilde{\pi}(x) = \pi'(x)$, for all $x \in \phi^{-1}(s')$, $s' \in \mathcal{S}_\tau$, and $\tau \in [t-1]$. Using this and the fact that $s \in \mathcal{S}_h$ and $t \geq h$, we have

$$\max_{\breve{\pi} \in \overline{\Pi}_{t,\epsilon}} \bar{d}^{\breve{\pi}}(s) = \bar{d}^{\tilde{\pi}}(s) = \bar{d}^{\pi'}(s) \leq \max_{\breve{\pi} \in \overline{\Pi}_{t-1,\epsilon}} \bar{d}^{\breve{\pi}}(s).$$

This shows Eq. (35) and completes the proof. □

F.1. Proof of Lemma 4.1

Proof of Lemma 4.1. Fix $h \in [H]$. We proceed by induction on t to show that

$$\forall t \in [h\,..\,H], \forall \pi \in \overline{\Pi}_{t,\epsilon}, \exists \pi' \in \overline{\Pi}_{h,\epsilon} : \forall s \in \mathcal{S}_h, \forall x \in \phi_\star^{-1}(s), \quad \pi(x) = \pi'(x). \tag{34}$$

For $t = h$, Eq. (36) holds trivially. Now, we suppose that Eq. (36) holds for $t \in [h\,..\,H-1]$, and show that it holds for $t+1$. By definition of $\overline{\Pi}_\epsilon^{(t+1)}$ (Eq. (33)), there exists $\tilde{\pi} \in \overline{\Pi}_{t,\epsilon}$ such that

$$\forall \tau \in [t], \forall s \in \mathcal{S}_\tau, \forall x \in \phi_\star^{-1}(s), \quad \pi(x) = \tilde{\pi}(x). \tag{35}$$

Now, by the induction hypothesis, there exists $\pi' \in \overline{\Pi}_{h,\epsilon}$ such that $\tilde{\pi}(x) = \pi'(x)$, for all $x \in \phi_\star^{-1}(s)$ and $s \in \mathcal{S}_h$. Combining this with Eq. (37) and the fact that $t \geq h$ implies that Eq. (36) holds for $t+1$, which concludes the induction.

Now, by instantiating Eq. (36) with $t = H$ and recalling that $\overline{\Pi}_\epsilon = \overline{\Pi}_\epsilon^{(H)}$ (by definition), we get that

$$\forall \pi \in \overline{\Pi}_\epsilon, \exists \pi' \in \overline{\Pi}_{h,\epsilon} : \forall s \in \mathcal{S}_h, \forall x \in \phi^{-1}(s), \quad \pi(x) = \pi'(x). \tag{36}$$

By Lemma F.1, this implies that for any $\mathfrak{s} \in \mathcal{S}_h \setminus \mathcal{S}_{h,\epsilon}$, $\max_{\pi \in \overline{\Pi}_\epsilon^{(h-1)}} \bar{d}^{\pi}(\mathfrak{s}) \leq \epsilon$. It follows that for all $\pi' \in \overline{\Pi}_\epsilon^{(h)}$ and $x \in \phi_\star^{-1}(\mathfrak{s})$, we have $\pi'(x) = \mathfrak{a}$, by definition of $\overline{\Pi}_{h,\epsilon}$; see Eq. (33). This together with Eq. (38) implies that $\pi(x) = \mathfrak{a}$ for all $x \in \phi_\star^{-1}(\mathfrak{s})$ and $\pi \in \overline{\Pi}_\epsilon$, as desired. □

F.2. Proof of Lemma 4.2 (Approximation for Truncated Policy Class)

Proof of Lemma 4.2. We will show that for all $t \in [H]$, $h \in [H]$, and $s \in \mathcal{S}_h$,

$$\max_{\pi \in \overline{\Pi}_{t-1,\epsilon}} \bar{d}^{\pi}(s) \leq \max_{\pi \in \overline{\Pi}_{t,\epsilon}} \bar{d}^{\pi}(s) + |\mathcal{S}_t|\epsilon. \tag{37}$$

With this established, summing Eq. (39) over t, telescoping, and using that $\sum_{t \in [H]} |\mathcal{S}_t| = S$ implies the desired result.

Let $t \in [H]$, $h \in [H]$ and $s \in \mathcal{S}_h$. Further, let $\tilde{\pi} \in \arg\max_{\pi \in \overline{\Pi}_{t-1,\epsilon}} \bar{d}^{\pi}(s)$ and let $\pi \in \overline{\Pi}_{t,\epsilon}$ be as in Eq. (33) for $\pi' = \tilde{\pi}$. First, suppose that $h \leq t$. Then, $\bar{d}^{\pi}(s) = \bar{d}^{\tilde{\pi}}(s)$ since $\pi|_{\mathcal{X}_\tau} \equiv \tilde{\pi}|_{\mathcal{X}_\tau}$ for all $\tau < t$, and so by our choice of $\tilde{\pi}$ and that $\pi \in \overline{\Pi}_{t,\epsilon}$, we have

$$\max_{\breve{\pi} \in \overline{\Pi}_{t-1,\epsilon}} \bar{d}^{\breve{\pi}}(s) = \bar{d}^{\tilde{\pi}}(s) = \bar{d}^{\pi}(s) \leq \max_{\breve{\pi} \in \overline{\Pi}_{t,\epsilon}} \bar{d}^{\breve{\pi}}(s).$$

Now suppose that $h > t$. We will use that I) $\pi|_{\mathcal{X}_\tau} = \tilde{\pi}|_{\mathcal{X}_\tau}$, for all $\tau \neq t$; and II) $\pi(x) = \tilde{\pi}(x)$, for all $x \in \phi_\star^{-1}(s')$ and $s' \in \mathcal{S}_{t,\epsilon}$, by definition of $\mathcal{S}_{t,\epsilon}$ and π, and Lemma F.1. We note that I) implies that $\bar{d}^{\tilde{\pi}}(s') = \bar{d}^{\pi}(s')$, for all $s' \in \mathcal{S}_t$, and the combination of I) and II) implies that $\overline{\mathbb{P}}^{\pi}[\boldsymbol{s}_h = s \mid \boldsymbol{s}_t = s'] = \overline{\mathbb{P}}^{\tilde{\pi}}[\boldsymbol{s}_h = s \mid \boldsymbol{s}_t = s']$, for all $s' \in \mathcal{S}_{t,\epsilon}$. Using these facts, we have

$$\begin{aligned}
\bar{d}^{\tilde{\pi}}(s) &= \sum_{s' \in \mathcal{S}_t} \overline{\mathbb{P}}^{\tilde{\pi}}[\boldsymbol{s}_h = s \mid \boldsymbol{s}_t = s'] \cdot \bar{d}^{\tilde{\pi}}(s'), \\
&= \sum_{s' \in \mathcal{S}_{t,\epsilon}} \overline{\mathbb{P}}^{\tilde{\pi}}[\boldsymbol{s}_h = s \mid \boldsymbol{s}_t = s'] \cdot \bar{d}^{\pi}(s') + \sum_{s' \in \mathcal{S}_t \setminus \mathcal{S}_{t,\epsilon}} \overline{\mathbb{P}}^{\tilde{\pi}}[\boldsymbol{s}_h = s \mid \boldsymbol{s}_t = s'] \cdot \bar{d}^{\tilde{\pi}}(s'),
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{s' \in \mathcal{S}_{t,\epsilon}} \bar{\mathbb{P}}^{\pi}[\boldsymbol{s}_h = s \mid \boldsymbol{s}_t = s'] \cdot \bar{d}^{\pi}(s') + \sum_{s' \in \mathcal{S}_t \setminus \mathcal{S}_{t,\epsilon}} \bar{\mathbb{P}}^{\tilde{\pi}}[\boldsymbol{s}_h = s \mid \boldsymbol{s}_t = s'] \cdot \bar{d}^{\tilde{\pi}}(s'), \\
&\leq \bar{d}^{\pi}(s) + \sum_{s' \in \mathcal{S}_t \setminus \mathcal{S}_{t,\epsilon}} \bar{\mathbb{P}}^{\tilde{\pi}}[\boldsymbol{s}_h = s \mid \boldsymbol{s}_t = s'] \cdot \bar{d}^{\tilde{\pi}}(s'), \\
&\leq \bar{d}^{\pi}(s) + |\mathcal{S}_t|\epsilon, \qquad (38)
\end{aligned}
$$

where the last inequality follows because $\bar{d}^{\tilde{\pi}}(s') \leq \epsilon$ for all $s' \in \mathcal{S}_t \setminus \mathcal{S}_{t,\epsilon}$, which follows from the definition of $\mathcal{S}_{t,\epsilon}$, Lemma F.1, and $\tilde{\pi} \in \overline{\Pi}_{t-1,\epsilon}$. Combining Eq. (40) with the fact that $\tilde{\pi} \in \arg\max_{\pi \in \overline{\Pi}_{t-1,\epsilon}} \bar{d}^{\pi}(s)$ and $\pi \in \overline{\Pi}_{t,\epsilon}$, we have that

$$
\max_{\breve{\pi} \in \overline{\Pi}_{t-1,\epsilon}} \bar{d}^{\breve{\pi}}(s) = \bar{d}^{\tilde{\pi}}(s) \leq \bar{d}^{\pi}(s) + |\mathcal{S}_t|\epsilon \leq \max_{\breve{\pi} \in \overline{\Pi}_{t,\epsilon}} \bar{d}^{\breve{\pi}}(s) + |\mathcal{S}_t|\epsilon.
$$

□

F.3. Proof of Lemma 4.3

Proof of Lemma 4.3. Let $\overline{\Pi}_{\mathtt{M}}$ be as in Lemma 4.2, and note that since $\Pi_{\mathtt{M}} \subseteq \overline{\Pi}_{\mathtt{M}}$, we have for any $s \in \mathcal{S}$,

$$
\max_{\pi \in \Pi_{\mathtt{M}}} d^{\pi}(s) \leq \max_{\pi \in \overline{\Pi}_{\mathtt{M}}} \bar{d}^{\pi}(s) \leq \max_{\pi \in \overline{\Pi}_{\epsilon}} \bar{d}^{\pi}(s) + S\epsilon. \qquad (39)
$$

where the last inequality follows by Lemma 4.2. Now, fix $s \in \mathcal{S}$ such that $\max_{\pi \in \Pi_{\mathtt{M}}} d^{\pi}(s) \geq \varepsilon$. Using Eq. (41) and that $\epsilon = \varepsilon/(2S)$, we have $\max_{\pi \in \overline{\Pi}_{\epsilon}} \bar{d}^{\pi}(s) \geq \varepsilon/2 \geq \epsilon$. Thus, since Ψ is a $(1/2, \epsilon)$-policy cover relative to $\overline{\Pi}_{\epsilon}$ in $\overline{\mathcal{M}}$ for all layers, there exists $\pi^{(s)} \in \Psi$ such that

$$
\frac{1}{2} \max_{\pi \in \overline{\Pi}_{\epsilon}} \bar{d}^{\pi}(s) \leq \bar{d}^{\pi^{(s)}}(s) = d^{\pi^{(s)}}(s),
$$

where the equality follows from the assumption that policies in Ψ never take the terminal action $\mathfrak{a}$. Combining this inequality with Eq. (41), we have that

$$
\max_{\pi \in \Pi_{\mathtt{M}}} d^{\pi}(s) \leq 2 d^{\pi^{(s)}}(s) + S\epsilon.
$$

Since $S\epsilon = \frac{\varepsilon}{2} \leq \frac{1}{2} \max_{\pi \in \Pi_{\mathtt{M}}} d^{\pi}(s)$, rearranging gives

$$
\frac{1}{4} \max_{\pi \in \Pi_{\mathtt{M}}} d^{\pi}(s) \leq d^{\pi^{(s)}}(s),
$$

which concludes the proof. □

Remark F.1. *The proof of Lemma 4.3 actually gives a result slightly stronger than what is stated in the lemma. Namely, it suffices for Ψ to be a $(1/2, \varepsilon/2)$-policy cover relative to $\overline{\Pi}_{\epsilon}$ in $\overline{\mathcal{M}}$ (as opposed to a $(1/2, \epsilon)$-policy cover). We state the weaker result because our analysis of Algorithm 2 does not take advantage of the stronger result.*

G. Proofs for Tabular MDPs

G.1. Proof of Lemma E.2 (MLE Guarantee for Tabular MDPs)

To prove a guarantee for the minimizer $\hat{f}^{(j,t)}$ for the conditional density estimation problem in Line 7 of Algorithm 7, we first derive the expression of the Bayes-optimal solution of this problem.

Lemma G.1. *Let $h \in [H]$, $t \in [h-1]$, $i \in [S]$, and consider the Bayes-optimal solution $P^{(i,t)}_{\mathsf{bayes}}$ of the problem in Line 7 of Algorithm 7; that is,*

$$
P^{(i,t)}_{\mathsf{bayes}} \in \operatorname*{arg\,max}_{P : \mathcal{S}_t \times \mathcal{S}_h \to \Delta(\mathcal{A} \times [S])} \mathbb{E}^{\mathtt{unif}(\Psi^{(t)}) \circ_t \pi_{\mathtt{unif}} \circ_{t+1} \hat{\pi}^{(i,t+1)}} \left[\log P(\boldsymbol{a}_t \mid \boldsymbol{s}_t, \boldsymbol{s}_h) \right]. \qquad (40)
$$

Then, for any $a \in \mathcal{A}$, $s \in \mathcal{S}_t$, and $s' \in \mathcal{S}_h$, $P^{(i,t)}_{\mathsf{bayes}}$ satisfies

$$P^{(i,t)}_{\mathsf{bayes}}(a \mid s, s') \coloneqq \frac{\overline{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a\right]}{\sum_{a' \in \mathcal{A}} \overline{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a'\right]}.$$

Proof of Lemma G.1. Fix $a \in \mathcal{A}$ and $(s, s') \in \mathcal{S}_t \times \mathcal{S}_h$. The solution $P^{(i,t)}_{\mathsf{bayes}}$ of the problem in Eq. (42) satisfies

$$\begin{aligned} P^{(i,t)}_{\mathsf{bayes}}(a \mid s, s') &= \mathbb{P}^{\pi_{\mathsf{unif}} \circ_{t+1} \hat{\pi}^{(i,t+1)}}\left[\boldsymbol{a}_t = a \mid \boldsymbol{s}_t = s, \boldsymbol{s}_h = s'\right], \\ &= \frac{\mathbb{P}^{\hat{\pi}^{(i,t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a\right] \cdot \mathbb{P}^{\pi_{\mathsf{unif}} \circ_{t+1} \hat{\pi}^{(i,t+1)}}\left[\boldsymbol{a}_t = a \mid \boldsymbol{s}_t = s\right]}{\sum_{a' \in \mathcal{A}} \mathbb{P}^{\hat{\pi}^{(i,t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = x, \boldsymbol{a}_t = a'\right] \cdot \mathbb{P}^{\pi_{\mathsf{unif}} \circ_{t+1} \hat{\pi}^{(i,t+1)}}\left[\boldsymbol{a}_t = a' \mid \boldsymbol{s}_t = s\right]}, \end{aligned} \tag{41}$$

where the last equality follows by Bayes Theorem; in particular the fact that

$$\mu[A \mid B, C] = \frac{\mu[B \mid A, C] \cdot \mu[A \mid C]}{\mu[B \mid C]}$$

applied with $A = \{\boldsymbol{a}_t = a\}$, $B = \{\boldsymbol{s}_h = s'\}$, $C = \{\boldsymbol{s}_t = s\}$, and $\mu \equiv \overline{\mathbb{P}}^{\pi_{\mathsf{unif}} \circ \hat{\pi}^{(i,t+1)}}$. Now, by combining Eq. (43) with the fact that $\boldsymbol{a}_t$ is independent of $\boldsymbol{s}_t$, we get that

$$\begin{aligned} P^{(i,t)}_{\mathsf{bayes}}(a \mid s, s') &= \frac{\mathbb{P}^{\hat{\pi}^{(i,t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a\right] \cdot \mathbb{P}^{\pi_{\mathsf{unif}} \circ_{t+1} \hat{\pi}^{(i,t+1)}}\left[\boldsymbol{a}_t = a\right]}{\sum_{a' \in \mathcal{A}} \mathbb{P}^{\hat{\pi}^{(i,t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a'\right] \cdot \mathbb{P}^{\pi_{\mathsf{unif}} \circ_{t+1} \hat{\pi}^{(i,t+1)}}\left[\boldsymbol{a}_t = a'\right]}, \\ &= \frac{\mathbb{P}^{\hat{\pi}^{(i,t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a\right]}{\sum_{a' \in \mathcal{A}} \mathbb{P}^{\hat{\pi}^{(i,t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a'\right]}, \end{aligned} \tag{42}$$

where Eq. (44) follows because $\boldsymbol{a}_t \sim \pi_{\mathsf{unif}}$. Now, since the partial policy $\hat{\pi}^{(i,t+1)}$ never takes the terminal action, we have $\mathbb{P}^{\hat{\pi}^{(i,t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a\right] = \overline{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a\right]$ (the left-hand side has $\mathbb{P}$ while the right-hand side has $\overline{\mathbb{P}}$) for all $a \in \mathcal{A}$ and $i \in [S]$. This, together with Eq. (44) implies

$$P^{(i,t)}_{\mathsf{bayes}}(a \mid s, s') = \frac{\overline{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a\right]}{\sum_{a' \in \mathcal{A}} \overline{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a'\right]}.$$

□

Proof of Lemma E.2. Fix $t \in [h-1]$ and $i \in [S]$. By Lemma G.1, $P^{(i,t)}_{\mathsf{bayes}} \in \{f : [S^2] \to \Delta_A\}$ is the Bayes-optimal solution of the conditional density estimation problem in Line 7 of Algorithm 7. And so, by a standard guarantee for log-loss conditional density estimation (see, e.g., Chen et al. (2022, Proposition E.2)),[10] there exists an absolute constant $C' > 0$ (independent of t, h, and other problem parameters) such that with probability at least $1 - \delta$,

$$\mathbb{E}^{\mathsf{unif}(\Psi^{(t)}) \circ_t \pi_{\mathsf{unif}} \circ_{t+1} \hat{\pi}^{(i,t+1)}}\left[\sum_{a \in \mathcal{A}} \left(\hat{f}^{(i,t)}(a \mid \boldsymbol{s}_t, \boldsymbol{s}_h)) - P^{(i,t)}_{\mathsf{bayes}}(a \mid \boldsymbol{s}_t, \boldsymbol{s}_h)\right)^2\right] \le \tilde{\varepsilon}^2_{\mathsf{stat}}(n, \delta), \tag{43}$$

where $\tilde{\varepsilon}^2_{\mathsf{stat}}(n, \delta) \coloneqq C' \log \mathcal{N}_{\mathcal{F}}(1/n) + C' \log(1/\delta)$ and $\mathcal{N}_{\mathcal{F}}(\varepsilon)$ denotes the ε-covering number of the set $\mathcal{F} \coloneqq \{f : [S]^2 \to \Delta_A\}$ in ℓ_∞-distance. It is easy to verify that $\mathcal{N}_{\mathcal{F}}(1/n) \le n^{AS^2}$, and so by setting $C^2 \coloneqq C'$ we have

$$\tilde{\varepsilon}^2_{\mathsf{stat}}(n, \delta) \le C^2 \cdot \varepsilon^2_{\mathsf{stat}}(n, \delta). \tag{44}$$

Now, since $\mathfrak{a}$ is never taken by the partial policies $(\hat{\pi}^{(j)}_{\tau:h-1})_{j \in [S]}$, $\tau \in [h-1]$, in Algorithm 2 or by the policies in $\Psi^{(2)}, \ldots, \Psi^{(h-1)}$ (by assumption), the guarantee in Eq. (45) also holds in $\overline{\mathcal{M}}$. Combining this with Eq. (46) completes the proof. □

G.2. Proof of Lemma E.3 (Local Optimality Guarantee)

Proof of Lemma E.3. Let $\varepsilon'_{\mathtt{stat}} \coloneqq C \cdot \varepsilon_{\mathtt{stat}}(n, \delta/(SH^2))$, where $\varepsilon_{\mathtt{stat}}(n,\delta)$ and $C > 0$ are as in Lemma E.2. We will show that for any $t \in [h-1]$ in Algorithm 7, there exists an event $\mathcal{E}_t$ of probability at least $1 - \delta/H^2$ under which the learned partial policies $\{\hat{\pi}^{(i,t)}\}_{i\in[S]}$ and $\{\hat{\pi}^{(i,t+1)}\}_{i\in[S]}$ are such that for any $i \in \mathcal{S}_h$, we have

$$\sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t)\left(\max_{a\in\mathcal{A}} Q_t^{\hat{\pi}^{(i,t+1)}}(s_t,a;i) - Q_t^{\hat{\pi}^{(,t+1)}}(s_t,\hat{\pi}^{(i,t)}(s_t);i)\right) \le 2SA\varepsilon'_{\mathtt{stat}}, \quad \forall s_t \in \mathcal{S}_t. \tag{45}$$

where $Q_t^{\hat{\pi}^{(i,t+1)}}(\cdot;i)$ is the Q-function at layer t with respect to the partial policy $\hat{\pi}^{(i,t+1)}$ for the BMDP $\overline{\mathcal{M}}$ with rewards $r_\tau^{(i)}(s) = \mathbf{1}\{s=i\}\cdot\mathbf{1}\{\tau = h\}$, for $\tau \in [h]$ (see Eq. (16)). This implies the desired result of the lemma, since $P_{\mathsf{FK}}^{(t)}(i \mid s,a) = Q_t^{\hat{\pi}^{(i,t+1)}}(s,a;i)$; see Eq. (17) and Eq. (22). We write (47) in terms of Q-functions (instead of $P_{\mathsf{FK}}^{(t)}$) to highlight similarities with the analysis of `MusIK` in the more general BMDP setting.

Fix $t \in [h-1]$ and $\mathfrak{i} \in \mathcal{S}_h$. Further, let $\mathcal{S}_t^+$ be the subset of states defined by

$$\mathcal{S}_t^+ \coloneqq \left\{ s \in \mathcal{S}_t : \sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s) \sum_{a\in\mathcal{A}} P^{(\mathfrak{i},t)}(\mathfrak{i} \mid s,a) > 0 \right\},$$
$$\text{where} \quad P^{(\mathfrak{i},t)}(s' \mid s,a) \coloneqq \overline{\mathbb{P}}^{\hat{\pi}^{(\mathfrak{i},t+1)}}[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = s, \boldsymbol{a}_t = a]. \tag{46}$$

By Lemma E.2 and Jensen's inequality, there is an event $\mathcal{E}_t^{(\mathfrak{i})}$ of probability at least $1 - \delta/(SH^2)$ under which the solution $\hat{f}^{(\mathfrak{i},t)}$ of the conditional density estimation problem in Line 7 of Algorithm 7 satisfies,

$$\overline{\mathbb{E}}_{\boldsymbol{s}_t\sim\mathtt{unif}(\Psi^{(t)}),\boldsymbol{a}'\sim\pi_{\mathtt{unif}}}\left[\sum_{s_h\in\mathfrak{S}_h} P^{(\mathfrak{i},t)}(s_h \mid \boldsymbol{s}_t, \boldsymbol{a}') \max_{a\in\mathfrak{A}} \left|\mathrm{err}^{(\mathfrak{i},t)}(a,\boldsymbol{s}_t,s_h)\right|\right] \le \varepsilon'_{\mathtt{stat}}, \tag{47}$$

where $\mathrm{err}^{(\mathfrak{i},t)}(a,s_t,s_h) \coloneqq \hat{f}^{(\mathfrak{i},t)}(a \mid s_t,s_h) - P_{\mathsf{bayes}}^{(\mathfrak{i},t)}(a \mid s_t,s_h)$ and

$$P_{\mathsf{bayes}}^{(\mathfrak{i},t)}(a \mid s,s') \coloneqq \frac{P^{(\mathfrak{i},t)}(s' \mid s,a)}{\sum_{a'\in\mathcal{A}} P^{(\mathfrak{i},t)}(s' \mid s,a')}. \tag{48}$$

In what follows, we condition on $\mathcal{E}_t \coloneqq \bigcap_{j\in[S]} \mathcal{E}_t^{(j)}$. Now, fix $\mathfrak{j} \in \mathcal{S}_t^+$. From Eq. (49), we have that

$$\begin{aligned}
SA\varepsilon'_{\mathtt{stat}} &\ge \sum_{\pi\in\Psi^{(t)}} \sum_{a'\in\mathfrak{A}} \sum_{s_h\in\mathfrak{S}_h} \bar{d}^{\pi}(\mathfrak{j}) P^{(\mathfrak{i},t)}(s_h \mid \mathfrak{j},a') \max_{a\in\mathcal{A}} \left|\mathrm{err}^{(\mathfrak{i},t)}(a,\mathfrak{j},s_h)\right|, \\
&\ge \sum_{\pi\in\Psi^{(t)}} \sum_{a'\in\mathfrak{A}} \bar{d}^{\pi}(\mathfrak{j}) P^{(\mathfrak{i},t)}(\mathfrak{i} \mid \mathfrak{j},a') \max_{a\in\mathcal{A}} \left|\mathrm{err}^{(\mathfrak{i},t)}(a,\mathfrak{j},\mathfrak{i})\right|, \\
&= \sum_{\pi\in\Psi^{(t)}} \sum_{a'\in\mathfrak{A}} \bar{d}^{\pi}(\mathfrak{j}) P^{(\mathfrak{i},t)}(\mathfrak{i} \mid \mathfrak{j},a') \max_{a\in\mathcal{A}} \left|\hat{f}^{(\mathfrak{i},t)}(a \mid \mathfrak{j},\mathfrak{i}) - P_{\mathsf{bayes}}^{(\mathfrak{i},t)}(a \mid \mathfrak{j},\mathfrak{i})\right|.
\end{aligned}$$

By rearranging and using the fact that $\sum_{\pi\in\Psi^{(t)}} \sum_{a'\in\mathfrak{A}} \bar{d}^{\pi}(\mathfrak{j}) P^{(\mathfrak{i},t)}(\mathfrak{i} \mid \mathfrak{j},a') > 0$ (since $\mathfrak{j} \in \mathcal{S}_t^+$), we get

$$\max_{a\in\mathfrak{A}} \left|\hat{f}^{(\mathfrak{i},t)}(a \mid \mathfrak{j},\mathfrak{i}) - P_{\mathsf{bayes}}^{(\mathfrak{i},t)}(a \mid \mathfrak{j},\mathfrak{i})\right| \le \frac{SA\varepsilon'_{\mathtt{stat}}}{\sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(\mathfrak{j}) \cdot \sum_{a'\in\mathcal{A}} P^{(\mathfrak{i},t)}(\mathfrak{i} \mid \mathfrak{j},a')}. \tag{49}$$

Now, let $\hat{a}^{(\mathfrak{i},t)}(s) \in \arg\max_{a\in\mathcal{A}} \hat{f}^{(\mathfrak{i},t)}(a \mid s,\mathfrak{i})$ and note that $\hat{a}^{(\mathfrak{i},t)}(s) = \hat{\pi}^{(\mathfrak{i},t)}(s)$, where $\hat{\pi}^{(\mathfrak{i},t)}$ is as in Algorithm 7. With this, Eq. (51) and the fact that $|\|y\|_\infty - \|z\|_\infty| \le \|y-z\|_\infty$, for all $y,z \in \mathbb{R}^A$, we have that

$$\begin{aligned}
\max_{a\in\mathfrak{A}} P_{\mathsf{bayes}}^{(\mathfrak{i},t)}(a \mid \mathfrak{j},\mathfrak{i}) &\le \hat{f}^{(\mathfrak{i},t)}(\hat{a}^{(\mathfrak{i},t)}(\mathfrak{j}) \mid \mathfrak{j},\mathfrak{i}) + \frac{SA\varepsilon'_{\mathtt{stat}}}{\sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(\mathfrak{j}) \cdot \sum_{a'\in\mathcal{A}} P^{(\mathfrak{i},t)}(\mathfrak{i} \mid \mathfrak{j},a')}, \\
&\le P^{(\mathfrak{i},t)}(\hat{a}^{(\mathfrak{i},t)}(\mathfrak{j}) \mid \mathfrak{j},\mathfrak{i}) + \frac{2SA\varepsilon'_{\mathtt{stat}}}{\sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(\mathfrak{j}) \cdot \sum_{a'\in\mathcal{A}} P^{(\mathfrak{i},t)}(\mathfrak{i} \mid \mathfrak{j},a')}. \quad \text{(by Eq. (51) again)}
\end{aligned} \tag{50}$$

With this in hand, we have that

$$\begin{aligned}
\max_{a\in\mathfrak{A}} Q_t^{\hat{\pi}^{(\mathrm{i},t+1)}}(\mathrm{j},a;\mathrm{i}) &= \max_{a\in\mathfrak{A}} \overline{\mathbb{P}}^{\hat{\pi}^{(\mathrm{i},t+1)}}[\boldsymbol{s}_h=\mathrm{i}\mid \boldsymbol{s}_t=\mathrm{j},\boldsymbol{a}_t=a],\\
&= \max_{a\in\mathcal{A}} P^{(\mathrm{i},t)}(\mathrm{i}\mid \mathrm{j},a), \quad \text{(by definition—see Eq. (48))}\\
&= \max_{a\in\mathfrak{A}} P^{(,t)}_{\mathsf{bayes}}(a\mid \mathrm{j},\mathrm{i}) \sum_{a'\in\mathcal{A}} P^{(\mathrm{i},t)}(\mathrm{i}\mid \mathrm{j},a'), \quad \text{(by def. of } P^{(i,t)}_{\mathsf{bayes}} \text{ in Eq. (50))}\\
&\le P^{(\mathrm{i},t)}(\hat{a}^{(\mathrm{i},t)}(\mathrm{j})\mid \mathrm{j},\mathrm{i}) \sum_{a'\in\mathcal{A}} P^{(\mathrm{i},t)}(\mathrm{i}\mid \mathrm{j},a') + \frac{2SA\varepsilon'_{\mathsf{stat}}}{\sum_{\pi\in\Psi^{(t)}}\bar{d}^{\pi}(\mathrm{j})}, \quad \text{(by Eq. (52))}\\
&= \overline{\mathbb{P}}^{\hat{\pi}^{(\mathrm{i},t+1)}}[\boldsymbol{s}_h=\mathrm{i}\mid \boldsymbol{s}_t=\mathrm{j},\boldsymbol{a}_t=\hat{a}^{(\mathrm{i},t)}(\mathrm{j})] + \frac{2SA\varepsilon'_{\mathsf{stat}}}{\sum_{\pi\in\Psi^{(t)}}\bar{d}^{\pi}(\mathrm{j})}, \quad \text{(by Eq. (50))}\\
&= Q_t^{\hat{\pi}^{(\mathrm{i},t+1)}}(\mathrm{j},\hat{\pi}^{(\mathrm{i},t)}(\mathrm{j});\mathrm{i}) + \frac{2SA\varepsilon'_{\mathsf{stat}}}{\sum_{\pi\in\Psi^{(t)}}\bar{d}^{\pi}(\mathrm{j})},
\end{aligned}$$

where the last equality follows by definition of $\hat{\pi}^{(\mathrm{i},t)}$ in Algorithm 7.

The argument above implies that

$$\sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t)\left(\max_{a\in\mathcal{A}} Q_t^{\hat{\pi}^{(,t+1)}}(s_t,a;) - Q_t^{\hat{\pi}^{(,t+1)}}(s_t,\hat{\pi}^{(,t)}(s_t);)\right) \le 2SA\varepsilon'_{\mathsf{stat}}, \quad \forall s_t\in\mathcal{S}_t^+. \tag{51}$$

On the other hand, for any $s_t\notin\mathcal{S}_t^+$, we have

$$\sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t)\max_{a\in\mathcal{A}} Q_t^{\hat{\pi}^{(,t+1)}}(s_t,a;) \le \sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t)\cdot\sum_{a'\in\mathcal{A}} P^{(\mathrm{i},t)}(\mathrm{i}\mid s_t,a') = 0,$$

by definition of $\mathcal{S}_t^+$. This, combined with the fact that $Q_t^{\hat{\pi}^{(\mathrm{i},t+1)}}(\cdot,\cdot;\mathrm{i})\ge 0$ implies that Eq. (53) also holds for $s_t\in\mathcal{S}_t\setminus\mathcal{S}_t^+$. Thus, we have that

$$\sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t)\left(\max_{a\in\mathcal{A}} Q_t^{\hat{\pi}^{(,t+1)}}(s_t,a;) - Q_t^{\hat{\pi}^{(,t+1)}}(s_t,\hat{\pi}^{(,t)}(s_t);)\right) \le 2SA\varepsilon'_{\mathsf{stat}}, \quad \forall s_t\in\mathcal{S}_t.$$

□

H. Proofs for Block MDPs

H.1. MLE Guarantee for Block MDPs

We now state and prove a guarantee for the minimizer $(\hat{f}^{(t)},\hat{\phi}^{(t)})$ of the conditional density estimation problem in Line 7 of Algorithm 2 under realizability. We first derive the expression of the Bayes-optimal solution of this problem (we express this solution as a function of probability measures in the extended BMDP, which will be convenient in the proof of Theorem 3.2).

Lemma H.1. *Let $h\in[H]$ and $t\in[h-1]$ be given, and define*

$$P^{(t)}_{\mathsf{bayes}}((a,j)\mid s,s') \coloneqq \frac{\overline{\mathbb{P}}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{s}_h=s'\mid \boldsymbol{s}_t=s,\boldsymbol{a}_t=a]}{\sum_{a'\in\mathcal{A},i\in[S]}\overline{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{s}_h=s'\mid \boldsymbol{s}_t=s,\boldsymbol{a}_t=a']}, \tag{52}$$

with $\hat{\pi}^{(j,t+1)}$ as in Algorithm 2. Consider the solution to the unconstrained maximum problem

$$\widetilde{P}^{(t)}_{\mathsf{bayes}} \in \operatorname*{arg\,max}_{\widetilde{P}:\mathcal{X}_t\times\mathcal{X}_h\to\Delta(\mathcal{A}\times[S])} \mathbb{E}_{\boldsymbol{i}_t\sim\mathsf{unif}([S])}\mathbb{E}^{\mathsf{unif}(\Psi^{(t)})\circ_t\pi_{\mathsf{unif}}\circ_{t+1}\hat{\pi}^{(\boldsymbol{i}_t,t+1)}}\left[\log\widetilde{P}^{(t)}((\boldsymbol{a}_t,\boldsymbol{i}_t)\mid \boldsymbol{x}_t,\boldsymbol{x}_h)\right]. \tag{53}$$

Then, for any $a\in\mathcal{A}$, $j\in[S]$, $x\in\mathcal{X}_t$, and $x'\in\mathcal{X}_h$, letting $s=\phi_\star(x)$ and $s'=\phi_\star(x')$, $\widetilde{P}^{(t)}_{\mathsf{bayes}}$ satisfies

$$\widetilde{P}^{(t)}_{\mathsf{bayes}}((a,j)\mid x,x') = P^{(t)}_{\mathsf{bayes}}((a,j)\mid s,s').$$

In addition, $(P^{(t)}_{\mathsf{bayes}}, \phi_\star)$ is the Bayes-optimal solution to the maximum likelihood problem in Line 7 of Algorithm 2; that is,

$$(P^{(t)}_{\mathsf{bayes}}, \phi_\star) \in \operatorname*{arg\,max}_{f:[S]^2\to\Delta(\mathcal{A}\times[S]),\phi\in\Phi} \mathbb{E}_{\boldsymbol{i}_t\sim\mathsf{unif}([S])}\mathbb{E}^{\mathsf{unif}(\Psi^{(t)})\circ_t\pi_{\mathsf{unif}}\circ_{t+1}\hat{\pi}^{(\boldsymbol{i}_t,t+1)}}\left[\log f((\boldsymbol{a}_t,\boldsymbol{i}_t)\mid\phi(\boldsymbol{x}_t),\phi(\boldsymbol{x}_h))\right].$$

Proof of Lemma H.1. Fix $a\in\mathcal{A}$, $j\in[S]$, and $(x,x')\in\mathcal{X}_t\times\mathcal{X}_h$. Further, let $\boldsymbol{i}_t\sim\mathsf{unif}([S])$ and $\mathbb{Q}$ denote the law over $(\boldsymbol{a}_t,\boldsymbol{i}_t,\boldsymbol{x}_t,\boldsymbol{x}_h)$ induced by first sampling $\boldsymbol{j}_t\sim\mathsf{unif}([S])$ then executing $\mathsf{unif}(\Psi^{(t)})\circ_t\pi_{\mathsf{unif}}\circ_{t+1}\hat{\pi}^{(\boldsymbol{i}_t,t+1)}$. With this, the solution $\widetilde{P}^{(t)}_{\mathsf{bayes}}$ of the problem in Eq. (55) satisfies

$$\begin{aligned}\widetilde{P}^{(t)}_{\mathsf{bayes}}((a,j)\mid x,x') &= \mathbb{Q}[\boldsymbol{a}_t=a,\boldsymbol{i}_t=j\mid\boldsymbol{x}_t=x,\boldsymbol{x}_h=x'],\\ &= \frac{\mathbb{Q}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a,\boldsymbol{i}_t=j]\cdot\mathbb{Q}[\boldsymbol{a}_t=a,\boldsymbol{i}_t=j\mid\boldsymbol{x}_t=x]}{\sum_{a'\in\mathcal{A}}\sum_{i\in[S]}\mathbb{Q}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a',\boldsymbol{i}_t=i]\cdot\mathbb{Q}[\boldsymbol{a}_t=a',\boldsymbol{i}_t=i\mid\boldsymbol{x}_t=x]},\end{aligned}\tag{54}$$

where the last equality follows by Bayes Theorem; in particular

$$\mu[A\mid B,C]=\frac{\mu[B\mid A,C]\cdot\mu[A\mid C]}{\mu[B\mid C]},$$

applied with $A=\{\boldsymbol{a}_t=a,\boldsymbol{i}_t=j\}$, $B=\{\boldsymbol{x}_h=x'\}$, $C=\{\boldsymbol{x}_t=x\}$, and $\mu\equiv\mathbb{Q}$. Now, by combining Eq. (56) with the fact that $\mathbb{Q}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a,\boldsymbol{i}_t=j]=\mathbb{P}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a]$ (note that the right-hand side is well-defined since $\hat{\pi}^{(j,t+1)}\in\Pi_{\mathsf{NM}}^{t+1:h-1}$), and using that $(\boldsymbol{a}_t,\boldsymbol{i}_t)$ is independent of $\boldsymbol{x}_t$, we get that

$$\begin{aligned}\widetilde{P}^{(t)}_{\mathsf{bayes}}((a,j)\mid x,x') &= \frac{\mathbb{P}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a]\cdot\mathbb{Q}[\boldsymbol{a}_t=a,\boldsymbol{i}_t=j]}{\sum_{a'\in\mathcal{A}}\sum_{i\in[S]}\mathbb{P}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a']\cdot\mathbb{Q}[\boldsymbol{a}_t=a',\boldsymbol{i}_t=i]},\\ &= \frac{\mathbb{P}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a]}{\sum_{a'\in\mathcal{A}}\sum_{i\in[S]}\mathbb{P}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a']},\end{aligned}\tag{55}$$

where Eq. (57) follows by the fact that $\mathbb{Q}[\boldsymbol{a}_t=a',\boldsymbol{i}_t=i']=\mathbb{Q}[\boldsymbol{a}_t=a'',\boldsymbol{i}_t=i'']$, for $i',i''\in[S],a',a''\in\mathcal{A}$.

Now, since the partial policies $\hat{\pi}^{(i,t+1)}$, $i\in[S]$, never take the terminal action, we have $\mathbb{P}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a]=\overline{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a]$ (the left-hand side has $\mathbb{P}$ while the right-hand side has $\overline{\mathbb{P}}$), for all $i\in[S]$. This, together with Eq. (57) implies

$$\widetilde{P}^{(t)}_{\mathsf{bayes}}((a,j)\mid x,x')=\frac{\overline{\mathbb{P}}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a]}{\sum_{a'\in\mathcal{A},i\in[S]}\overline{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a']}.\tag{56}$$

Note that since the outputs of the (potentially non-Markovian) partial policies $\{\hat{\pi}^{(i,t+1)},i\in[S]\}\subseteq\Pi_{\mathsf{NM}}^{t+1:h-1}$ depend only on $\boldsymbol{x}_{t+1:h-1}$, and not on $\boldsymbol{x}_{1:t}$, we have

$$\begin{aligned}&\overline{\mathbb{P}}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a]\\ &=\sum_{x''\in\mathcal{X}_{t+1}}\overline{\mathbb{P}}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_{t+1}=x'',\boldsymbol{x}_t=x,\boldsymbol{a}_t=a]\cdot\overline{\mathbb{P}}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{x}_{t+1}=x''\mid\boldsymbol{x}_t=x,\boldsymbol{a}_t=a],\\ &=\sum_{x''\in\mathcal{X}_{t+1}}\overline{\mathbb{P}}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_{t+1}=x'',\boldsymbol{x}_t=x,\boldsymbol{a}_t=a]\cdot\overline{\mathbb{P}}[\boldsymbol{x}_{t+1}=x''\mid\boldsymbol{s}_t=\phi_\star(x),\boldsymbol{a}_t=a],\\ &=\sum_{x''\in\mathcal{X}_{t+1}}\overline{\mathbb{P}}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{x}_{t+1}=x'']\cdot\overline{\mathbb{P}}[\boldsymbol{x}_{t+1}=x''\mid\boldsymbol{s}_t=\phi_\star(x),\boldsymbol{a}_t=a],\quad(\text{since }\hat{\pi}^{(j,t+1)}\in\Pi_{\mathsf{NM}}^{t+1:h-1})\\ &=\overline{\mathbb{P}}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{x}_h=x'\mid\boldsymbol{s}_t=\phi_\star(x),\boldsymbol{a}_t=a],\\ &=q(x'\mid\phi_\star(x'))\cdot\overline{\mathbb{P}}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{s}_h=\phi_\star(x')\mid\boldsymbol{s}_t=\phi_\star(x),\boldsymbol{a}_t=a].\end{aligned}$$

This, together with Eq. (58) implies that $\widetilde{P}^{(t)}_{\mathsf{bayes}}(\cdot \mid x, x') \equiv P^{(t)}_{\mathsf{bayes}}(\cdot \mid \phi_\star(x), \phi_\star(x'))$ (after canceling the terms involving $q(x' \mid \phi_\star(x'))$), where $P^{(t)}_{\mathsf{bayes}}$ is as in Eq. (54). Now that we have established Eq. (54), we show the second claim of the lemma. The population version of the problem in Line 7 of Algorithm 2 becomes equivalent to the following optimization problem:

$$\max_{f:[S]^2\to\Delta(\mathcal{A}\times[S]),\phi\in\Phi} \mathbb{E}_{\boldsymbol{i}_t\sim\mathsf{unif}([S])}\mathbb{E}^{\mathsf{unif}(\Psi^{(t)})\circ_t\pi_{\mathsf{unif}}\circ_{t+1}\hat{\pi}^{(\boldsymbol{i}_t,t+1)}}\left[\log f((\boldsymbol{a}_t,\boldsymbol{i}_t)\mid\phi(\boldsymbol{x}_t),\phi(\boldsymbol{x}_h))\right]. \tag{57}$$

Note that the value of this problem is always at least that of Eq. (55). On the other hand, by Eq. (54), the value of the objective in Eq. (59) with the pair $(f,\phi) = (P^{(t)}_{\mathsf{bayes}},\phi_\star)$ matches the optimal value of the problem in Eq. (55), and so $(P^{(t)}_{\mathsf{bayes}},\phi_\star)$ is indeed a solution of Eq. (59). □

Lemma H.2 (MLE guarantee)**.** *Let $n \ge 1$ and $\delta \in (0,1)$, and define $\varepsilon_{\mathsf{stat}}(n,\delta) := n^{-1/2}\sqrt{S^3A\log n + \log(|\Phi|/\delta)}$. Further, let $1 \le t < h \le H$ and suppose that Φ satisfies Assumption 2.1 and that the policies in $\Psi^{(t)}$ never take the terminal action $\mathfrak{a}$. Then, there exists an absolute constant $C > 0$ (independent of t, h, and other problem parameters) such that the MLE $(\hat{f}^{(t)},\hat{\phi}^{(t)})$ of the conditional density estimation problem in Line 7 of Algorithm 2 satisfies with probability at least $1-\delta$,*

$$\begin{aligned}&\overline{\mathbb{E}}_{\boldsymbol{i}_t\sim\mathsf{unif}([S])}\overline{\mathbb{E}}^{\mathsf{unif}(\Psi^{(t)})\circ_t\pi_{\mathsf{unif}}\circ_{t+1}\hat{\pi}^{(\boldsymbol{i}_t,t+1)}}\left[\sum_{a\in\mathcal{A},j\in[S]}\left(\hat{f}^{(t)}((a,j)\mid\hat{\phi}^{(t)}(\boldsymbol{x}_t),\hat{\phi}^{(t)}(\boldsymbol{x}_h)) - P^{(t)}_{\mathsf{bayes}}((a,j)\mid\boldsymbol{s}_t,\boldsymbol{s}_h)\right)^2\right]\\ &\quad\le C^2\cdot\varepsilon^2_{\mathsf{stat}}(n,\delta).\end{aligned}$$

where $P^{(t)}_{\mathsf{bayes}}$ is as in Lemma H.1.

Proof of Lemma H.2. Fix $t \in [h-1]$. By Lemma H.1, $(P^{(t)}_{\mathsf{bayes}},\phi_\star)$ is the Bayes-optimal solution of the conditional density estimation problem in Line 7 of Algorithm 2. And so, by Assumption 2.1 and a standard MLE guarantee for log-loss conditional density estimation (see e.g. Chen et al. (2022, Proposition E.2)),[10] there exists an absolute constant $C' > 0$ (independent of t, h, and other problem parameters) such that with probability at least $1-\delta$,

$$\begin{aligned}&\mathbb{E}_{\boldsymbol{i}_t\sim\mathsf{unif}([S])}\mathbb{E}^{\mathsf{unif}(\Psi^{(t)})\circ_t\pi_{\mathsf{unif}}\circ_{t+1}\hat{\pi}^{(\boldsymbol{i}_t,t+1)}}\left[\sum_{a\in\mathcal{A},j\in[S]}\left(\hat{f}^{(t)}((a,j)\mid\hat{\phi}^{(t)}(\boldsymbol{x}_t),\hat{\phi}^{(t)}(\boldsymbol{x}_h)) - P^{(t)}_{\mathsf{bayes}}((a,j)\mid\boldsymbol{s}_t,\boldsymbol{s}_h)\right)^2\right]\\ &\quad\le\tilde{\varepsilon}^2_{\mathsf{stat}}(n,\delta),\end{aligned} \tag{58}$$

where $\tilde{\varepsilon}^2_{\mathsf{stat}}(n,\delta) := C'\log\mathcal{N}_{\mathcal{F}}(1/n) + C'\log(|\Phi|/\delta)$ and $\mathcal{N}_{\mathcal{F}}(\varepsilon)$ denotes the ε-covering number of the set $\mathcal{F} := \{f : [S]^2 \to \Delta([S]\times\mathcal{A})\}$ in ℓ_∞-distance. It is easy to verify that $\mathcal{N}_{\mathcal{F}}(1/n) \le n^{AS^3}$, and so by setting $C^2 := C'$, we have

$$\tilde{\varepsilon}^2_{\mathsf{stat}}(n,\delta) \le C^2\cdot\varepsilon^2_{\mathsf{stat}}(n,\delta). \tag{59}$$

Now, since $\mathfrak{a}$ is never taken by the partial policies $(\hat{\pi}^{(i,\tau)})_{i\in[S],\tau\in[h-1]}$, in Algorithm 2 or by the policies in $\Psi^{(2)},\dots,\Psi^{(h-1)}$ (by assumption), the guarantee in Eq. (60) also holds in $\overline{\mathcal{M}}$. Combining this with Eq. (61) completes the proof. □

H.2. Proof of Theorem 3.2 (Main Guarantee for MusIK)

Proof of Theorem 3.2. Let $\epsilon := \varepsilon/(2S)$. Let $\varepsilon_{\mathsf{stat}}(\cdot,\cdot)$ and C be as in Theorem E.3 (note that C is an absolute constant independent of all problem parameters). Let $\mathcal{E}_h$ be the success event of Theorem E.3 for $h \in [H]$ and ϵ, and define $\mathcal{E} := \bigcap_{h\in[H]}\mathcal{E}_h$. Note that by the union bound we have $\mathbb{P}[\mathcal{E}] \ge 1-\delta$. For n large enough such that $8AS^4HC\varepsilon_{\mathsf{stat}}(n,\frac{\delta}{H^2}) \le \epsilon$ (which is implied by the condition on n in the theorem's statement for $c = 2^5C$), Theorem E.3 implies that under $\mathcal{E}$, the output $\Psi^{(1)},\dots,\Psi^{(H)}$ of MusIK are $(1/2,\epsilon)$-policy covers relative to $\overline{\Pi}_\epsilon$ in $\overline{\mathcal{M}}$ for layers 1 to H, respectively. Thus, by Lemma 4.3, the desired result holds under $\mathcal{E}$.

The parameter n in Theorem 3.2 represents the input to MusIK used to generate an approximate $(1/4,\varepsilon)$-policy cover relative to Π_{M} in $\mathcal{M}$ at all layers. MusIK passes n to all of IKDP invocations (see Line 3 of Algorithm 1). Since, for any layer $h \in [H]$,

[10] Technically, (Chen et al., 2022, Proposition E.2) bounds the Hellinger distance, which immediately implies a bound on MSE.

the corresponding IKDP instance in MusIK requires n trajectories for each layer $t \in [h-1]$ (see Line 1 of Algorithm 2), the total number of trajectories needed by MusIK in the setting of Theorem 3.2 is

$$\#\text{ of trajectories} = \widetilde{O}(1) \cdot \frac{A^2 S^{10} H^4 \left(AS^3 + \log(|\Phi| H^2/\delta)\right)}{\varepsilon^2}.$$

□

H.3. Proof of Theorem E.3 (Main Guarantee for IKDP)

Before proving Theorem E.3, we first define the Q- and V-functions corresponding to certain 'fictitious' rewards we introduce for the analysis. These functions will be instrumental in our proofs. We then present a generalized performance difference lemma that holds for the non-Markovian partial policies of MusIK (since the policies are non-Markovian the standard performance difference lemma does not give us something useful for the proof of Theorem E.3). In Appendix H.3.3, we bound the errors appearing on the RHS of our generalized performance difference lemma. Finally, we present the proof of Theorem E.3 in Appendix H.3.4

H.3.1. The Q- and V-functions

For $t, h \in [H]$, $a \in \mathcal{A}$, and $s' \in \mathcal{S}_h$, define $r_t(\cdot; s') : \mathcal{X}_t \to \{0, 1\}$ as

$$r_t(x; s') = \mathbf{1}\{\phi_\star(x) = s'\}. \tag{60}$$

This can be interpreted as a reward function that takes value 1 whenever the latent state is s'. For $t \in [H]$ and any two partial policies $\pi^{(t)} \in \Pi_{\mathsf{NM}}^{t:h-1}$ and $\pi^{(t+1)} \in \Pi_{\mathsf{NM}}^{t+1:h-1}$, we define the corresponding tth layer Q- and V-functions in $\overline{\mathcal{M}}$ as

$$Q_t^{\pi^{(t+1)}}(x, a; s') \coloneqq r_t(x; s') + \overline{\mathbb{E}}^{\pi^{(t+1)}}\left[\sum_{\tau=t+1}^{h} r_\tau(\boldsymbol{x}_\tau; s') \,\middle|\, \boldsymbol{x}_t = x, \boldsymbol{a}_t = a\right], \tag{61}$$

$$\text{and } V_t^{\pi^{(t)}}(x; s') \coloneqq r_t(x; s') + \overline{\mathbb{E}}^{\pi^{(t)}}\left[\sum_{\tau=t+1}^{h} r_\tau(\boldsymbol{x}_\tau; s') \,\middle|\, \boldsymbol{x}_t = x, \boldsymbol{a}_t = \pi^{(t)}(x)\right]. \tag{62}$$

These match the standard definitions of the Q- and V- functions for Markovian policies, albeit with action-independent rewards. Note that we only define $Q_\tau^{\pi^{(t+1)}}$ and $V_\tau^{\pi^{(t)}}$ for $\tau = t$, as the policies involved are non-Markovian, and are undefined on layers $\tau < t$.

Useful properties of the Q- and V-functions. Given the definition of the rewards in (62), the Q-function in (63) can be expressed in terms of certain conditional probabilities for visiting latent states, which will be useful throughout the proof.

Lemma H.3. *For any $s' \in \mathcal{S}_h$, $x \in \mathcal{X}_t$, $a \in \mathcal{A}$, and $\pi^{(t+1)} \in \Pi_{\mathsf{NM}}^{t+1,h-1}$, we have*

$$Q_t^{\pi^{(t+1)}}(x, a; s') = \overline{\mathbb{P}}^{\pi^{(t+1)}}[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = \phi_\star(x), \boldsymbol{a}_t = a]. \tag{63}$$

For IKDP's partial policies $\{\hat{\pi}^{(i,\tau)} : i \in [S]\} \subseteq \Pi_{\mathsf{NM}}^{\tau:h-1}$, for $\tau \in [h-1]$, the corresponding V-functions satisfy an identity similar to (65).

Lemma H.4. *For any $i \in [S]$, $s' \in \mathcal{S}_h$, $x \in \mathcal{X}_t$, and $a \in \mathcal{A}$, we have*

$$V_t^{\hat{\pi}^{(i,t)}}(x; s') = \overline{\mathbb{P}}^{\hat{\pi}^{(\hat{\imath}(x),t+1)}}[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = \phi_\star(x), \boldsymbol{a}_t = \hat{a}(x)],$$

where $(\hat{a}(x), \hat{\imath}(x)) \in \arg\max_{(a',i')} \hat{f}^{(t)}((a', i') \mid \hat{\phi}^{(t)}(x), i)$ and $(\hat{f}^{(t)}, \hat{\phi}^{(t)})$ are defined as in Line 7 of Algorithm 2.

Note that unlike the Q-function in Lemma H.3, it is not the case that the V-function in Lemma H.4 depends on $x \in \mathcal{X}_t$ only through $\phi_\star(x)$.

Proof of Lemma H.3. By definition of the reward functions, we have that $r_t(\cdot, s') \equiv 0$, for all $t < h$, and

$$\sum_{\tau=t+1}^{h} r_\tau(\boldsymbol{x}_\tau; s') = r_h(\boldsymbol{x}_h; s') = \mathbb{I}\{\phi_\star(\boldsymbol{x}_h) = s'\}.$$

Therefore,

$$\begin{aligned} Q_t^{\pi^{(t+1)}}(x,a;s') &= r_t(x;s') + \overline{\mathbb{E}}^{\pi^{(t+1)}}\left[\sum_{\tau=t+1}^{h} r_\tau(\boldsymbol{x}_\tau;s') \;\middle|\; \boldsymbol{x}_t = x, \boldsymbol{a}_t = a\right], \\ &= 0 + \overline{\mathbb{E}}^{\pi^{(t+1)}}\left[r_h(\boldsymbol{x}_h;s') \mid \boldsymbol{x}_t = x, \boldsymbol{a}_t = a\right], \\ &= \overline{\mathbb{P}}^{\pi^{(t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = \phi_\star(x), \boldsymbol{a}_t = a\right], \end{aligned}$$

where the last equality follows from the fact that, while $\pi^{(t+1)} \in \Pi_{\mathsf{NM}}^{t+1:h-1}$ is non-Markovian, it only depends on the observations at layers $t+1$ to $h-1$. □

Proof of Lemma H.4. By definition of the reward functions, we have that $r_t(\cdot, s') \equiv 0$, for all $t < h$, and

$$\sum_{\tau=t+1}^{h} r_\tau(\boldsymbol{x}_\tau;s') = r_h(\boldsymbol{x}_h;s') = \mathbb{I}\{\phi_\star(\boldsymbol{x}_h) = s'\}. \tag{64}$$

Therefore,

$$\begin{aligned} V_t^{\hat{\pi}^{(i,t)}}(x;s') &= r_t(x;s') + \overline{\mathbb{E}}^{\hat{\pi}^{(i,t)}}\left[\sum_{\tau=t+1}^{h} r_\tau(\boldsymbol{x}_\tau;s') \;\middle|\; \boldsymbol{x}_t = x, \boldsymbol{a}_t = \hat{\pi}^{(i,t)}(x)\right], \\ &= 0 + \overline{\mathbb{E}}^{\hat{\pi}^{(i,t)}}\left[r_h(\boldsymbol{x}_h;s') \mid \boldsymbol{x}_t = x, \boldsymbol{a}_t = \hat{\pi}^{(i,t)}(x)\right], \quad \text{(by (66))} \\ &= \overline{\mathbb{E}}^{\hat{\pi}^{(i,t)} \circ_{t+1} \hat{\pi}^{(i(x),t+1)}}\left[r_h(\boldsymbol{x}_h;s') \mid \boldsymbol{x}_t = x, \boldsymbol{a}_t = \hat{a}(x)\right], \\ &= \overline{\mathbb{E}}^{\hat{\pi}^{(i(x),t+1)}}\left[r_h(\boldsymbol{x}_h;s') \mid \boldsymbol{x}_t = x, \boldsymbol{a}_t = \hat{a}(x)\right], \quad \text{(since } \hat{\pi}^{(j,t+1)} \in \Pi_{\mathsf{NM}}^{t+1,h-1}, \forall j) \\ &= \overline{\mathbb{P}}^{\hat{\pi}^{(i(x),t+1)}}\left[\boldsymbol{s}_h = s' \mid \boldsymbol{s}_t = \phi_\star(x), \boldsymbol{a}_t = \hat{a}(x)\right], \quad \text{(by (62) and } \hat{\pi}^{(j,t+1)} \in \Pi_{\mathsf{NM}}^{t+1,h-1}, \forall j) \end{aligned} \tag{65}$$

where (67) follows by the definition of $\hat{\pi}^{(i,t)}$ in Line 7 of Algorithm 2.

□

H.3.2. Generalized Performance Difference Lemma for MusIK's Non-Markovian Policies

We now present a generalized performance difference lemma that holds for the non-Markovian partial policies used in MusIK/IKDP. In what follows, we use the convention that for any $\pi \in \Pi_{\mathsf{NM}}^{l:r}$ and $\tau \in [l\,..\,r]$, $\pi(x_{l:\tau}) = \mathfrak{a}$, for any $x_{l:\tau} \in \overline{\mathcal{X}}_l \times \cdots \times \overline{\mathcal{X}}_\tau$ such that $x_\tau = \mathfrak{t}_\tau$ (or equivalently $\phi_\star(x_\tau) = \mathfrak{t}_\tau$).[11]

Lemma H.5. *Fix $h \in [H]$, and let us adopt the convention that $\hat{\pi}^{(i,h)} \equiv \pi_{\mathsf{unif}}, \forall i \in [S]$. The partial policies $\hat{\pi}^{(i,t)}$ produced by* IKDP *for $i \in [S]$, $t \in [h-1]$ satisfy for any $\mathfrak{s} \in \mathcal{S}_h$,*

$$\min_{i\in[S]} \overline{\mathbb{E}}\left[V_1^{\pi_\star^{(\mathfrak{s})}}(\boldsymbol{x}_1;\mathfrak{s}) - V_1^{\hat{\pi}^{(i,1)}}(\boldsymbol{x}_1;\mathfrak{s})\right] \le \sum_{t=1}^{h-1} \min_{i\in[S]} \max_{j\in[S]} \overline{\mathbb{E}}^{\pi_\star^{(\mathfrak{s})}}\left[\mathbb{I}\{\boldsymbol{s}_t \in \mathcal{S}_{t,\epsilon}\}\left(Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t, \pi_\star^{(\mathfrak{s})}(\boldsymbol{x}_t);\mathfrak{s}) - V_t^{\hat{\pi}^{(i,t)}}(\boldsymbol{x}_t;\mathfrak{s})\right)\right],$$

where $\pi_\star^{(\mathfrak{s})} \in \arg\max_{\pi\in\overline{\Pi}_\epsilon} \bar{d}^\pi(\mathfrak{s})$ and $Q_t^{\hat{\pi}^{(j,t+1)}}(\cdot;\mathfrak{s})$ (resp. $V_t^{\hat{\pi}^{(i,t)}}(\cdot;\mathfrak{s})$) is defined as in (63) (resp. (64)) with $\pi^{(t+1)} = \hat{\pi}^{(j,t+1)}$ (resp. $\pi^{(t)} = \hat{\pi}^{(i,t)}$).

Proof of Lemma H.5. Let $\mathfrak{s} \in \mathcal{S}_h$ be fixed. We proceed by backwards induction to show that for all $\tau \in [h-1]$, the learned partial policies $\hat{\pi}^{(1,\tau)}, \ldots, \hat{\pi}^{(S,\tau)} \in \Pi_{\mathsf{NM}}^{\tau:h-1}$ have the property that

$$\min_{i\in[S]} \overline{\mathbb{E}}^{\pi_\star^{(\mathfrak{s})}}\left[V_\tau^{\pi_\star^{(\mathfrak{s})}}(\boldsymbol{x}_\tau;\mathfrak{s}) - V_\tau^{\hat{\pi}^{(i,\tau)}}(\boldsymbol{x}_\tau;\mathfrak{s})\right] \le \Sigma_\tau, \tag{66}$$

where $\Sigma_\tau \coloneqq \sum_{t=\tau}^{h-1} \min_{i\in[S]} \max_{j\in[S]} \overline{\mathbb{E}}^{\pi_\star^{(\mathfrak{s})}}\left[Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t, \pi_\star^{(\mathfrak{s})}(\boldsymbol{x}_t);\mathfrak{s}) - V_t^{\hat{\pi}^{(i,t)}}(\boldsymbol{x}_t;\mathfrak{s})\right]$.

[11] We recall that we have assumed the state $\mathfrak{t}_h$ emits itself as an observation.

Base case. The base case (i.e. $\tau = h-1$) reduces to showing that for any $\mathfrak{s} \in \mathcal{S}_h$,

$$\overline{\mathbb{E}}^{\pi_\star^{(\mathfrak{s})}}\left[V_{h-1}^{\pi_\star^{(\mathfrak{s})}}(\boldsymbol{x}_{h-1};\mathfrak{s})\right] \le \max_{j\in[S]} \overline{\mathbb{E}}^{\pi_\star^{(\mathfrak{s})}}\left[Q_{h-1}^{\hat{\pi}^{(j,h)}}(\boldsymbol{x}_{h-1}, \pi_\star^{(\mathfrak{s})}(\boldsymbol{x}_{h-1});\mathfrak{s})\right].$$

This holds with equality regardless of how $\{\hat{\pi}^{(i,h)}\}_{i\in[S]}$ are chosen, since the reward function for layer h is independent of the actions taken at that layer.

Inductive step. Now, let $t \in [h-2]$ and suppose that Eq. (68) holds for $\tau = t+1$, and we show that it holds for $\tau = t$. Fix $\mathfrak{s} \in \mathcal{S}_{h,\epsilon}$ and let $\pi_\star \equiv \pi_\star^{(\mathfrak{s})}$ to simplify notation. Further, fix $\mathfrak{i} \in [S]$ and let $\mathfrak{j}$ be the minimizer of the left-hand side of Eq. (68) for $\tau = t+1$. By Lemma 4.1, we know that $\pi_\star(x_t) = \mathfrak{a}$ for all $x_t \in \operatorname{supp} q(\cdot \mid s_t)$ and $s_t \in \mathcal{S}_t \setminus \mathcal{S}_{t,\epsilon}$. Therefore, since the V-function at layer $t+1$ is zero on the terminal state $\mathfrak{t}_{t+1}$ (see definition of the V-function in (64)), we have

$$\begin{aligned}
\overline{\mathbb{E}}^{\pi_\star}\left[V_{t+1}^{\hat{\pi}^{(\mathfrak{j},t+1)}}(\boldsymbol{x}_{t+1};\mathfrak{s})\right] &= \sum_{s_t\in\mathfrak{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}^{\pi_\star}\left[V_{t+1}^{\hat{\pi}^{(\mathfrak{j},t+1)}}(x_{t+1};\mathfrak{s})\middle|\, \boldsymbol{s}_t = s_t\right], \\
&= -\sum_{s_t\in\mathfrak{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[r_t(\boldsymbol{x}_t;\mathfrak{s})\right] \\
&\quad + \sum_{s_t\in\mathfrak{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[Q_t^{\hat{\pi}^{(\mathfrak{j},t+1)}}(\boldsymbol{x}_t, \pi_\star(\boldsymbol{x}_t);\mathfrak{s})\right], \\
&= \sum_{s_t\in\mathfrak{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[Q_t^{\hat{\pi}^{(\mathfrak{j},t+1)}}(\boldsymbol{x}_t, \pi_\star(\boldsymbol{x}_t);\mathfrak{s})\right], \qquad (67)
\end{aligned}$$

where the last equality follows by the fact that $t < h$ and that the reward $r_t(\cdot;\mathfrak{s}) \equiv 0$ in this case. Combining Eq. (69) with Eq. (68), we have

$$\begin{aligned}
&\sum_{s_t\in\mathfrak{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[Q_t^{\hat{\pi}^{(\mathfrak{j},t+1)}}(\boldsymbol{x}_t, \pi_\star(\boldsymbol{x}_t);\mathfrak{s}) - V_t^{\hat{\pi}^{(\mathfrak{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right] \\
&\ge \overline{\mathbb{E}}^{\pi_\star}\left[V_{t+1}^{\hat{\pi}^{(\mathfrak{j},t+1)}}(\boldsymbol{x}_{t+1};\mathfrak{s})\right] - \sum_{s_t\in\mathcal{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[V_t^{\hat{\pi}^{(\mathfrak{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right], \quad \text{(by Eq. (69))} \\
&\ge \overline{\mathbb{E}}^{\pi_\star}[V_{t+1}^{\pi_\star}(\boldsymbol{x}_{t+1};\mathfrak{s})] - \sum_{s_t\in\mathcal{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[V_t^{\hat{\pi}^{(\mathfrak{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right] - \Sigma_{t+1}, \quad \text{(by induction)} \\
&= \sum_{s_t\in\mathfrak{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}^{\pi_\star}\left[V_{t+1}^{\pi_\star}(\boldsymbol{x}_{t+1};\mathfrak{s})\middle|\, \boldsymbol{s}_t = s_t\right] - \sum_{s_t\in\mathcal{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[V_t^{\hat{\pi}^{(\mathfrak{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right] - \Sigma_{t+1}, \\
&= \sum_{s_t\in\mathfrak{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[V_t^{\pi_\star}(\boldsymbol{x}_t;\mathfrak{s}) - V_t^{\hat{\pi}^{(\mathfrak{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right] - \Sigma_{t+1}, \qquad (68)
\end{aligned}$$

where in the last step we used that the rewards are zero except at layer h and that $\pi_\star(x_t) = \mathfrak{a}$ for $x_t \in \operatorname{supp} q(\cdot \mid s_t)$ with $s_t \in \mathcal{S}_t \setminus \mathcal{S}_{t,\epsilon}$ (by Lemma 4.1). For such an x_t, we also have that $V_t^{\pi_\star}(x_t;\mathfrak{s}) = 0$, and so since $V_t^{\hat{\pi}^{(\mathfrak{i},t)}}(\cdot;\mathfrak{s})$ is non-negative, Eq. (70) implies that

$$\begin{aligned}
&\overline{\mathbb{E}}^{\pi_\star}\left[V_t^{\pi_\star}(\boldsymbol{x}_t;\mathfrak{s}) - V_t^{\hat{\pi}^{(\mathfrak{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right] \\
&\le \sum_{s_t\in\mathfrak{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[V_t^{\pi_\star}(\boldsymbol{x}_t;\mathfrak{s}) - V_t^{\hat{\pi}^{(\mathfrak{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right], \\
&\le \Sigma_{t+1} + \sum_{s_t\in\mathfrak{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[Q_t^{\hat{\pi}^{(\mathfrak{j},t+1)}}(\boldsymbol{x}_t, \pi_\star(\boldsymbol{x}_t);\mathfrak{s}) - V_t^{\hat{\pi}^{(\mathfrak{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right], \\
&\le \Sigma_{t+1} + \max_{j\in[S]} \sum_{s_t\in\mathfrak{S}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t, \pi_\star(\boldsymbol{x}_t);\mathfrak{s}) - V_t^{\hat{\pi}^{(\mathfrak{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right]. \qquad (69)
\end{aligned}$$

Recall that $\mathfrak{i}$ was chosen arbitrarily in $[S]$, and so taking the min over $\mathfrak{i} \in [S]$ on both sides of (71) implies the desired result. □

H.3.3. Local Error Guarantee

The following lemma, which is a restatement of Lemma E.4, gives us a way of bounding the error terms appearing on the right-hand side of the inequality in Lemma H.5.

Lemma H.6 (Restatement of Lemma E.4)**.** *Let $\epsilon, \delta \in (0,1)$, $h \in [H]$, and suppose Φ satisfies Assumption 2.1. If the policies in $\Psi^{(2)} \cup \cdots \cup \Psi^{(h-1)}$ never take the terminal action $\mathfrak{a}$, then for any $t \in [h-1]$, there is an event $\mathcal{E}_t$ of probability at least $1 - \frac{\delta}{H^2}$ under which the partial policies $\{\hat{\pi}^{(j,\tau)}\}_{j\in[S],\tau\in[h-1]}$, constructed during the call to $\texttt{IKDP}(\Psi^{(1)}, \dots, \Psi^{(h-1)}, \Phi, n)$ are such that for any $s_h \in \mathcal{S}_h$ there exists $\mathfrak{i} \in [S]$ that satisfies*

$$0 \le \sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[\max_{a\in\mathcal{A},j\in[S]} Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t, a; s_h) - V_t^{\hat{\pi}^{(\cdot,t)}}(\boldsymbol{x}_t; s_h)\right] \le 2S^3AC\varepsilon_{\texttt{stat}}(n, \tfrac{\delta}{H^2}), \quad \forall s_t \in \mathcal{S}_t, \tag{70}$$

where $\varepsilon_{\texttt{stat}}(\cdot,\cdot)$ and $C > 0$ are as in Lemma H.2; here $C > 0$ is an absolute constant independent of problem parameters.

Proof of Lemma H.6. To simplify notation throughout the proof, let

$$P^{(t)}(s_h \mid s_t, a) \coloneqq \frac{1}{S}\sum_{j\in[S]} \overline{\mathbb{P}}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{s}_h = s_h \mid \boldsymbol{s}_t = s_t, \boldsymbol{a}_t = a]. \tag{71}$$

By Lemma H.2 and Jensen's inequality, we have that with probability at least $1 - \delta/H^2$, the solution $(\hat{f}^{(t)}, \hat{\phi}^{(t)})$ of the conditional density estimation problem in Line 7 of Algorithm 2 satisfies,

$$\overline{\mathbb{E}}_{\boldsymbol{s}_t\sim\texttt{unif}(\Psi^{(t)}),\boldsymbol{x}_t\sim q(\cdot|\boldsymbol{s}_t),\boldsymbol{a}'\sim\pi_{\texttt{unif}}}\left[\sum_{s_h\in\mathcal{S}_h,x_h} P^{(t)}(s_h \mid \boldsymbol{s}_t, \boldsymbol{a}')q(x_h \mid s_h)\max_{a\in\mathcal{A},j\in[S]}\left|\mathrm{err}^{(t)}(a, j, \boldsymbol{x}_t, x_h)\right|\right] \le \varepsilon'_{\texttt{stat}}, \tag{72}$$

where $\varepsilon'_{\texttt{stat}} \coloneqq C \cdot \varepsilon_{\texttt{stat}}(n, \frac{\delta}{H^2})$, C is an absolute constant independent of t, h, and other problem parameters,

$$\mathrm{err}^{(t)}(a, j, x_t, x_h) \coloneqq \hat{f}^{(t)}((a,j) \mid \hat{\phi}^{(t)}(x_t), \hat{\phi}^{(t)}(x_h)) - P^{(t)}_{\texttt{bayes}}((a,j) \mid \phi_\star(x_t), \phi_\star(x_h)),$$

and finally

$$P^{(t)}_{\texttt{bayes}}((a,j) \mid s_t, s_h) \coloneqq \frac{\overline{\mathbb{P}}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{s}_h = s_h \mid \boldsymbol{s}_t = s_t, \boldsymbol{a}_t = a]}{\sum_{a'\in\mathcal{A},i\in[S]}\overline{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{s}_h = s_h \mid \boldsymbol{s}_t = s_t, \boldsymbol{a}_t = a']}. \tag{73}$$

We denote this event by $\mathcal{E}_t$. Note that to rewrite the result of Lemma H.2 as (74), we use that the policies $\hat{\pi}^{(j,t+1)}$, $j \in [S]$, while non-Markovian, only depend on $\boldsymbol{x}_{t+1}, \dots, \boldsymbol{x}_{h-1}$. Moving forward, we condition on $\mathcal{E}_t$.

Fix $\mathfrak{s}' \in \mathcal{S}_h$ and let $\mathfrak{i} \in \arg\max_{i\in[S]}\sum_{x_h}\mathbb{I}\{\hat{\phi}^{(t)}(x_h) = i\}q(x_h \mid \mathfrak{s}')$, and note that

$$\sum_{x_h}\mathbb{I}\{\hat{\phi}^{(t)}(x_h) = \mathfrak{i}\}q(x_h \mid \mathfrak{s}') \ge \frac{1}{S}. \tag{74}$$

Further, let $\mathcal{S}_t^+$ be the subset of states defined by

$$\mathcal{S}_t^+ \coloneqq \left\{\tilde{s} \in \mathcal{S}_t : \sum_{\pi\in\Psi^{(t)}}\bar{d}^{\pi}(\tilde{s})\sum_{a\in\mathcal{A}} P^{(t)}(\mathfrak{s}' \mid \tilde{s}, a) > 0\right\}.$$

Now, fix $\mathfrak{s} \in \mathcal{S}_t^+$. From Eq. (74), we have that

$$\begin{aligned}
&\sum_{\pi\in\Psi^{(t)},x_t,x_h,a'\in\mathfrak{A}} \bar{d}^{\pi}(\mathfrak{s})q(x_t \mid \mathfrak{s})P^{(t)}(\mathfrak{s}' \mid \mathfrak{s}, a')q(x_h \mid \mathfrak{s}')\max_{a\in\mathcal{A},j\in[S]}\left|\mathrm{err}^{(t)}(a, j, x_t, x_h)\right| \\
&\le \sum_{\pi\in\Psi^{(t)},x_t,x_h,a'\in\mathcal{A},s_t\in\mathcal{S}_t,s_h\in\mathcal{S}_h} \bar{d}^{\pi}(s_t)q(x_t \mid s_t)P^{(t)}(s_h \mid s_t, a')q(x_h \mid s_h)\max_{a\in\mathcal{A},j\in[S]}\left|\mathrm{err}^{(t)}(a, j, x_t, x_h)\right|, \\
&\le SA\varepsilon'_{\texttt{stat}}.
\end{aligned} \tag{75}$$

Applying Eq. (76) within Eq. (77) implies that

$$\begin{aligned}
&SA\varepsilon'_{\mathsf{stat}} \\
&\geq \sum_{\substack{\pi\in\Psi^{(t)},x_t,a'\in\mathcal{A},\\ x_h:\hat{\phi}^{(t)}(x_h)=\mathrm{i}}} \bar{d}^{\pi}(\mathfrak{s})q(x_t\mid\mathfrak{s})P^{(t)}(\mathfrak{s}'\mid\mathfrak{s},a')q(x_h\mid\mathfrak{s}') \max_{a\in\mathcal{A},j\in[S]}\left|\hat{f}^{(t)}((a,j)\mid\hat{\phi}^{(t)}(x_t),\mathrm{i})-P^{(t)}_{\mathsf{bayes}}((a,j)\mid\mathfrak{s},\mathfrak{s}')\right|, \\
&\geq \frac{1}{S}\sum_{\pi\in\Psi^{(t)},x_t,a'\in\mathfrak{A}} \bar{d}^{\pi}(\mathfrak{s})q(x_t\mid\mathfrak{s})P^{(t)}(\mathfrak{s}'\mid\mathfrak{s},a') \max_{a\in\mathcal{A},j\in[S]}\left|\hat{f}^{(t)}((a,j)\mid\hat{\phi}^{(t)}(x_t),\mathrm{i})-P^{(t)}_{\mathsf{bayes}}((a,j)\mid\mathfrak{s},\mathfrak{s}')\right|.
\end{aligned}$$

By rearranging and using that $\sum_{\pi\in\Psi^{(t)}}\bar{d}^{\pi}(\mathfrak{s})\sum_{a'\in\mathcal{A}}P^{(t)}(\mathfrak{s}'\mid\mathfrak{s},a')>0$ (since $\mathfrak{s}\in\mathcal{S}^{+}_t$), we get

$$\bar{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot\mid\mathfrak{s})}\left[\max_{a\in\mathcal{A},j\in[S]}\left|\hat{f}^{(t)}((a,j)\mid\hat{\phi}^{(t)}(\boldsymbol{x}_t),\mathrm{i})-P^{(t)}_{\mathsf{bayes}}((a,j)\mid\mathfrak{s},\mathfrak{s}')\right|\right]\leq\frac{S^2A\varepsilon'_{\mathsf{stat}}}{\sum_{\pi\in\Psi^{(t)}}\bar{d}^{\pi}(\mathfrak{s})\sum_{a'\in\mathcal{A}}P^{(t)}(\mathfrak{s}'\mid\mathfrak{s},a')}. \tag{76}$$

Now, let $\hat{a}^{(\mathrm{i},t)}(x_t),\hat{\imath}^{(\mathrm{i},t)}(x_t)\in\arg\max_{a\in\mathcal{A},j\in[S]}\hat{f}^{(t)}((a,j)\mid\hat{\phi}^{(t)}(x_t),\mathrm{i})$ and note that $\hat{a}^{(\mathrm{i},t)}(x_t)=\hat{\pi}^{(\mathrm{i},t)}(x_t)$, where $\hat{\pi}^{(\mathrm{i},t)}(x_t)$ is defined as in Algorithm 2. With this, Eq. (78), and the fact that $|\|y\|_\infty-\|z\|_\infty|\leq\|y-z\|_\infty$, for all $y,z\in\mathbb{R}^{A\times S}$ we have

$$\begin{aligned}
&\max_{a\in\mathfrak{A},j\in[S]}P^{(t)}_{\mathsf{bayes}}((a,j)\mid\mathfrak{s},\mathfrak{s}') \\
&\leq\bar{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot\mid\mathfrak{s})}\left[\max_{a\in\mathcal{A},j\in[S]}\hat{f}^{(t)}((a,j)\mid\hat{\phi}^{(t)}(\boldsymbol{x}_t),\mathrm{i})\right]+\frac{S^2A\varepsilon'_{\mathsf{stat}}}{\sum_{\pi\in\Psi^{(t)},a'\in\mathfrak{A}}\bar{d}^{\pi}(\mathfrak{s})P^{(t)}(\mathfrak{s}'\mid\mathfrak{s},a')}, \\
&=\bar{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot\mid\mathfrak{s})}\left[\hat{f}^{(t)}((\hat{a}^{(\mathrm{i},t)}(\boldsymbol{x}_t),\hat{\imath}^{(\mathrm{i},t)}(\boldsymbol{x}_t))\mid\hat{\phi}^{(t)}(\boldsymbol{x}_t),\mathrm{i})\right]+\frac{S^2A\varepsilon'_{\mathsf{stat}}}{\sum_{\pi\in\Psi^{(t)},a'\in\mathfrak{A}}\bar{d}^{\pi}(\mathfrak{s})P^{(t)}(\mathfrak{s}'\mid\mathfrak{s},a')}, \\
&\leq\bar{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot\mid\mathfrak{s})}\left[P^{(t)}_{\mathsf{bayes}}((\hat{a}^{(\mathrm{i},t)}(\boldsymbol{x}_t),\hat{\imath}^{(\mathrm{i},t)}(\boldsymbol{x}_t))\mid\mathfrak{s},\mathfrak{s}')\right]+\frac{2S^2A\varepsilon'_{\mathsf{stat}}}{\sum_{\pi\in\Psi^{(t)}}\bar{d}^{\pi}(\mathfrak{s})\sum_{a'\in\mathcal{A}}P^{(t)}(\mathfrak{s}'\mid\mathfrak{s},a')}.
\end{aligned}\tag{77}$$

Now, observe that from the definition of $P^{(t)}_{\mathsf{bayes}}$ in Eq. (75), we have that for all $s_t\in\mathcal{S}_t$, $a\in\mathcal{A}$, $j\in[S]$, and $x_t\in\operatorname{supp}q(\cdot\mid s_t)$,

$$Q^{\hat{\pi}^{(j,t+1)}}_t(x_t,a;\mathfrak{s}')=\bar{\mathbb{P}}^{\hat{\pi}^{(j,t+1)}}[\boldsymbol{s}_h=\mathfrak{s}'\mid\boldsymbol{s}_t=s_t,\boldsymbol{a}_t=a]=P^{(t)}_{\mathsf{bayes}}((a,j)\mid s_t,\mathfrak{s}')\sum_{a'\in\mathcal{A}}S\cdot P^{(t)}(\mathfrak{s}'\mid s_t,a'), \tag{78}$$

where we have used that $\sum_{i\in[S],a'\in\mathfrak{A}}\bar{\mathbb{P}}^{\hat{\pi}^{(i,t+1)}}[\boldsymbol{s}_h=\mathfrak{s}'\mid\boldsymbol{s}_t=s_t,\boldsymbol{a}_t=a']=S\cdot\sum_{a'\in\mathcal{A}}P^{(t)}(\mathfrak{s}'\mid s_t,a')$ by definition of $P^{(t)}$ in (73). Combining this with Eq. (79), we have

$$\begin{aligned}
&\bar{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot\mid\mathfrak{s})}\left[\max_{a\in\mathcal{A},j\in[S]}Q^{\hat{\pi}^{(j,t+1)}}_t(\boldsymbol{x}_t,a;\mathfrak{s}')\right] \\
&\leq\bar{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot\mid\mathfrak{s})}\left[P^{(t)}_{\mathsf{bayes}}((\hat{a}^{(\mathrm{i},t)}(\boldsymbol{x}_t),\hat{\imath}^{(\mathrm{i},t)}(\boldsymbol{x}_t))\mid\mathfrak{s},\mathfrak{s}')\right]\cdot\sum_{a'\in\mathcal{A}}S\cdot P^{(t)}(\mathfrak{s}'\mid\mathfrak{s},a')+\frac{2S^3A\varepsilon'_{\mathsf{stat}}}{\sum_{\pi\in\Psi^{(t)}}\bar{d}^{\pi}(\mathfrak{s})},\quad\text{(by (80) \& (79))} \\
&=\bar{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot\mid\mathfrak{s})}\left[\bar{\mathbb{P}}^{\hat{\pi}^{(\hat{\imath}(\boldsymbol{x}_t),t+1)}}[\boldsymbol{s}_h=\mathfrak{s}'\mid\boldsymbol{s}_t=\mathfrak{s},\boldsymbol{a}_t=\hat{a}^{(\mathrm{i},t)}(\boldsymbol{x}_t)]\right]+\frac{2S^3A\varepsilon'_{\mathsf{stat}}}{\sum_{\pi\in\Psi^{(t)}}\bar{d}^{\pi}(\mathfrak{s})},\quad\text{(where }\hat{\imath}(x)\coloneqq\hat{\imath}^{(\mathrm{i},t)}(x)\text{)} \\
&=\bar{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot\mid\mathfrak{s})}\left[V^{\hat{\pi}^{(\mathrm{i},t)}}_t(\boldsymbol{x}_t;\mathfrak{s}')\right]+\frac{2S^3A\varepsilon'_{\mathsf{stat}}}{\sum_{\pi\in\Psi^{(t)}}\bar{d}^{\pi}(\mathfrak{s})},
\end{aligned}$$

where the first equality uses Eq. (80) once more and the second equality follows from Lemma H.4 and the definition of $\hat{\pi}^{(\mathrm{i},t)}$ in Algorithm 2. Summarizing, we have shown that

$$\sum_{\pi\in\Psi^{(t)}}\bar{d}^{\pi}(s_t)\bar{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot\mid s_t)}\left[\max_{a\in\mathcal{A},j\in[S]}Q^{\hat{\pi}^{(j,t+1)}}_t(\boldsymbol{x}_t,a;')-V^{\hat{\pi}^{(,t)}}_t(\boldsymbol{x}_t;')\right]\leq 2S^3A\varepsilon'_{\mathsf{stat}},\quad\forall s_t\in\mathcal{S}^{+}_t. \tag{79}$$

We now show that the LHS of (81) is larger than 0. We have that for all $s_t\in\mathcal{S}_t$,

$$\max_{a\in\mathfrak{A},j\in[S]}P^{(t)}_{\mathsf{bayes}}((a,j)\mid s_t,\mathfrak{s}')\geq\bar{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot\mid s_t)}\left[P^{(t)}_{\mathsf{bayes}}((\hat{a}^{(\mathrm{i},t)}(\boldsymbol{x}_t),\hat{\imath}^{(\mathrm{i},t)}(\boldsymbol{x}_t))\mid s_t,\mathfrak{s}')\right].$$

Combining this with Eq. (80) implies that for all $s_t \in \mathcal{S}_t$

$$\overline{\mathbb{E}}_{\boldsymbol{x}_t \sim q(\cdot|s_t)}\left[V_t^{\hat{\pi}^{(\mathrm{i},t)}}(\boldsymbol{x}_t;\mathfrak{s}')\right] \le \overline{\mathbb{E}}_{\boldsymbol{x}_t \sim q(\cdot|s_t)}\left[\max_{a\in\mathfrak{A},j\in[S]} Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t,a;\mathfrak{s}')\right]. \tag{80}$$

Therefore, we have

$$0 \le \sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[\max_{a\in\mathcal{A},j\in[S]} Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t,a;') - V_t^{\hat{\pi}^{(,t)}}(\boldsymbol{x}_t;')\right] \le 2S^3A\varepsilon'_{\mathtt{stat}}, \quad \forall s_t \in \mathcal{S}_t^+. \tag{81}$$

On the other hand, for any $s_t \notin \mathcal{S}_t^+$, we have $\sum_{\pi\in\Psi^{(t)},a'\in\mathfrak{A}} \bar{d}^{\pi}(s_t)P^{(t)}(\mathfrak{s}' \mid s_t,a') = 0$ (by definition of $\mathcal{S}_t^+$), and so by (80) and (82), we have

$$\begin{aligned}
\sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[V_t^{\hat{\pi}^{(\mathrm{i},t)}}(\boldsymbol{x}_t;\mathfrak{s}')\right] &\le \sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[\max_{a\in\mathfrak{A},j\in[S]} Q_t^{\hat{\pi}^{(j,t+1)}}(x_t,a;\mathfrak{s}')\right], \\
&\le S\sum_{\pi\in\Psi^{(t)},a'\in\mathfrak{A}} \bar{d}^{\pi}(s_t)P^{(t)}(\mathfrak{s}' \mid s_t,a'), \quad \text{(by (80))} \\
&= 0.
\end{aligned}$$

This implies that Eq. (83) also holds for $s_t \in \mathcal{S}_t \setminus \mathcal{S}_t^+$, giving the desired result. □

H.3.4. Proof of Theorem E.3

Proof of Theorem E.3. In light of Lemma H.5, it suffices to show that, for any $t \in [h-1]$, there is an event $\mathcal{E}_t$ which occurs with probability at least $1-\delta/H^2$, under which for any $s \in [S]$,

$$\sigma_t := \min_{i\in[S]}\max_{j\in[S]} \mathbb{E}^{\pi_\star^{(s)}}\left[\mathbb{I}\{\boldsymbol{s}_t \in \mathcal{S}_{t,\epsilon}\}\left(Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t,\pi_\star^{(s)}(\boldsymbol{x}_t);s) - V_t^{\hat{\pi}^{(i,t)}}(\boldsymbol{x}_t;s)\right)\right] \le \frac{\epsilon}{2H}, \tag{82}$$

where $\pi_\star^{(s)} \in \arg\max_{\pi\in\overline{\Pi}_\epsilon} \bar{d}^{\pi}(s)$ and $V_\tau^{\pi}(\cdot;s)$ is the V-function at layer $\tau \in [h-1]$ with respect to the partial policy π for the BMDP $\overline{\mathcal{M}}$ with rewards $r_t(x;s) = \mathbf{1}\{\phi_\star(x) = s\}$, $t \in [h]$—see Definition in (64). By summing Eq. (84) over $t = 1,\dots,h-1$, and using Lemma H.5 together with a union bound, we will be able to prove the desired result.

Fix $t \in [h-1]$ and let $\mathcal{E}_t$ be the event of Lemma H.6. Recall that $\mathbb{P}[\mathcal{E}_t] \ge 1-\delta/H^2$. In what follows, we condition on $\mathcal{E}_t$ and prove (84). Fix $\mathfrak{s} \in \mathcal{S}_{h,t}$ and let $\pi_\star \equiv \pi_\star^{(\mathfrak{s})}$. Further, let $\mathfrak{i}$ be as in Lemma H.6 with $s_h = \mathfrak{s}$. Since $\Psi^{(t)}$ is an $(1/2,\epsilon)$-policy cover relative to $\overline{\Pi}_\epsilon$ at layer t and $\pi_\star \in \overline{\Pi}_\epsilon$, we have that

$$\bar{d}^{\pi_\star}(s_t) \le \max_{\tilde{\pi}\in\overline{\Pi}_\epsilon} \bar{d}^{\tilde{\pi}}(s_t) \le 2\sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t), \quad \forall s_t \in \mathcal{S}_{t,\epsilon}. \tag{83}$$

The last inequality and the definition of $\mathcal{S}_{t,\epsilon}$ implies that for all $s_t \in \mathcal{S}_{t,\epsilon}$, $\sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t) > 0$. This, together with Lemma H.6 (in particular, the left-hand side inequality in (72)) implies that

$$\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[\max_{a\in\mathfrak{A},j\in[S]} Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t,a;\mathfrak{s}) - V_t^{\hat{\pi}^{(\mathrm{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right] \ge 0. \tag{84}$$

Thus, for any $s_t \in \mathcal{S}_{t,\epsilon}$, we have

$$\begin{aligned}
&\bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[\max_{a\in\mathfrak{A},j\in[S]} Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t,a;\mathfrak{s}) - V_t^{\hat{\pi}^{(\mathrm{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right], \\
&= \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[\max_{a\in\mathfrak{A},j\in[S]} Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t,a;\mathfrak{s}) - V_t^{\hat{\pi}^{(\mathrm{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right], \quad \text{(justified below)} \\
&\le 2\sum_{\pi\in\Psi^{(t)}} \bar{d}^{\pi}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[\max_{a\in\mathfrak{A},j\in[S]} Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t,a;\mathfrak{s}) - V_t^{\hat{\pi}^{(\mathrm{i},t)}}(\boldsymbol{x}_t;\mathfrak{s})\right], \quad \text{(by (86) and (85))}
\end{aligned} \tag{85}$$

$$\leq 4S^3AC\varepsilon_{\mathtt{stat}}(n,\delta/H^2), \tag{86}$$

for some absolute constant $C>0$; the last inequality follows by Lemma H.6 (in particular, the right-hand side in inequality in Eq. (72)). Now, Eq. (87) follows from the fact that

$$\max_{a\in\bar{\mathfrak{A}}} Q_t^{\hat{\pi}^{(j,t+1)}}(x_t,a;\mathfrak{s}) = \max_{a\in\mathcal{A}} Q_t^{\hat{\pi}^{(j,t+1)}}(x_t,a;\mathfrak{s}), \quad \forall j\in[S],$$

since $Q_t^{\hat{\pi}^{(j,t+1)}}(x_t,\mathfrak{a};\mathfrak{s})=0$. On the other hand, by definition of σ_t in (84), we have

$$\begin{aligned}
\sigma_t &\leq \sum_{s_t\in\bar{\mathfrak{S}}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\max_{j\in[S]}\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t,\pi_\star(\boldsymbol{x}_t);\mathfrak{s}) - V_t^{\hat{\pi}^{(i,t)}}(\boldsymbol{x}_t;\mathfrak{s})\right],\\
&\leq \sum_{s_t\in\bar{\mathfrak{S}}_{t,\epsilon}} \bar{d}^{\pi_\star}(s_t)\overline{\mathbb{E}}_{\boldsymbol{x}_t\sim q(\cdot|s_t)}\left[\max_{a\in\bar{\mathfrak{A}},j\in[S]} Q_t^{\hat{\pi}^{(j,t+1)}}(\boldsymbol{x}_t,a;\mathfrak{s}) - V_t^{\hat{\pi}^{(i,t)}}(\boldsymbol{x}_t;\mathfrak{s})\right],\\
&\leq 4S^4AC\varepsilon_{\mathtt{stat}}(n,\frac{\delta}{H^2}). \quad \text{(by (88))}
\end{aligned} \tag{87}$$

Now, by choosing n large enough such that $8AS^4HC\varepsilon_{\mathtt{stat}}(n,\frac{\delta}{H^2})\leq\epsilon$ (as in the theorem's statement), we get

$$\sigma_t \leq \frac{\epsilon}{2H}. \tag{88}$$

Thus, under the event $\mathcal{E}' \coloneqq \mathcal{E}_1\cup\cdots\cup\mathcal{E}_{h-1}$ (note that $\mathbb{P}[\mathcal{E}']\geq 1-\delta/H$ by a union bound), we have by Lemma H.5 and Eq. (90) that

$$\min_{i\in[S]}\overline{\mathbb{E}}^{\pi_\star}\left[V_1^{\pi_\star}(\boldsymbol{x}_1;\mathfrak{s}) - V_1^{\hat{\pi}^{(i,1)}}(\boldsymbol{x}_1;\mathfrak{s})\right] \leq \sum_{t=1}^{h-1}\sigma_t \leq \frac{\epsilon}{2}. \tag{89}$$

Note that $\overline{\mathbb{E}}^{\pi_\star}[V_1^{\pi_\star}(\boldsymbol{x}_1;\mathfrak{s})] = \max_{\pi\in\bar{\Pi}_\epsilon}\bar{d}^{\pi}(\mathfrak{s})$ and $\overline{\mathbb{E}}^{\pi_\star}[V_1^{\hat{\pi}^{(i,1)}}(\boldsymbol{x}_1;\mathfrak{s})] = \bar{d}^{\hat{\pi}^{(i,1)}}(\mathfrak{s})$, by definition of $\pi_\star$ and the V-function. Thus, (91) implies that

$$\max_{i\in[S]}\bar{d}^{\hat{\pi}^{(i,1)}}(\mathfrak{s}) \geq \max_{\pi\in\bar{\Pi}_\epsilon}\bar{d}^{\pi}(\mathfrak{s}) - \frac{\epsilon}{2} \geq \frac{1}{2}\max_{\pi\in\bar{\Pi}_\epsilon}\bar{d}^{\pi}(\mathfrak{s}),$$

where the last inequality follows from the fact that $\max_{\pi\in\bar{\Pi}_\epsilon}\bar{d}^{\pi}(\mathfrak{s})\geq\epsilon$, since $\mathfrak{s}\in\mathcal{S}_{h,\epsilon}$. This means that $\Psi^{(h)}=\{\hat{\pi}^{(i,1)}:i\in[S]\}$ is a $(1/2,\epsilon)$-policy cover relative to $\bar{\Pi}_\epsilon$ for layer h in $\bar{\mathcal{M}}$, which completes the proof. □

I. Proofs for Reward-Based RL

Lemma I.1. *Let $n\geq 1$ and $\delta\in(0,1)$, and define $\varepsilon_{\mathtt{stat}}(n,\delta)\coloneqq n^{-1/2}\sqrt{SA\log n+\log(|\Phi|/\delta)}$. Further, suppose that Assumption 2.1 and Assumption B.1 hold. Then, there exists an absolute constant $C>0$ such that for all $h\in H$ the solution $(\hat{f}^{(h)},\hat{\phi}^{(h)})$ of the least-squares problem in Line 6 of Algorithm 4 satisfies with probability at least $1-\delta$,*

$$\mathbb{E}^{\mathtt{unif}(\Psi^{(h)})}\left[\max_{a\in\mathcal{A}}\left(\hat{f}^{(h)}(\hat{\phi}^{(h)}(\boldsymbol{x}_h),a) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,a)\right)^2\right] \leq C^2\cdot\varepsilon^2_{\mathtt{stat}}(n,\delta).$$

Proof of Lemma I.1. Fix $h\in[h-1]$ and let $\tilde{f}^{(h)}_{\mathsf{bayes}}$ be as in Lemma I.2. By Lemma I.2, $(\tilde{f}^{(h)}_{\mathsf{bayes}},\phi_\star)$ is the Bayes-optimal solution of the least-square problem in Line 6 of Algorithm 4. And so, by Assumption 2.1 and a standard guarantee for least-square regression (see e.g. (Van de Geer and van de Geer, 2000)), there exists an absolute constant $C'>0$ (independent of h and any other problem parameter) such that with probability at least $1-\delta$,

$$\mathbb{E}^{\mathtt{unif}(\Psi^{(h)})}\left[\max_{a\in\mathcal{A}}\left(\hat{f}^{(h)}(\hat{\phi}^{(h)}(\boldsymbol{x}_h),a) - \tilde{f}^{(h)}_{\mathsf{bayes}}(\phi_\star(\boldsymbol{x}_h),a)\right)^2\right] \leq \tilde{\varepsilon}^2_{\mathtt{stat}}(n,\delta),$$

where $\tilde{\varepsilon}^2_{\mathsf{stat}}(n,\delta) \coloneqq C' \log \mathcal{N}_{\mathcal{F}}(1/n) + C' \log(|\Phi|/\delta)$ and $\mathcal{N}_{\mathcal{F}}(1/n)$ denotes the $\frac{1}{n}$-covering number of the set $\mathcal{F} \coloneqq \{f : [S] \times \mathcal{A} \to \mathbb{R}_{\geq 0}\}$ in ℓ_∞ distance. It is easy to verify that $\mathcal{N}_{\mathcal{F}}(1/n) \leq n^{AS}$, and so by setting $C^2 \coloneqq C'$, we have

$$\tilde{\varepsilon}^2_{\mathsf{stat}}(n,\delta) \leq C^2 \cdot \varepsilon^2_{\mathsf{stat}}(n,\delta).$$

Now, by the expression of $\tilde{f}^{(h)}_{\mathsf{bayes}}$ in Eq. (93) and Assumption B.1, we have that $\tilde{f}^{(h)}_{\mathsf{bayes}}(\phi_\star(x), a) = Q^{\hat{\pi}^{(h+1)}}_h(x,a)$, which completes the proof. □

Lemma I.2. *Let $h \in [H]$ and consider of the unconstrained problem*

$$f^{(h)}_{\mathsf{bayes}} \in \operatorname*{arg\,min}_{f : \mathcal{X}_h \times \mathcal{A} \to \mathbb{R}_{\geq 0}} \mathbb{E}^{\mathsf{unif}(\Psi^{(h)}) \circ_h \pi_{\mathsf{unif}} \circ_{h+1} \hat{\pi}^{(h+1)}} \left[\left(f(\boldsymbol{x}_h, \boldsymbol{a}_h) - \sum_{\tau=h}^{H} \boldsymbol{r}_\tau \right)^2 \right], \tag{90}$$

where $(\boldsymbol{r}_h)$ are the reward random variables and $\hat{\pi}^{(h+1)} \in \Pi^{h+1:H}_{\mathsf{M}}$ is as in Algorithm 4. Then, under Assumption B.1 for any $a \in \mathcal{A}$, $x \in \mathcal{X}_h$, and $s = \phi_\star(x)$, $f^{(h)}_{\mathsf{bayes}}$ satisfies

$$f^{(h)}_{\mathsf{bayes}}(x,a) = \tilde{f}^{(h)}_{\mathsf{bayes}}(s,a) \coloneqq \bar{r}_h(s,a) + \mathbb{E}^{\hat{\pi}^{(h+1)}} \left[\sum_{\tau=h+1}^{H} r_\tau(\boldsymbol{x}_\tau, \hat{\pi}^{(\tau)}(\boldsymbol{x}_\tau)) \middle| \boldsymbol{s}_h = s, \boldsymbol{a}_h = a \right]. \tag{91}$$

Further, $(\tilde{f}^{(h)}_{\mathsf{bayes}}, \phi_\star)$ is the Bayes-optimal solution of the problem in Line 6 of Algorithm 4; that is,

$$(\tilde{f}^{(h)}_{\mathsf{bayes}}, \phi_\star) \in \operatorname*{arg\,min}_{\tilde{f} : [S] \times \mathcal{A} \to \mathbb{R}_{\geq 0}, \phi \in \Phi} \mathbb{E}^{\mathsf{unif}(\Psi^{(h)}) \circ_h \pi_{\mathsf{unif}} \circ_{h+1} \hat{\pi}^{(h+1)}} \left[\left(\tilde{f}(\phi(\boldsymbol{x}_h), \boldsymbol{a}_h) - \sum_{\tau=h}^{H} \boldsymbol{r}_\tau \right)^2 \right].$$

Proof of Lemma I.2. Fix $a \in \mathcal{A}$ and $x \in \mathcal{X}_h$, and let $s = \phi_\star(x)$. The least-squares solution $f^{(h)}_{\mathsf{bayes}}$ of the problem in Eq. (92) is given by

$$\begin{aligned}
f^{(h)}_{\mathsf{bayes}}(x,a) &= \mathbb{E}^{\hat{\pi}^{(h+1)}} \left[\sum_{\tau=h}^{H} \boldsymbol{r}_\tau \middle| \boldsymbol{x}_h = x, \boldsymbol{a}_h = a \right], \\
&= \mathbb{E}[\boldsymbol{r}_h \mid \boldsymbol{x}_h = x, \boldsymbol{a}_h = a] + \mathbb{E}^{\hat{\pi}^{(h+1)}} \left[\sum_{\tau=h+1}^{H} \boldsymbol{r}_\tau \middle| \boldsymbol{x}_h = x, \boldsymbol{a}_h = a \right], \\
&= \bar{r}_h(s,a) + \mathbb{E}^{\hat{\pi}^{(h+1)}} \left[\sum_{\tau=h+1}^{H} \boldsymbol{r}_\tau \middle| \boldsymbol{x}_h = x, \boldsymbol{a}_h = a \right], \quad \text{(by Assumption B.1)} \\
&= \bar{r}_h(s,a) + \mathbb{E}^{\hat{\pi}^{(h+1)}} \left[\sum_{\tau=h+1}^{H} \boldsymbol{r}_\tau \middle| \boldsymbol{s}_h = s, \boldsymbol{a}_h = a \right], \\
&= \tilde{f}^{(h)}_{\mathsf{bayes}}(s,a),
\end{aligned} \tag{92}$$

where Eq. (94) follows by the Block MDP assumption. Now that we have established Eq. (93), we show the second claim of the lemma. The unconstrained population version of the problem in Line 6 of Algorithm 4 becomes equivalent to the following problem:

$$\min_{\tilde{f} : [S] \times \mathcal{A} \to \mathbb{R}_{\geq 0}, \phi \in \Phi} \mathbb{E}^{\mathsf{unif}(\Psi^{(h)}) \circ_h \pi_{\mathsf{unif}} \circ_{h+1} \hat{\pi}^{(h+1)}} \left[\left(\tilde{f}(\phi(\boldsymbol{x}_h), \boldsymbol{a}_h) - \sum_{\tau=h}^{H} \boldsymbol{r}_\tau \right)^2 \right]. \tag{93}$$

Note that the value of this problem is always at least that of Eq. (92). On the other hand, by Eq. (93), the value of the objective in Eq. (95) with the pair $(\tilde{f}, \phi) = (\tilde{f}^{(h)}_{\mathsf{bayes}}, \phi_\star)$ matches the optimal value of the problem Eq. (92), and so $(\tilde{f}^{(h)}_{\mathsf{bayes}}, \phi_\star)$ is indeed a solution of Eq. (95). □

We now restate and prove a slightly more detailed version of Theorem B.1.

Theorem I.1 (Restatement of Theorem B.1)**.** *Let α, ε, $\delta \in (0,1)$ be given. Further, let $\varepsilon_{\mathtt{stat}}(\cdot,\cdot)$ and $C > 0$ be as in Lemma I.1 (C is an absolute constant independent of problem parameters) and suppose that Assumptions 2.1 and B.1 hold, and that for all $h \in [H]$:*

1. *$\Psi^{(h)}$ is a (α,ϵ)-approximate cover for layer h, where $\epsilon := \varepsilon/(2SH^2)$.*
2. *$|\Psi^{(h)}| \le S$.*

Then, as long as n is chosen such that $4S^2HC\varepsilon_{\mathtt{stat}}(n,\delta/H)/\alpha$, we have that with probability at least $1-\delta$, the policy $\hat{\pi}^{(1)}$ outputed by Algorithm 4 satisfies

$$\mathbb{E}^{\hat{\pi}^{(1)}}\left[\sum_{h=1}^{H} \boldsymbol{r}_h\right] \ge \max_{\pi\in\Pi_{\mathtt{M}}} \mathbb{E}^{\pi}\left[\sum_{h=1}^{H} \boldsymbol{r}_h\right] - \varepsilon.$$

In particular, the total number of sampled trajectories required by the algorithm is

$$\widetilde{O}(1)\cdot\frac{H^3S^4(SA+\log(|\Phi|/\delta))}{\alpha^2\varepsilon^2}.$$

Proof of Theorem I.1. We proceed by induction to show that for any $h \in [H]$, there is an event $\mathcal{E}_h$ of probability at least $1-\delta/H$ under which the learned partial policy $\hat{\pi}^{(h)}$ is such that

$$\mathbb{E}^{\pi_\star}\left[Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\pi_\star(\boldsymbol{x}_h)) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\hat{\pi}^{(h)}(\boldsymbol{x}_h))\right] \le \frac{\varepsilon}{H}, \tag{94}$$

where $\pi_\star \in \arg\max_{\pi\in\Pi_{\mathtt{M}}}\mathbb{E}^{\pi}[\sum_{h=1}^{H}\boldsymbol{r}_h]$ is the optimal policy and

$$Q_h^{\pi}(x,a) := \mathbb{E}^{\pi}\left[\sum_{\tau=h}^{H}\boldsymbol{r}_\tau \,\middle|\, \boldsymbol{x}_h = x, \boldsymbol{a}_h = a\right],$$

is the Q-function corresponding to the rewards $(\boldsymbol{r}_h)$ and the policy π. Once we establish Eq. (96) for all $h \in [H]$, we will apply the performance difference lemma to obtain the desired result.

Fix $h \in [H]$. By Lemma I.2, there is an event $\mathcal{E}_h$ of probability at least $1-\delta/H$ under which the solution $(\hat{f}^{(h)},\hat{\phi}^{(h)})$ of the least-squares regression problem on Line 6 of Algorithm 4 satisfies,

$$\mathbb{E}_{\boldsymbol{s}_h\sim\mathtt{unif}(\Psi^{(h)})}\mathbb{E}_{\boldsymbol{x}_h\sim q(\cdot|\boldsymbol{s}_h)}\left[\max_{a\in\mathcal{A}}\left|\hat{f}^{(h)}(\hat{\phi}^{(h)}(\boldsymbol{x}_h),a) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,a)\right|\right] \le C\cdot\varepsilon_{\mathtt{stat}}(n,\tfrac{\delta}{H}), \tag{95}$$

for some absolute constant C independent of h and other problem parameters. Let $\widetilde{\mathcal{S}}_{h,\epsilon} \subseteq \mathcal{S}_h$ be the subset of states s such that $\max_{\pi\in\Pi_{\mathtt{M}}} d^{\pi}(s) < \epsilon$. Moving forward, we let $\varepsilon'_{\mathtt{stat}} := C\cdot\varepsilon_{\mathtt{stat}}(n,\frac{\delta}{H})$ and fix $\mathfrak{s}\in\widetilde{\mathcal{S}}_{h,\epsilon}$. From Eq. (97) and that $|\Psi^{(h)}| \le S$, we have

$$\sum_{\pi\in\Psi^{(h)}} d^{\pi}(\mathfrak{s})\mathbb{E}_{\boldsymbol{x}_h\sim q(\cdot|\mathfrak{s})}\left[\max_{a\in\mathcal{A}}\left|\hat{f}^{(h)}(\hat{\phi}^{(h)}(\boldsymbol{x}_h),a) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,a)\right|\right] \le S\varepsilon'_{\mathtt{stat}}.$$

Now, let $\hat{\pi}^{(h)}(x_h) \in \arg\max_{a\in\mathcal{A}}\hat{f}^{(h)}(\hat{\phi}^{(h)}(x_h),a)$. With this and the fact that $|\|y\|_\infty - \|z\|_\infty| \le \|y-z\|_\infty$, for all $y,z\in\mathbb{R}^A$, we have

$$\begin{aligned}\sum_{\pi\in\Psi^{(h)}} d^{\pi}(\mathfrak{s})\mathbb{E}_{\boldsymbol{x}_h\sim q(\cdot|\mathfrak{s})}\left[\max_{a\in\mathcal{A}} Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,a)\right] &\le \sum_{\pi\in\Psi^{(h)}} d^{\pi}(\mathfrak{s})\mathbb{E}_{\boldsymbol{x}_h\sim q(\cdot|\mathfrak{s})}\left[\hat{f}^{(h)}(\hat{\phi}^{(h)}(\boldsymbol{x}_h),\hat{\pi}^{(h)}(\boldsymbol{x}_h))\right] + S\varepsilon'_{\mathtt{stat}},\\ &\le \sum_{\pi\in\Psi^{(h)}} d^{\pi}(\mathfrak{s})\mathbb{E}_{\boldsymbol{x}_h\sim q(\cdot|\mathfrak{s})}\left[Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\hat{\pi}^{(h)}(\boldsymbol{x}_h))\right] + 2S\varepsilon'_{\mathtt{stat}}.\end{aligned}$$

Thus, since $\Psi^{(h)}$ is a (α,ϵ)-approximate policy cover and $\mathfrak{s}\in\widetilde{\mathcal{S}}_{h,\epsilon}$, we have that

$$2S\varepsilon'_{\mathtt{stat}} \ge \sum_{\pi\in\Psi^{(h)}} d^{\pi}(\mathfrak{s})\mathbb{E}_{\boldsymbol{x}_h\sim q(\cdot|\mathfrak{s})}\left[\max_{a\in\mathcal{A}} Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,a) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\hat{\pi}^{(h)}(\boldsymbol{x}_h))\right],$$

$$\geq \alpha d^{\pi_\star}(\mathfrak{s})\mathbb{E}_{\boldsymbol{x}_h\sim q(\cdot|\mathfrak{s})}\left[\max_{a\in\mathcal{A}} Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,a) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\hat{\pi}^{(h)}(\boldsymbol{x}_h))\right].$$

We have just shown that

$$d^{\pi_\star}(s_h)\mathbb{E}_{\boldsymbol{x}_h\sim q(\cdot|s_h)}\left[\max_{a\in\mathcal{A}} Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,a) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\hat{\pi}^{(h)}(\boldsymbol{x}_h))\right] \leq 2S\varepsilon'_{\mathtt{stat}}/\alpha, \quad \forall s_h \in \widetilde{\mathcal{S}}_{h,\epsilon}. \tag{96}$$

On the other hand, for any $s_h \notin \widetilde{\mathcal{S}}_{h,\epsilon}$, we have $d^{\pi_\star}(s_h) < \epsilon$. Using this and the fact that $Q^{\hat{\pi}^{(h+1)}}(x,a) \in [0,H]$, we have

$$d^{\pi_\star}(s_h)\mathbb{E}_{\boldsymbol{x}_h\sim q(\cdot|s_h)}\left[\max_{a\in\mathcal{A}} Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,a)\right] \leq H\epsilon, \quad \forall s_h \notin \widetilde{\mathcal{S}}_{h,\epsilon}.$$

Combining this with Eq. (98) and that the Q-function is non-negative (by Assumption B.1), we have

$$\begin{aligned}
&\mathbb{E}^{\pi_\star}\left[Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\pi_\star(\boldsymbol{x}_h)) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\hat{\pi}^{(h)}(\boldsymbol{x}_h))\right]\\
&\leq \mathbb{E}^{\pi_\star}\left[\max_{a\in\mathcal{A}} Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,a) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\hat{\pi}^{(h)}(\boldsymbol{x}_h))\right],\\
&= \mathbb{E}^{\pi_\star}\left[\max_{a\in\mathcal{A}} Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,a) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\hat{\pi}^{(h)}(\boldsymbol{x}_h))\right],\\
&\leq d^{\pi_\star}(s_h)\mathbb{E}_{\boldsymbol{x}_h\sim q(\cdot|s_h)}\left[\max_{a\in\mathcal{A}} Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,a) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\hat{\pi}^{(h)}(\boldsymbol{x}_h))\right],\\
&\leq 2S^2\varepsilon'_{\mathtt{stat}}/\alpha + HS\epsilon,\\
&\leq 2S^2\varepsilon'_{\mathtt{stat}}/\alpha + \varepsilon/(2H),
\end{aligned} \tag{97}$$

where the last inequality follows by the fact that $\epsilon = \varepsilon/(2H^2S)$. Now, by choosing n large enough such that $4S^2HC\varepsilon_{\mathtt{stat}}(n,\delta/H)/\alpha \leq \varepsilon$ (as in the theorem's statement), we have that $\varepsilon'_{\mathtt{stat}} \leq \varepsilon\alpha/(4HS^2)$ (by definition of $\varepsilon'_{\mathtt{stat}}$) and so Eq. (99) implies

$$\mathbb{E}^{\pi_\star}\left[Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\pi_\star(\boldsymbol{x}_h)) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\hat{\pi}^{(h)}(\boldsymbol{x}_h))\right] \leq \frac{\varepsilon}{H}. \tag{98}$$

Recall that this inequality holds under the event $\mathcal{E}_h$. On the other hand, by the performance difference lemma (Kakade, 2003) and the definition of $\pi_\star$, the V-function $V_1^\pi(x) \coloneqq \mathbb{E}^\pi[\sum_{h=1}^H \boldsymbol{r}_h \mid \boldsymbol{x}_1 = x]$ satisfies

$$\begin{aligned}
\mathbb{E}[V_1^{\hat{\pi}^{(1)}}(\boldsymbol{x}_1)] - \max_{\pi\in\Pi_{\mathsf{M}}}\mathbb{E}[V_1^\pi(\boldsymbol{x}_1)] &= \mathbb{E}[V_1^{\hat{\pi}^{(1)}}(\boldsymbol{x}_1)] - \mathbb{E}[V_1^{\pi_\star}(\boldsymbol{x}_1)],\\
&= \mathbb{E}^{\pi_\star}\left[Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\pi_\star(\boldsymbol{x}_h)) - Q_h^{\hat{\pi}^{(h+1)}}(\boldsymbol{x}_h,\hat{\pi}^{(h)}(\boldsymbol{x}_h))\right].
\end{aligned}$$

Thus by Eq. (100), we have that under the event $\mathcal{E} \coloneqq \bigcup_{h=1}^H \mathcal{E}_h$,

$$\mathbb{E}[V_1^{\hat{\pi}^{(1)}}(\boldsymbol{x}_1)] - \max_{\pi\in\Pi_{\mathsf{M}}}\mathbb{E}[V_1^\pi(\boldsymbol{x}_1)] \leq \varepsilon.$$

The desired suboptimality result follow by the fact that $\mathbb{P}[\mathcal{E}] \geq 1-\delta$.

Sample complexity of PSDP. In order to satisfy the condition $4S^2HC\varepsilon_{\mathtt{stat}}(n,\delta/H)/\alpha \leq \varepsilon$ in the theorem statement (where C is some absolute constant), n needs to be larger than $N = \widetilde{O}(1)\cdot(H^2S^4(SA + \log(|\Phi|/\delta))/(\alpha\varepsilon)^2)$, where $\widetilde{O}$ hides log-factors in $1/\varepsilon, A, S, H$, and $\log|\Phi|$. Since n represents the number of sampled trajectories per layer in PSDP, the total number of sampled trajectories in the latter is simply $N_{\mathsf{PSDP}} = HN$. □